



Universidad
Nacional
de Loja

Universidad Nacional de Loja

Facultad de la Energía, las Industrias y los Recursos

Naturales No Renovables

Carrera de Ingeniería en Sistemas

**Diseño de aplicación web mediante el patrón de diseño Factory
para Micro-Learning del lenguaje de señas ecuatoriano**

Trabajo de Titulación
previa a la obtención del
título de Ingeniero en
Sistemas

AUTOR:

Jonnathan Damián Espinoza Erráez

DIRECTOR:

Ing. Edison Leonardo Coronel Romero, Mg.Sc.

Loja – Ecuador

2022

Certificación

Loja, 18 de octubre de 2022

Ing. Edison Leonardo Coronel Romero, Mg. Sc.

DIRECTOR DEL TRABAJO DE TITULACIÓN

CERTIFICO:

Que he revisado y orientado todo el proceso de elaboración del Trabajo de Titulación denominado: **Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del Lenguaje de señas ecuatoriano**, previo a la obtención del título de **Ingeniero en Sistemas**, de la autoría del estudiante **Jonnathan Damián Espinoza Erráez**, con **cédula de identidad Nro. 0107022006**, una vez que el trabajo cumple con todos los requisitos exigidos por la Universidad Nacional de Loja, para el efecto, autorizo la presentación del mismo para su respectiva sustentación y defensa.

Ing. Edison Leonardo Coronel Romero, Mg. Sc.

DIRECTOR DEL TRABAJO DE TITULACIÓN

Autoría

Yo, **Jonnathan Damián Espinoza Erráez**, declaro ser el autor del presente Trabajo de Titulación y eximo expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos de posibles reclamos o acciones legales por el contenido del mismo. Adicionalmente acepto y autorizo a la Universidad Nacional de Loja, la publicación de mi Trabajo de Titulación en el Repositorio Institucional - Biblioteca Virtual.

Firma:

Cédula de Identidad: 0107022006

Fecha: 07 de diciembre de 2022

Correo electrónico: jonnathan.espinoza@unl.edu.ec

Teléfono: 0989614685

Carta de autorización por parte del autor, para la consulta de producción parcial o total, y/o publicación electrónica del texto completo, del Trabajo de Titulación

Yo **Jonnathan Damián Espinoza Erráez**, declaró ser el autor del Trabajo de Titulación denominado: **Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del Lenguaje de señas ecuatoriano**; autorizo al Sistema Bibliotecario de la Universidad Nacional de Loja para que, con fines académicos, muestre al mundo la producción intelectual de la Universidad, a través de la visibilidad de su contenido de la siguiente manera en el Repositorio Institucional:

Los usuarios pueden consultar el contenido de este trabajo en el Repositorio Institucional, en las redes de información del país y del exterior, con los cuales tenga convenio la Universidad.

La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia del Trabajo de Titulación que realice un tercero.

Para constancia de esta autorización, en la ciudad de Loja, a los siete días del mes de diciembre de dos mil veintidós.

Firma:

Autor: Jonnathan Damián Espinoza Erráez

Cédula: 0107022006

Dirección: Loja

Correo electrónico: jonnathan.espinoza@unl.edu.ec

Celular: 0989614685

DATOS COMPLEMENTARIOS:

Director del Trabajo de Titulación: Ing. Edison Leonardo Coronel Romero, Mg.Sc.

Dedicatoria

El Trabajo de Titulación lo dedico con infinito amor a mis padres Luis Gerardo y Natalia Beatriz quienes supieron cuidarme y protegerme de los males, con su cariño y bondad me inculcaron buenos valores, guiándome por el camino de la vida, la educación, la sabiduría; y junto con mi hermano Kevin Andrés que es siempre mi apoyo y fuerza en todas mis actividades y por ser el cómplice en todo lo que realizo.

A toda mi familia que con su apoyo moral e incondicional hicieron posible el poder alcanzar tan prestigiada meta de mi vida.

A mis compañeros y amigos que estuvieron en el proceso de aprendizaje y compartieron varios momentos en el paso por esta etapa de formación. Finalmente quiero dedicar este trabajo de investigación a todos los que formaron parte de mi institución tanto profesional como personal como los docentes de la carrera y tutores de pasantías pre profesionales.

Jonnathan Damián Espinoza Erráz

Agradecimiento

Sobre todo, quiero agradecer a mis padres por su apoyo en mi educación y la realización de mis sueños, y a mi familia y amigos que siempre me han apoyado y comprendido. Agradezco infinitamente a Dios por lo que me ha brindado, especialmente por darme la oportunidad de adquirir los conocimientos necesarios para desarrollar este proyecto.

Agradezco mucho a mi líder del Trabajo de Titulación, Ing. Edison Leonardo Coronel Mg Sc, por su sabia orientación y excelente asesoría a lo largo del desarrollo de este trabajo de investigación. Finalmente quiero agradecer a la Universidad Nacional de Loja, en especial a la Carrera de Ingeniería en Sistemas y a su personal administrativo y docente. Han sido parte de mi crecimiento para alcanzar todas mis metas como estudiante.

Jonnathan Damián Espinoza Erráz

Índice de Contenidos

Portada	i
Certificación	ii
Autoría	iii
Carta de autorización	iv
Dedicatoria	v
Agradecimiento	vi
Índice de Contenidos	vii
Índice de Tablas	ix
Índice de Figuras	xi
Índice de Anexos	xii
1. Título	1
2. Resumen	2
2.1. Abstract.....	3
3. Introducción	4
4. Marco teórico	6
4.1. Lengua de Señas	6
4.2. Lengua de Señas Ecuatoriana	6
4.3. Modelo conceptual para el diseño web para personas con discapacidad auditiva... ..	7
4.4. Micro-learning	7
4.5. Patrones de Diseño.....	10
4.6. Patrón de Diseño Factory.....	11
4.7. Comparativa de la metodología de desarrollo tradicional vs ágil	11
4.8. Comparativa entre metodologías de desarrollo ágil.....	13
4.9. Programación Extrema.....	13
4.10. Comparativa de base de datos relacionales vs NoSQL	15
4.11. Comparativa entre bases de datos NoSQL.....	16
4.12. Node.js	17
4.13. API	17

4.14.	REST.....	19
4.15.	Autenticación por token	22
4.16.	JSON Web Token.....	23
4.17.	Trabajos relacionados	24
5.	Metodología	26
5.1.	Proceso.....	26
5.2.	Métodos	27
5.3.	Técnicas.....	27
5.4.	Estándares.....	28
5.5.	Metodologías de desarrollo de software	28
5.6.	Participantes	28
6.	Resultados.....	30
6.1.	Objetivo 1	30
6.2.	Objetivo 2.....	37
6.3.	Objetivo 3.....	51
7.	Discusión	61
7.1.	Desarrollo de la propuesta alternativa	61
7.2.	Valoración técnica económica ambiental.....	64
8.	Conclusiones.....	67
9.	Recomendaciones.....	68
10.	Bibliografía.....	69
11.	Anexos	74

Índice de Tablas:

Tabla 1. Características del Microlearning	8
Tabla 2. Materiales empleados en el ámbito del microlearning [14]	9
Tabla 3. Clasificación de los Patrones de Diseño [15].....	10
Tabla 4. Diferencias entre metodologías Tradicionales y Ágiles.....	12
Tabla 5. Comparación de metodologías de desarrollo ágil.....	13
Tabla 6. Prácticas de la metodología XP [26].....	14
Tabla 7. Comparación de las bases de datos Relacionales y NoSQL.....	16
Tabla 8. Comparación de diferentes bases de datos NoSQL.....	16
Tabla 9. Comparación de XML-RPC/SOAP y servicio REST [36]	19
Tabla 10. Características de los métodos HTTP [34]	21
Tabla 11. Trabajos relacionados	24
Tabla 12. Recurso Humano	28
Tabla 13. Requerimientos funcionales	32
Tabla 14. Requerimientos no funcionales	33
Tabla 15. Historias de usuario.....	34
Tabla 16. Tecnologías utilizadas en el desarrollo de la aplicación web	37
Tabla 17. Planificación de las iteraciones de la metodología XP.....	38
Tabla 18. Estructura de carpetas en el back-end	41
Tabla 19. Estructura de carpetas en el front-end con React.....	44
Tabla 20. Código de la clase controllersFactory	47
Tabla 21. Código del endpoint de la API	47
Tabla 22. Código para hacer peticiones al servidor desde el cliente	48
Tabla 23. Contenido de la aplicación web.....	53
Tabla 24. Script en Python para transformar video a gif.....	54
Tabla 25. Casos de pruebas funcionales	55
Tabla 26. Resumen de Pruebas de Aceptación	57
Tabla 27. Criterio para el estado de las Pruebas de Aceptación	57
Tabla 28. Resumen de las encuestas a profesionales en LSEC	58
Tabla 29. Valoración técnica.....	64
Tabla 30. Valoración económica de talento humano.....	65
Tabla 31. Valoración económica de recursos de hardware y software	65
Tabla 32. Valoración económica de servicios	66

Tabla 33. Presupuesto final66

Índice de Figuras:

Figura 1. Beneficios del micro-learning [14].	8
Figura 2. Anatomía de la cápsula de aprendizaje [14].	9
Figura 3. Ejemplo de la implementación del Patrón de Diseño Factory [15].	11
Figura 4. El proceso de la programación extrema [25].	15
Figura 5. Estructura de la arquitectura REST.	19
Figura 6. Tendencia de búsqueda en Stack Overflow de JSON frente XML [37].	22
Figura 7. Proceso de autenticación por cookies y tokens.	23
Figura 8. Proceso de autenticación con JWT [40].	23
Figura 9. Ejemplo de JWT [38].	24
Figura 10. Estructura JWT [39].	24
Figura 11. Wireframe para enseñar lengua de señas.	31
Figura 12. Wireframe para evaluar conocimiento de señas.	32
Figura 13. Diagrama de casos de uso de la aplicación desarrollada.	35
Figura 14. Prototipo de la cápsula de aprendizaje.	36
Figura 15. Prototipo de la tarjeta de aprendizaje.	37
Figura 16. Control de versiones con GitKraken.	39
Figura 17. Arquitectura de la aplicación web.	40
Figura 18. Estructura de carpetas en el back-end.	40
Figura 19. Diagrama de clases de la API.	42
Figura 20. Diagrama de clases del patrón Factory Method de la solución informática.	43
Figura 21. Estructura de carpetas en el front-end con React.	43
Figura 22. Diagrama de despliegue.	44
Figura 23. Diagrama de componentes de la aplicación web.	45
Figura 24. Total, de pruebas realizadas a los endpoints de la API.	49
Figura 25. Ejemplo de test aplicado al endpoint de la API.	49
Figura 26. Total, de pruebas front-end.	50
Figura 27. Prueba unitaria de la interfaz para crear cuenta de usuario.	50
Figura 28. Resultado de las Pruebas de Integración de la aplicación web.	51
Figura 29. Creación y despliegue de la base de datos en MongoDB Atlas.	52
Figura 30. Despliegue de la API en Render Hosting.	52
Figura 31. Despliegue de la interfaz de usuario de React en Vercel.	53

Índice de Anexos:

Anexo 1. Elicitación de Requerimientos.....	74
Anexo 2. Especificación de Requerimientos de Software	81
Anexo 3. Prototipo de Interfaz de Usuario	94
Anexo 4. Descripción de Historias de Usuario	102
Anexo 5. Descripción del Diagrama de Clases	112
Anexo 6. Descripción del Diagrama de Componentes.....	113
Anexo 7. Desarrollo de la Aplicación Web	114
Anexo 8. Plan de pruebas de la API	155
Anexo 9. Plan de pruebas unitarias del Front-end	163
Anexo 10. Plan de Pruebas de Integración.....	170
Anexo 11. Plan de Pruebas Funcionales	180
Anexo 12. Plan de Pruebas de Aceptación.....	206
Anexo 13. Encuestas a maestros en lengua de señas ecuatoriana	220
Anexo 14. Certificado de la Asociación de Personas Sordas Virgen del Cisne	236
Anexo 15. Manual de Usuario.....	237
Anexo 16. Manual de Administrador	250
Anexo 17. Prueba de aceptación a profesional de la educación	274
Anexo 18. Certificado de traducción del resumen.....	284

1. Título

Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano

2. Resumen

El Trabajo de Titulación (TT) consistió en diseñar una aplicación web de micro-learning mediante el patrón de diseño Factory para el aprendizaje de Lengua de Señas Ecuatoriana (LSEC), creando una plataforma escalable con el uso de patrones de diseño con la finalidad de motivar a la población de 18 a 30 años en obtener conocimientos del vocabulario LSEC y así mejorar la comunicación e inclusión social con la comunidad sorda.

La primera etapa del TT, empezó con la selección de los actores involucrados en la aplicación web y los requerimientos para la misma, que se obtuvieron mediante la técnica de la entrevista y el estudio de documentación. Por consiguiente, en la segunda etapa del TT inició el desarrollo del software por medio de la metodología Programación Extrema (XP), generando una API REST. Finalmente, en la última etapa se logró crear una aplicación web funcional mediante el micro-learning usando como recurso los GIF para la enseñanza de Lengua de Señas Ecuatoriana, luego de la utilización de las pruebas de aceptación determinando la funcionalidad idónea de la solución informática con la aplicación de tecnologías como MongoDB, Express.js, React.js y Node.js.

Palabras claves: API, metodología xp, micro-learning, cápsulas de aprendizaje, tarjetas de aprendizaje.

2.1. Abstract

As part of this research, a micro-learning web application was designed based on the Factory design pattern for learning Ecuadorian Sign Language (LSEC), created a scalable platform using design patterns, and motivating a group of 18-30-year-olds to become familiar with the LSEC vocabulary, thereby improving communication and social inclusion with the deaf community.

The first stage of this project, consisted on the selection of the stakeholders involved in the web application and the requirements, which were obtained through the interview technique and the documentation study. Consequently, in the second stage of the project, a development of the software was carried out by means of the Extreme Programming (XP) methodology, generating a REST API. Finally, in the last stage, we were able to develop a functional web application through micro-learning using GIFs as a resource for teaching Ecuadorian Sign Language using GIFs, after evaluating whether the software solution met the criteria for suitable functionality using technologies such as MongoDB, Express.js, React.js and Node.js.

Keywords: API, xp methodology, micro-learning, learning capsules, learning cards.

3. Introducción

La inclusión de las personas con discapacidad busca nuevos enfoques y herramientas para solucionar muchos de los problemas que sufre este colectivo. El informe [1] establece que en la emergencia sanitaria por la pandemia del virus COVID-19, las personas con discapacidad que ya son las más marginadas de la sociedad, pueden ver agravada esta situación. El coronavirus ha puesto en manifiesto las dificultades que tienen las personas con discapacidad auditiva en su vida cotidiana: acceder a la mayoría de información televisiva, comprar en los supermercados, recibir atención médica, ir a la farmacia e incluso tomar clases en línea, ya que en lo general se lo realiza a través de llamadas orales. Existen tres maneras para vencer estos obstáculos, la primera es la lectura de los labios, la segunda opción es tener un intérprete que lo acompañe en cada momento, y por último es mediante el lenguaje de señas.

El lenguaje de señas es un sistema lingüístico visual, espacial, manual y gestual, condicionado por factores lingüísticos, culturales, históricos y sociales, siendo utilizado por personas sordas y sordociegas [2]. En Ecuador en su Art. 47, de la Constitución establecida en 2008, se reconoce a la Lengua de Señas Ecuatoriana (LSEC) como derecho de las nuevas formas de comunicación de las personas con discapacidad auditiva, beneficiando de manera especial a la comunidad sorda. Por ello, se han efectuado varias investigaciones, dentro de las cuales resaltan la guía enfocada en la LSEC publicada por la Asociación Ponce León y el Diccionario Oficial "Gabriel Román", con una página web donde contiene alrededor de 5.000 palabras de LSEC incluyendo gráficos y videos explicativos. De acuerdo con los estudios expuestos anteriormente, no presentan una metodología de enseñanza dentro de la web; por consiguiente, el trabajo investigativo tiene como reto social lograr crear una aplicación web para enseñar LSEC mediante el micro-learning.

El presidente de la Asociación de Personas Sordas Virgen del Cisne de la ciudad de Loja, mediante entrevistas realizadas, manifestó que el problema en la pandemia para las personas con discapacidad auditiva fue el de comprar en tiendas y farmacias porque los que atendían no sabían lengua de señas y tenían que escribir en un papel para poder comunicarse. También, se encontró que las personas sordas tienen libre acceso a la educación de la LSEC, ya que están en su derecho, por tal motivo se necesita que las personas sin discapacidad aprendan lengua de señas de forma voluntaria y comprendan el valor de la inclusión. Por ello se necesitan aplicaciones que motiven a las personas ecuatorianas aprender LSEC (para mayor información de la entrevista, revisar el Anexo 1. Elicitación de Requerimientos).

Por consiguiente, en el Trabajo de Titulación (TT) se diseña una aplicación web para enseñar de forma básica la lengua de señas ecuatoriana mediante el micro-learning a personas sin

discapacidad de 18 a 30 años. Para que esta sea escalable y en un futuro cualquier desarrollador pueda mejorar la aplicación, se propone la utilización de patrones de diseño.

Los patrones de diseño son soluciones a los problemas de desarrollo [3]. El rápido desarrollo de la industria de la tecnología significa que las prácticas de desarrollo de software evolucionan constantemente. Antes las pruebas de software se realizaban después de haber terminado la codificación, pero ahora se introducen las pruebas dentro de las fases de los métodos ágiles, entonces los patrones de diseño garantizan la validez del código probado por los desarrolladores, siendo menos propenso a cometer errores de código.

La aplicación web desarrollada en el presente TT se basa en una API Rest por su escalabilidad, seguridad mediante tokens e independencia de tecnologías. Para mejorar la legibilidad del código y la creación de objetos, se eligió el patrón de Diseño Factory a fin de instanciar las clases de los controladores dentro de las rutas de la API.

El Trabajo de Titulación tiene por objetivo general el de “Diseñar una aplicación web de Micro-Learning mediante el patrón de diseño Factory para el aprendizaje del Lenguaje de Señas Ecuatoriano”, y para cumplirlo se definieron tres objetivos específicos: 1. Realizar la ingeniería de requerimientos para la aplicación web de Micro-Learning dedicada al aprendizaje del Lenguaje de Señas Ecuatoriano, 2. Desarrollar la aplicación web de Micro-Learning para el aprendizaje del Lenguaje de Señas Ecuatoriano y 3. Evaluar la aplicación web en un entorno simulado para determinar la fiabilidad del diseño de la aplicación web.

Las secciones del TT se describen a continuación. La sección Marco Teórico proporciona diecisiete capítulos de información bibliográfica referente al objeto de estudio. La sección de Metodología describe los procesos, métodos, técnicas y estándares, así como los recursos utilizados y los participantes involucrados en el TT. La sección de Resultados se describen las evidencias obtenidas durante el desarrollo del TT, divididas en tres apartados, uno por cada objetivo. La parte de Discusión analiza los resultados obtenidos y presenta las contribuciones del autor basándose en los resultados y evidencias bibliográficas. La sección de conclusiones presenta los principales datos encontrados a lo largo de la ejecución del TT. Finalmente, la sección Recomendaciones, brinda sugerencias y aspectos a considerar para trabajos futuros.

4. Marco teórico

Esta sección recoge la información bibliográfica más relevante para la elaboración del Trabajo de Titulación (TT).

4.1. Lengua de Señas

Alonso y Valmaseda, citados en [4], definen la lengua de señas (LS) como una forma no vocal del lenguaje hablado que desarrollan de forma natural las personas sordas. En concreto, la LS es el idioma que, a través de expresiones visuales y gestos la comunidad sorda se puede comunicar.

La sintaxis de la LS difiere del lenguaje oral, por ejemplo, la LS Mexicana es diferente a la del español, ya que la primera se construye señalando tiempo-lugar, objetos-sujeto-verbo, siendo que en español la secuencia usual se da en términos del sujeto-verbo-objetos-tiempo-lugar [5]. Dicha sintaxis varía en cada lugar o región, teniendo en Ecuador su propia Lengua de Señas.

4.1.1. Características de la lengua de señas

Eulalia Eguiguren, citado en [6] señala algunas ventajas de la LS:

- Las cuerdas vocales no se utilizan para la comunicación.
- Tiene su propia estructura gramatical.
- Posee percepción visual y expresión gesto espacial.
- Independencia del lenguaje hablado.
- Visual.
- El conocimiento y la información pueden expresarse simultáneamente a través de diferentes canales.

La desventaja de la LS es que no todas las palabras tienen signos. Como tal, el uso del alfabeto dactilológico es fundamental para representar los conceptos que el emisor desea [6].

4.2. Lengua de Señas Ecuatoriana

La comunidad sorda en Ecuador utiliza la Lengua de Señas Ecuatoriana (LSE) para comunicarse, en inglés se utiliza las siglas LSEC para representar esta lengua [7]. La LSEC fue reconocida por primera vez como un derecho de comunicación en 1998 mediante el Art. 53 de la Constitución Política de la República del Ecuador, y en el 2008 con la reformatión de la Constitución de la República del Ecuador se formula el mismo derecho a través del Art. 47 [8].

En la investigación [9], a través de la recopilación de datos de 11 ciudades del país, en sus resultados indican que el vocabulario de la LSEC tiene influencia de la Lengua de Señas

Americana (ASL) un 30 %, la Lengua de Señas de España un 20% y las antiguas señas ecuatorianas un 50%; además la LSEC posee dos dialectos diferenciados que es el de la zona costera y la zona de montaña.

La primera guía sobre la LSEC fue publicada en 1988 por la Asociación Ponce de León, en donde recopila doscientas cincuenta señas usadas en Quito y Guayaquil. La obra se basa en fotografías y dibujos. En 2012 se publicó el Diccionario de Lengua de Señas Ecuatoriana bajo la coordinación de la Federación Nacional de Sordos del Ecuador (FENASEC). También se cuenta con un diccionario LSEC en la web con 5.000 palabras que incluye gráficos y videos explicativos [7].

4.3. Modelo conceptual para el diseño web para personas con discapacidad auditiva

La difusión de información por Internet se basa más en la vista que en el oído, por lo que este medio obviamente sería ideal para las personas con discapacidad auditiva. Sin embargo, la discapacidad auditiva no solo causa la dificultad en la comprensión del habla en la comunicación oral, sino que también conduce a déficit significativo en el procesamiento de cualquier información verbal, ya sea hablada o escrita [10].

Fajardo Bravo en [10] establece tres aspectos que se deben tener en cuenta para diseñar una página web accesible para personas con discapacidad auditiva, las cuales son:

- Cantidad y tipo de información que procesan los sordos: No sobrecargar el sitio con palabras escritas e imágenes, utilizar términos más comunes o de uso más frecuente.
- Estrategias de procesamiento de los sordos: Utilizar elementos intuitivos y no sobrecargar la página web de información, utilizar íconos o pequeñas palabras universales y evitar el uso de mucho texto.
- Actividades cognitivas que demanda la web: Facilitar la navegación y comprensión web evitando usar demasiados enlaces.

4.4. Micro-learning

El micro-learning o microaprendizaje en español, emerge del micro contenido de pequeños fragmentos de información digital. Se basa en el uso de pequeñas unidades de contenido de aprendizaje y tecnología flexible que lo hacen más accesible a las personas en momentos específicos y situaciones cotidianas [11]. Este enfoque representa una forma de aprendizaje didáctico que utiliza medios digitales para proporcionar contenido pequeño, coherente y autónomo para actividades de aprendizaje cortas [12]. El objetivo principal es que los usuarios adquieran nuevos conocimientos mediante contenidos multimedia, como: videos, afiches, audios, imágenes, etc. [13].

De acuerdo a la definición de Gassler en [14] el microaprendizaje posee ciertas características con base en los microcontenidos (nuggets) y actividades (pasos), los cuales se detallan en la Tabla 1.

Tabla 1.
Características del Microlearning

Microaprendizaje	Características
Microcontenido	Los microcontenidos son pequeños (por ejemplo, caben en pantallas pequeñas).
	Están centradas en un tema (un solo hecho o concepto).
	Poseen una estructura sencilla y fácil de entender (carga cognitiva reducida).
Actividades	Son breves (segundos en lugar de minutos).
	Son interactivas (requieren la intervención del usuario).
	Proporcionan retroalimentación (como reacción directa a la intervención del usuario).
	Provocan la reflexión de los alumnos.

Con las características de la Tabla 1, los alumnos eligen su propio ritmo e integran las actividades de aprendizaje en sus rutinas diarias [12]. Estas características dan beneficios al emplearlo, como se visualiza en la Figura 1 tomado de [15].



Figura 1. Beneficios del micro-learning [15].

Para impartir el contenido de corta duración dentro del ámbito del micro-learning, se lo realiza mediante las Cápsulas de Aprendizaje. Estos varían según el contexto en el que se desarrollaron y los materiales multimedia utilizados durante la fase de aprendizaje. Sin embargo, estas cápsulas tienen tres actividades cognitivas: aprender, practicar y evaluar [15]. La estructura de la cápsula se visualiza en la Figura 2.

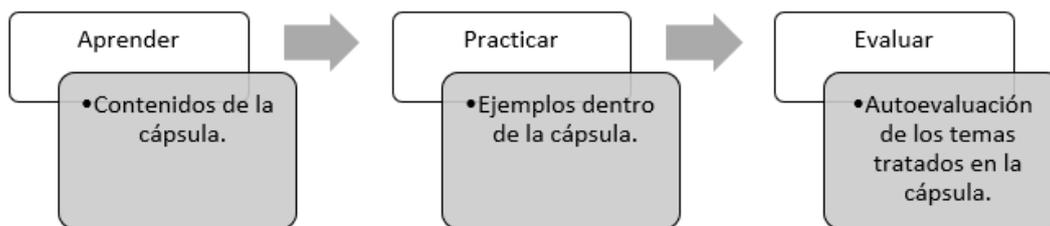


Figura 2. Anatomía de la cápsula de aprendizaje [15].

4.4.1. Materiales multimedia en micro-learning

Daniel Gómez en [15], extrae una lista de materiales para representar los microcontenidos, las cuales se presentan en la Tabla 2.

Tabla 2.
Materiales empleados en el ámbito del microlearning [15]

Material multimedia	Descripción
Videos	El contenido de video se presenta a los estudiantes de manera dinámica, atractiva y emocional. Se deben considerar las siguientes propiedades: <ul style="list-style-type: none"> • Manténgalo breve y no más de 10 minutos. • Sea específico y conciso en lo que quiere transmitir. • Dinámico. • Tiene normas. • Emociones para afectar emocionalmente a los alumnos.
Cuadros de texto	El propósito de usar texto es explicar temas que los elementos dinámicos no pueden cubrir. Aquí hay algunas recomendaciones para crear este tipo de contenido: <ul style="list-style-type: none"> • Oraciones cortas y fluidas. • Lenguaje simple. • Introduzca sólo las palabras que necesita. • Texto conciso y valioso. • Escribir de forma clara, directa y precisa. • Párrafos cortos. • Voz activa.
Imágenes	El propósito de las imágenes es mejorar el aprendizaje al complementar o representar conceptos simples. Aquí hay algunas recomendaciones para crear este tipo de contenido: <ul style="list-style-type: none"> • El texto de las imágenes debe ser mínimo y nítido/de alta resolución. • Las imágenes se asocian con cuadros de texto para asociar conceptos y transmitir la idea que se expresa.
Tarjetas de aprendizaje	Estos también se llaman flashcards o mapas interactivos. Están diseñados para ayudar a los alumnos a recordar información específica. Estas tarjetas tienen una imagen en un lado de la tarjeta y el texto correspondiente en el otro lado. Ayuda a los alumnos a practicar la memorización de oraciones, palabras o frases. E-learning utiliza este recurso para estudiar idiomas, problemas matemáticos, historia y más.
Piezas de audio	Consiste en grabaciones de audio entregadas en formato episódico. Este elemento es una alternativa rentable al video para superar la palabra escrita.
Afiches digitales	Los carteles, pósteres y afiches digitales deben combinar elementos gráficos como imágenes y texto para que sean efectivos. Se lo emplea para transmitir contenido formativo de manera breve, enfocados en una temática en específico.

4.5. Patrones de Diseño

Brad Appleton, citado por Pressman en [3] define los patrones de diseño como: una solución comprobada para problemas recurrentes dentro de un contexto específico de necesidades en competencia. Dicho de otra manera, un patrón de diseño es una solución que se aplica a diferentes problemas de diseño de software.

Los patrones de diseño pueden cubrir una amplia gama de abstracciones y aplicaciones, por ende, dentro de la Ingeniería de Software se define las siguientes categorías según [3]:

- Patrones arquitectónicos: Enfocados en resolver problemas de diseño estructurales.
- Patrones de datos: Enfocados en solucionar problemas orientados al modelado de datos.
- Patrones de componentes: también conocidos como patrones de diseño, que están enfocados en resolver problemas relacionados al desarrollo de subsistemas y componentes.
- Patrones de diseño de la interfaz: Resuelven problemas relacionados a la interfaz de usuario final.
- Patrones web App: Están enfocadas en resolver los problemas que surgen al desarrollar web Apps y por lo general incorporan los patrones de las otras categorías.

El presente Trabajo de Titulación hace uso del término “Patrón de Diseño” para referenciar a la categoría “Patrón de Componentes”.

En [16] se exponen por primera vez los patrones de diseño del desarrollo de software, recopilando veintitrés patrones de diseño, además de definir tres categorías de acuerdo al propósito del patrón: creación, estructural y de comportamiento. Los criterios de la clasificación junto con los patrones de diseño, se muestran en la Tabla 3 tomada de [16].

Tabla 3.
Clasificación de los Patrones de Diseño [16]

		Propósito		
		De Creación	Estructurales	De Comportamiento
Ámbito	Clase	Factory Method	Adapter (de clases)	Interpreter Template Method
	Objeto	Abstract Factory Builder Prototype Singleton	Adapter (de objetos) Bridge Composite Decorator Facade Flyweight Proxy	Chain of Responsibility Command Iterator Mediator Memento Observer State

				Strategy Visitor
--	--	--	--	---------------------

4.6. Patrón de Diseño Factory

El propósito del patrón de diseño Factory es establecer una interfaz para crear un objeto, permitiendo que la clase asigne la creación de objetos a sus subclasses [16]. Dicho de otra manera, el patrón Factory provee un método abstracto de creación de un objeto delegando a las subclasses su creación efectiva.

El patrón Factory se aplica cuando [16]:

- Una clase quiere especificar los objetos que instancian sus subclasses.
- La clase no conoce los objetos que debe crear.
- La clase delega la responsabilidad a una de varias clases auxiliares.

La Figura 3 muestra un ejemplo de implementación del patrón Factory. Product define una interfaz de los objetos que crea el método de fabricación, ConcreteProduct hereda la interfaz Product, Creator especifica el método de fabricación y retorna un objeto de tipo Product, finalmente, ConcreteCreator establece el método de fabricación para retornar una instancia de ConcreteProduct.

Como se puede visualizar en la Figura 3, lo esencial de este patrón es la utilización de una clase intermedia, en este caso Creator que provee objetos. Los objetos que se desean instanciar deben tener los mismos métodos, por tal razón en la Programación Orientada a Objetos (POO) se crea una clase abstracta o interfaz.

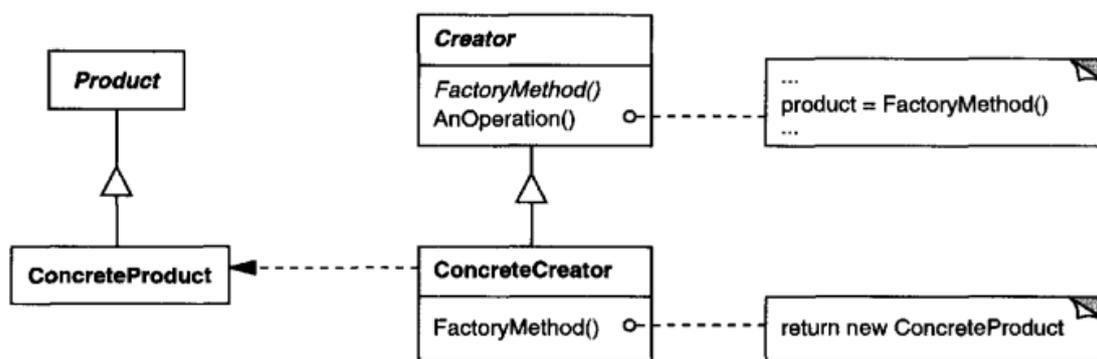


Figura 3. Ejemplo de la implementación del Patrón de Diseño Factory [16].

4.7. Comparativa de la metodología de desarrollo tradicional vs ágil

Se analizaron los estudios [17], [18], [19], [20], [21] y [22] para realizar la comparación entre las metodologías tradicionales y ágiles. La Tabla 4 muestra la abstracción de las características de estas metodologías.

Tabla 4.
Diferencias entre metodologías Tradicionales y Ágiles

Aspecto	Metodología	
	Tradicional	Ágil
Enfoque	Predictivo	Adaptativo
Tamaño del proyecto	Grande	Pequeño
Tamaño de los equipos de trabajo	Grandes	Pequeños (<10 integrantes)
Roles de los integrantes	Específicos	Flexibles
Resistencia al cambio durante el proyecto	Alta	Baja
Énfasis	Orientado a los procesos	Orientado a las personas
Tiempo del proyecto	Largo	Corto
Políticas/normas/ control	Numerosos	Pocos
Tipo de contrato con el cliente	Fijo	Flexible
Énfasis en la arquitectura del software	Muy alto	Bajo
Cantidad de documentación	Extensa	Poca
Entrega del producto	Al final del proyecto	Entrega constantes de prototipos
Frecuencia de comunicación y forma de interacción con el cliente	Bajo/mediante reuniones	Constante/ forma parte del equipo de desarrollo
Centro de la metodología	Procesos: roles, actividades, artefactos	Las personas y el trabajo en equipo
Costo y envergadura de los proyectos	Altos	Bajos
Momento en que se define la arquitectura del software	Etapas tempranas del proyecto	Se define y mejora durante el proyecto
Control de Calidad	Planificación difícil y control estricto. Pruebas difíciles y tardías	Control permanente de requisitos, diseño y soluciones. Pruebas permanentes
Éxito de medición	Conformación de planificar	Valor de negocio
Estilo de gestión	Autocrático	Descentralizada
Modelo de desarrollo	Modelo de ciclo de vida	Modelo evolutivo de entrega
Requisitos de usuario	Detallado y definido antes de la codificación /implementación	Entrada interactiva
Pruebas	Después de completar la codificación	Cada iteración

A través del análisis de la Tabla 4, se determinó que la metodología más apropiada para implementar en el Trabajo de Titulación (TT), son las metodologías ágiles, debido a que está orientada a proyectos pequeños, es adaptable por lo que no le afecta en gran medida los cambios, no necesita de un gran equipo de trabajo, además el tiempo de desarrollo es corto. Estas características van conforme al tipo de TT planteado, porque el desarrollo de la misma está a cargo una sola persona en un plazo de 400 horas.

4.8. Comparativa entre metodologías de desarrollo ágil

Se tomó en cuenta los trabajos [23], [18], [20] y [24] para obtener las características de las metodologías ágiles más importantes. En la Tabla 5 se muestran estas características.

Tabla 5.
Comparación de metodologías de desarrollo ágil

Características	XP	SCRUM	DSDM	FDD	ASD
Enfoque	Iterativo, incremental	Incremental, iterativo	Iterativo	Iterativo	Iterativo, incremental
Periodo del ciclo de interacción	1 a 6 semanas	2 a 4 semanas	En 20% del tiempo total 80% de producto	2 días a 2 semanas	4 a 8 semanas
Tamaño del proyecto	Pequeño	Grande	Todo tipo	Gran escala	Pequeño
Complejidad del problema	Sencillo	Complejos	Complejos	Complejos	Sencillo
Tamaño del equipo	Pequeño	Pequeño	Pequeño/Mediano	Pequeño	Mediano
Curva de aprendizaje	Rápida	Rápida	Lenta	Rápida	Rápida
Documentación	Documentación básica	Documentación básica	Más que la metodología SCRUM y XP	Más extensa que entre todos	Documentación básica
Principales Prácticas	Simplicidad, Programación en pares	Reuniones de SCRUM	Prototipado, MoSCoW, Time boxing	Desarrollo por características, Modelado de objetos, utilización de diagramas UML	Basado en características, Time boxing, Risk Driven
Involucramiento del usuario/cliente	Totalmente involucrado	Mediante el propietario del producto	Mediante lanzamientos frecuentes	Mediante informes	Mediante lanzamientos frecuentes

De acuerdo a las características expuestas en la Tabla 5, se escogió la metodología XP, por que cuenta con el enfoque iterativo, grupos de equipos pequeños, se adapta a los cambios de requerimientos y su documentación es necesaria para las etapas del Trabajo de Titulación.

4.9. Programación Extrema

Kent Beck citado en [25] define a la Programación Extrema (XP) como un método de desarrollo de software que sigue los postulados del Manifiesto Ágil, anteponiendo la adaptabilidad en lugar de seguir un plan.

La metodología XP se enfoca en la prueba y error de desarrollar productos de software funcionales, permitiendo a los clientes participar activamente en todo el proceso para el éxito de la metodología, promoviendo el trabajo en equipo y creando un buen ambiente [25].

4.9.1. Prácticas de la programación extrema

La XP está orientada al desarrollo del software cuando los requerimientos cambian rápidamente, por ello la importancia de las pruebas, participación activa del cliente, la simplicidad y la refactorización del código [3]. Las prácticas de la XP se presentan en la Tabla 6 que fueron tomadas de [26].

Tabla 6.
Prácticas de la metodología XP [26]

Principio o práctica	Descripción
Planeación incremental	Los requerimientos se registran en las historias de usuario y los desarrolladores dividen estas historias en “tareas” de desarrollo.
Liberaciones pequeñas	Al principio, se desarrollan funcionalidades mínimas que agregan valor a la empresa. Las versiones del sistema se publican periódicamente y, con el tiempo, se agregan funciones a la primera versión.
Diseño simple	Se hace suficiente diseño solo para cumplir con los requisitos actuales.
Desarrollo de la primera prueba	Las pruebas de unidades automatizadas se utilizan cuando se escriben pruebas para una nueva característica, antes de que se implemente la nueva característica.
Refactorización	Se espera que todos los desarrolladores continúen refactorizando su código y encuentren mejoras en el código lo antes posible. Lo anterior hace que el código sea simple y mantenible.
Programación en pares	Los desarrolladores codifican en parejas, cada uno revisando el trabajo del otro.
Propiedad colectiva	Los desarrolladores codifican en parejas en todas las áreas del sistema, cada uno responsable de todo el código.
Integración continua	Una vez completada la tarea, se integrará en todo el sistema. Después de dicha integración, todas las pruebas unitarias del sistema pasan.
Ritmo sustentable	Grandes cantidades de tiempo se consideran inaceptables porque el tiempo libre reduce la calidad y la productividad del código.
Cliente en sitio	El cliente es miembro del equipo de desarrollo y es responsable de enviar los requerimientos de su sistema al equipo para su implementación.

4.9.2. Proceso de la programación extrema

La programación extrema contiene cuatro actividades estructuradas: planificación, diseño, codificación y prueba [3]. En la Figura 4 que fue tomada de [3], muestra cada actividad de XP, destacando las tareas clave de cada actividad.

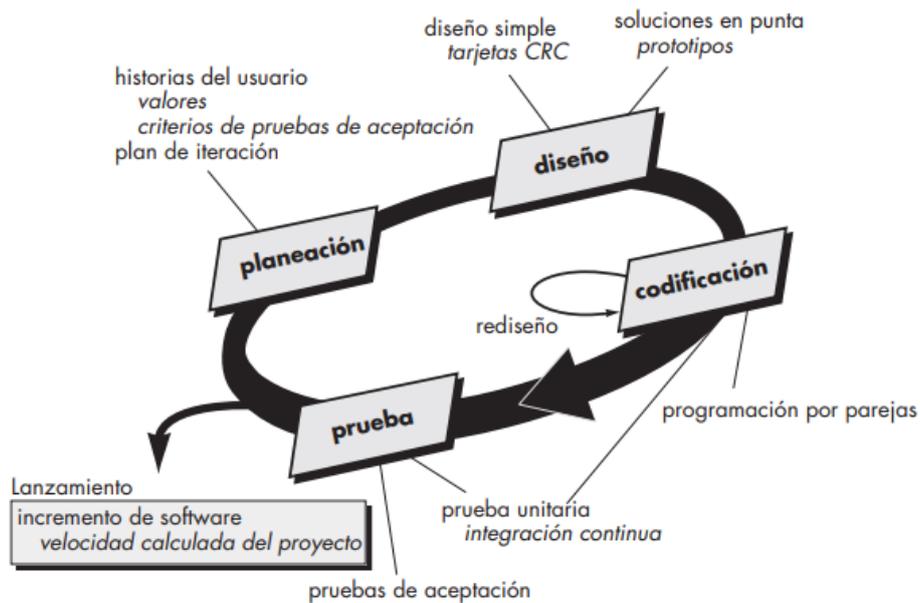


Figura 4. El proceso de la programación extrema [3].

A continuación, se detalla las actividades claves de la XP:

- **Planificación:** Esta actividad comienza escuchando al cliente, comprendiendo el contexto del negocio y logrando así obtener los requisitos a través de User Stories. El equipo de desarrollo asigna tareas, estimaciones de carga de trabajo, recursos necesarios para ejecutar, planes de entrega, iteraciones, emparejamiento y reuniones diarias [23].
- **Diseño:** Esta es la etapa en la que el equipo del proyecto evalúa las Historia de Usuario y la divide en tareas. Las tareas representan una característica diferente del software, y se pueden diseñar pruebas unitarias para verificar cada tarea. Estas tareas están representadas por una tarjeta CRC (Clase-Responsabilidad-Colaborador) [3].
- **Codificación:** Es la programación de pares, las pruebas unitarias e integración. En esta etapa la disponibilidad del cliente es alta, para resolver las dudas que puedan surgir durante la jornada laboral.
- **Prueba:** Se realiza pruebas unitarias por cada una de las tareas identificadas en las historias de usuario. En estas pruebas se evalúa cada uno de los métodos y clases [3].

4.10. Comparativa de base de datos relacionales vs NoSQL

Se analizó los estudios [27] y [28] para realizar una comparación entre las bases de datos SQL y NoSQL. La Tabla 7 presenta las características extraídas del análisis de estas dos bases de datos.

Tabla 7.
Comparación de las bases de datos Relacionales y NoSQL

Base de Datos Relacional	NoSQL
Las bases de datos relacionales son escalables verticalmente	Las bases de datos NoSQL son escalables horizontalmente
Sigue la propiedad ACID	Sigue la propiedad BASE
Disminuye el rendimiento cuando se trata de una gran cantidad de datos semiestructurados y no estructurados (Big Data)	Rendimiento, escalabilidad y flexibilidad necesarios para el Big Data que proporcionan las bases de datos NoSQL
Escalabilidad limitada	Sin límite de escalabilidad
Adecuado para aplicaciones financieras, es decir, las transacciones bancarias requieren la propiedad ACID	Adecuado para sitios de medios sociales y grandes aplicaciones web como la actualización del estado de Facebook o los comentarios de Tweets
La gestión del cambio es difícil debido a la rigidez del esquema	Libre de esquemas y la gestión de cambios es fácil
Se utiliza el SQL estándar para consultar los datos	No hay estándares para la consulta de datos
Garantiza la seguridad de las transacciones	Algunos problemas de seguridad
Problema de replicación de datos	Soporta la replicación automática de datos

De acuerdo a las características presentadas en la Tabla 7, se determinó que la mejor opción de base de datos para la presente solución informática es la base de datos NoSQL debido a su escalabilidad horizontal, puede procesar una gran cantidad de datos de manera eficiente, sigue el enfoque BASE (Basically Available, Soft state, Eventually consistent) que permiten administrar varias formas de datos masivos facilitando el manejo de la Big Data.

4.11. Comparativa entre bases de datos NoSQL

Para seleccionar la base de datos NoSQL se revisó las investigaciones [29] y [30]. La abstracción de las características se presenta en la Tabla 8. Se eligió la base de datos MongoDB porque soporta el escalado de los datos entre múltiples nodos, la carga se distribuye de forma equitativa entre los nodos y, si se interrumpe el equilibrio, la carga se redistribuye automáticamente de forma equitativa; los datos se pueden representar en formato JSON, ya que tienen un alto rendimiento y flexibilidad.

Tabla 8.
Comparación de diferentes bases de datos NoSQL

Característica	Almacén de Columnas Anchas (Cassandra)	Almacén de Documentos (MongoDB)	Almacén de Valores Clave (Redis)
Modelo de base de datos	Almacén de columnas anchas	Almacén de documentos	Almacén de claves y valores
Tabla	Familia de columnas	Colección	Conjunto Hash, Lista, Conjunto, Conjunto ordenado y Cadena
Valor	Filas	Documentos	Par de valores clave
Leer operaciones	Lento	Rápido	Rápido

Escribir operaciones	Rápido	Rápido	Rápido
Borrar operaciones	Rápido	Rápido	Rápido
Lenguaje	Java	C++	C
Scripts del lado del servidor	No	JavaScript	Lua
Disparadores	Si	No	No
Clave foránea	No	Normalmente, no se utiliza, sin embargo, la operación con DBRef se puede hacer	No

4.12. Node.js

Según [31], define a Node.js como un entorno de ejecución del lenguaje de programación JavaScript. Es altamente utilizada para crear servicios web de forma rápida. La naturaleza de Node.js con su modelo de E/S no bloqueante, se adapta bien a la comunicación entre el cliente y el servidor en una aplicación web [32]. El código JavaScript en el lado del navegador está sujeto a varias restricciones de seguridad en tiempo de ejecución y el funcionamiento del sistema del cliente está limitado. Node.js utiliza programación asíncrona y basada en eventos, y está diseñado para servicios de red [33].

4.12.1. Express JS

Express JS es un middleware que conecta el cliente y el servidor a través de rutas. Su funcionalidad es vital en una aplicación web MVC (modelo-vista-controlador), ya que el middleware separa el modelo (base de datos) de la vista (interfaz web) [32]. Proporciona características adicionales a Node.js para crear APIs sobre HTTP, añadiendo enrutamiento y representación de vistas dinámicas HTML.

4.13. API

Interfaz de Programación de Aplicaciones o por sus siglas API, es una interfaz basada en protocolos que posibilitan la comunicación entre aplicaciones o servicios, independientemente del lenguaje de programación o tecnología en la que están construidas; la interacción de las APIs es esencialmente una solicitud y respuesta entre el cliente (interfaz de usuario) y el servidor [34]. Las APIs brindan flexibilidad al desarrollar software, ya que simplifican el diseño arquitectónico, se adaptan a los cambios en el proyecto permitiendo ahorrar tiempo y dinero.

Se puede clasificar a las APIs en función a su distribución [35]:

- Iterativas o privadas: son creadas y utilizadas dentro de una misma compañía, es decir, los servicios de la API no se exponen al público.
- Externas o públicas: los servicios se ponen a disposición de cualquier persona que las quiera utilizar.

- Partner: está en el intermedio de una API privada y pública, ya que, son utilizadas por diferentes compañías.

La clasificación de las APIs también puede ser clasificada en función a qué están orientadas [35]:

- APIs de servicios Web: utilizan los servicios JSON-RPC, SOAP, XML-RPC o REST para proporcionar su servicio por medio de una dirección web.
- APIs basadas en librerías: estas APIs tiene su importancia para los programadores en determinado lenguaje de programación, porque al usar librerías facilita la codificación.
- APIs de sistemas operativos: permiten conocer cómo están estructuradas las funcionalidades o servicios de un determinado Sistema Operativo.

En el proyecto desarrollado se utilizó las APIs de Servicio Web, por ende, a continuación, se habla más a profundidad de estos.

4.13.1. APIs de Servicio Web

Los servicios web son un tipo de API que a través de una URL proporciona el acceso a un servicio en la red. Por lo general se utiliza el protocolo de transporte HTTP en el intercambio de datos entre los diferentes sistemas, y para ello, la API debe de seguir una arquitectura de software, a fin de facilitar información comprensible para otras aplicaciones [35].

Los servicios web válidos para este tipo de APIs, son [35]:

- **SOAP** (Simple Object Access Protocol): protocolo de comunicación para transmitir datos entre diversos componentes de software, basándose en XML (eXtensible Markup Language).
- **XML-RPC** (Remote Procedure Calls): protocolo de llamada a procedimientos remotos, que utiliza XML y HTTP para la descripción de los datos.
- **JSON-RPC**: similar a XML-RPC, pero utiliza JSON en vez de XML para la representación de los datos.
- **REST** (Representational State Transfer): encargada de representar la transferencia de datos mediante el protocolo HTTP.

Principalmente los servicios webs REST y SOAP son los más utilizados para la creación de APIs. Si bien ambos servicios son muy adecuados para las comunicaciones, hay una pequeña diferencia entre ambos [34]. En la siguiente sección se aclara la diferencia entre estos.

4.13.2. Rest vs XML-RPC/SOAP

Las tecnologías XML-RPC y SOAP proporcionaron la solución de la interconectividad de los sistemas a través de un lenguaje universal (XML), pero tenían varios problemas en

comparación al estándar líder de hoy en día (REST) [36]. Aunque SOAP sigue un protocolo y REST es un estilo arquitectónico para desarrollar servicios web, poseen ciertas diferencias al momento de escoger la tecnología, tal cual se presenta en la Tabla 9.

Tabla 9.
Comparación de XML-RPC/SOAP y servicio REST [36]

XML-RCP/SOAP	REST
Había que crear clientes SOAP específicos para cada lenguaje de programación. aunque el XML era universal, había que codificar un nuevo cliente para analizar el WSDL y entender cómo funcionaba el servicio.	REST es completamente agnóstico a la tecnología y no requiere clientes especiales, sólo un lenguaje de programación capaz de conectarse a través del protocolo elegido (por ejemplo, http, Ftp, etc.).
El cliente necesita saber todo sobre el servicio antes de iniciar la interacción (de ahí el WSDL mencionado anteriormente).	El cliente sólo necesita conocer el punto final raíz principal, y con los hipermedios proporcionados en la respuesta, es posible el autodescubrimiento.
Como el servicio se utilizaba desde el código fuente del cliente y llamaba a una función o método específico desde el código del servidor, el acoplamiento entre esos dos sistemas era demasiado grande. una reescritura del código del servidor probablemente llevaría a una reescritura en el código del cliente.	La interfaz es independiente de la implementación; el código completo del lado del servidor puede ser reescrito y la interfaz del API no tendrá que ser cambiada.

Con base en la Tabla 9, para desarrollar la aplicación web en el objetivo 2, se eligió el estilo arquitectónico REST, debido a que presenta una mejor escalabilidad del proyecto, independientemente del lenguaje de programación escogido.

4.14. REST

Transferencia de Estado Representacional o por sus siglas REST, es una arquitectura de software que representa la transmisión de información a través del protocolo HTTP. Se basa en la arquitectura cliente-servidor, y mediante los métodos: get, post, delete y put, se encarga de tratar los objetos (leer, crear, borrar y actualizar) [35]. En la Figura 5 se presenta el esquema de la arquitectura REST.

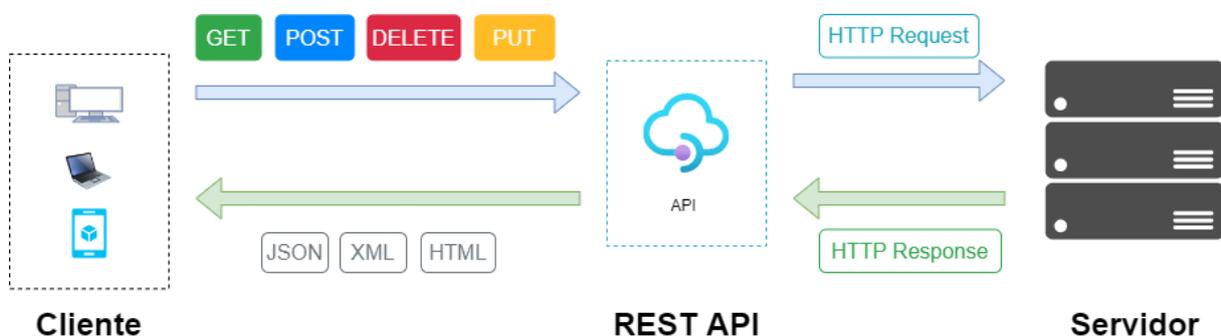


Figura 5. Estructura de la arquitectura REST.

4.14.1. Restricciones REST

Roy Fielding citado en [34] definió seis restricciones para REST, los cuales se describen a continuación:

- **Cliente-servidor:** Es la separación del cliente y servidor para que escalen de forma independiente entre sí. De este modo, se mejora la flexibilidad bajo la interfaz de usuario y la escalabilidad al simplificar los componentes del servidor.
- **Interfaz uniforme:** Desacopla y simplifica la arquitectura para que pueda evolucionar de forma independiente. Se definieron cuatro restricciones: manipulación de recursos mediante representaciones, mensajes autodescriptivos e hipermedia como motor del estado de la aplicación (HATEOAS) e identificación de recursos.
- **Sistema de capas:** En esta arquitectura se compone de capas jerárquicas para que cada uno de los componentes no pueda ver más allá de la capa inmediata hacia la que están conectados. Se aplica una restricción por capas para encerrar los servicios heredados. La desventaja de este sistema es que aumenta la sobrecarga y el retraso en el procesamiento de datos.
- **Caché:** Esta restricción define que la respuesta se declare como almacenable en caché (o no almacenable en caché) explícitamente (o implícitamente). La respuesta almacenable en caché requiere que el cliente reutilice la respuesta. El almacenamiento en caché mejora el rendimiento en el lado del cliente y la escalabilidad en el lado del servidor.
- **Sin estado (Stateless):** Cualquier petición del cliente no se guardará. No se hará una sesión ni un historial de las peticiones del cliente. No se requiere la gestión de recursos. Esta restricción aporta las propiedades de escalabilidad, visibilidad y fiabilidad.
- **Código en demanda:** Funcionalidad que se añade bajo REST a la descarga y ejecución de código. Permite al cliente reducir las funcionalidades que se necesitan para ser pre implementadas. Esta restricción mejora la extensibilidad del sistema.

4.14.2. Métodos en REST

Para las solicitudes y respuestas, REST utiliza HTTP y mediante los métodos: post, get, put y delete, proporciona las acciones CRUD (crear, recuperar, actualizar y eliminar). A continuación se habla de cada método [36]:

- **GET:** Acceder a un recurso en modo de solo lectura.
- **POST:** Normalmente se utiliza para enviar un nuevo recurso al servidor (acción de creación).

- **PUT:** Normalmente se utiliza para actualizar un recurso determinado (acción de actualización).
- **DELETE:** Sirve para eliminar un recurso.
- **HEAD:** No forma parte de las acciones de CRUD, pero el verbo se utiliza para preguntar si un determinado recurso existe sin devolver ninguna de sus representaciones.
- **OPTIONS:** No forma parte de las acciones de CRUD, pero se utiliza para recuperar una lista de verbos disponibles en un recurso determinado (es decir, ¿qué puede hacer el cliente con un recurso específico?).

Los métodos descritos anteriormente pueden ser seguros e idempotentes. En la Tabla 10 se presentan las características de estos métodos, que fue extraída de [34]. Los métodos idempotentes son aquellos que pueden ser llamados tantas veces sin ningún efecto sobre los datos almacenados, mientras que los métodos seguros no actualizan los recursos y son solo de lectura.

Tabla 10.
Características de los métodos HTTP [34]

Método HTTP	Idempotentes	Seguro
GET	Si	Si
POS	No	No
PUT	Si	No
DELETE	Si	No

4.14.3. Lenguaje de transporte

Durante años, el estándar para intercambiar la información entre el front-end y back-end fue XML, ya que proporcionaba una forma agnóstica de expresar datos que podían enviarse fácilmente. Hoy en día, hay un nuevo estándar que ha ganado popularidad entre los desarrolladores, dicho estándar es JSON (ver Figura 6) [36].

A continuación, se detalla las principales ventajas que ofrece JSON [36]:

- Es ligero.
- Es legible para los humanos.
- Admite diferentes tipos de datos.

Como el protocolo en el que se basa REST es HTTP, los desarrolladores pueden utilizar un mecanismo llamado negociación de contenido para permitir que los clientes especifiquen cuál de los formatos soportados quieren recibir, permitiendo una mayor flexibilidad en la API [36].

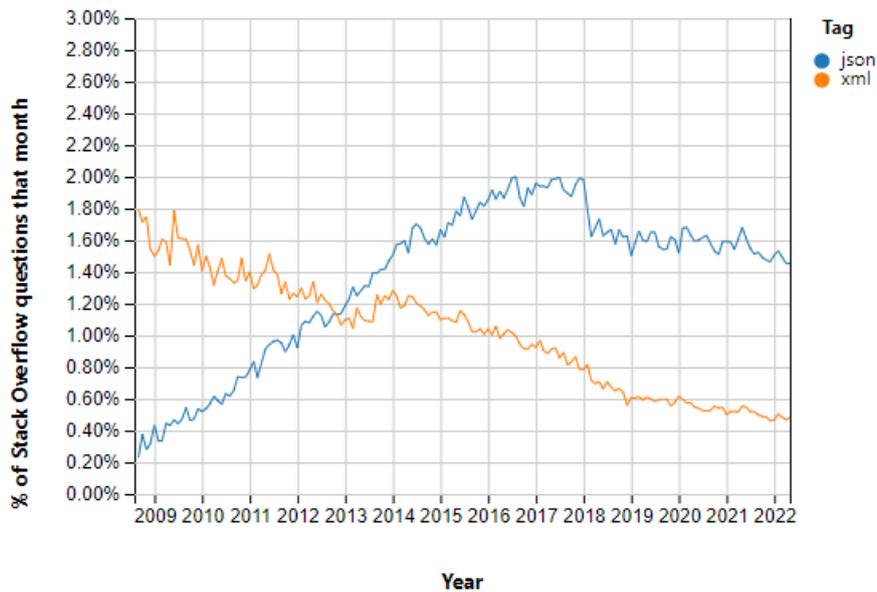


Figura 6. Tendencia de búsqueda en Stack Overflow de JSON frente XML [37].

La aplicación web desarrollada en el segundo objetivo, utiliza el formato JSON para intercambiar los datos entre la interfaz de usuario y la API, por las ventajas mencionadas anteriormente.

4.15. Autenticación por token

Se han introducido varios métodos de autenticación a lo largo de los años, cada uno de los cuales intenta brindar un servicio más seguro que el otro; primero se sugiere la autenticación basada en contraseña, luego la autenticación biométrica basada en sesión y finalmente la autenticación basada en token [38].

Los tokens de autenticación no tienen estado, son seguros, escalables y están diseñados para facilitar a los usuarios sin sobrecargar el servidor [39], por estas características se elige esta técnica cuando se crea una API web, servicios REST o un sitio web.

El mecanismo tradicional de una aplicación guarda el ID de sesión en el servidor que contiene la identidad del usuario a partir de la cookie de sesión. La técnica basada en tokens, aplica un enfoque moderno diseñado para resolver los problemas de ID de sesión; al utilizar tokens como sustituto de los ID de sesión se pueden reducir la carga del servidor, gestionar los permisos, y como un nuevo y mejor mecanismo para soportar arquitecturas distribuidas o multiplataforma y basadas en la nube [39].

En la Figura 7 se ilustra el proceso de autenticación basado en cookies y tokens. Al usar tokens, esta se envía al usuario y se guarda en una cookie o en el almacenamiento local del navegador; al ejecutar una solicitud posterior se recupera el token y se manda de nuevo al servidor en la cabecera HTTP.

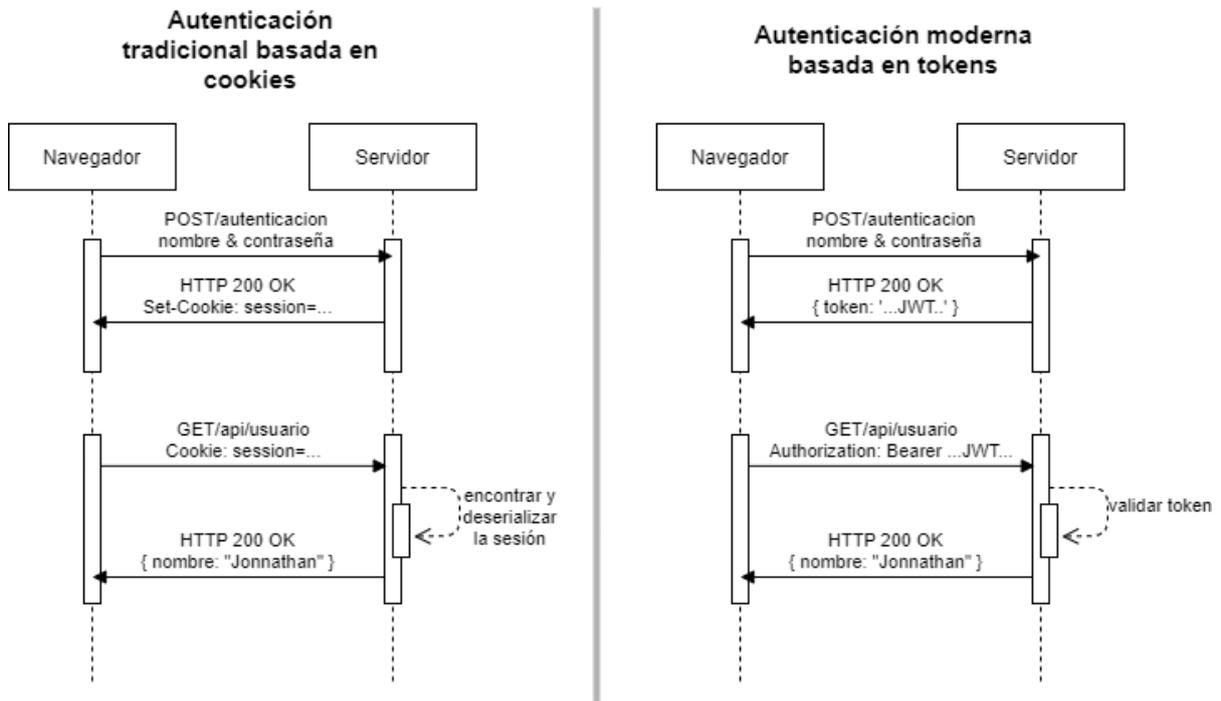


Figura 7. Proceso de autenticación por cookies y tokens.

4.16. JSON Web Token

JSON Web Token o por sus siglas JWT, es un estándar RFC 7519 en donde define la forma de transmitir datos de manera segura a través de un objeto JSON [38]. JWT funciona de la misma manera que una contraseña, cuando el usuario se conecta con éxito, el servidor le dará un token y este se puede guardar en las cookies o el almacenamiento local del navegador. El token se utiliza para acceder a ciertas páginas, el usuario enviará el token de vuelta como prueba de que el usuario ha iniciado la sesión con éxito. La Figura 8 presenta el proceso de autenticación con JWT, obtenida de [40].

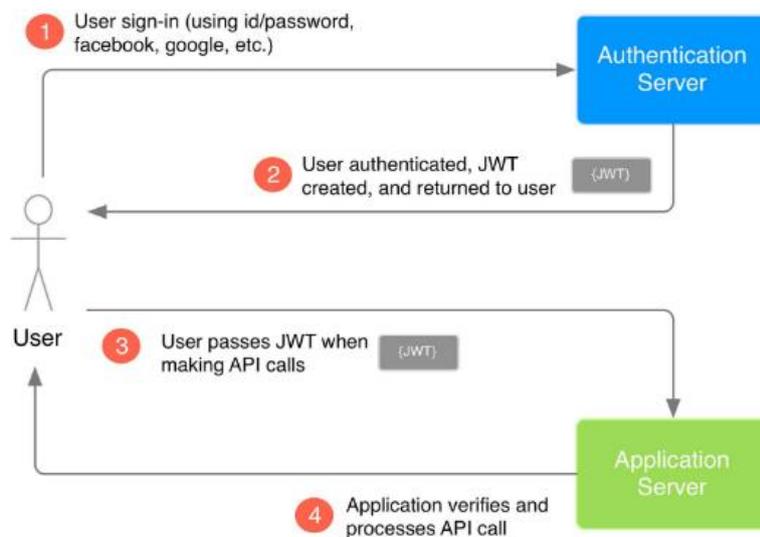


Figura 8. Proceso de autenticación con JWT [40].

JWT contiene tres partes separadas por un punto: cabecera (Header), carga útil (Payload) y las firmas (Signature). Cuando las tres partes se conectan con el punto, se crea un token que se puede enviar de vuelta en la cabecera de autenticación HTTP con la palabra clave Bearer [38], tal cual se presenta en la Figura 9.

Authorization: Bearer i72mkx267.lusowng396.hdon7vsj

Figura 9. Ejemplo de JWT [38].

La Figura 10 indica la estructura de JWT. En la sección Header consta de dos partes: el algoritmo utilizado y el tipo de token, luego se codifica en Base64. La segunda parte, Payload, contiene el claim, que suele ser datos adicionales del usuario, de la misma manera se codifica en Base64. La tercera parte Signature, contiene la cabecera codificada, la carga útil codificada, el secreto, el algoritmo que se ha definido en la cabecera y el signo.

```

Header: Algorithms and Token Types
{
  "alg": "HS256",
  "typ": "JWT"
}
Payload: Data
{
  "iss": "#appbackend",
  "user": "alam",
  "pass": "rahasia"
}
Signature: results from Hash
{
  "Base64-encoded(Header.Payload)" + "key" + "Algorithm"
}

```

Figura 10. Estructura JWT [39].

4.17. Trabajos relacionados

Es muy importante conocer algunos trabajos desarrollados en el tema de creación de aplicaciones web para la enseñanza de lengua de señas y micro-learning. Por tal motivo, se ha realizado una revisión de algunos trabajos en Ecuador y de otros países. Los resultados correspondientes se muestran en la Tabla 11.

Tabla 11.
Trabajos relacionados

Titulo	Descripción	Referencia
Aplicación web para la enseñanza del lenguaje de señas, módulo básico para niños del segundo año de educación básica de la unidad educativa de sordos	La aplicación web está orientada en la enseñanza de la lengua de señas a nivel básico, mediante la presentación de videos e imágenes con su respectivo significado. En cuanto a la evaluación del aprendizaje recibido por la niñez participante en el estudio estuvo basado en un examen de opción múltiple para selección de la respuesta correcta.	[41]

del gobierno provincial de Imbabura		
Desarrollo de un portal web para la enseñanza-aprendizaje del lenguaje de señas en la ciudad de Guayaquil	El portal web está enfocado en brindar conocimientos básicos del lenguaje de señas con la ayuda de videos dirigidos al público en general y personas con discapacidad auditiva. Los usuarios al final de los videos impartidos no son evaluados por el portal web.	[42]
Learning Application for Malaysian Sign Language: Content Design, User Interface and Usability	La aplicación web creada en Malasia contiene elementos de nivel básico en lengua de señas, comenzando desde las animaciones y audios, luego se encuentra el módulo de ejercicios de palabras donde el usuario da su respuesta a las preguntas generadas, a partir de la selección de 240 palabras aleatorias y finalmente esta aplicación brinda ejercicios de ortografía con la propuesta de 160 palabras aleatorias que deben ser escogidas por el usuario para rellenar las preguntas.	[43]
Web-Based Application for Learning Malaysian Sign Language	Es una aplicación web creativa que representa los signos del lenguaje de señas en animaciones 2D e imágenes con la facilidad de ser convertidas en GIF animado, haciendo posible la compatibilidad en plataformas móviles, de escritorio y navegadores web.	[44]
Design of e-learning courseware for hearing impaired (HI) students	Aplicación basada en e-learning donde el contenido de enseñanza de nivel básico está encapsulado en tres módulos. Se utilizan las imágenes GIF debido al beneficio en el procesamiento cognitivo de los estudiantes en el aprendizaje del lenguaje de señas y otra forma de representar los signos básicos de la lengua de señas son los videos. Además, contiene módulos para realizar ejercicios personalizados, un ejemplo a citar es el arrastre de la palabra correcta con la imagen.	[45]

5. Metodología

El presente Trabajo de Titulación (TT) logró obtener como resultado el diseño de la aplicación web utilizando el patrón de diseño Factory, el cual surgió basándose en el problema de investigación y la pregunta: **¿Diseñar una aplicación web basada en el patrón de diseño Factory facilitará la creación de una app para el aprendizaje del LSE?**

Se definió el proceso del proyecto, métodos, técnicas, estándares, metodologías de desarrollo de software y materiales, las cuales se describen a continuación:

5.1. Proceso

Para cumplir con el objetivo principal del TT, se siguió el siguiente proceso:

- 1) Proceso de Ingeniería de Requerimientos
 - a) Se determinó los actores que intervienen en la aplicación web (ver **la sección Resultados, punto 6.1.2 del Objetivo 1**).
 - b) Se realizó un estudio de documentación sobre trabajos similares para obtener recomendaciones o materiales visuales dentro de la aplicación web (ver **la sección Resultados, punto 6.1.3 del Objetivo 1**).
 - c) Se establecieron los requerimientos de la aplicación web conforme al estándar IEEE 830 (ver **la sección Resultados, punto 6.1.4 del Objetivo 1**).
 - d) Se validó los requerimientos mediante la técnica de prototipado (ver **la sección Resultados, punto 6.1.4.5 del Objetivo 1**).
- 2) Desarrollo del módulo de software
 - a) Se modeló el diseño arquitectónico basándose en los requerimientos de la aplicación web (ver **la sección Resultados, punto 6.2.2 del Objetivo 2**).
 - b) Se diseñó el módulo de software mediante diagramas de clases y paquetes utilizando notación UML (ver **la sección Resultados, punto 6.2.2.1 y 6.2.2.2 del Objetivo 2**).
 - c) Se codificó la aplicación web utilizando la metodología Programación Extrema (XP) (ver **la sección Resultados, punto 6.2.3 del Objetivo 2**).
 - d) Se realizaron pruebas unitarias de la aplicación web, correspondiente a la fase 4 de la metodología de desarrollo XP (ver **la sección Resultados, punto 6.2.4 del Objetivo 2**).
- 3) Evaluación del diseño de software
 - a) Se generaron pruebas de integración de cliente con el servidor (ver **la sección Resultados, punto 6.3.1 del Objetivo 3**).
 - b) Se realizó el despliegue de la aplicación web en un entorno simulado (ver **la sección Resultados, punto 6.3.2 del Objetivo 3**).

- c) Se generaron pruebas funcionales de la aplicación web (ver **la sección Resultados, punto 6.3.3 del Objetivo 3**).
- d) Se realizó pruebas de aceptación del diseño de la aplicación web (ver **la sección Resultados, punto 6.3.4 del Objetivo 3**).

5.2. Métodos

5.2.1. Analítico

El presente método consiste en separar un todo en sus elementos para observar sus causas y los efectos [46] la cual fue utilizada para descomponer el TT en diversas etapas, acompañadas de actividades para realizar cada objetivo específico (ver la sección de Metodología, apartado 5.1). Cada una de estas etapas y actividades se desarrollaron en la sección de Resultados.

5.2.2. Científico

Se toman las etapas del método científico para desarrollar el Trabajo de Titulación: Definición del problema, planteamiento de la hipótesis (ver sección Metodología), recolección y análisis de datos (ver la sección de Resultados), contrastación de datos con la hipótesis (ver la sección de Discusión, apartado 7.1), y por último, elaboración de las conclusiones (ver la sección de Conclusiones) [47].

5.2.3. Estudios de Caso

Este método permitió identificar en trabajos relacionados los materiales visuales y recomendaciones para presentar la lengua de señas [48] (ver sección Trabajos relacionados y la sección de Resultados, punto 6.1.3 del Objetivo 1).

5.2.4. Experimentales Controlados

5.2.4.1. Simulación

El presente método hace la ejecución del producto con datos artificiales [49], permitiendo evaluar la aplicación web en un entorno simulado, cumpliendo el objetivo tres del presente Trabajo de Titulación (ver la sección de Resultados, punto 6.3.2 del Objetivo 3).

5.2.5. Encuestas

Este método permitió recolectar datos sobre las pruebas de aceptación de la aplicación web [48], dentro del proceso de evaluación del diseño de software (ver Anexo 12. Plan de Pruebas de Aceptación).

5.3. Técnicas

5.3.1. Entrevista

Es una de las técnicas utilizadas para la elicitación de requerimientos, se lo empleó en el proceso de la ingeniería de requerimientos de la aplicación desarrollada, la cual fue aplicada

a un intérprete de Lengua de Señas Ecuatoriana (LSEC) y al presidente de la Asociación de Personas Sordas “Virgen del Cisne”, con la finalidad de obtener información sobre la enseñanza de la LSEC a las personas ecuatorianas y requerimientos de la aplicación web (ver la sección Resultados el punto 6.1.4 del Objetivo 1).

5.3.2. Estudio de documentación

Se empleó el estudio de documentación como técnica en el análisis de los trabajos relacionados a fin de extraer materiales y métodos para enseñar lengua de señas mediante el micro-learning plasmando los resultados en wireframes y elaborar un prototipo de la aplicación web (ver la sección Resultados el punto 6.1.3 del Objetivo 1).

5.3.3. Reuniones

Se empleó la técnica de las reuniones con el presidente de la Asociación de Personas Sordas “Virgen del Cisne”, el intérprete de la lengua de señas ecuatoriana, el investigador del TT y el tutor del TT para validar el prototipo de la aplicación (ver la sección Resultados el punto 6.1.4.5 del Objetivo 1), los requerimientos (ver la sección Resultados el punto 6.1.4 del Objetivo 1), avances de codificación (ver la sección Resultados el punto 6.2.1 del Objetivo 2) y los temas educativos (ver la sección Resultados el punto 6.3.2 del Objetivo 3).

5.4. Estándares

5.4.1. IEEE 830

Se utilizó el estándar IEEE 830 para la especificación de los requerimientos de la aplicación web en el primer objetivo específico del Trabajo de Titulación (ver la sección Resultados el punto 6.1.4 del Objetivo 1).

5.5. Metodologías de desarrollo de software

La codificación de la aplicación web se realizó bajo la metodología de desarrollo Programación Extrema (XP), cumpliendo a cabalidad el objetivo 2 (ver la sección Resultados el punto 6.2). Para más información sobre la metodología, véase el punto 4.9 del Marco teórico.

5.6. Participantes

Las personas que estuvieron inmersas en el desarrollo del TT se pueden observar en la Tabla 12.

Tabla 12.
Recurso Humano

Nombre	Descripción
Jonnathan Damián Espinoza Erráez	Estudiante a cargo de la ejecución del TT.
Ing. Edison Leonardo Coronel Romero, Mg. Sc.	Docente y director del TT.

Luis Alberto Quichimbo	Intérprete de la Lengua de Señas Ecuatoriana, colaborando en el primer y tercer objetivo por medio de su asesoramiento y entrevistas en donde se obtuvieron los requerimientos de la aplicación web.
Darwin Alfredo Jumbo Quichimbo	Presidente de la Asociación de Personas Sordas “Virgen del Cisne”, colaborando en el primer y tercer objetivo para validar la aceptación de la aplicación web por medio de su asesoramiento.

6. Resultados

Para el desarrollo del TT se planteó un objetivo general y para su cumplimiento se establecieron tres objetivos específicos: Objetivo I, se realiza la especificación de requerimientos a la solución informática. En el Objetivo II, se desarrolla la aplicación web basándose en los requerimientos obtenidos en el objetivo I, dicho desarrollo se sigue la metodología XP; y finalmente, en el Objetivo III se realizan las pruebas de aceptación a la aplicación web.

6.1. Objetivo 1

Realizar la ingeniería de requerimientos para la aplicación web de Micro-Learning dedicada al aprendizaje del Lenguaje de Señas Ecuatoriano.

Para realizar el objetivo 1, se elaboró la especificación de requerimientos en donde se obtuvieron las historias de usuario que están dentro de la primera fase de la metodología XP.

Los requerimientos se obtuvieron por medio de entrevistas y con el estudio de documentación se elaboró el diseño de un prototipo de la aplicación web. A continuación, se detallan los resultados obtenidos.

6.1.1. Definir el grupo objetivo que utilizará la aplicación web

El grupo objetivo que va dirigido la aplicación es a personas sin discapacidad a partir de los 18 años hasta los 30 años de edad, ya que la aplicación está enfocada a motivar a las personas interesadas aprender lengua de señas ecuatoriana (LSEC) de forma básica y aumentar el número de hablantes de LSEC, por ello la investigación se enfoca solo en el diseño de la plataforma para subir contenido de micro-learning de LSEC.

6.1.2. Establecer los actores de la aplicación web

La aplicación web tiene los siguientes actores:

- **Usuario final:** El usuario que requiere aprender lengua de señas ecuatoriana, persona sin discapacidad entre 18 y 30 años.
- **Usuario administrador:** El usuario que gestionará el contenido educativo de la aplicación.

6.1.3. Estudio de documentación

Las cápsulas de aprendizaje del micro-learning posee tres fases: aprender, practicar y evaluar (ver **la sección del Marco teórico, punto 4.4**). En la fase de aprender, para elaborar el contenido se puede utilizar diferentes herramientas multimedia, sin embargo, al enseñar Lengua de Señas (LS) se debe de elegir el material adecuado a fin de transmitir el conocimiento deseado. Se estudiaron los trabajos [41], [43], [44], [45], [50], [51] y [52] con el

propósito de obtener recomendaciones de cómo diseñar una aplicación web dirigida a enseñar lengua de señas. Cada autor utiliza diferentes herramientas, las más comunes para representar las señas son: videos, animación 2D, imágenes estáticas, GIF, audio y texto. Para enseñar LS, los autores llevan a cabo una enseñanza visual con elementos animados, ya que permite a los estudiantes realizar más procesamiento cognitivo. En la Figura 11 se presenta un wireframe que ilustra una abstracción del diseño con los diferentes materiales que varios de los investigadores utilizan en su solución informática para mostrar una seña.

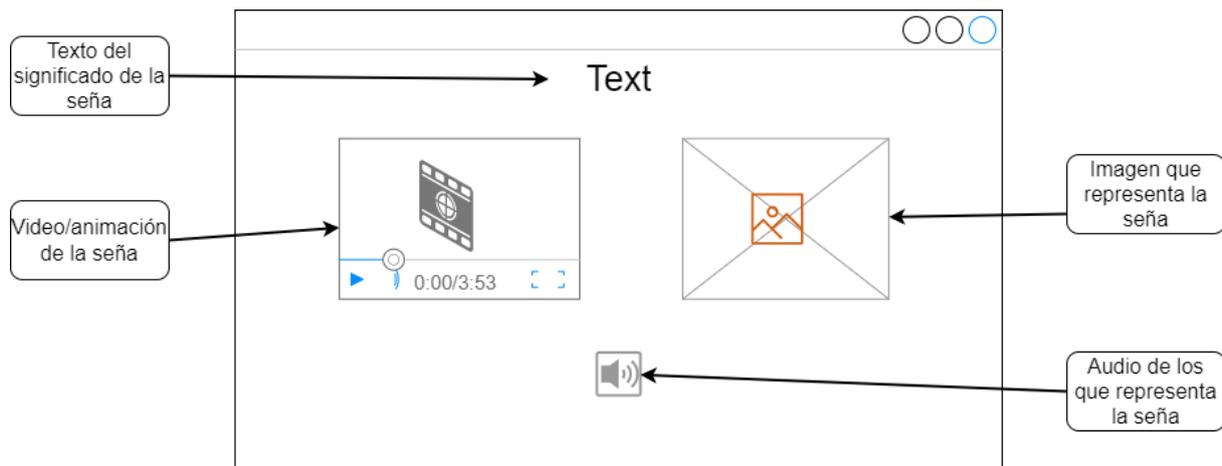


Figura 11. Wireframe para enseñar lengua de señas.

En la fase de practicar y evaluar se hace necesario una metodología que permita al aprendiz poner a prueba sus conocimientos. Se analizó los documentos [53], [54] y [55] en donde los autores investigan métodos para crear pruebas de vocabulario de lengua de señas. Las características de estas, están basadas en cuatro tipos: elegir entre imágenes, elegir entre palabras, elegir correcto o incorrecto y escribir palabras. En cuanto a los estímulos que provocan las respuestas, las pruebas utilizan lo siguiente: signos animados o dibujados, imágenes de objetos, imágenes de objetos asociados a palabras y oraciones escritas.

En la Figura 12 se muestra un wireframe que presenta las características de una prueba para evaluar vocabulario de lengua de señas, que se obtuvo mediante la observación de los diseños de los artículos analizados. Como se puede observar, las pruebas contienen: una pregunta en texto relacionada con lo que se va a resolver, un video o imagen de la seña, un conjunto de respuestas ya sean en texto o imagen.

Las pruebas permiten medir el nivel de aprendizaje del estudiante y poner en práctica sus conocimientos. Como se menciona en [53] son pocas las evaluaciones existentes con referencia a la lengua de señas; esto se debe a que la población no oyente es relativamente pequeña a la oyente, por tal motivo, se puede adaptar las pruebas existentes de un lenguaje de señas a otro o del lenguaje hablado. Por consiguiente, para el desarrollo de la fase de

práctica se hace uso de las tarjetas de aprendizaje, donde el usuario final debe de elegir entre correcto o incorrecto, como se sugiere en [54].

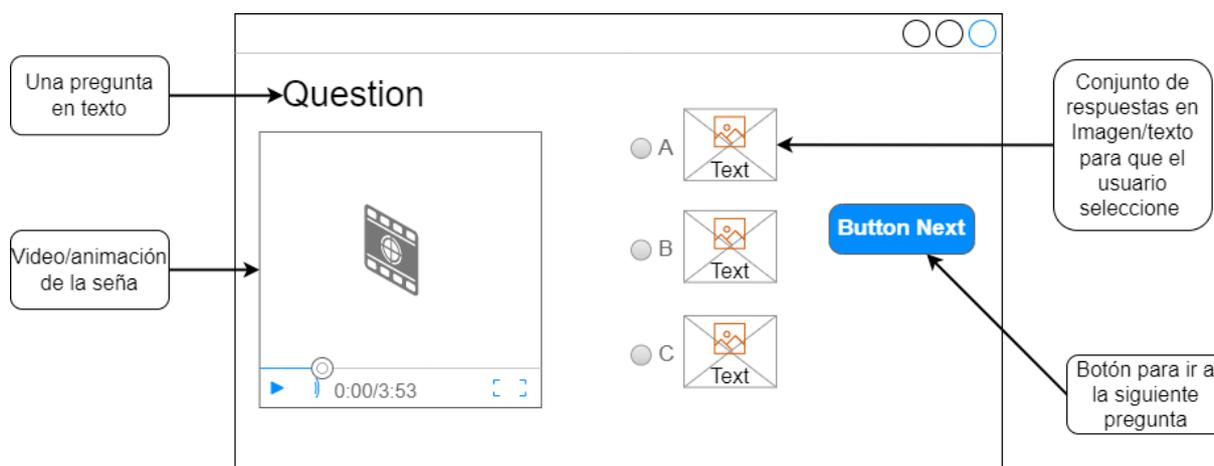


Figura 12. Wireframe para evaluar conocimiento de señas.

6.1.4. Requerimientos de la aplicación web

En el presente apartado se describen los requerimientos funcionales y no funcionales que se establecieron siguiendo el estándar IEEE 830 (véase el **Anexo 2**. Especificación de Requerimientos de Software), mismos que fueron obtenidos por medio del estudio de documentación y de entrevistas realizadas a un intérprete de la lengua de señas ecuatoriana y al presidente de la Asociación de Personas Sordas “Virgen del Cisne” (ver **Anexo 1**. Elicitación de Requerimientos). Así mismo, las historias de usuario, el diagrama de casos de usos y prototipo de la aplicación.

6.1.4.1. Requerimientos funcionales

La Tabla 13 muestra los requerimientos funcionales de la aplicación web para enseñar lengua de señas ecuatoriana mediante el micro-learning. Es importante mencionar que se tomaron en cuenta las características del micro-learning, las recomendaciones que se obtuvieron dentro del estudio de documentación y las sugerencias de los entrevistados.

Tabla 13.
Requerimientos funcionales

Código	Requerimiento	Descripción
RF01	Crear cuenta de usuario	El usuario final podrá crear una cuenta en el sistema, con su nombre, correo electrónico y contraseña.
RF02	Autenticación de usuario	El usuario final y administrador deben de identificarse con el correo electrónico y contraseña para ingresar al sistema.
RF03	Cerrar sesión de usuario	El usuario final y administrador podrá cerrar sesión en el sistema e ingresar con otra cuenta.
RF04	Gestión de módulos	El usuario administrador podrá crear, visualizar, actualizar y borrar módulos. Para crear el módulo deberá de ingresar: número de módulo y nombre.

RF05	Gestión de lecciones	El usuario administrador podrá crear, visualizar, actualizar y borrar lecciones. Para crear una lección deberá de ingresar: número de módulo, nombre de la lección e icono.
RF06	Gestión de tarjetas de aprendizaje	El usuario administrador podrá crear, visualizar, actualizar y borrar tarjetas de aprendizaje. Para crear una tarjeta de aprendizaje deberá de ingresar: nombre de la lección a la que pertenece, pregunta, gif de la seña, respuestas de selección múltiple y la respuesta correcta. En las tarjetas de aprendizaje, se visualizará el gif de la seña que el usuario final deberá de identificar la seña y elegir una respuesta. Las tarjetas de aprendizaje no darán puntaje, son solo un medio para que el estudiante pueda practicar y recordar lo aprendido, pero al completar todas las tarjetas de aprendizaje dentro de una lección, está se marcará como aprendido.
RF07	Visualizar lecciones completadas.	El usuario final podrá ver su avance de lecciones completadas.
RF08	Gestión de cápsulas de aprendizaje	El usuario administrador podrá crear, visualizar, actualizar y borrar cápsulas de aprendizaje. Para crear una cápsula de aprendizaje deberá de ingresar: nombre de la lección a la que pertenece, nombre de la cápsula, gif de la seña e imagen representativa de la seña. La cápsula de aprendizaje se mostrará al ingresar a la lección.

6.1.4.2. Requerimientos no funcionales

En la Tabla 14 se muestran los requerimientos no funcionales de la aplicación web para enseñar lengua de señas ecuatoriana mediante el micro-learning.

Tabla 14.
Requerimientos no funcionales

Código	Requerimiento	Descripción
RNF01	Rendimiento	El sistema debe evitar que las consultas y otros procesos afecten significativamente el rendimiento de la base de datos y el tráfico de la red.
RNF02	Seguridad	El sistema cuenta con autenticación de usuario vía correo electrónico y contraseña. El sistema garantiza la seguridad de la información y los datos que maneja, como correos electrónicos y contraseñas. Controles que permiten al personal autorizado acceder a la información a través de Internet con el fin de visualizar y cargar información en el sistema.
RNF03	Fiabilidad	El sistema debería poder restaurar los datos afectados en caso de un error en el tiempo de respuesta y mostrar un mensaje de error al usuario.
RNF04	Disponibilidad	Los sistemas (back-end y front-end) deben estar disponibles las 24 horas del día, los 7 días de la semana, dada la disponibilidad de los servicios de hospedaje contratados, y deben restaurarse en el menor tiempo posible en caso de falla del software del sistema.
RNF05	Mantenibilidad	Los sistemas (back-end y front-end) necesitan documentación que sea fácil de actualizar para que se pueda realizar el mantenimiento.
RNF06	Portabilidad	El front-end deberá ser compatible con la mayoría de los navegadores de internet.

RNF07	Usabilidad	El sistema debe proporcionar una interfaz de usuario intuitiva y fácil de usar que haga que el proceso sea fácil de entender y ejecutar. También debería funcionar con cualquier navegador web.
-------	------------	---

6.1.4.3. Historias de usuario

La Tabla 15 presenta el resumen de las historias de usuario de la aplicación web para enseñar lengua de señas ecuatoriana mediante el micro-learning. El desarrollo y estructura de cada historia de usuario está en el **Anexo 4**. Descripción de Historias de Usuario.

Tabla 15.
Historias de usuario

Identificador de la historia	Nombre	Rol	Característica / Funcionalidad
HU01	Crear cuenta usuario final	Como usuario final	Quiero crear una cuenta de usuario para poder acceder al contenido de la aplicación web y aprender Lengua de Señas Ecuatoriana
HU02	Iniciar sesión usuario final	Como usuario final	Quiero ingresar a la aplicación web mediante correo electrónico y contraseña para poder acceder al contenido
HU03	Cerrar sesión usuario final	Como usuario final	Quiero cerrar sesión para poder salir de la aplicación web
HU04	Crear módulos	Como usuario administrador	Quiero crear módulos de contenido para poder seccionar el contenido de la aplicación web
HU05	Visualizar módulos	Como usuario administrador	Quiero visualizar los módulos de contenido creados para poder conocer el listado de los módulos
HU06	Actualizar módulos	Como usuario administrador	Quiero editar los módulos de contenido creados para poder actualizar la información de éstas
HU07	Borrar módulos	Como usuario administrador	Quiero eliminar los módulos creados para poder borrar definitivamente de la base de datos
HU08	Crear lección	Como usuario administrador	Quiero crear una nueva lección para poder organizar los temas y agregar los microcontenidos
HU09	Visualizar lección	Como usuario administrador	Quiero visualizar las lecciones creadas para poder conocer el listado de las lecciones
HU10	Actualizar lección	Como usuario administrador	Quiero editar la lección creada para poder actualizar la información de estas
HU11	Borrar lección	Como usuario administrador	Quiero eliminar las lecciones creadas para poder borrar definitivamente de la base de datos
HU12	Crear ejercicios	Como usuario administrador	Quiero crear una tarjeta de aprendizaje para poder agregar en una lección
HU13	Visualizar ejercicio	Como usuario administrador	Quiero visualizar las tarjetas de aprendizaje creadas para poder conocer el listado de las pruebas
HU14	Actualizar ejercicio	Como usuario administrador	Quiero editar la prueba creada para poder actualizar la información de estas.

HU15	Borrar ejercicio	Como usuario administrador	Quiero eliminar las tarjetas de aprendizaje creadas para poder borrar definitivamente de la base de datos.
HU16	Visualizar lecciones completadas	Como usuario final	Quiero ver mis avances en las lecciones para identificar las lecciones estudiadas.
HU17	Crear cápsula de aprendizaje	Como usuario administrador	Quiero crear una nueva cápsula de aprendizaje para poder agregar a una lección.
HU18	Visualizar cápsula de aprendizaje	Como usuario administrador	Quiero visualizar las cápsulas de aprendizaje creadas para poder conocer el listado de estas.
HU19	Actualizar cápsulas de aprendizaje	Como usuario administrador	Quiero editar las cápsulas de aprendizaje creadas para poder actualizar la información de estos.
HU20	Borrar cápsulas de aprendizaje	Como usuario administrador	Quiero eliminar las cápsulas de aprendizaje creadas para poder borrar definitivamente de la base de datos.
HU21	Visualizar cápsulas de aprendizaje en la interfaz de usuario final	Como usuario final	Quiero visualizar los microcontenidos separados en módulos y lecciones para saber el orden en que se debe seguir cada sección y tener un aprendizaje eficaz.
HU22	Realizar ejercicios	Como usuario final	Quiero realizar ejercicios para verificar lo aprendido.

6.1.4.4. Casos de uso

La Figura 13 muestra el diagrama de casos de uso de la aplicación web.

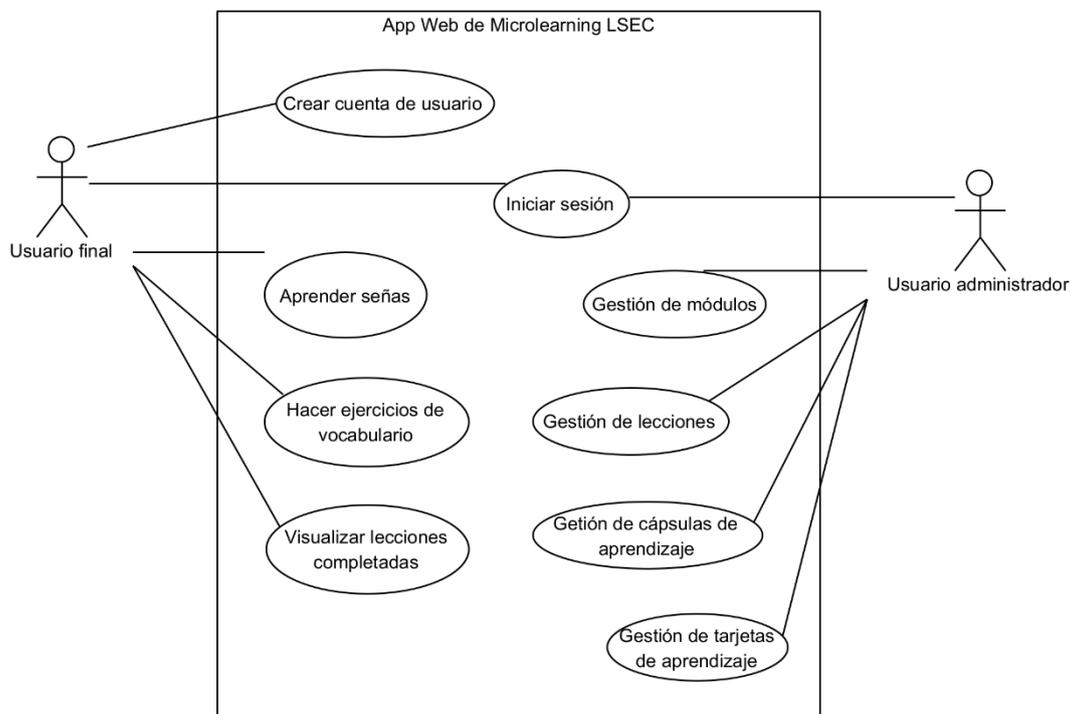


Figura 13. Diagrama de casos de uso de la aplicación desarrollada.

6.1.4.5. Prototipo de la aplicación web

Al analizar las características principales del micro-learning, el estudio de documentación y la entrevista, se diseñó un prototipo de la aplicación web, mismo que sirve para la codificación (ver **Anexo 3**. Prototipo de Interfaz de Usuario). A fin de enseñar lengua de señas, se hace uso de las cápsulas de aprendizaje, en la Figura 14 se presenta el diseño de estos y el material utilizado para enseñar es: tema de aprendizaje, GIF de la seña y una imagen que representa el significado de la seña. Para que el usuario final pueda realizar ejercicios, se elaboró las tarjetas de aprendizaje, cuya característica está basada en elegir la respuesta correcta; la Figura 15 muestra el diseño de esta y el material utilizado es: una pregunta en texto, el GIF de la seña, y las opciones que el usuario deberá de elegir.

Cabe mencionar, que el prototipo también sigue las recomendaciones que establece Fajardo Bravo en su investigación para diseñar páginas web para personas con discapacidad auditiva, con la finalidad de complementar estos consejos con las Pautas de Accesibilidad al Contenido Web (WCAG). Para más información sobre las recomendaciones de Fajardo, véase el **punto 4.3** del Marco teórico.

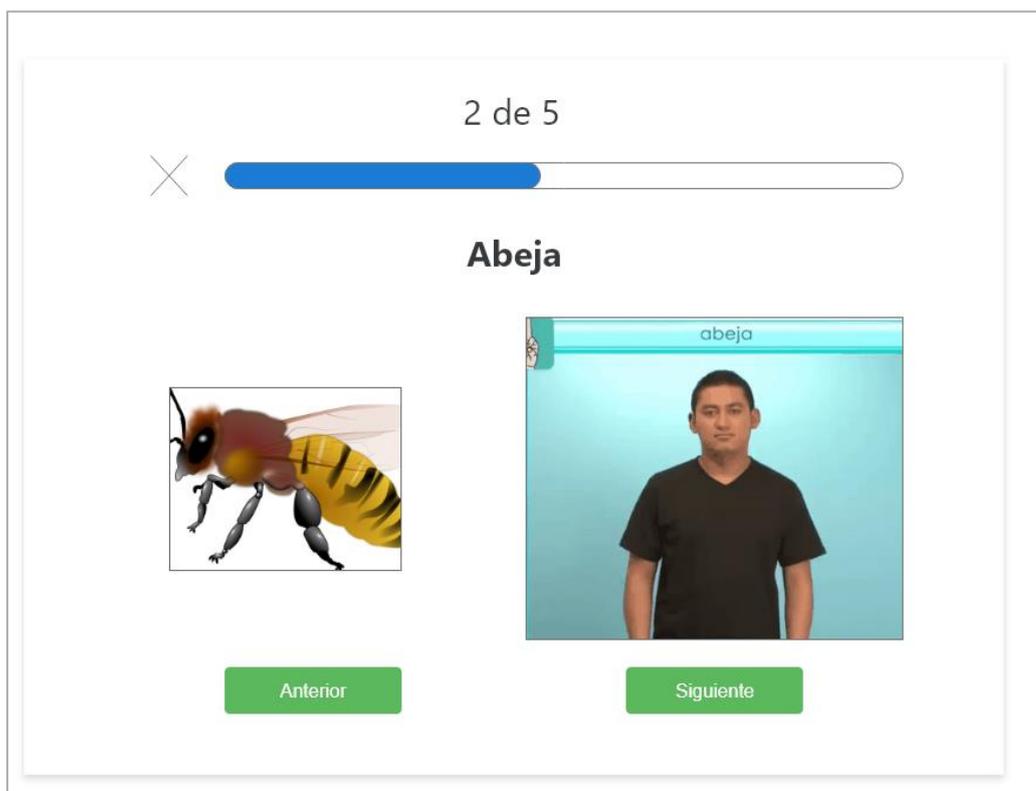


Figura 14. Prototipo de la cápsula de aprendizaje.



Figura 15. Prototipo de la tarjeta de aprendizaje.

6.2. Objetivo 2

Desarrollar la aplicación web de Micro-Learning para el aprendizaje del Lenguaje de Señas Ecuatoriano.

En el presente apartado se describe el proceso de desarrollo de la solución informática utilizando la metodología XP, la cual se seleccionó de acuerdo con sus características en el desarrollo ágil (para más información véase la sección del Marco teórico el **punto 4.7** y **4.8**). Para saber más sobre la metodología, véase el **punto 4.9** del Marco teórico. Las tecnologías utilizadas para la codificación se presentan en la Tabla 16. Para un mayor detalle, observar el proceso de la metodología en el **Anexo 7**. Desarrollo de la Aplicación Web.

Tabla 16.
Tecnologías utilizadas en el desarrollo de la aplicación web

Tecnología	Logo	Empleo
MongoDB		Sistema de base de datos NoSQL, para almacenar y gestionar los modelos de datos.
Express.js		Framework para crear el servidor web y las rutas de la API.
React.js		Librería para crear componentes de interfaz de usuario.
Node.js		Entorno de ejecución del lado del servidor utilizando JavaScript.

6.2.1. Planificación

En la fase de la planificación se definieron las tareas basándose en las historias de usuario. Se plantearon tres iteraciones cuya duración de cumplimiento es de dos semanas cada una. La Tabla 17 presenta la planificación realizada en cada iteración.

Primero se utilizó Miro y por medio de su pizarra virtual se estableció el orden de los requerimientos a desarrollar. Una vez fijado que requerimiento se va a codificar primero, se empleó Jira Software, que es una plataforma que permite gestionar las tareas y llevar un proceso ordenado de las mismas, todo esto basado en metodologías ágiles.

Tabla 17.
Planificación de las iteraciones de la metodología XP

No. Iteración	Clave	Historia de Usuario
1	LSEC-24	Crear cuenta usuario final
	LSEC-25	Iniciar sesión usuario final
	LSEC-26	Cerrar sesión usuario final
	LSEC-7	Crear lección
	LSEC-8	Visualizar lección
	LSEC-9	Actualizar lección
	LSEC-11	Borrar lección
	LSEC-12	Crear módulos
	LSEC-13	Visualizar módulos
	LSEC-14	Actualizar módulos
	LSEC-15	Borrar módulos
2	LSEC-17	Visualizar microcontenido
	LSEC-16	Crear microcontenido
	LSEC-18	Actualizar microcontenido
	LSEC-19	Borrar microcontenido
	LSEC-27	Visualizar cápsulas de aprendizaje en la interfaz de usuario final
3	LSEC-20	Crear ejercicio
	LSEC-21	Visualizar ejercicio
	LSEC-22	Actualizar ejercicio
	LSEC-23	Borrar ejercicio
	LSEC-28	Visualizar lecciones completadas

Por medio de Git y bajo el flujo de trabajo Git Flow, se mantuvo un mayor control y organización de las versiones de la aplicación en el proceso de desarrollo. La Figura 16 muestra los diferentes commits realizados, esto en relación con las iteraciones.

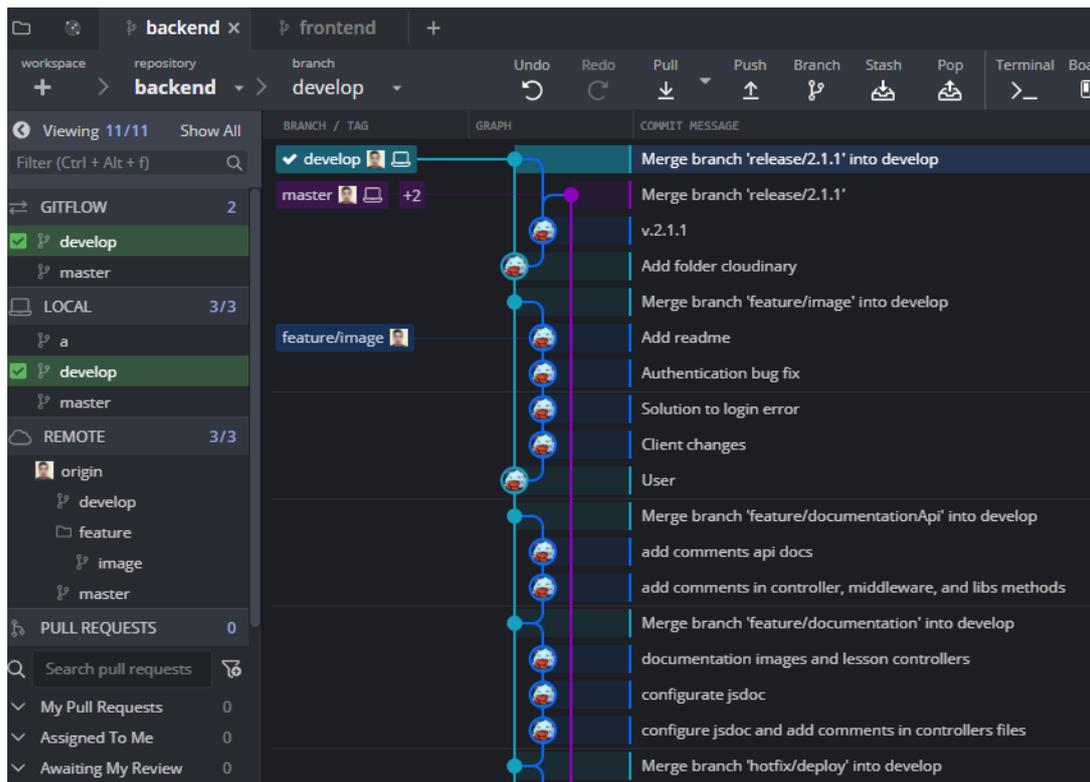


Figura 16. Control de versiones con GitKraken.

6.2.2. Diseño

De acuerdo con la metodología XP, el diseño se realiza en toda la ejecución del proyecto. En la fase de diseño se definió la arquitectura API REST como se muestra en la Figura 17. Para mayor información sobre la arquitectura, revisar los **puntos 4.13** y **4.14** del Marco teórico. Se implementa la autenticación por tokens por su seguridad y escalabilidad, véase los **puntos 4.15** y **4.16** de la sección del Marco teórico.

La aplicación está dividida en Front-end y Back-end, por consiguiente, se crearon dos proyectos que están alojados en GitHub. Las direcciones para acceder a los repositorios se presentan a continuación:

- **Back-end:** <https://github.com/JonnathanE/api-microlearning-LSEC>
- **Front-end:** <https://github.com/JonnathanE/microlearning-LSEC>

En el Back-end se construye la API que provee las funciones o servicios para que otro software lo pueda utilizar. Esto permite crear aplicaciones flexibles y escalables, porque los datos se pueden compartir desde cualquier sitio o aplicación con mayor facilidad. El objetivo de esto, es obtener las peticiones del usuario y por medio de los controladores realizar acceder a la base de datos. Las tecnologías utilizadas para su construcción son: Node.js, JavaScript y express.js.

Se empleó la base de datos MongoDB por sus características de escalabilidad y rapidez en realizar consultas. Para mayor información, revisar la sección del Marco teórico **punto 4.10 y 4.11. Comparativa entre bases de datos NoSQL.**

El Front-end constituye la interfaz de usuario (HTML, CSS y JavaScript) y es la encargada de realizar las peticiones HTTP a la API. Para su construcción se utilizó la librería de React.js.

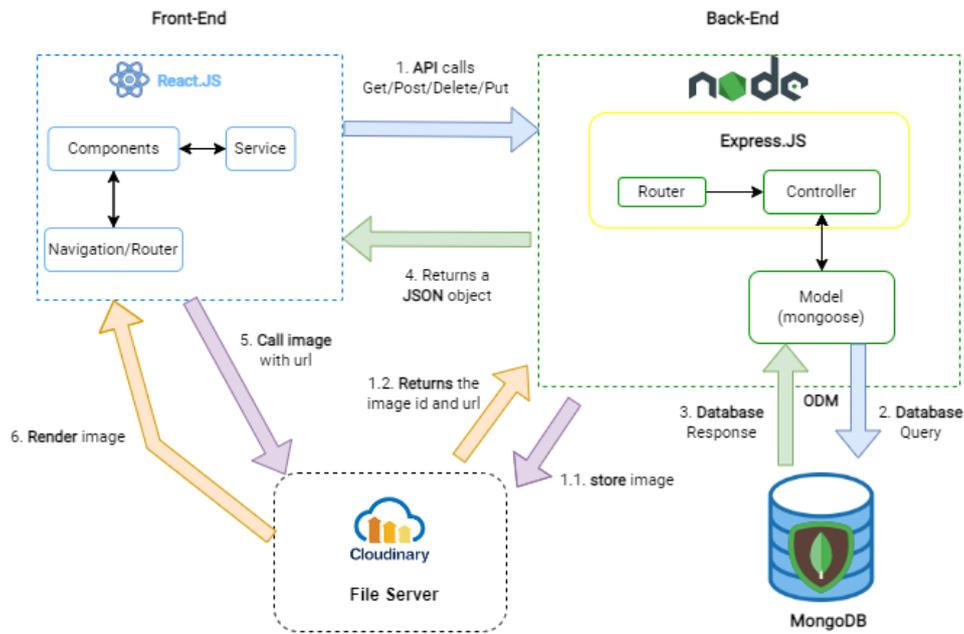


Figura 17. Arquitectura de la aplicación web.

6.2.2.1. Diseño del back-end

La estructura de carpetas del back-end se observa en la Figura 18 y la descripción de cada una de estas, se detalla en la Tabla 18.

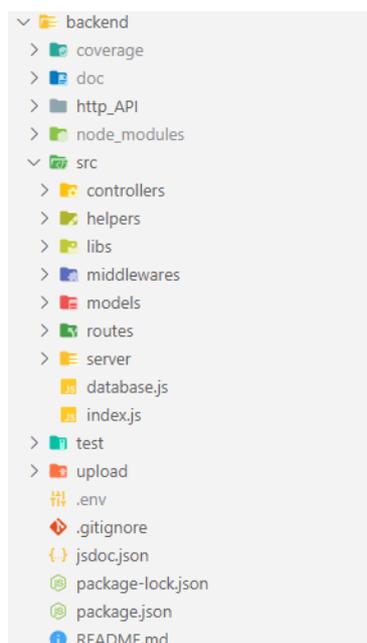


Figura 18. Estructura de carpetas en el back-end.

Tabla 18.
Estructura de carpetas en el back-end

Nombre	Descripción
coverage	Carpeta en donde se almacena información de las pruebas unitarias.
doc	Carpeta de la documentación de la API
http_API	Carpeta que almacena las peticiones para probar en Postman
node_modules	Carpeta que guarda las dependencias del sistema.
src	Carpeta en donde se encuentra la lógica de la API.
src/controllers	Carpeta que contiene todos los controladores de la API.
src/helpers	Carpeta que contiene scripts de especificación de errores.
src/libs	Carpeta que almacena scripts de configuraciones.
src/middlewares	Carpeta que contiene scripts que se ejecutan antes de recibir una petición del cliente.
src/models	Carpeta que almacena los modelos de la base de datos.
src/routes	Carpeta que contiene los endpoints de la API.
src/server	Carpeta que almacena el script de configuración del servidor con express.
src/database.js	Archivo en donde se configura mongoose.
src/index.js	Script que el servidor ejecuta para iniciar la API.
test	Carpeta que almacena las pruebas unitarias.
upload	Carpeta que sirve para almacenar los archivos (imágenes y gifs) de manera temporal hasta que se suban al servidor de archivos.
.env	Archivo que contiene variables de entorno.
.gitignore	Fichero para especificar carpetas o archivos para que git los ignore.
jsdoc.json	Archivo que contiene las configuraciones para generar la documentación.
package-lock.json	Fichero con información de la aplicación.
package.json	Fichero que contiene la configuración de la aplicación web y los scripts para ejecutar la aplicación.
README.md	Archivo que contiene información acerca de la aplicación.

Se elaboró un diagrama de clases a fin de describir la estructura de la API en el back-end, mostrando los atributos, relaciones y operaciones de las clases. En la Figura 19 se observa el diagrama de clases final. Para una mejor visualización del diagrama, véase el **Anexo 5**. Descripción del Diagrama de Clases.

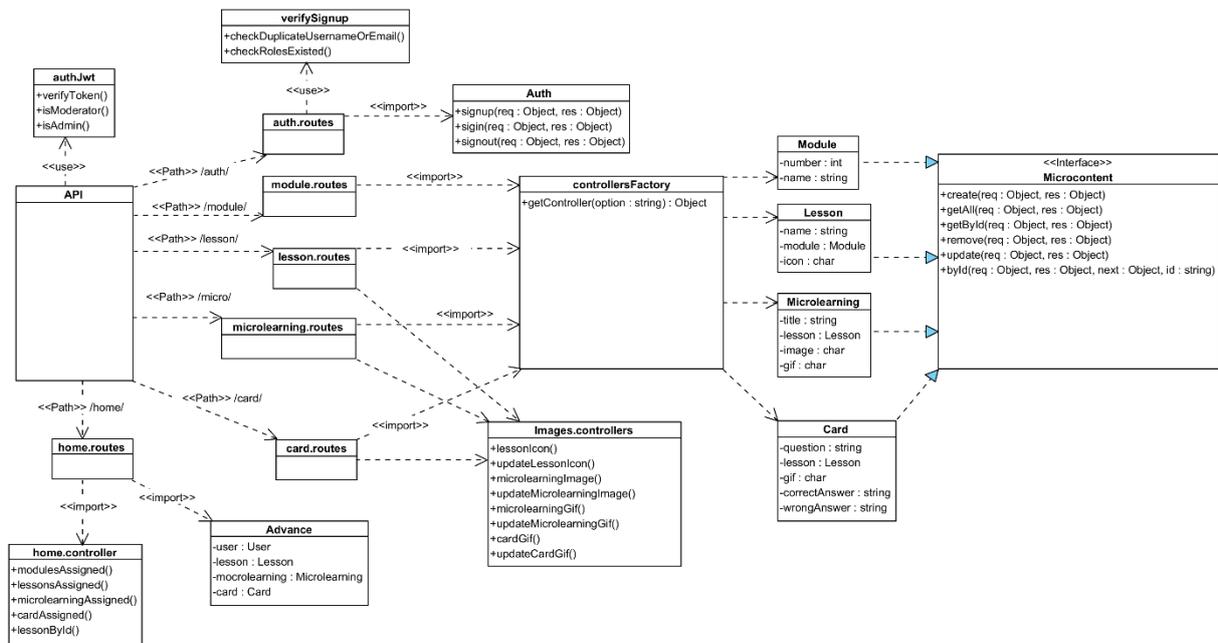


Figura 19. Diagrama de clases de la API.

Como lo indica el patrón de diseño Factory, en vez de instanciar un objeto directamente con el operador new, se crea mediante un método que proviene de la fábrica. En concreto, los objetos se instancian desde el método fábrica. Para referenciar a los objetos que devuelve la fábrica, se lo va a denominar producto. Aunque solo se cambia el lugar donde se invoca el objeto, ahora se puede sobrescribir el método fábrica en una subclase y cambiar la clase de los productos creados por el método. Para mayor información sobre los patrones de diseño y el método factory revisar el **punto 4.5 y 4.6** de la sección Marco teórico.

En la Figura 20 se presenta el diagrama de clases del patrón Factory Method implementada en el presente Trabajo de Titulación. Las clases Module, Lesson, Microlearning y Card deben implementar la interfaz Microcontent, que declara los métodos: create, getAll, getById, remove, update y byId. Cada clase implementa estos métodos de forma diferente. El método fábrica getController dentro de la clase ControllersFactory devuelve el objeto indicado.

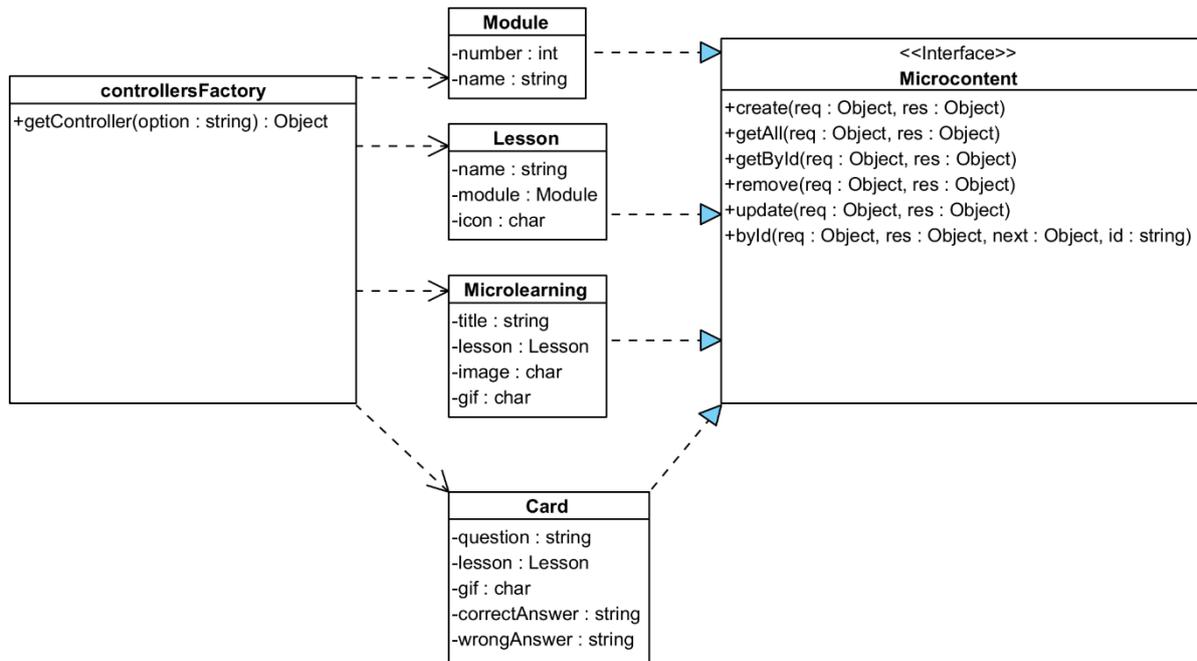


Figura 20. Diagrama de clases del patrón Factory Method de la solución informática.

6.2.2.2. Diseño del front-end

El front-end es la interfaz del usuario, mismo que se siguió el prototipo de la aplicación para la codificación (ver **Anexo 3. Prototipo de Interfaz de Usuario**).

La estructura de carpetas del front-end se presenta en la Figura 21 y la descripción de cada uno está en la Tabla 19.

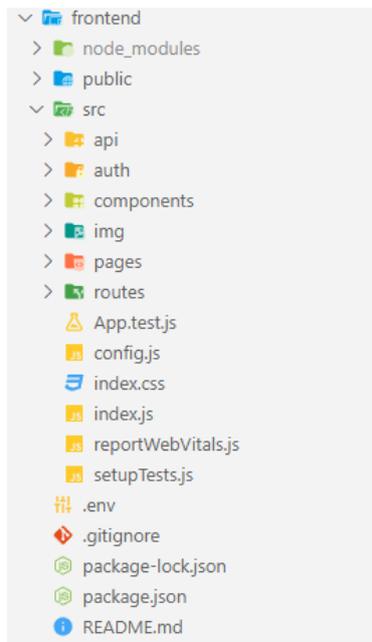


Figura 21. Estructura de carpetas en el front-end con React.

Tabla 19.
Estructura de carpetas en el front-end con React

Nombre	Descripción
node_modules	Carpeta que guarda las dependencias de la aplicación web.
public	Carpeta que guarda el archivo index HTML, estilos y archivos públicos.
src	Carpeta en donde se encuentran los elementos de React.
src/api	Carpeta que contiene los scripts de las peticiones http con axios.
src/helpers	Carpeta que contiene scripts de especificación de errores.
src/auth	Carpeta que contiene los archivos para autenticar mediante los estados globales usando el hook useContext
src/components	Carpeta que almacena los componentes jsx para utilizarlo en las páginas.
src/img	Carpeta que contiene imágenes.
src/pages	Carpeta que contiene las páginas web en las que el usuario interactúa.
src/routes	Carpeta que almacena las rutas privadas y públicas mediante la librería react-router-dom.
src/config.js	Archivo de configuraciones.
src/index.js	Script que el servidor ejecuta para iniciar la aplicación.
src/index.css	Archivo en css que provee estilos de manera global.
test	Carpeta que guarda las pruebas unitarias.
.env	Archivo que almacena las variables de entorno.
.gitignore	Fichero en donde se especifican las carpetas y archivos para que git los ignore.
package-lock.json	Fichero que almacena la información de las dependencias de la aplicación.
package.json	Fichero que contiene la configuración de la aplicación web y los scripts para ejecutar la aplicación.
README.md	Archivo que contiene información acerca de la aplicación.

La Figura 22 presenta el diagrama de despliegue en donde se observa la relación de los diferentes nodos de la aplicación web desarrollada. La aplicación está diseñada para ejecutar React.js en un servidor web y la API en un servidor de aplicaciones. La separación se hizo para que sea fácilmente escalable.

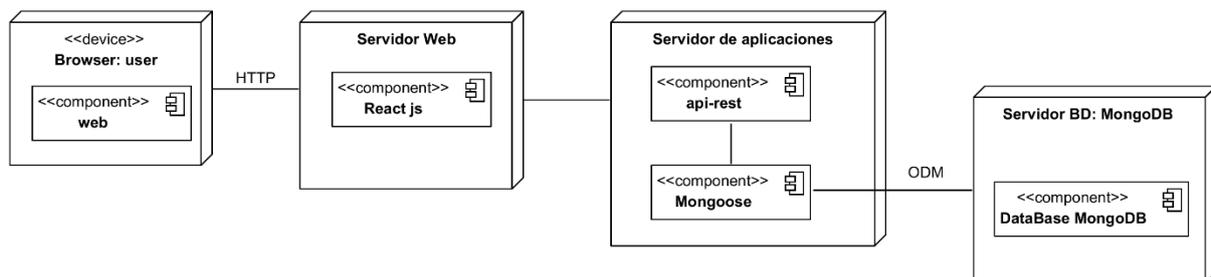


Figura 22. Diagrama de despliegue.

Finalmente, se desarrolló un diagrama de componentes de la aplicación web. La Figura 23 muestra la interacción de cada componente generado en el desarrollo del software, para una mejor visualización, véase el **Anexo 6**. Descripción del Diagrama de Componentes.

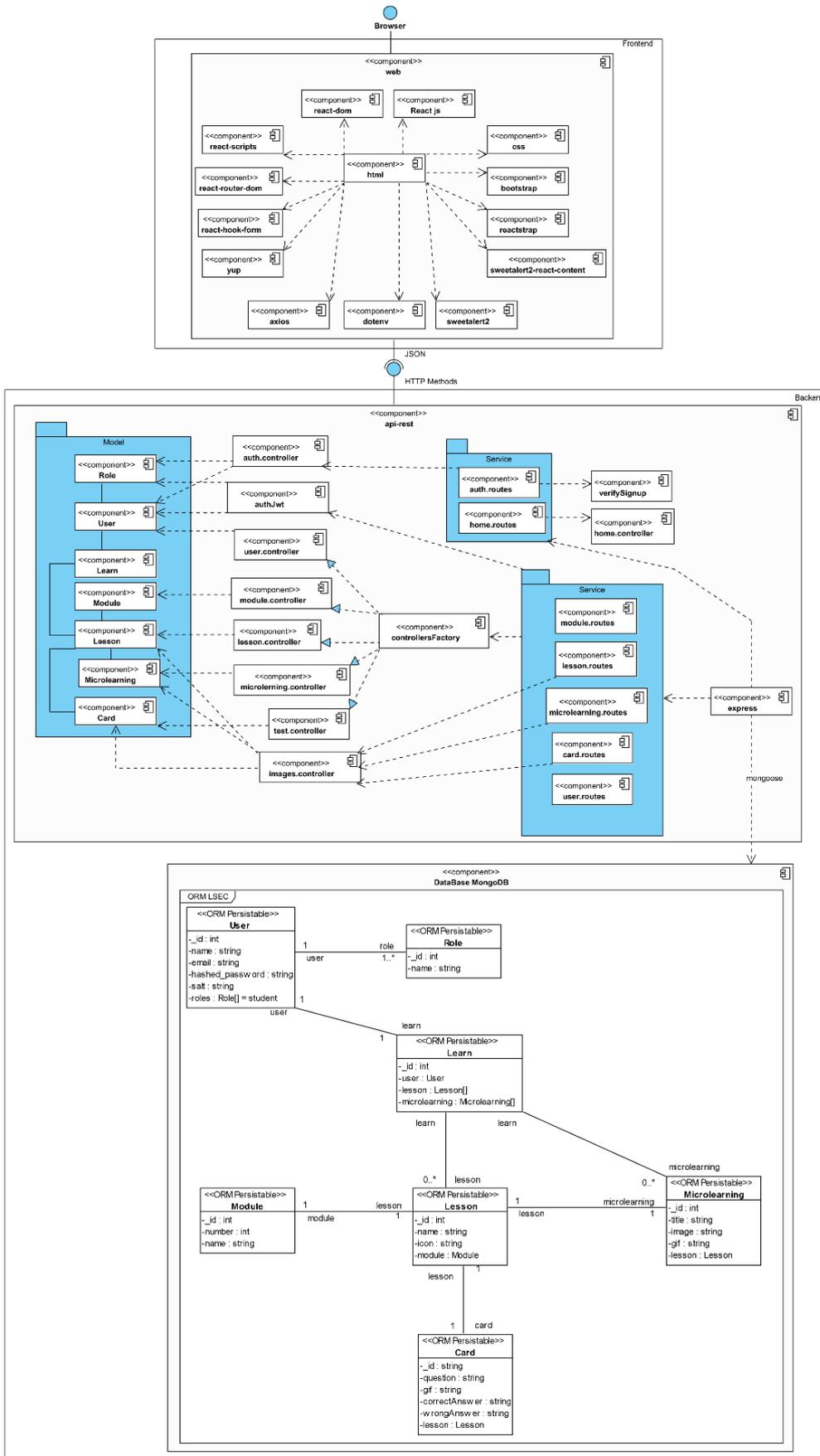


Figura 23. Diagrama de componentes de la aplicación web.

6.2.3. Codificación

Para facilitar la comprensión del código y como buena práctica de desarrollo de software, se optó por usar los siguientes estándares:

- Estándares para código:
 - El nombre de variables y métodos se escribieron basándose en el estilo de escritura “Camel Case”.
 - El nombre de componentes de React, la primera letra se escribió con mayúscula tal como lo sugiere la documentación de React.
 - Se escribió el código en inglés.
- Estándares para la Base de Datos:
 - Se siguió el estilo de escritura “Pascal Case” para nombrar las colecciones de MongoDB.
 - Los documentos de las colecciones se escribieron en minúsculas.
 - Los nombres de los documentos y colecciones se escribieron en inglés.
- Pautas para la Accesibilidad al Contenido Web (WCAG) en su versión 2.0:
 - Imágenes: usar el atributo alt para su descripción.
 - Los enlaces de hipertexto: usar texto simple y claro.
 - Página titulada: incluir título y descripción de cada página.
 - Organización de página: usar encabezados (<h1>), regiones/puntos de referencia, listas (, , y <dl>) y estructura consistente.
 - Secuencia significativa: diseñar una navegación intuitiva.

6.2.3.1. Codificación del back-end

JavaScript permite codificar en diferentes paradigmas de programación y para implementar el patrón Factory se utilizó clases en los controladores. Las librerías más destacadas que se instalaron son: express.js, mongoose, cors, crypto-js, jsdoc, jest y supertest.

En la Tabla 20 se presenta el código realizado para hacer la fábrica que proveerá los objetos. Se tiene las siguientes funciones:

- Importación de los controladores que se desea instanciar.
- Creación de la clase **controllersFactory**.
- La funcionalidad de **obtener** un controlador que se denomina **getController()**, recibe el parámetro **option**, que es un string. Se verifica si el parámetro recibe el nombre del controlador que desea retornar y devuelve un objeto del mismo, caso contrario se retorna un valor nulo.

Tabla 20.
Código de la clase controllersFactory

```
// importar
const User = require('../controllers/user.controller');
const Module = require('../controllers/module.controller');
const Lesson = require('../controllers/lesson.controller');
const Microlearning = require('../controllers/microlearning.controller');
const Card = require('../card.controller');
/**
 * Clase para administrar el patrón de diseño de fábrica
 */
class controllersFactory {
  /**
   * Método para obtener un controlador: usuario, módulo, lección, microaprendizaje, tarjeta
   * @param {string} option Nombre del controlador para regresar
   * @returns {Object}
   */
  getController = (option) => {
    if (option === 'user') return new User();
    if (option === 'module') return new Module;
    if (option === 'lesson') return new Lesson;
    if (option === 'microlearning') return new Microlearning;
    if (option === 'card') return new Card;
    return null;
  }
}
module.exports = controllersFactory;
```

La implementación del patrón Factory se utilizó en los endpoints de la API. En la Tabla 21 se muestra el código ejemplo en donde se hace uso de la clase **controllersFactory**, teniendo la siguiente estructura:

- Importar la clase **controllersFactory**.
- Se crea un objeto **controllersFactory** en una variable llamada **factory**.
- Por medio de la variable **factory** se llama a la función **getController('microlearning')**, pasándole el nombre del controlador que va a retornar. El objeto devuelto se almacena en una variable llamada **micro**.
- Se utiliza la variable **micro** para llamar a las funciones **create** o **getAll**.

Tabla 21.
Código del endpoint de la API

```
const { verifyToken, isAdmin, isModerator } = require('../middlewares/authJwt');
const controllersFactory = require('../controllers/controllersFactory'); // importar la fabrica

const factory = new controllersFactory(); // creo el objeto fabrica
const micro = factory.getController('microlearning'); // llamo al controlador microlearning

router.post('/', [verifyToken, isAdmin], micro.create); // utilizo el método devuelto para crear un microlearning
router.get('/', micro.getAll); // // utilizo el método devuelto para obtener un microlearning
```

```
// método para obtener el parámetro
router.param('microId', micro.byId);

module.exports = router;
```

6.2.3.2. Codificación del front-end

La codificación de la interfaz de usuario está desarrollada bajo la biblioteca React.js. Las librerías más destacadas que se instalaron son: axios, chart.js, sweetalert2, react-hook-form, yup y react-query.

Para hacer peticiones al servidor se hace uso de Axios, que es un cliente HTTP basado en promesas. En la Tabla 22 se presenta un ejemplo de código que hace peticiones al servidor.

Tabla 22.
Código para hacer peticiones al servidor desde el cliente

```
import axios from 'axios'; // importar la axios
import { API } from '../config'; // importar la url de la API

const user = JSON.parse(localStorage.getItem("user")); // obtener el usuario del localStorage
const TOKEN = user && user.token; // si existe un token lo almacena en una constante

// se crea un objeto axios con los atributos necesarios para conectarse a la API
export const userRequest = axios.create({
  baseURL: API,
  headers: {Authorization: `Bearer ${TOKEN}`}
});

// método para crear un Microlearning, como parámetro recibe el Id del microlearning
export const addMicrolearning = async micro => {
  const {data} = await userRequest.post('/micro', micro);
  return data;
}

// método para obtener todos los Microlearnings
export const getMicrolearnings = async () => {
  const {data} = await publicRequest.get('/micro');
  return data
}
```

6.2.4. Pruebas

6.2.4.1. Pruebas unitarias en el back-end

Las pruebas unitarias realizadas en cada iteración de la metodología XP permitieron comprobar el correcto funcionamiento de la API. Se utilizó el Framework **Jest** como herramienta de testing y automatización de la misma. Además, se realizaron los test a los endpoints y por ello se precisó instalar el módulo **SuperTest** que provee funcionalidades para realizar peticiones HTTP. Esto permitió verificar los diferentes escenarios al ejecutar llamadas

al servidor, constatando así el tipo de respuesta basada en REST, la información que se necesita enviar, los datos devueltos que sean correctos y la autenticación de los tokens.

Las pruebas realizadas a la API se determinaron con base en las Historias de Usuario y la identificación de varios escenarios que pueden vulnerar el sistema. Se realizaron un total de 73 pruebas, que se puede observar en la Figura 24 con el tiempo de ejecución total y su aprobación. En la Figura 25 se muestra un ejemplo de escritura de test realizada.

Para mayor detalle sobre las pruebas, véase el **Anexo 8. Plan de pruebas de la API.**

```
Test Suites: 5 passed, 5 total
Tests:      73 passed, 73 total
Snapshots:  0 total
Time:       36.078 s
Ran all test suites.
```

Figura 24. Total, de pruebas realizadas a los endpoints de la API.

```
1 test('CP46 update the parameter of the microcontent title', async () => {
2     const micros = await api
3         .get('/api/micro/')
4         .expect(200)
5         .expect('Content-Type', /application\/json/)
6     const updateMicro = {
7         title: 'Micro 137',
8     }
9     const response = await api
10        .put(`/api/micro/${micros.body[0]._id}`)
11        .send(updateMicro)
12        .set('authorization', `Bearer ${tokenAdmin}`)
13        .expect('Content-Type', /application\/json/)
14        .expect(200)
15        expect(response.body.message).toBe('El microcontenido se actualizado correctamente');
16    });
```

Figura 25. Ejemplo de test aplicado al endpoint de la API.

6.2.4.2. Pruebas unitarias en el front-end

Las pruebas unitarias para el front-end consiste en probar el renderizado de la interfaz de usuario, en concreto, lo que el usuario final ve. Actualmente, es primordial desarrollar este tipo de pruebas debido a que los nuevos frameworks y paradigmas de programación dedicados a las interfaces introducen lógica más compleja en sus algoritmos a comparación de años atrás. React proporciona la herramienta de testing **Jest** para automatizar sus pruebas, ya que estas dos fueron desarrolladas por Facebook. Para llevar a cabo la renderización de los componentes en su unidad, se utilizó **React Testing Library**, que es un módulo que dota de funciones para acceder al Document Object Model (DOM) e interactuar con el documento HTML.

Las pruebas se desarrollaron de acuerdo a las Historias de Usuario, como se trata de interfaces, se consideró los componentes generados a lo largo del desarrollo. Los resultados se observan en la Figura 26, teniendo un total de 59 pruebas. En la Figura 27 se muestra un ejemplo de escritura de test al front-end.

Véase el Anexo 9. Plan de pruebas unitarias del Front-end, para obtener un mayor detalle de las pruebas unitarias en el front-end.

```
Test Suites: 30 passed, 30 total
Tests:      59 passed, 59 total
Snapshots:  0 total
Time:       19.15 s, estimated 26 s
Ran all test suites.
```

Figura 26. Total, de pruebas front-end.

```
1 import '@testing-library/jest-dom'
2 import { render, screen, fireEvent } from '@testing-library/react';
3 import { MemoryRouter } from 'react-router-dom';
4 import Signup from './Signup';
5
6 beforeEach(() => {
7   render(<MemoryRouter><Signup /></MemoryRouter>)
8 })
9
10 describe('<Signup />', () => {
11
12   test('CP01 render signup user', () => {
13     screen.getByText('Email')
14     screen.getByText('Password')
15     screen.getByTestId('btnSubmit')
16     screen.getByTestId('linkLogin')
17   });
18
19   test('CP02 input name change', () => {
20     const inputPassword = screen.getByTestId('name')
21     expect(inputPassword.value).toBe("");
22     fireEvent.change(inputPassword, { target: { value: 'myname' } });
23     expect(inputPassword.value).toBe('myname');
24   });
25
26   test('CP03 input email change', () => {
27     const inputEmail = screen.getByTestId('email')
28     expect(inputEmail.value).toBe("");
29     fireEvent.change(inputEmail, { target: { value: 'jonathan@test.com' } });
30     expect(inputEmail.value).toBe('jonathan@test.com');
31   });
32
33   test('CP04 input password change', () => {
34     const inputPassword = screen.getByTestId('password')
35     expect(inputPassword.value).toBe("");
36     fireEvent.change(inputPassword, { target: { value: 'mypassword' } });
37     expect(inputPassword.value).toBe('mypassword');
38   });
39
40 });
```

Figura 27. Prueba unitaria de la interfaz para crear cuenta de usuario.

6.3. Objetivo 3

Evaluar la aplicación web en un entorno simulado para determinar la fiabilidad del diseño de la aplicación web.

En el presente apartado se detalla el proceso que se llevó a cabo para el cumplimiento del objetivo 3: Pruebas de integración, funcionalidad y aceptación.

6.3.1. Pruebas de Integración

Con las pruebas de integración se verificó el correcto funcionamiento entre los distintos componentes después de haber sido probados unitariamente, esto con la finalidad de comprobar la interacción de las interfaces de usuario con la API. Se identificaron y ejecutaron 55 casos de pruebas, que están estrechamente relacionadas con los requerimientos funcionales. Para la ejecución de las mismas, se usó el framework de testing **Cypress** que incluye librerías de aserciones, de mocks y la automatización de test 2e2 (end-to-end). La Figura 28 muestra el resultado de la realización de las pruebas, mostrando la aceptación de cada una de ellas y su tiempo total de ejecución.

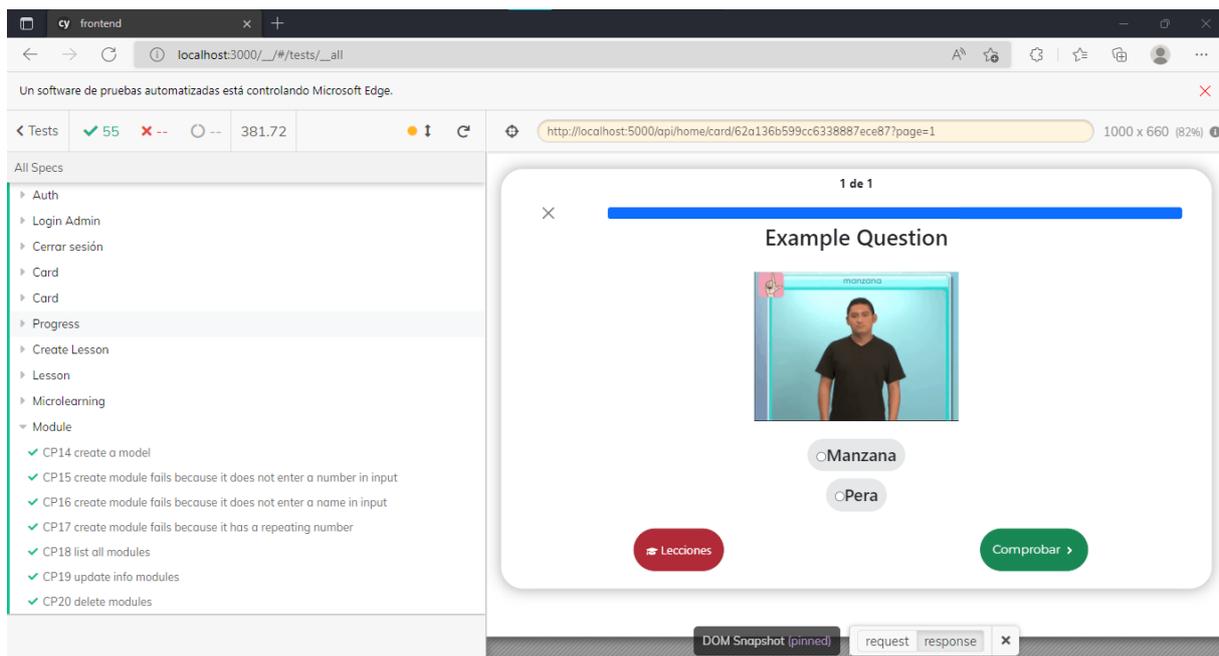


Figura 28. Resultado de las Pruebas de Integración de la aplicación web.

Para mayor información sobre el plan de pruebas de integración, revisar el **Anexo 10**. Plan de Pruebas de Integración.

6.3.2. Despliegue de la aplicación en un entorno simulado

Se desplegó la aplicación web en un servidor, para realizar las pruebas funcionales y de aceptación, en función de lo establecido en el objetivo 3. El despliegue de la base de datos,

se realizó en **MongoDB Atlas**, que se lo conoce como un servicio en la nube propiamente para MongoDB. En la Figura 29 se observa la base de datos creada y en ejecución.

El proyecto del back-end se desplegó en el servicio web de **Render Hosting** y el front-end en **Vercel**, como se observa en la Figura 30 y Figura 31 respectivamente. Para acceder a la aplicación web se puede ingresar a la siguiente dirección web: <https://learn-lsec.vercel.app/>.

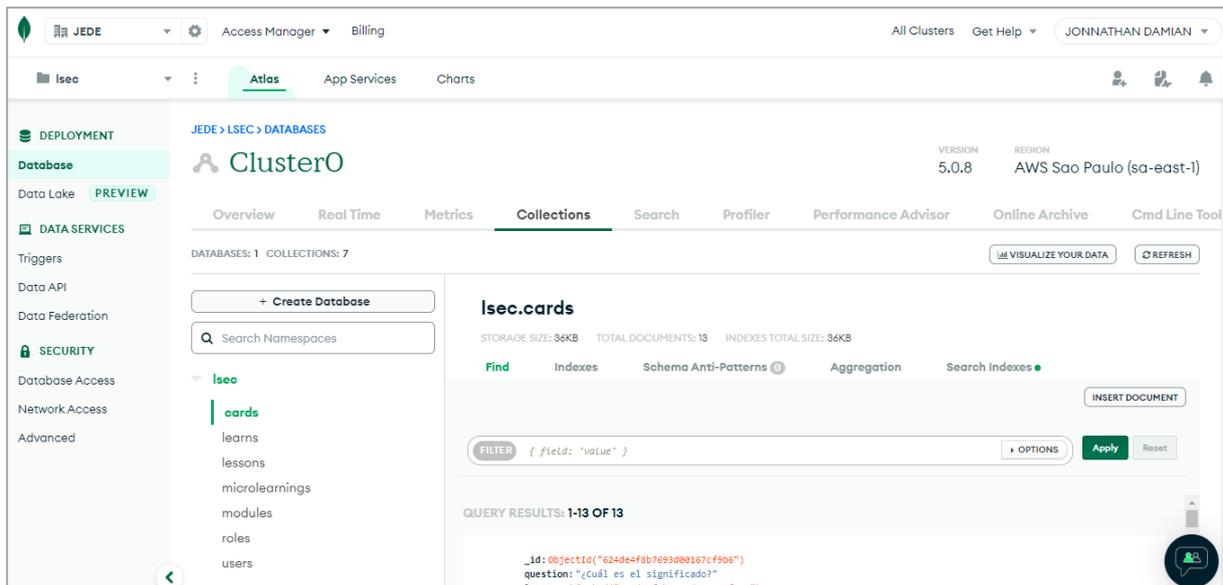


Figura 29. Creación y despliegue de la base de datos en MongoDB Atlas.

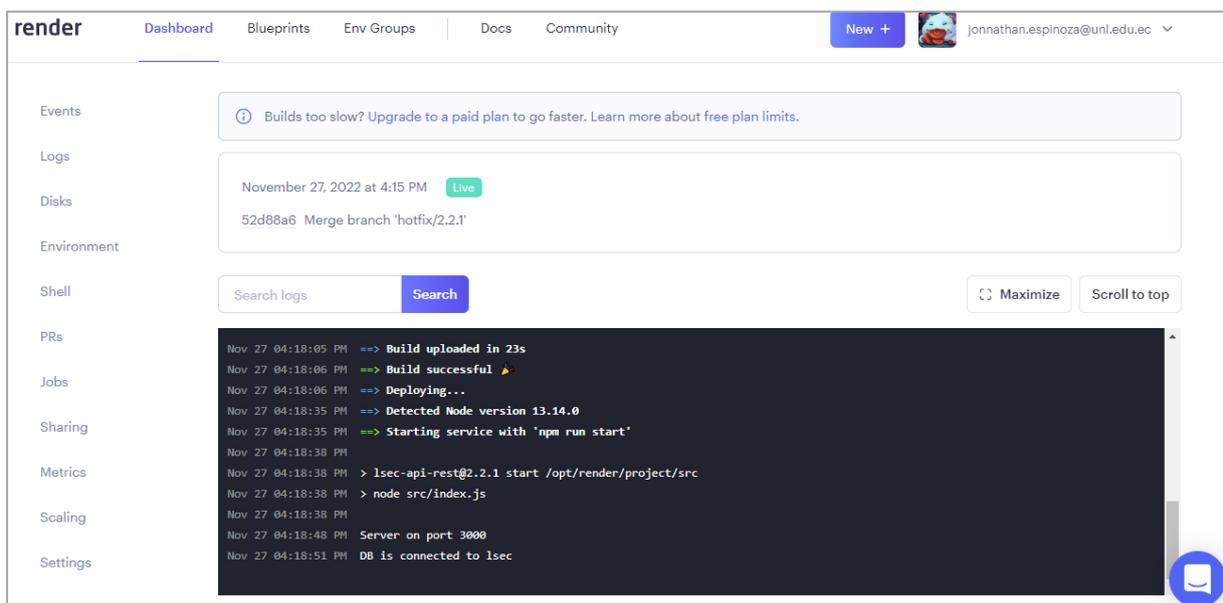


Figura 30. Despliegue de la API en Render Hosting.

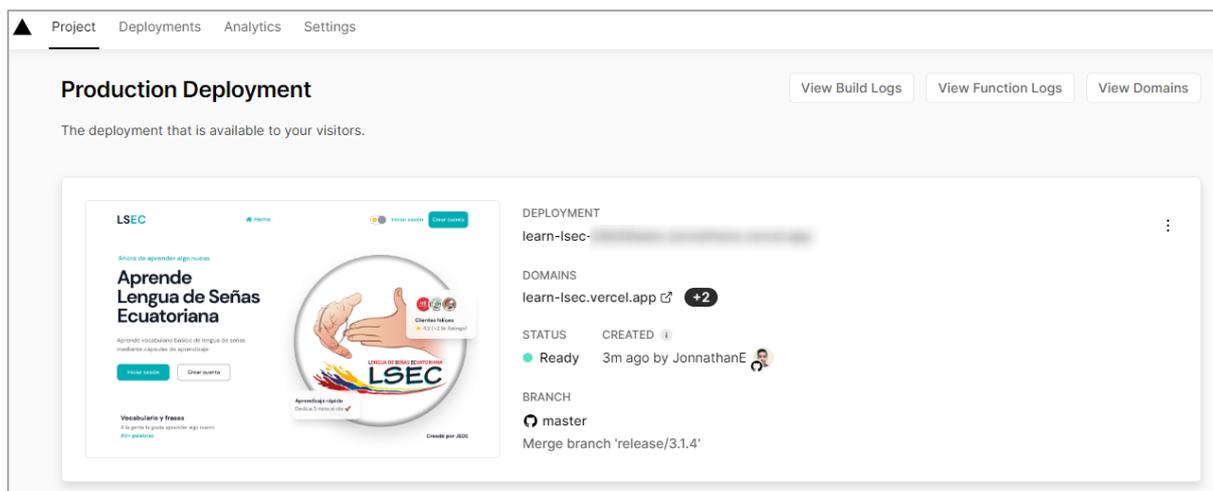


Figura 31. Despliegue de la interfaz de usuario de React en Vercel.

Una vez desplegada la aplicación web en producción, se procedió a crear el contenido educativo con la finalidad de realizar las pruebas de aceptación. En la entrevista realizada al presidente de la Asociación de Personas Sordas Virgen del Cisne, se mencionó que solo las asociaciones pueden llevar a cabo los cursos de enseñanza de la LSEC, por tal motivo, cualquier software independiente, debe de tener temas básicos dentro de ella (ver **Anexo 1. Elicitación de Requerimientos punto 1.7**). Con la ayuda del presidente se determinó y elaboró los temas a utilizar en un entorno simulado. Cabe recordar que la solución informática posee un panel administrativo para que el usuario administrador pueda gestionar (crear, ver, actualizar y eliminar) el contenido educativo, haciendo que estos no sean estáticos, por ello, con fines académicos y para evaluar el diseño de la aplicación se escogieron los temas de la Tabla 23.

Tabla 23.
Contenido de la aplicación web

Módulo	Lecciones	Cápsulas de Aprendizaje	Tarjetas de Aprendizaje
Módulo 1	Saludos	Buenas tardes, Buenas noches, Buenos días, Hola, Adiós	Buenas tardes, Buenas noches
	Frutas	Pera, Manzana, Naranja, Mandarina	Pera, Naranja
	Frasas	¿Cómo te sientes?, ¿En qué le puedo ayudar?, ¿Está bien?, ¿Quién me puede ayudar?	¿En qué le puedo ayudar?, ¿Quién me puede ayudar?
Módulo 2	Familia	Padre, Mamá, Hermano, Hermana, Abuela	Padre, Hermano
	Números	Uno, Dos, Tres, Cuatro, Cinco, Seis (1), Seis (2), Siete (1), Siete (2), Ocho (1), Ocho (2), Nueve (1), Nueve (2), Diez (1), Diez (2)	Seis (1), Diez (1)
	Animales	Abeja, Perro, Gato, Vaca	Abeja, Gato
	Frasas 2	Por favor, Por favor (2), Un momento por favor, Su número, por favor, Por favor, necesito información, Pase, por favor, Gracias, ¿Quién me puede ayudar?,	Por favor, Por favor (2), Un momento por favor, Su número, por favor, Por favor, necesito información, Pase, por favor, Gracias, ¿Quién me puede

		¿En qué le puedo ayudar?, Prestar, Amigo(a)	ayudar?, ¿En qué le puedo ayudar?, Prestar, Amigo(a)
Módulo 3	Medicina	Hospital, Doctor, Vacuna, Mascarilla, lavarse, Contagiar, Contagioso(a), Enfermar, Enfermo, Enfermedad, Tos, Fiebre, Ayudar, Ayuda	Hospital, Doctor, Vacuna, Mascarilla, lavarse, Contagiar, Contagioso(a), Enfermar, Enfermo, Enfermedad, Tos, Fiebre, Ayudar, Ayuda
	Direcciones	Calle, Dirección, ¿Dónde?, ¿Cómo llegó?, Dirección, Terminal de buses, Taxi	Calle, Dirección, ¿Dónde?, ¿Cómo llegó?, Dirección, Terminal de buses, Taxi
	Preguntas	¿Cómo?, ¿Por qué?, ¿Cuándo?, ¿Dónde?, ¿Para qué?	¿Cómo?, ¿Por qué?, ¿Cuándo?, ¿Dónde?, ¿Para qué?
	Transacciones Monetarias	¿Cuánto cuesta?, Sin dinero, Por favor necesito un certificado bancario	¿Cuánto cuesta?, Sin dinero, Por favor necesito un certificado bancario

Como se menciona dentro de los requerimientos, las señas se deben de presentar en un GIF. Si bien existen diferentes maneras de generar un GIF ya sea a través de animaciones o videos, por motivos de agilidad, se utilizó los videos del Diccionario de Lengua de Señas Ecuatoriano “Gabriel Román” y por medio de un script en Python se convirtió en GIF. En la Tabla 24 se muestra el script realizado para transformar los videos.

Tabla 24.
Script en Python para transformar video a gif

```
# Import libraries
from pytube import YouTube
from pytube.request import stream
from moviepy.editor import VideoFileClip

# Section to download the youtube video
link = input('Insert video link: ')
yt = YouTube(link)
print(yt.title)
print('download video')
stream = yt.streams.filter(progressive=True, file_extension='mp4').first().download()

# Section to convert video to gif
print('Convert video to gif')
video = yt.title + '.mp4'
print(video)
clip = VideoFileClip(video)
clip.write_gif(yt.title+'.gif', fps=10)
print('Ready')
```

6.3.3. Pruebas funcionales

Para verificar que la aplicación web satisface los requerimientos funcionales, se realizaron las Pruebas Funcionales. El resumen de los casos de prueba se observa en la Tabla 25.

Para mayor detalle sobre las pruebas, observar el **Anexo 11**. Plan de Pruebas Funcionales.

Tabla 25.
Casos de pruebas funcionales

Nro. Caso de prueba	Descripción del caso de prueba	RF relacionados	Resultado	Observación
CP01	Se probará la respuesta correcta al crear cuenta de usuario.	RF01	SI	
CP02	Se probará la respuesta al crear cuenta con los campos vacíos.	RF01	SI	
CP03	Se probará la respuesta correcta al iniciar sesión.	RF02	Parcialmente	Se corrigió la funcionalidad de guardar el token en el Local Storage porque la API devolvía el ID del usuario, lo que era una brecha de seguridad.
CP04	Se probará la respuesta al iniciar sesión con una contraseña incorrecta.	RF02	SI	
CP05	Se probará la respuesta correcta al crear un nuevo módulo.	RF04	SI	
CP06	Se probará la respuesta correcta al visualizar todos los módulos creados.	RF04	SI	
CP07	Se probará la respuesta correcta al actualizar los datos de un módulo creado.	RF04	SI	
CP08	Se probará la respuesta correcta al eliminar un módulo creado.	RF04	SI	
CP09	Se probará la respuesta correcta al crear una nueva lección.	RF05	SI	
CP10	Se probará la respuesta correcta al visualizar todas las lecciones creadas.	RF05	SI	
CP11	Se probará la respuesta correcta al actualizar los datos de una lección creada.	RF05	SI	

CP12	Se probará la respuesta correcta al eliminar una lección creada.	RF05	SI	
CP13	Se probará la respuesta correcta al crear una cápsula de aprendizaje.	RF08	SI	
CP14	Se probará la respuesta correcta al visualizar todas las cápsulas de aprendizaje creadas.	RF08	SI	
CP15	Se probará la respuesta correcta al actualizar los datos de una cápsula de aprendizaje.	RF08	SI	
CP16	Se probará la respuesta correcta al eliminar una cápsula de aprendizaje creada.	RF08	SI	
CP17	Se probará la respuesta correcta al crear una tarjeta de aprendizaje.	RF06	SI	
CP18	Se probará la respuesta correcta al visualizar todas las tarjetas de aprendizaje creadas.	RF06	SI	
CP19	Se probará la respuesta correcta al actualizar los datos de una tarjeta de aprendizaje.	RF06	SI	
CP20	Se probará la respuesta correcta al eliminar una tarjeta de aprendizaje creada.	RF06	SI	
CP21	Se probará la respuesta correcta al cerrar sesión.	RF03	Parcialmente	Se corrigió las rutas privadas porque la app permitía ingresar al usuario cuando el token se había caducado, por tal motivo se creó la funcionalidad de refrescar el token.
CP22	Se probará la repuesta al listar todas las lecciones completadas por el usuario.	RF07	SI	

6.3.4. Pruebas de Aceptación

El propósito de las pruebas de aceptación es garantizar que el diseño de la aplicación web desarrollada en este documento sea aceptable. Para su aplicación se tomó una muestra por el método no probabilístico por conveniencia de 24 estudiantes pertenecientes a la carrera de ingeniería en sistemas/computación de la Universidad Nacional de Loja, cuya edad varía entre los 18 a 25 años de edad. Luego de la presentación y manejo del software, se llenó la encuesta y se aplicaron los parámetros resumidos en la Tabla 26. Además de estos parámetros, en la

encuesta aplicada se solicitó su nombre, identificación, dirección de correo electrónico y sugerencias para la veracidad de los resultados.

Para mayor detalle sobre el procedimiento y ejecución de las pruebas de aceptación, observar el **Anexo 12**. Plan de Pruebas de Aceptación.

Tabla 26.
Resumen de Pruebas de Aceptación

No.	Parámetros	Porcentaje (%)		
		Sí	Parcialmente	No
1	¿Considera que la interfaz de la aplicación es intuitiva?	66,7	33,3	0
2	¿Se entiende el contenido dentro de la aplicación web?	91,7	8,3	0
3	¿Resulta fácil entender el resultado de una acción?	87,5	12,5	0
4	¿Son apropiados los mensajes presentados por el sistema?	62,5	37,5	0
5	¿La aplicación presenta claramente los errores?	50	41,7	8,3
6	¿Permite una cómoda navegación dentro de la aplicación?	66,7	33,3	0
7	¿Es fácil identificar un objeto o acción?	83,3	16,7	0
8	¿La interfaz está diseñada para facilitar la realización de las tareas de la mejor forma posible?	75	25	0
9	¿El uso de GIFs (imágenes animadas) ayuda en la comprensión de la lengua de señas?	66,7	33,3	0
10	¿Los repasos o prácticas por medio de la selección múltiple en la aplicación web, sirven como un medio de practicar la lengua de señas?	79,2	20,8	0
11	¿La aplicación web garantiza su operabilidad al no presentar fallas?	37,5	62,5	0
12	¿La funcionalidad de la aplicación web, permite el aprendizaje de la lengua de señas?	87,5	12,5	0
PROMEDIO		71,19	28,12	0,69

Como se ve en la Tabla 26, el resultado para los criterios **sí**, **parcialmente** y **no** es de 71,19%; 28,12% y 0,69% respectivamente. Las pruebas de aceptación se clasifican como **positivas** cuando se comparan con los criterios de estado de la Tabla 27.

Tabla 27.
Criterio para el estado de las Pruebas de Aceptación

Estado de las Pruebas de Aceptación	Criterio		
	Sí	Parcialmente	No
Positivo	$\geq 70\%$	$\leq 30\%$	$< 1\%$
Negativo	$< 70\%$	$> 30\%$	$> 1\%$

Uno de los inconvenientes que el usuario final encontró, fue el momento de cargar la siguiente cápsula de aprendizaje, ya que el GIF no se actualiza al mismo tiempo, lo que provoca confusión. Dicho problema no se había tenido en cuenta hasta ese momento, por lo cual se procedió a corregir inmediatamente. La solución a dicho problema, fue la de crear un loader para los gifs y se utilizó los hooks `Suspense` de React. Además, se implementaron algunas

sugerencias de los encuestados: tener todas las alertas centradas, mejorar los espacios en cada componente, agregar el modo oscuro y añadir más contenido en las cápsulas.

Posteriormente, se realizó una presentación de la aplicación web a 4 profesores de la lengua de señas ecuatoriana pertenecientes a la Asociación de Personas Sordas Virgen del Cisne, quienes cuentan con la certificación de la FENASEC, cuya edad varía entre los 24 a 30 años de edad y de los cuales 3 son personas con discapacidad auditiva. Dicha presentación tuvo el objetivo de obtener el nivel de aceptación del software por parte de profesionales en la enseñanza de la LSEC.

Los profesionales después de probar el software, llenaron una encuesta sobre la aceptación de la aplicación web. En la Tabla 28 se muestra el resumen de los resultados.

Tabla 28.
Resumen de las encuestas a profesionales en LSEC

No.	Parámetros	Porcentaje (%)				
		Totalmente de acuerdo	De acuerdo	Indeciso	En desacuerdo	Totalmente en desacuerdo
P1	¿La aplicación web al momento de usarla fue?	0	33	67	0	0
P2	¿Considera que la aplicación es atractiva e intuitiva?	0	100	0	0	0
P3	¿Considera que la aplicación web es fácil de usar para cualquier persona con capacidad de manejar un navegador web?	0	100	0	0	0
P4	¿La funcionalidad de la aplicación web, permite el aprendizaje de la lengua de señas?	0	100	0	0	0
P5	¿Presentar las lecciones en contenidos cortos (Microlearning), ayuda a la enseñanza de la lengua de señas?	33	67	0	0	0
P6	¿El uso de GIFs (imágenes animadas) ayuda en la comprensión de la lengua de señas?	33	67	0	0	0
P7	¿Los repaos o prácticas por medio de la selección múltiple en la aplicación web, sirven como un medio de practicar la lengua de señas?	33	67	0	0	0
P8	¿Considera que la aplicación web fomenta el aprendizaje de la lengua de señas?	33	67	0	0	0

P9	¿La aplicación web garantiza su operabilidad al no presentar fallas?	0	100	0	0	0
P10	¿Haría uso cotidiano de la aplicación?	0	100	0	0	0
P11	¿Recomendaría a otros usuarios que usen la aplicación?	67	33	0	0	0
PROMEDIO		18,09	75,82	6,09	0	0

Con las preguntas P1, P2, P3 y P9 de la Tabla 28, se determina que, el 33% de los expertos consideran que la aplicación web es Fácil de usar y el 67% no tuvo complicaciones en usar la aplicación. El 100% de los expertos están De Acuerdo en que la aplicación es atractiva e intuitiva, cualquier persona con capacidad de manejar un navegador web podría utilizar sin problema la aplicación web y que la aplicación opera sin fallas. Como resultado de este análisis, se establece que la aplicación web es fácil de usar y es intuitiva para el usuario.

Con las preguntas P4, P5, P6, P7, P8, P10 y P11 de la Tabla 28, se determina que, el 100% de los expertos están De Acuerdo en que la aplicación web permite el aprendizaje de la lengua de señas ecuatoriana y la ocuparían cotidianamente. El 33% indicaron que están Totalmente de Acuerdo que el uso de la metodología del microlearning ayudaría en la enseñanza de la lengua de señas ecuatoriana y el 67% están De Acuerdo; estos mismos porcentajes se repiten en la aceptación de los GIFs como medio visual para presentar el patrón de la lengua de señas, en la implementación de prácticas de retroalimentación por medio de selección múltiple, y consideran que la aplicación fomenta el aprendizaje de la lengua de señas ecuatoriana. El 67% están Totalmente de Acuerdo en recomendar la aplicación web y el 33% están De Acuerdo. Como resultado de este análisis, se establece que la aplicación web fomenta el aprendizaje básico de la lengua de señas ecuatoriana mediante el micro-learning.

Para mayor información sobre las encuestas realizadas a los profesionales en la educación de la LSEC, ver el **Anexo 13**. Encuestas a maestros en lengua de señas ecuatoriana.

Para validar los resultados de las encuestas a los profesionales en la educación de la lengua de señas ecuatoriana, el presidente de la Asociación de Personas Sordas Virgen del Cisne otorgó un certificado, el cual se puede observar en el **Anexo 14**. Certificado de la Asociación de Personas Sordas Virgen del Cisne.

Finalmente, se realizó una prueba a los docentes de la carrera de Pedagogía de las Ciencias Experimentales - Informática de la Universidad Nacional de Loja, con la finalidad de obtener recomendaciones sobre el uso del micro-learning para enseñar LSEC (ver **Anexo 17**. Prueba de aceptación a profesional de la educación). La recomendación fue la siguiente: “Es importante sugerir que la imagen debe estar en relación con la Señá, considerando que eso

facilita el proceso de enseñanza-aprendizaje. También sugiero que el recuadro de la persona que realiza la Señal, sea más grande y claro, para su ejercicio y comprensión”.

Al finalizar las pruebas de aceptación, se generó tres manuales que son:

- **Manual de usuario** que se encuentra en el **Anexo 15**. Manual de Usuario.
- **Manual de administrador** que se encuentra en el **Anexo 16**. Manual de Administrador.
- **Manual de programador** que se observa en la siguiente dirección web:
<https://github.com/JonnathanE/api-microlearning-LSEC> (README.md) y
<https://github.com/JonnathanE/microlearning-LSEC> (README.md).

7. Discusión

7.1. Desarrollo de la propuesta alternativa

A continuación, se evalúa los resultados obtenidos durante la ejecución del TT.

7.1.1. Objetivo 1: Realizar la ingeniería de requerimientos para la aplicación web de Micro-Learning dedicada al aprendizaje del Lenguaje de Señas Ecuatoriano.

El desarrollo del primer objetivo está enfocado en obtener los requerimientos e historias de usuario, como lo indica la primera fase de la metodología XP; y para su obtención se realizaron dos técnicas: estudio de documentación y entrevistas.

El propósito del estudio de documentación es obtener recomendaciones para diseñar una aplicación web dedicada a la educación de la lengua de señas (ver **la sección de Resultados**, el punto **6.1.3**). Para ello, se tomaron los trabajos relacionados, en los cuales, los autores se basan en mostrar una retroalimentación inmediata y continua a través de una enseñanza visual. Para tener un contexto de cómo elaborar la interfaz con el micro-learning, se utilizó la observación para extraer los elementos utilizados y plasmarlo en un wireframe cada uno de las fases: aprender, practicar y evaluar.

Dentro de la fase aprender, se encontró, que los autores [41], [43], [44], [45], [50], [51] y [52] llevan a cabo una enseñanza visual con elementos animados porque permite a los estudiantes realizar más procesamiento cognitivo y los materiales utilizados para presentar la seña son: videos, animación 2D, imágenes estáticas y dinámicas. En la fase de practicar y evaluar, se realizó una investigación de metodologías de evaluación de lengua de señas y la estrategia que los investigadores [53], [54] y [55] han utilizado, es adaptar las pruebas existentes de un lenguaje de señas a otro u hablado, basándose en cuatro tipos: elegir entre imágenes, elegir entre palabras, elegir correcto o incorrecto y escribir palabras. Por tal motivo, para el desarrollo de la fase práctica se eligió crear tarjetas de aprendizaje donde el usuario final debe de elegir entre correcto o incorrecto.

Para indagar sobre el proceso de enseñanza de la LSEC, las metodologías y herramientas, se realizó una entrevista a dos expertos en la Lengua de Señas Ecuatoriana (ver **Anexo 1**. Elicitación de Requerimientos). Junto con un prototipo realizado a partir de los wireframes, se determinó que la aplicación del micro-learning es una buena herramienta de apoyo para enseñar lengua de señas por la utilización de unidades pequeñas de contenido de aprendizaje en la web. Además, que los mejores recursos que se pueden presentar las señas son las imágenes animadas en formato de GIF o videos, corroborando así el estudio de documentación.

Para crear una aplicación accesible también se tomó las recomendaciones de I. Fajardo en su trabajo “Accesibilidad Cognitiva de los Sordos a la Web”, en el que propone una guía para desarrollar aplicaciones web accesibles para personas con discapacidad auditiva. Creando así una interfaz amigable para los usuarios finales (ver **Anexo 3**. Prototipo de Interfaz de Usuario).

Después del análisis del estudio de documentación y las entrevistas, se realizó la Especificación de Requerimientos de Software basándose en el estándar IEEE 830-1998, en donde se definió las funcionalidades, restricciones y las historias de usuario del sistema (ver **Anexo 2**. Especificación de Requerimientos de Software).

7.1.2. Objetivo 2: Desarrollar la aplicación web de Micro-Learning para el aprendizaje del Lenguaje de Señas Ecuatoriano.

Se utilizó la metodología XP para el desarrollo de la aplicación web mediante cuatro fases: Planificación, Diseño, Codificación y Pruebas. La estructura del proyecto se generó basándose en la arquitectura Rest API, justamente por su escalabilidad y mantenibilidad. Las tecnologías de Node.js y React.js fueron seleccionados por su curva de aprendizaje sencilla, fácil de implementar, conocimiento propio de la tecnología de JavaScript y para mantener un mismo lenguaje de programación; mientras que la base de datos MongoDB fue elegida por medio de una investigación (ver el **punto 4.10 y 4.11 de la sección de Marco Teórico**), en donde resultó ser la más eficiente al realizar búsquedas y recuperar información.

La fase de planeación se realizó a través de especificación de historias de usuario (ver **la sección de Resultados**, el punto **6.2.1**); primero se llevó a cabo la planificación de los requerimientos con la herramienta Miro, en donde se estableció las versiones de la aplicación y el número de iteraciones. Posteriormente, se establecieron tareas, y para mantener un control y seguimiento de cada una de ellas se utilizó Jira Software. Para crear la API y la interfaz de usuario, se dividió el proyecto en Back-end y Front-end.

En el back-end, la fase de diseño, se completó por medio de un diagrama de clases, además de diseñar el patrón Factory para gestionar los controladores de la API (ver **la sección de Resultados**, el punto **6.2.2.1**). La fase de codificación se realizó en Node.js y para implementar el patrón Factory, se trabajó bajo el paradigma de Programación Orientada a Objetos, sin embargo, el lenguaje de JavaScript no proporciona la creación de clases abstractas o de interfaz, por ello, de forma individual se verificó si todas las clases poseían los mismos métodos (ver **la sección de Resultados**, el punto **6.2.3.1**). La fase de pruebas, se ejecutó de manera correcta a través de la tecnología Jest y para comprobar los endpoints se precisó la instalación del módulo SuperTest, que provee funcionalidades para ejecutar

peticiones HTTP, verificando así el tipo de respuesta en REST, los datos devueltos y la autenticación por tokens (ver **la sección de Resultados**, el punto **6.2.4.1**).

En el front-end, en la fase de diseño se elaboró un diagrama de componentes (ver **la sección de Resultados**, el punto **6.2.2.2**), la fase de codificación, se desarrolló basándose en el prototipo de la aplicación. En la fase de pruebas, se realizó con las tecnologías Jest y React Testing Library, que vienen instaladas por defecto en React (ver **la sección de Resultados**, el punto **6.2.4.2**).

7.1.3. Objetivo 3: Evaluar la aplicación web en un entorno simulado para determinar la fiabilidad del diseño de la aplicación web.

El objetivo se cumplió con la elaboración de las pruebas de integración, funcionales y de aceptación. En las pruebas de integración (ver **la sección de Resultados**, el punto **6.3.1**) se determinó el correcto funcionamiento de la API con el cliente, obteniendo resultados favorables gracias a la herramienta de Cypress que permite generar y automatizar pruebas e2e (end-to-end) como si fuera un simulador de usuario final en tiempo real. Las pruebas funcionales (ver **la sección de Resultados**, el punto **6.3.3**) permitieron determinar que los requerimientos funcionales se cumplieron en su totalidad, alcanzando resultados positivos, ya que se siguió el Marco de Desarrollo de la Junta de Andalucía (MADEJA).

Las pruebas de aceptación (ver **la sección de Resultados**, el punto **6.3.4**) se aplicaron a estudiantes de la carrera de Ingeniería en Sistemas/Computación, ya que reúnen las características del usuario final, mismos que llenaron una encuesta luego de la presentación y manipulación del software. El 71,19% indicaron, con un criterio de **Si**, que la interfaz es atractiva, intuitiva, fácil de usar, aceptación de los gifs como recurso para mostrar las señas y que la aplicación web mediante el micro-learning permite el aprendizaje de la lengua de señas ecuatoriana. Dichos resultados también se contrastan con los trabajos relacionados [41], [42], [43] y [45] que tienen por objetivo el aprendizaje-enseñanza de la lengua de señas por medio de software utilizando videos e imágenes animadas; no obstante, en el presente trabajo de titulación, todo el material visual está inmerso dentro de la metodología micro-learning, haciendo que la solución informática sea escalable gracias a la implementación de un panel administrativo. Además, los encuestados manifestaron que, al momento de practicar mediante tarjetas de aprendizaje, sería bueno incrementar más opciones de selección múltiple y que estas sean aleatorias para aumentar la dificultad.

Durante la encuesta a los usuarios finales se encontró una limitación dentro de la aplicación, al momento de pasar al siguiente contenido, el GIF no se actualiza al mismo tiempo. El inconveniente se suscita en la interfaz gráfica y la forma idónea para resolverlo fue la de

implementar el hook Suspense de React, que permite cargar los componentes bajo demanda (ver **Anexo 12**. Plan de Pruebas de Aceptación).

7.2. Valoración técnica económica ambiental

7.2.1. Valoración técnica

Los recursos técnicos que se emplearon en el desarrollo del TT se observan en la Tabla 29.

Tabla 29.
Valoración técnica

Recurso	Descripción	Link
Visual Paradigm Community	Software que se utilizó para la construcción de los diagramas UML (ver el punto 6.2.2 de la sección de Resultados).	https://www.visual-paradigm.com/
Adobe XD	Herramienta utilizada en la creación del prototipo de la interfaz gráfica de la aplicación web (ver Anexo 3. Prototipo de Interfaz de Usuario).	https://www.adobe.com/la/products/xd.html
Miro	Software que se utilizó para planificar las versiones de la aplicación web y el orden de ejecución de los requerimientos (ver Anexo 7. Desarrollo de la Aplicación Web).	https://miro.com/es/
Jira Software	Software en línea que se utilizó para la administración y seguimiento de las historias de usuario (ver el punto 6.2.1 de la sección de Resultados).	https://www.atlassian.com/es/software/jira
Zoom	Videochat que permitió hacer reuniones, revisiones del Trabajo de Titulación y entrevistas.	https://zoom.us
Node.js	Entorno de ejecución para JavaScript de lado del servidor, el cual sirvió para desarrollar la API de la aplicación web.	https://nodejs.org/es
React.js	Biblioteca de JavaScript que se utilizó para construir la interfaz de usuario en el front-end.	https://es.reactjs.org/
Visual Studio Code	Editor de código que fue utilizado para la codificación de la aplicación web.	https://code.visualstudio.com
Git	Se utilizó Git para el control de versiones del software de la aplicación, además de seguir el modelo GitFlow.	https://git-scm.com/
GitHub	Se utilizó la plataforma GitHub para alojar los proyectos (API y el Front-end).	https://github.com
GitKraken	Es un software que provee un GUI de Git intuitiva y un CLI de Git, el cual se utilizó para gestionar GitFlow y realizar comandos de Git.	https://www.gitkraken.com
sweetalert2	Librería de mensajes de alerta en la interfaz de usuario, el cual se instaló en el proyecto de front-end y utilizarlo dentro de los componentes de React.js.	https://sweetalert2.github.io
Jest	Es un marco de pruebas para JavaScript y se utilizó para ejecutar las pruebas unitarias del back-end y el front-end.	https://jestjs.io
MongoDB Atlas	Es un servicio en la nube que se utilizó para alojar la base de datos de MongoDB.	https://www.mongodb.com/es

Render Hosting	Servicio de computación en la nube que se utilizó para desplegar los servicios de la API.	https://render.com/
Vercel	Servicio de computación en la nube que se utilizó para desplegar la interfaz de usuario (React).	https://vercel.com/
Cloudinary	Servicio de gestión de imágenes basados en la nube. Se utilizó para alojar las imágenes y los gifs de la aplicación web.	https://cloudinary.com/

7.2.2. Valoración económica

En el desarrollo del Trabajo de Titulación, se invirtió en: talento humano, recursos de hardware y software, servicios e imprevistos.

En la Tabla 30 se muestra el costo invertido en talento humano. La asesoría del docente como tutor, es asumida por la Universidad Nacional de Loja.

Tabla 30.
Valoración económica de talento humano

Talento humano	Número de horas	Valor por hora	Subtotal
Investigador del Trabajo de Titulación	400	\$ 8,00	\$ 3.200,00
Director del proyecto	72	-	-
Subtotal			\$ 3.200,00
Imprevistos (10% del subtotal)			\$ 320,00
Total			\$ 3.520,00

Los recursos de hardware y software que fueron utilizados para el desarrollo de la aplicación web, se presentan en la Tabla 31.

Tabla 31.
Valoración económica de recursos de hardware y software

Recursos	Número de meses	Valor por	Cantidad	Subtotal
Laptop personal	5	\$ 30,00	1	\$ 150
Vercel	2	\$ -	1	\$ -
Render Hosting	2	\$ -	1	\$ -
MongoDB Atlas	2	\$ -	1	\$ -
Cloudinary	3	\$ -	1	\$ -
Miro Software	4	\$ -	1	\$ -
Jira Software	4	\$ -	1	\$ -
Postman	4	\$ -	1	\$ -
Git	4	\$ -	1	\$ -
GitHub	4	\$ -	1	\$ -
GitKraken	4	\$ -	1	\$ -
Visual Paradigm Community	4	\$ -	1	\$ -
Adobe XD	1	\$ -	1	\$ -
Zoom	5	\$ -	1	\$ -

Visual Studio Code	4	\$ -	1	\$ -
Lenguaje de programación	4	\$ -	1	\$ -
MongoDB	4	\$ -	1	\$ -
Subtotal				\$ 150
Imprevistos (10% del subtotal)				\$ 15
Total				\$ 165

El valor económico invertido en los servicios para el desarrollo del Trabajo de Titulación, se muestra en la Tabla 32.

Tabla 32.
Valoración económica de servicios

Servicios	Número de meses	Valor por	Cantidad	Subtotal
Internet	5	\$ 20,00	1	\$ 100
Subtotal				\$ 100
Imprevistos (10% del subtotal)				\$ 10
Total				\$ 110

El presupuesto final del desarrollo del Trabajo de Titulación es de \$3.795,00. En la Tabla 33 se presentan los detalles. Cabe recordar que el presupuesto total del TT, representa a un entorno académico.

Tabla 33.
Presupuesto final

Servicios	Subtotal
Talento humano	\$ 3.520,00
Recursos de hardware y software	\$ 165,00
Servicios	\$ 110,00
Total	\$ 3.795,00

8. Conclusiones

Esta sección describe los eventos más relevantes encontrados en la ejecución del TT:

- El uso del Patrón de Diseño Factory en la gestión de los controladores de la API permite crear objetos desde la fábrica, haciendo que el código sea fácil de comprender y de implementar, facilitando así la creación de la aplicación web; además el código puede ser modificado ante posibles cambios en los requerimientos.
- El uso del estudio de documentación como técnica de elicitación de requerimientos ayudó a definir el diseño de las cápsulas y tarjetas de aprendizaje para la validación del micro-learning en la aplicación desarrollada.
- La ejecución de la metodología XP permitió llevar a cabo un proceso metódico y adaptable del desarrollo de la aplicación web mediante sus cuatro fases: planificación, diseño, codificación y pruebas; cumpliendo a cabalidad con las 21 historias de usuario planteadas en la especificación de requerimientos.
- El método del micro-learning proporciona a la aplicación un entorno favorable y didáctico para aprender lengua de señas ecuatoriana por medio de imágenes animadas en formato GIF, corroborando su aplicabilidad mediante las pruebas de aceptación de los usuarios finales con un promedio del 71,19%.
- Las tarjetas de aprendizaje es un material multimedia para medir el nivel de vocabulario de lengua de señas ecuatoriana, logrando una aceptación del 79,2% por parte del usuario final.

9. Recomendaciones

Después de completar el TT, las recomendaciones son:

- Utilizar el estudio de documentación como una técnica para obtener requerimientos de usuario, ya que proporciona una visión general del proyecto a partir de trabajos relacionados y por medio de wireframes extraer componentes de interfaz de usuario.
- Para el desarrollo del front-end con React, basar la programación de acuerdo a un prototipo de interfaz de usuario.
- Utilizar la técnica de la paginación con MongoDB para devolver grandes cantidades de datos y no tener un retardo en la interfaz de usuario.
- Optimizar los GIF antes de almacenar en la base de datos para que al renderizarlo no se demore en cargar en la interfaz de usuario.
- Para la planificación de los contenidos educativos, en el caso de la lengua de señas ecuatoriana, enfocarse en incluir temas básicos por motivos de que la FENASEC es la única autorizada en realizar cursos de manera profesional.
- Dentro de los contenidos educativos de lengua de señas, incluir temas de salud, emergencia, seguridad y frases diarias.
- El recuadro de la persona que realiza la Seña, sea más grande y claro, para su ejercicio y comprensión.
- La imagen representativa de la seña debe ser clara y tener relación con la seña.

Se recomienda para trabajos a futuros:

- Desarrollar una aplicación móvil, que consuma los datos de la API desarrollada en el presente Trabajo de Titulación.
- Implementar los videos como medio visual para presentar las señas.
- Implementar más alternativas de autenticación de usuarios a través de servicios como Google o Facebook, mejorando así la experiencia de usuario.

10. Bibliografía

Referencias Bibliográficas

- [1] S. Meresman and H. Ullmann, "COVID-19 y las personas con discapacidad en América Latina: mitigar el impacto y proteger derechos para asegurar la inclusión hoy y mañana," *CEPAL*, vol. Políticas, no. 237, 2020.
- [2] RAE, "Definición de lengua de signos - Diccionario panhispánico del español jurídico - RAE." [Online]. Available: <https://dpej.rae.es/lema/lengua-de-signos>. [Accessed: 29-Aug-2020].
- [3] R. S. Pressman, *Ingeniería del Software. Un Enfoque Practico*, 7th ed. 2010.
- [4] M. Montaña, "Capacitación en Lengua de Señas a los estudiantes de Ciencias de la Educación de la Pontificia Universidad Católica del Ecuador, sede Esmeraldas," Pontificia Universidad Católica del Ecuador, 2014.
- [5] J. Varela-Barraza, C. Huerta, and O. Tello, "Apuntes para una Teoría de la Lengua de Señas," *Conductual*, vol. 5, no. 2, 2017.
- [6] J. R. Álava Jiménez, "Sonidos en Silencio: Comunicación No verbal para Facilitar el Proceso de Enseñanza-aprendizaje de los Jóvenes con Discapacidad Auditiva el Ecuador," Universidad Central del Ecuador, 2014.
- [7] K. Nasevilla, "Aportes lingüísticos para la sistematización de la Lengua de Señas de Quito," Pontificia Universidad Católica Del Ecuador, 2015.
- [8] Á. D. Serrano Morán, "Evaluación de la presencia de la lengua de señas en los noticieros de los canales de televisión ecuatoriana," Pontificia Universidad Católica Del Ecuador, 2020.
- [9] D. Eberle, E. Parks, S. Eberle, and J. Parks, "Sociolinguistic Survey Report of the Ecuadorian Deaf Community," no. August, p. 37, 2012.
- [10] I. Fajardo Bravo, J. Cañas, A. Antolí, and L. Salmerón, "Accesibilidad Cognitiva de los Sordos a la Web," 2002.
- [11] J. Salinas Ibáñez and V. Marín, "Pasado, presente y futuro del microlearning como estrategia para el desarrollo profesional," *Campus Virtuales*, vol. 3, no. 2, pp. 46–61, 2014.
- [12] B. Göschlberger and P. A. Bruck, "Gamification in mobile and workplace integrated MicroLearning," *ACM Int. Conf. Proceeding Ser.*, no. December 2017, pp. 545–552, 2017, doi: 10.1145/3151759.3151795.

- [13] S. Liao and C. Zhu, "Micro-learning based on social networking," *Proc. 2nd Int. Conf. Comput. Sci. Netw. Technol. ICCSNT 2012*, pp. 1163–1166, 2012, doi: 10.1109/ICCSNT.2012.6526131.
- [14] G. Gassler, T. Hug, and C. Glahn, "Integrated Micro Learning - An outline of the basic method and first results," *Interact. Comput. Aided Learn.*, no. January 2004, pp. 1–7, 2004.
- [15] D. Gomez, "Propuesta metodológica para la creación de cápsulas de aprendizaje para adultos mayores," Universidad de Cuenca, 2021.
- [16] G. Erich, H. Richard, J. Ralph, and V. John, *Patrones de Diseño*, vol. 09. 2007.
- [17] J. R. Molina Ríos, M. P. Zea Ordóñez, M. J. Contenido Segarra, and F. G. García Zerda, "Comparación de Metodologías en Aplicaciones Web," *3C Tecnol.*, vol. 7, no. 1, Mar. 2018, doi: 10.17993/3ctecno.2018.v7n1e25.1-19.
- [18] R. Figueroa-Díaz, C. Sólis, and A. Cabrera, "Metodologías tradicionales vs. metodologías ágiles." 2007, doi: 10.13140/RG.2.1.2897.3206.
- [19] M. B. Molina, C. H. Vite, and D. J. Cuesta, "Metodologías ágiles frente a las tradicionales en el proceso de desarrollo de software," *Espirales Rev. Multidiscip. Investig. ISSN 2550-6862*, vol. 2, no. 17, pp. 114–121, 2018, doi: 10.31876/re.v2i17.269.
- [20] S. M. Velásquez Restrepo, J. D. Vahos Montoya, M. E. Gómez Adasme, E. J. Restrepo Zapata, A. A. Pino Martínez, and S. Londoño Marín, "Una revisión comparativa de la literatura acerca de metodologías tradicionales y modernas de desarrollo de software," *Rev. CINTEX*, vol. 24, no. 2, pp. 13–23, 2019, doi: 10.33131/24222208.334.
- [21] H. R. González, "Análisis y diseño de un modelo con integración de una metodología ágil en el nivel 2 de CMMI," Bogotá, 2017.
- [22] C. Boaventura José, E. Peña Herrera, P. Verdecia Vicet, and Y. Fustiel Alvarez, "Elección entre una metodología ágil y tradicional basado en técnicas de soft computing," *Rev. Cuba. Ciencias Informáticas*, vol. 10, pp. 145–158, 2016.
- [23] X. Quiñónez-Ku Cavada, J. Casierra Cavada, L. Herrera-Izquierdo, and J. Mera Quiroz, "Análisis Comparativo de metodologías Ágiles de desarrollo de software : una revisión bibliográfica Comparative analysis of agile software development methodologies : a bibliographic review," *ResearchGate*, no. June, 2019.
- [24] A. López Gil, "Estudio comparativo de metodologías tradicionales y ágiles para proyectos de Desarrollo de Software," p. 139, 2018.

- [25] J. C. Salazar, Á. Tovar, J. C. Linares, A. Lozano, and L. Valbuena, "Scrum versus XP: similitudes y diferencias," *Tecnol. Investig. y Acad.*, vol. 6, no. 2, pp. 29–37, 2018.
- [26] I. Sommerville, *Ingeniería de Software*, 9th ed. 2011.
- [27] W. Khan, W. Ahmad, B. Luo, and E. Ahmed, "SQL database with physical database tuning technique and NoSQL graph database comparisons," *Proc. 2019 IEEE 3rd Inf. Technol. Networking, Electron. Autom. Control Conf. ITNEC 2019*, no. Itnec, pp. 110–116, 2019, doi: 10.1109/ITNEC.2019.8729264.
- [28] K. Sahatqija, J. Ajdari, X. Zenuni, B. Raufi, and F. Ismaili, "Comparison between relational and NOSQL databases," *2018 41st Int. Conv. Inf. Commun. Technol. Electron. Microelectron. MIPRO 2018 - Proc.*, pp. 216–221, 2018, doi: 10.23919/MIPRO.2018.8400041.
- [29] A. Gupta, S. Tyagi, N. Panwar, S. Sachdeva, and U. Saxena, "NoSQL databases: Critical analysis and comparison," *2017 Int. Conf. Comput. Commun. Technol. Smart Nation, IC3TSN 2017*, vol. 2017-October, pp. 293–299, 2018, doi: 10.1109/IC3TSN.2017.8284494.
- [30] E. Tang and Y. Fan, "Performance comparison between five NoSQL databases," *Proc. - 2016 7th Int. Conf. Cloud Comput. Big Data, CCBBD 2016*, pp. 105–109, 2017, doi: 10.1109/CCBD.2016.030.
- [31] S. Wang, L. Zhu, and M. Cheng, "Docker-based web server instructional system," *Proc. - 18th IEEE/ACIS Int. Conf. Comput. Inf. Sci. ICIS 2019*, pp. 285–289, 2019, doi: 10.1109/ICIS46139.2019.8940219.
- [32] D. Laksono, "Testing Spatial Data Deliverance in SQL and NoSQL Database Using NodeJS Fullstack Web App," *Proc. - 2018 4th Int. Conf. Sci. Technol. ICST 2018*, vol. 1, pp. 1–5, 2018, doi: 10.1109/ICSTC.2018.8528705.
- [33] L. Liang, L. Zhu, W. Shang, D. Feng, and Z. Xiao, "Express supervision system based on NodeJS and MongoDB," *Proc. - 16th IEEE/ACIS Int. Conf. Comput. Inf. Sci. ICIS 2017*, pp. 607–612, 2017, doi: 10.1109/ICIS.2017.7960064.
- [34] A. Soni and V. Ranga, "API features individualizing of web services: REST and SOAP," *Int. J. Innov. Technol. Explor. Eng.*, vol. 8, no. 9 Special Issue, pp. 664–671, 2019, doi: 10.35940/ijitee.I1107.0789S19.
- [35] S. Plaza Estévez, N. Ramírez Lamela, and C. Acosta Morales, "API de servicios web orientados a accesibilidad," Universidad Complutense de Madrid, 2016.

- [36] F. Doglio, *REST API Development with Node.js*, Second. La Paz, 2018.
- [37] Stack Overflow, "Stack Overflow Trends," 2022. [Online]. Available: <https://insights.stackoverflow.com/trends?tags=json%2Cxml>.
- [38] Y. Balaj, "Token-Based vs Session-Based Authentication : A survey," no. September, pp. 1–6, 2017.
- [39] A. Rahmatulloh, R. Gunawan, and F. Nursuwars, "Performance comparison of signed algorithms on JSON Web Token," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 550, no. 1, 2019, doi: 10.1088/1757-899X/550/1/012023.
- [40] M. Hidayat, R. Dimas, and A. Siswanto, "Design of Restaurant Billing System (E Bill Resto) by Applying Synchronization of Data Billing in Branch Companies to Main Companies Based on Rest API," *Proceeding - ICoSTA 2020 2020 Int. Conf. Smart Technol. Appl. Empower. Ind. IoT by Implement. Green Technol. Sustain. Dev.*, 2020, doi: 10.1109/ICoSTA48221.2020.1570615039.
- [41] E. Sevilla, "Aplicación web para la enseñanza del lenguaje de señas, módulo básico para niños del segundo año de educación básica de la unidad educativa de sordos del gobierno provincial de Imbabura," Universidad Técnica del Norte, Ibarra, 2015.
- [42] T. Pérez, "Desarrollo de un portal web para la enseñanza-aprendizaje del lenguaje de señas en la ciudad de Guayaquil," Universidad de Guayaquil, Guayaquil, 2014.
- [43] S. A. Mokhtar and S. A. Siti Mashitah, "Learning Application for Malaysian Sign Language: Content Design, User Interface and Usability," in *Proceedings of the 9th International Conference on Ubiquitous Information Management and Communication*, 2015, doi: 10.1145/2701126.2701142.
- [44] S. A. Mokhtar, S. A. Siti Sarah, and S. A. Siti Mashitah, "Web-Based Application for Learning Malaysian Sign Language," in *Proceedings of the 11th International Conference on Ubiquitous Information Management and Communication*, 2017, doi: 10.1145/3022227.3022235.
- [45] S. F. Saud and Z. A. Nasruddin, "Design of e-learning courseware for hearing impaired (HI) students," in *2016 4th International Conference on User Science and Engineering (i-USER)*, 2016, pp. 271–276, doi: 10.1109/IUSER.2016.7857973.
- [46] E. Cabezas, D. Andrade, and J. Torres, *Introducción a la Metodología de la Investigación Científica*, Primera. Universidad de las Fuerzas Armadas ESPE, 2018.
- [47] T. Otzen, C. Manterola, I. R. Núñez, and M. García-Domínguez, "The scientific method.

- Its relevance in conducting clinical research,” *Int. J. Morphol.*, vol. 35, no. 3, pp. 1031–1036, 2017, doi: 10.4067/S0717-95022017000300035.
- [48] M. G. Bocco, J. A. Cruz Lemus, and M. G. Piattini Velthuis, *Métodos de investigación en ingeniería del software*. Madrid: RA-MA, S. A, 2014.
- [49] G. E. Barchini, “Métodos ‘I + D’ de la Informática,” *Rev. Informática Educ. y Medios Audiovisuales*, vol. 2, no. 5, pp. 16–24, 2005.
- [50] J. Ohene-Djan, R. Zimmer, M. Gorle, and S. Naqvi, “A personalisable electronic book for video-based sign language education,” *Educ. Technol. Soc.*, vol. 6, no. 4, pp. 86–99, 2003.
- [51] J. Ohene-Djan and S. Naqvi, “An adaptive WWW-based system to teach British Sign Language,” in *Proceedings - 5th IEEE International Conference on Advanced Learning Technologies, ICALT 2005*, 2005, vol. 2005, pp. 127–129, doi: 10.1109/ICALT.2005.44.
- [52] E. Efthimiou and S. E. Fotinea, “An environment for deaf accessibility to educational content,” in *New Trends in ICT and Accessibility - Proceedings of the 1st International Conference in Information and Communication Technology and Accessibility, ICTA 2007*, 2007, pp. 125–130.
- [53] W. Mann, P. Roy, and G. Morgan, “Adaptation of a vocabulary test from British Sign Language to American Sign Language,” *Lang. Test.*, vol. 33, no. 1, pp. 3–22, 2016, doi: 10.1177/0265532215575627.
- [54] A. Miranda *et al.*, “Bilingual alphabetisation of deaf children and progress assessment: A prototype of a web system based on a phonic methodology,” in *Proceedings of the IADIS International Conference WWW/Internet 2010*, 2010, pp. 350–354.
- [55] J. P. López *et al.*, “Design and development of sign language questionnaires based on video and web interfaces,” *Artif. Intell. Eng. Des. Anal. Manuf. AIEDAM*, vol. 33, no. 4, pp. 429–441, 2019, doi: 10.1017/S0890060419000374.

11. Anexos

Anexo 1. Elicitación de Requerimientos

Este documento presenta el proceso de extracción de requerimientos. Luego analiza la recopilación de requerimientos a través de técnicas de entrevista, análisis y validación.

1. Entrevista

La entrevista se aplicó al Sr. Luis Alberto Quichimbo intérprete de la Lengua de Señas Ecuatoriana (LSEC) y miembro de la Asociación de Personas Sordas “Virgen del Cisne”, y al Sr. Darwin Quichimbo presidente de la Asociación de Personas Sordas “Virgen del Cisne” y miembro de la Federación Nacional de Personas Sordas del Ecuador (FENASEC). La entrevista se realizó en tres sesiones, la primera a través de la aplicación de Google Meet para hablar sobre la enseñanza de la LSEC y las otras dos a través de la aplicación Zoom, en donde se presentó un prototipo para la validación de requerimientos.

El objetivo de la entrevista es obtener información sobre el proceso de enseñanza de la lengua de señas, metodología de enseñanza dentro de la Asociación y materiales.

En la primera entrevista se habló sobre los problemas que la comunidad sorda tuvo durante la pandemia del COVID-19, en la cual el Sr. Darwin Quichimbo quien es una persona con discapacidad auditiva, manifestó que los problemas ya vienen desde antes de la pandemia; a las personas sordas se les dificulta comprar en tiendas, farmacias, recibir atención médica y recibir clases. Por ello necesitan que las personas ecuatorianas se inscriban a los cursos de lengua de señas que imparte la asociación, con el fin aumentar el número de hablantes en LSEC. Las personas que más se inscriben en los cursos son niños y adolescentes, y el total de estudiantes que se inscribieron en el último curso realizado hasta la fecha de la entrevista, fueron ocho estudiantes de 6 a 12 años de edad. Además, manifestó que, si se implementara una aplicación que enseñe lo básico de la lengua de señas a las personas sin discapacidad auditiva, ayudaría a que más personas se interesen en aprender este lindo lenguaje y que promulgaría a seguir cursos que la asociación imparte.

El link para tener acceso a la entrevista completa es: <https://bit.ly/3yD4jZ9>

Con la primera entrevista se determinó que la aplicación del presente Trabajo de Titulación estará enfocada a enseñar lengua de señas de forma básica a personas sin discapacidad de 18 a 30 años de edad, con la finalidad de motivar e incrementar el número de hablantes de LSEC y probar la aceptación del micro-learning como método de enseñanza (en trabajos futuros se espera que la aplicación desarrollada amplie el grupo de personas y que también

sirva para personas con discapacidad auditiva que no tengan acceso a los cursos de Lengua de Señas).

En la segunda entrevista se habló sobre los métodos de enseñanza de lengua de señas para obtener los requerimientos de la aplicación. A continuación, se presentan las preguntas de la entrevista y las respuestas resumidas.

Link para ver la entrevista completa: <https://bit.ly/3g7AWYR>

1.1. (EP01) ¿Cuál es el proceso de enseñanza de la lengua de señas?

La enseñanza de la lengua de señas se lo da a través de cursos que organiza la FENASEC, estos se lo realizan de forma presencial. Sólo la FENASEC tiene la autorización de impartir los cursos.

Primero se hace la inscripción de participantes (estudiantes), las personas con discapacidad auditiva tienen el derecho de recibir el curso de forma gratuita, mientras que las personas sin discapacidad deben pagar por el curso correspondiente.

Una vez que se hayan inscrito los estudiantes, la FENASEC envía módulos a los participantes; conforme se va avanzando el curso, se imparten cuatro módulos, que parte desde un nivel básico, mediano y superior. Y Al finalizar el curso se los otorga un certificado de aprobación.

1.2. (EP02) ¿Qué metodologías se usar para enseñar la lengua de señas?

Primero se enseña lo básico de la lengua de señas como, por ejemplo, el alfabeto, los números, la familia, verbos. Después de tener los conceptos fundamentales, se los enseña estructurar oraciones y texto. Como enseñanza final se enseña desarrollar ideas y diálogos en lengua de señas.

Se proyecta cada uno de las señas en donde se muestra la palabra, la seña y una imagen representativa del objeto. Después de cada proyección se realiza prácticas.

En la pandemia del COVID-19 se realiza los cursos virtualmente a través de la herramienta zoom y se realiza la misma práctica que de forma presencial.

1.3. (EP03) ¿Qué tipo de materiales utilizan para enseñar lengua de señas?

En el curso desarrollado de forma presencial, se utiliza los siguientes materiales:

- módulos (libros),
- proyecciones (diapositivas),
- videos,
- imágenes.

En la forma virtual se utiliza la herramienta de zoom en donde se proyecta de la misma manera proyecciones y el profesor indica las señas.

1.4. (EP04) De los materiales utilizados para enseñar lengua de señas. ¿Cuál resulta más efectiva para que las personas aprendan más rápido?

Los videos ayudan a visualizar de mejor forma el patrón de la seña, por lo que resulta más efectivo.

Las imágenes estáticas son de ayuda siempre y cuando se tenga un guía o especialista en la lengua de señas para que verifique si el estudiante ejecuta de buena manera la seña.

1.5. (EP05) ¿Existe algunos problemas o dificultades al enseñar lengua de señas?

El problema más habitual que el estudiante tiene, es la dificultad de ejecutar la seña y esto se lo soluciona a través de la práctica.

En el caso de los cursos virtuales, la latencia de red es un problema porque se pierde la fluidez al ejecutar la seña y se tiene que repetir reiteradas veces.

1.6. (EP06) ¿Tomaría en cuenta el microaprendizaje (Microlearning) como una metodología o herramienta para enseñar lengua de señas?

El uso del microlearning sería una buena herramienta de apoyo para enseñar lengua de señas, por lo cual debería ser aplicada por los profesores de la lengua de señas para seguir el mismo proceso, pero de una forma ya digital. Además de contribuir con la motivación de seguir aprendiendo lengua de señas.

1.7. (EP07) ¿Qué le parece si pudiéramos desarrollar una aplicación web basada en el Microlearning para enseñar lengua de señas y que importancia le daría usted a esta aplicación?

Los entrevistados supieron manifestar que la idea parece muy interesante porque el mundo está cada vez digitalizándose. Esta aplicación debe ser utilizado por los profesores de la lengua de señas porque sería una herramienta de gran apoyo.

Se debe tener en cuenta que la enseñanza de la lengua de señas sólo lo realiza la FENASEC, por lo que la aplicación debe ser utilizada por ellos. Por ello si todo el contenido que se enseña en los cursos está presente de la aplicación, debería incluir una funcionalidad para verificar si el usuario (estudiante) haya cancelado el pago correspondiente. Pero si se quiere hacer de forma gratuita el contenido, esto debe presentar contenido básico para motivar a las personas aprender más de la lengua de señas.

1.8. (EP08) ¿Cuáles son sus expectativas sobre la confiabilidad de la aplicación web?

Los chicos tienen que actualizarse y utilizar la tecnología, para que las personas tengan acceso.

1.9. (EP09) ¿Cuáles son sus expectativas sobre la capacidad(rendimiento) de la aplicación?

Al mostrar un prototipo de la aplicación web (ver Anexo 3: Prototipo de interfaz de usuario), los entrevistados manifestaron que sería una buena herramienta para los profesores y contribuir en la enseñanza de la lengua de señas.

2. Lista inicial de requerimientos

Con los datos obtenidos de la entrevista y el estudio de documentación, se obtiene una lista inicial de requerimientos, los cuales se presentan en la Tabla 34.

Tabla 34.
Lista inicial de requerimientos.

Número	Requerimientos	Solicitado/ Inferido	Pregunta
01	El usuario podrá crear una cuenta con: nombre, correo electrónico y contraseña	Solicitado	EP07
02	El usuario deberá identificarse para poder acceder a la aplicación	Solicitado	EP07
03	El administrador podrá crear módulos y lecciones	Inferido	EP02
04	El administrador podrá crear ejercicios de vocabulario por medio de selección múltiple	Inferido	EP01
05	El administrador podrá consultar los avances del usuario en la aplicación web	Solicitado	EP05
06	El administrador podrá consultar los avances por el nombre del usuario o correo electrónico	Solicitado	EP05
07	El administrador podrá buscar usuarios	Solicitado	EP01
08	El administrador podrá gestionar lecciones (crear, modificar, buscar y eliminar)	Inferido	EP06
09	El administrador podrá gestionar módulos (crear, modificar, buscar y eliminar)	Inferido	EP06
10	El administrador podrá ingresar imágenes o gifs en las lecciones.	Inferido	EP04
11	El funcionamiento de la aplicación web deberá de sincronizar los datos evitando retardos.	Inferido	EP08

12	El sistema ofrecerá información guía para el manejo del sistema	Inferido	EP08
13	El usuario podrá cerrar sesión	Inferido	EP01
14	Las lecciones deben contener imagen representativa, gif o video, palabra	Inferido	EP04

3. Matriz de iteración

La matriz de iteraciones evalúa los requerimientos e identifica conflictos o superposiciones. La matriz representa dos entradas, cada una de las cuales contiene todos los requerimientos y, por lo tanto, están relacionadas entre sí. • Superposición: Hay redundancia cuando se maneja el mismo aspecto del sistema entre r-1 y r-2. Representado por la letra S. • Conflicto: Si hay un conflicto entre r-1 y r-2, tendrá un problema de consistencia interna. Se representa con la letra C.

En la Tabla 35 se presenta la matriz de iteración de los requerimientos iniciales.

Tabla 35.
Matriz de iteración

R	01	02	03	04	05	06	07	08	09	10	11	12	13	14
01														
02														
03								S	S					
04														
05						S								
06					S		S							
07						S								
08			S											
09			S											
10														S
11														
12														
13														
14										S				

La Tabla 36 presenta los conflictos encontrados y a partir del análisis, se obtienen nuevos requerimientos.

Tabla 36.
Superposiciones y conflictos

N°	Requerimiento	Error	Descripción	Requerimiento final
1	03, 08	S	El requerimiento 08 detalla al requerimiento 03	El administrador podrá gestionar lecciones (crear, consultar, modificar, buscar y eliminar)
2	03, 09	S	El requerimiento 09 detalla al requerimiento 03	El administrador podrá gestionar módulos (crear, modificar, buscar y eliminar)
3	05, 06	S	El requerimiento 06 tiene operaciones complementarias al requerimiento 05	El administrador podrá consultar los avances del usuario en la aplicación web por medio del nombre de usuario o correo electrónico
4	06, 07	S	El requerimiento 06 tiene operaciones complementarias al requerimiento 07	El administrador podrá realizar la búsqueda de usuarios por medio de nombre de usuario o correo electrónico
5	10, 14	S	El requerimiento 14 tiene operaciones complementarias al requerimiento 10	El administrador debe ingresar en las lecciones una imagen representativa al tema, video o gif y la palabra descriptora

4. Lista final de requerimientos

Una vez solucionado los conflictos, la Tabla 37 presenta los requerimientos finales para el desarrollo de la aplicación web.

Tabla 37.
Lista final de requerimientos

Código	Requerimientos
01	Crear cuenta de usuario
02	Autenticación de usuario
03	Cerrar sesión de usuario
04	Gestión de módulos

05	Gestión de lecciones
06	Gestión de tarjetas de aprendizaje (ejercicios)
07	Visualizar avance de lecciones por parte del estudiante
08	Gestión de cápsulas de aprendizaje (micro contenido)

5. Anexo



Figura 32. Entrevista vía online.



Figura 33. Entrevista para los requerimientos

Especificación de Requerimientos de Software IEEE 830

Proyecto: Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano

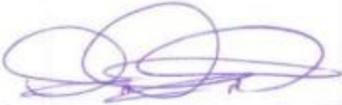
Versión: 3.0.0

Fecha: 12/06/2022

Ficha de documento

Fecha	Revisión	Autor	Verificado y Validado
24/06/2021	1.0.0	Jonnathan Espinoza	Ing. Edison Coronel Romero
08/07/2021	2.0.0	Jonnathan Espinoza	Ing. Edison Coronel Romero
13/06/2022	3.0.0	Jonnathan Espinoza	Ing. Edison Coronel Romero

Documento validado por las partes en fecha:

Por el cliente	Por el Director de Trabajo de Titulación
 Darwin Alfredo Jumbo Quichimbo Presidente de la Asociación de Personas Sordas "Virgen del Cisne"	Ing. Edison Leonardo Coronel Romero Director de Trabajo de Titulación

1. Introducción

Este documento es una Especificación de Requerimientos de Software (ERS) para el diseño de aplicaciones web utilizando el patrón de diseño Fábrica de micro-learning de Lengua de Señas Ecuatoriana. Esta especificación se construye en base a las pautas del estándar IEEE 830.

1.1. Propósito

El propósito de este documento es definir especificaciones funcionales y no funcionales para el diseño de aplicaciones web utilizando el patrón de diseño Fábrica de micro-learning de Lengua de Señas Ecuatoriana.

1.2. Alcance

Esta solución informática tiene como objetivo diseñar una aplicación web para la enseñanza de la lengua de signos a través del micro aprendizaje. Por lo tanto, los requisitos están dirigidos a los usuarios del sistema para continuar con el desarrollo de la aplicación y ocuparse de su automatización.

1.3. Personal involucrado

Tabla 38.
Desarrollador de la aplicación

Nombre	Jonnathan Espinoza
Rol	Programador y analista
Categoría Profesional	Estudiante
Responsabilidad	Desarrollo del software y analista de la solución informática.
Información de contacto	jonnathan.espinoza@unl.edu.ec

Tabla 39.
Director del Trabajo de Titulación

Nombre	Edison Coronel
Rol	Director del Trabajo de Titulación
Categoría Profesional	Ingeniero en Sistemas
Responsabilidad	Asesorar el Trabajo de Titulación
Información de contacto	edison.coronel@unl.edu.ec

1.4. Definiciones, acrónimos y abreviaturas

Tabla 40.
Definiciones, acrónimos y abreviaturas

Nombre	Descripción
Usuario	Persona que hace uso de la aplicación web para el aprendizaje de la Lengua de Señas Ecuatoriana

Administrador	Perona que administrará y gestionará la aplicación web
ERS	Especificación de Requerimientos Software
RF	Requerimiento Funcional
RNF	Requerimiento No Funcional
HTTP	Hypertext Transfer Protocol

1.5. Referencias

Tabla 41.
Referencias ERS

Título del Documento	Referencia
Standard IEEE 830 - 1998	IEEE

1.6. Resumen

Este documento consta de tres secciones. La primera sección lo presenta y proporciona una descripción general de las especificaciones de recursos del sistema.

En la segunda sección del documento, con el fin de comprender, sin entrar en demasiados detalles, las principales funciones que debe realizar el sistema, los datos involucrados y los factores, restricciones, suposiciones y dependencias que influyen en él.

Finalmente, la tercera sección del documento define en detalle los requerimientos que debe cumplir el sistema.

2. Descripción general

2.1. Perspectiva de la aplicación

La solución informática está diseñada para funcionar en un entorno web, lo que la hace rápida y eficiente de usar.

2.2. Funcionalidad de la aplicación

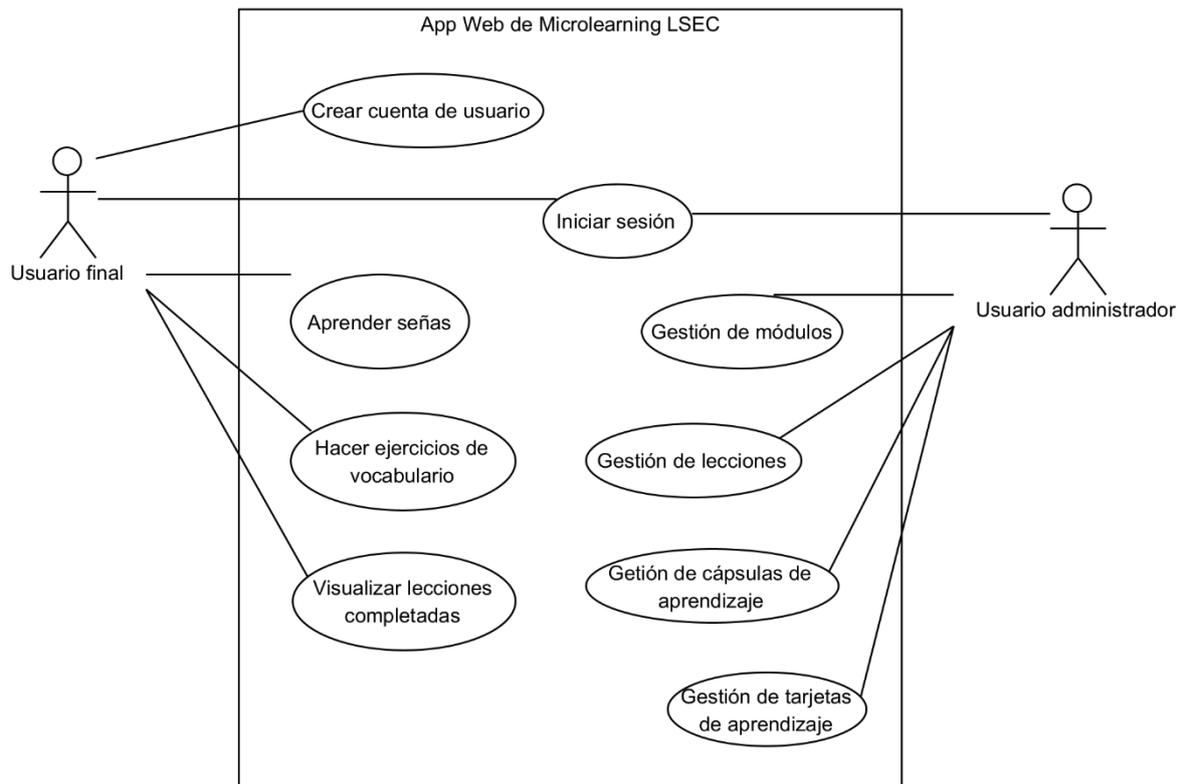


Figura 34. Casos de uso.

2.3. Características de usuarios

Tabla 42.
Características del usuario final

Tipo de usuario	Usuario final
Formación	N/A
Actividades	<ul style="list-style-type: none"> • Crear una cuenta en la aplicación web. • Ingresar a las cápsulas de aprendizaje. • Practicar con las tarjetas de conocimiento. • Ver las lecciones completadas.

Tabla 43.
Características del usuario administrador

Tipo de usuario	Usuario administrador
Formación	N/A
Actividades	Crear, modificar y eliminar el contenido (cápsulas, tarjetas, módulos y lecciones) de la aplicación web.

2.4. Restricciones de la aplicación

- El software se desarrollará basándose en la metodología XP.
- El software deberá tener un diseño simple.

- Se creará una API.
- Tecnología back-end: JavaScript, Nodejs, Express, MongoDB.
- Tecnología front-end: React JS, CSS, Bootstrap.

2.5. Suposiciones y dependencias de la aplicación

- Se supone que los requisitos descritos en este documento son estables.
- La computadora que ejecuta el sistema debe cumplir con los requisitos anteriores para una operación adecuada.

3. Requerimientos específicos

3.1. Requerimientos comunes de las interfaces

3.1.1. Interfaces de usuario

La interfaz de usuario consta de una serie de ventanas que contienen botones, listas, cuadros de texto, imágenes, gifs, etc. Esto debe escribirse especialmente para el sistema propuesto y lo muestra el navegador de Internet.

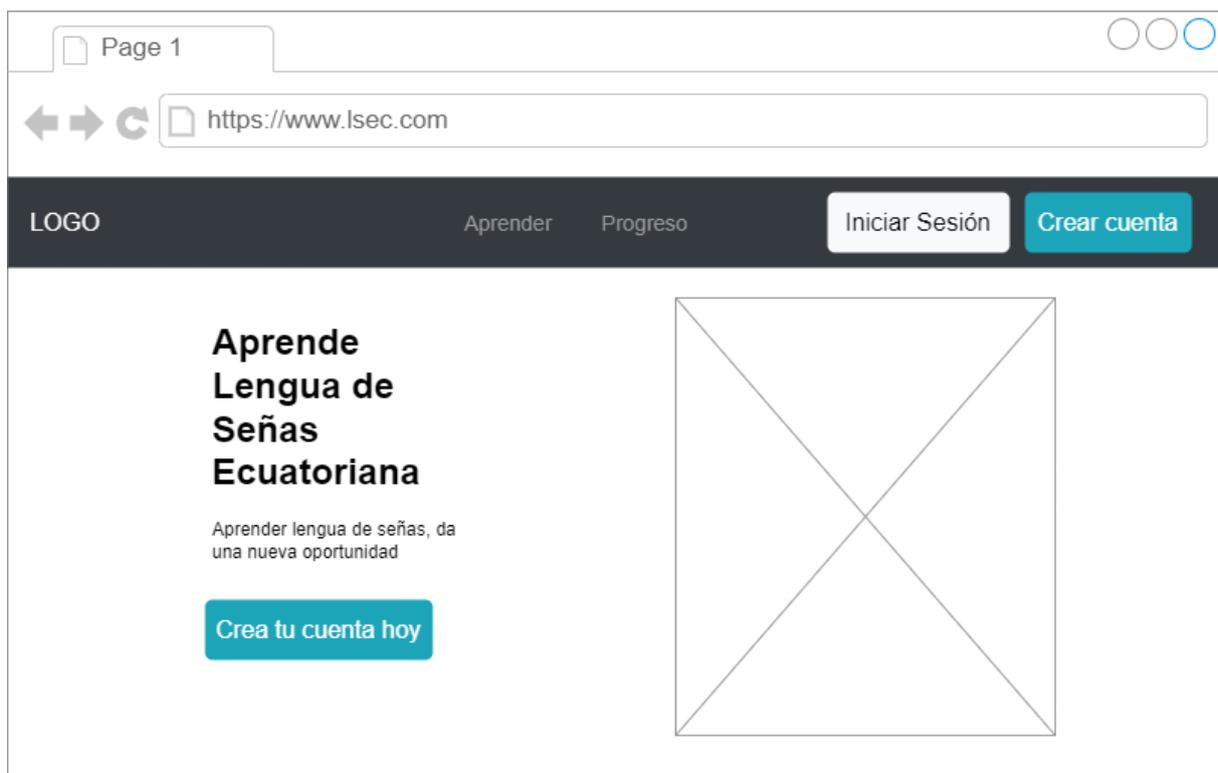


Figura 35. Wireframe página principal.

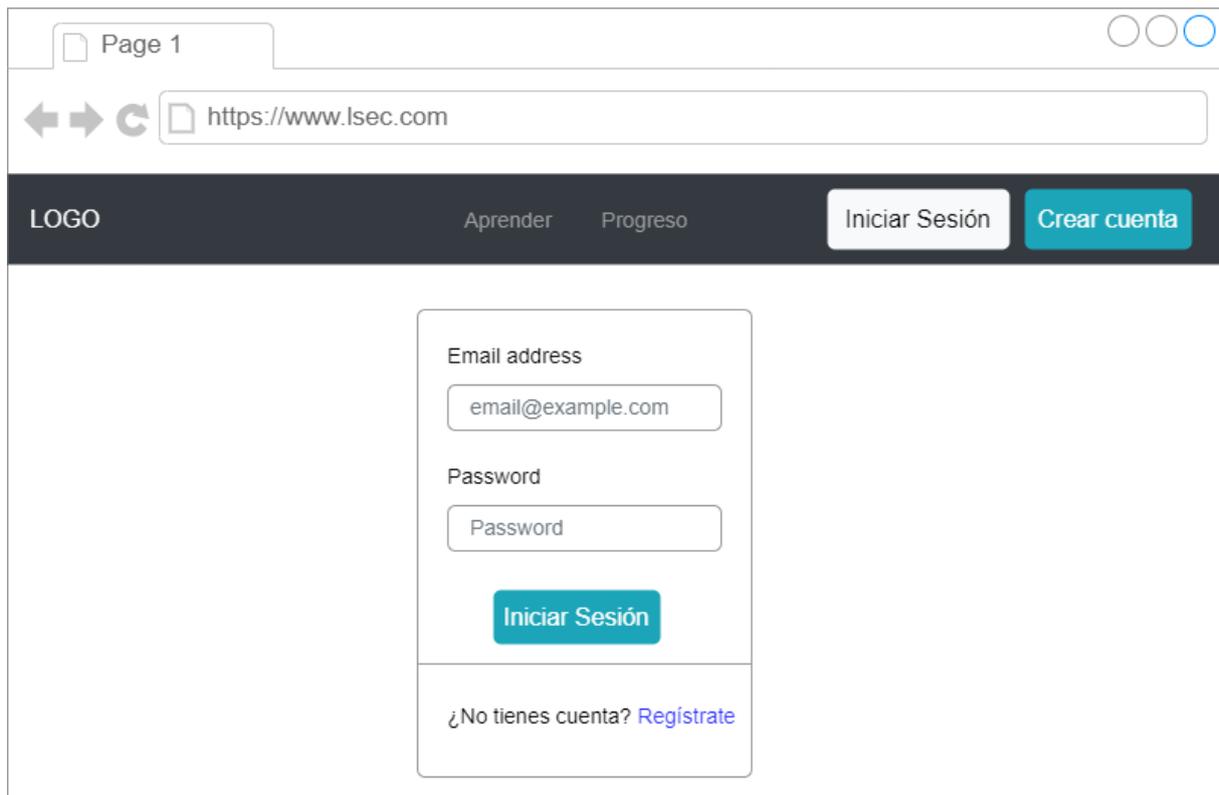


Figura 36. Wireframe iniciar sesión.

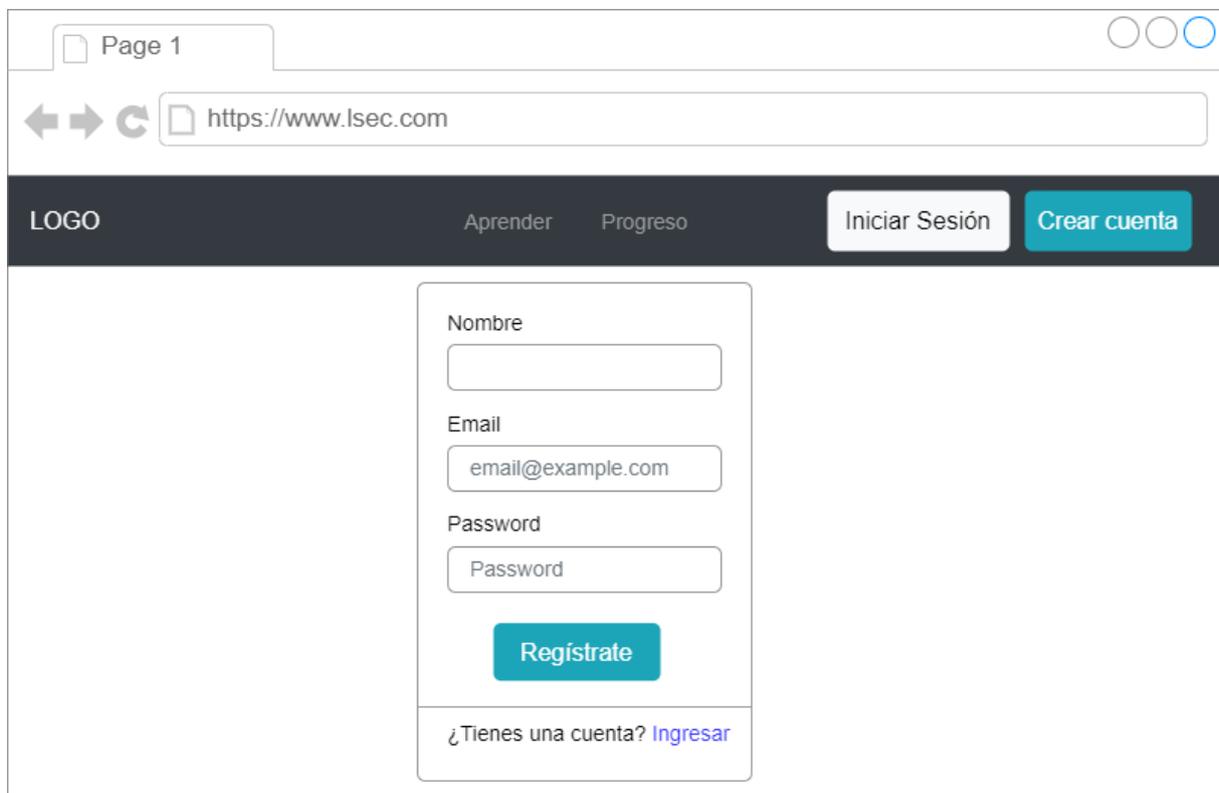


Figura 37. Wireframe crear cuenta de usuario.

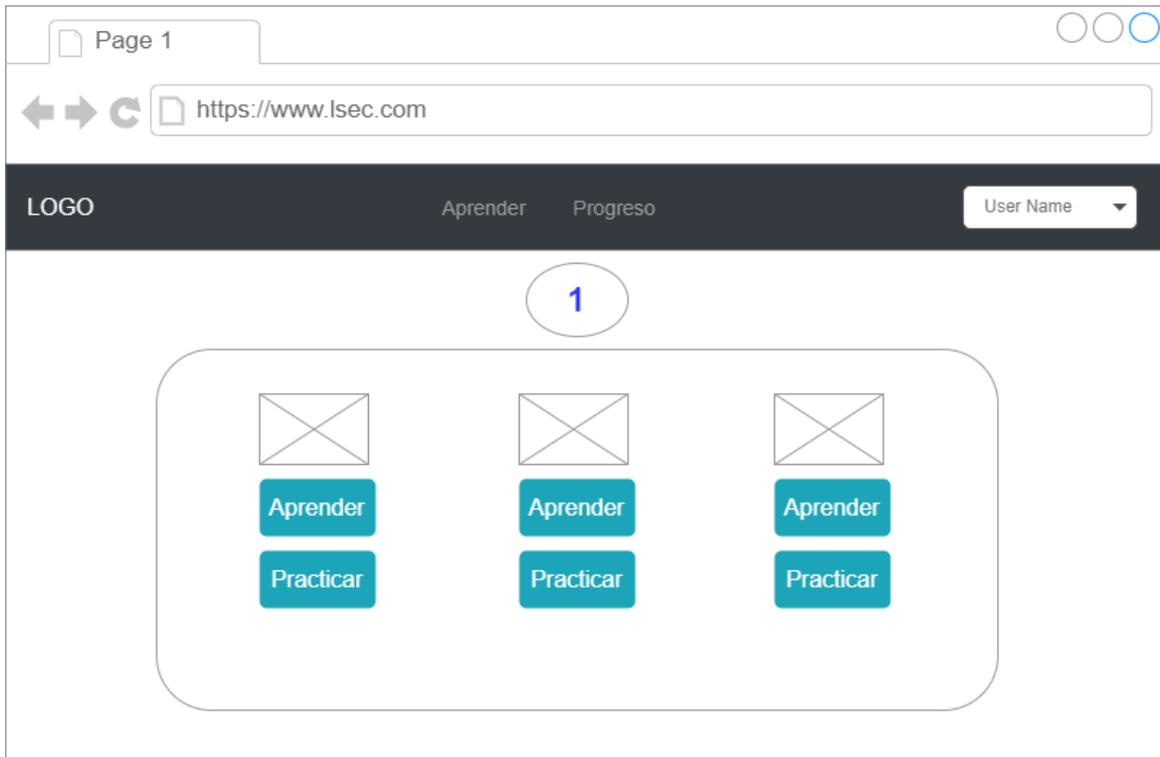


Figura 38. Wireframe interfaz de contenido.

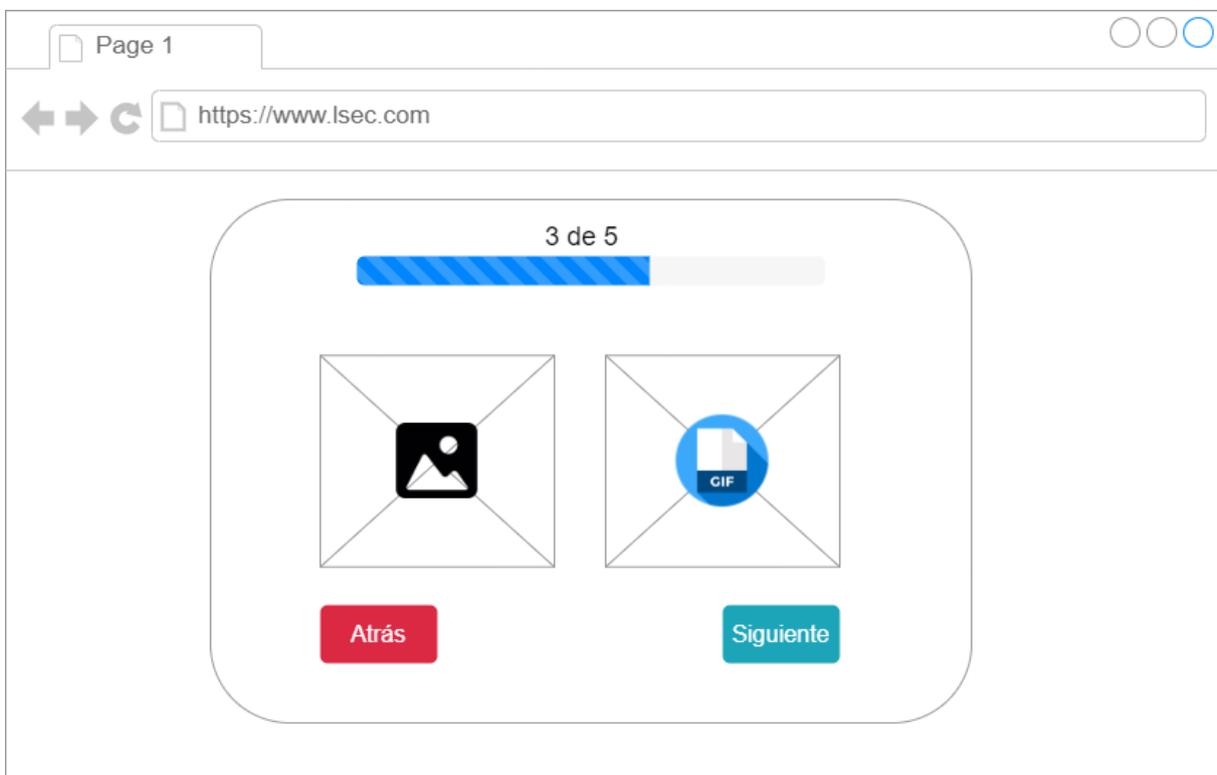


Figura 39. Wireframe cápsula de aprendizaje.



Figura 40. Wireframe tarjeta de aprendizaje.

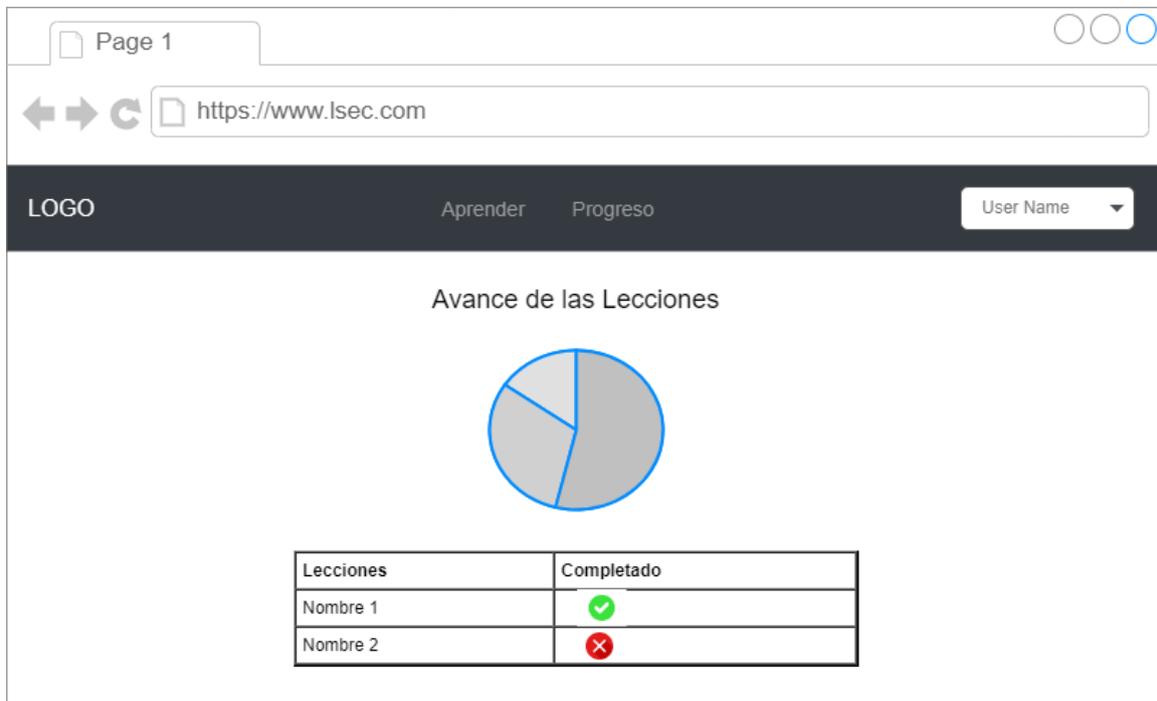


Figura 41. Wireframe ver lecciones completadas.

3.1.2. Interfaces de hardware

Se necesita:

- Computador con acceso a internet.

3.1.3. Interfaces de software

- Sistema Operativo: Windows 7 o superior.
- Navegador de internet: Chrome o Mozilla.

3.1.4. Interfaces de comunicación

El back-end y front-end se comunicarán mediante el protocolo HTTP mediante las peticiones GET, POST y DELETE. Para transmitir la información se utilizará JSON y para la seguridad JWT.

3.2. Requerimientos funcionales

Tabla 44.
Requerimiento funcional crear cuenta de usuario

Identificación de requerimiento	RF01
Nombre del requerimiento	Crear cuenta de usuario
Descripción del requerimiento	El usuario final podrá crear cuenta en el sistema a través de un formulario. Los datos para el registro son: nombre, correo electrónico y contraseña.
Prioridad del requerimiento	Alta

Tabla 45.
Requerimiento funcional autenticación de usuario

Identificación de requerimiento	RF02
Nombre del requerimiento	Autenticación de usuario
Descripción del requerimiento	El usuario final y administrador debe de identificarse con el correo electrónico y contraseña para acceder al sistema.
Prioridad del requerimiento	Alta

Tabla 46.
Requerimiento funcional cerrar sesión de usuario

Identificación de requerimiento	RF03
Nombre del requerimiento	Cerrar sesión de usuario

Descripción del requerimiento	El usuario final y administrador podrá cerrar sesión en el sistema e ingresar con otra cuenta.
Prioridad del requerimiento	Medio

Tabla 47.
Requerimiento funcional gestión de módulos

Identificación de requerimiento	RF04
Nombre del requerimiento	Gestión de módulos
Descripción del requerimiento	El usuario administrador podrá crear, visualizar, actualizar y borrar módulos. Para crear el módulo deberá de ingresar: número de módulo y nombre.
Prioridad del requerimiento	Alta

Tabla 48.
Requerimiento funcional gestión de lecciones

Identificación de requerimiento	RF05
Nombre del requerimiento	Gestión de lecciones
Descripción del requerimiento	El usuario administrador podrá crear, visualizar, actualizar y borrar lecciones. Para crear una lección deberá de ingresar: número de módulo, número de lección, nombre de la lección e icono.
Prioridad del requerimiento	Alta

Tabla 49.
Requerimiento funcional gestión de tarjetas de aprendizaje

Identificación de requerimiento	RF06
Nombre del requerimiento	Gestión de tarjetas de aprendizaje
Descripción del requerimiento	El usuario administrador podrá crear, visualizar, actualizar y borrar tarjetas de aprendizaje. Para crear una tarjeta de aprendizaje deberá de ingresar: nombre de la lección a la que pertenece, pregunta, gif de la seña, respuestas de selección múltiple y la respuesta correcta. En las tarjetas de aprendizaje, se visualizará el gif de la seña que el usuario final deberá de identificar la seña y elegir una respuesta. Las tarjetas de aprendizaje no darán puntaje, son solo un medio para que el estudiante pueda practicar y recordar lo aprendido, pero al completar todas las tarjetas de aprendizaje dentro de una lección, está se marcará como aprendido.
Prioridad del requerimiento	Alta

Tabla 50.
Requerimiento funcional visualizar lecciones completadas

Identificación de requerimiento	RF07
Nombre del requerimiento	Visualizar lecciones completadas.
Descripción del requerimiento	El usuario final podrá ver su avance de lecciones completadas.
Prioridad del requerimiento	Baja

Tabla 51.
Requerimiento funcional gestión de cápsulas de aprendizaje

Identificación de requerimiento	RF08
Nombre del requerimiento	Gestión de cápsulas de aprendizaje
Descripción del requerimiento	<p>El usuario administrador podrá crear, visualizar, actualizar y borrar cápsulas de aprendizaje.</p> <p>Para crear una cápsula de aprendizaje deberá de ingresar: nombre de la lección a la que pertenece, nombre de la cápsula, gif de la seña e imagen representativa de la seña.</p> <p>La cápsula de aprendizaje se mostrará al ingresar a la lección.</p>
Prioridad del requerimiento	Alta

3.3. Requerimientos no funcionales

Tabla 52.
Requerimientos no funcionales

Código	Requerimiento	Descripción
RNF01	Rendimiento	El sistema debe evitar que las consultas y otros procesos afecten significativamente el rendimiento de la base de datos y el tráfico de la red.
RNF02	Seguridad	<p>El sistema cuenta con autenticación de usuario vía correo electrónico y contraseña.</p> <p>El sistema garantiza la seguridad de la información y los datos que maneja, como correos electrónicos y contraseñas.</p> <p>Controles que permiten al personal autorizado acceder a la información a través de Internet con el fin de visualizar y cargar información en el sistema.</p>
RNF03	Fiabilidad	El sistema debería poder restaurar los datos afectados en caso de un error en el tiempo de respuesta y mostrar un mensaje de error al usuario.
RNF04	Disponibilidad	Los sistemas (back-end y front-end) deben estar disponibles las 24 horas del día, los 7 días de la semana, dada la disponibilidad de los servicios de hospedaje contratados, y deben restaurarse en el menor tiempo posible en caso de falla del software del sistema.
RNF05	Mantenibilidad	Los sistemas (back-end y front-end) necesitan documentación que sea fácil de actualizar para que se pueda realizar el mantenimiento.

RNF06	Portabilidad	El front-end deberá ser compatible con la mayoría de los navegadores de internet.
RNF07	Usabilidad	El sistema debe proporcionar una interfaz de usuario intuitiva y fácil de usar que haga que el proceso sea fácil de entender y ejecutar. También debería funcionar con cualquier navegador web.

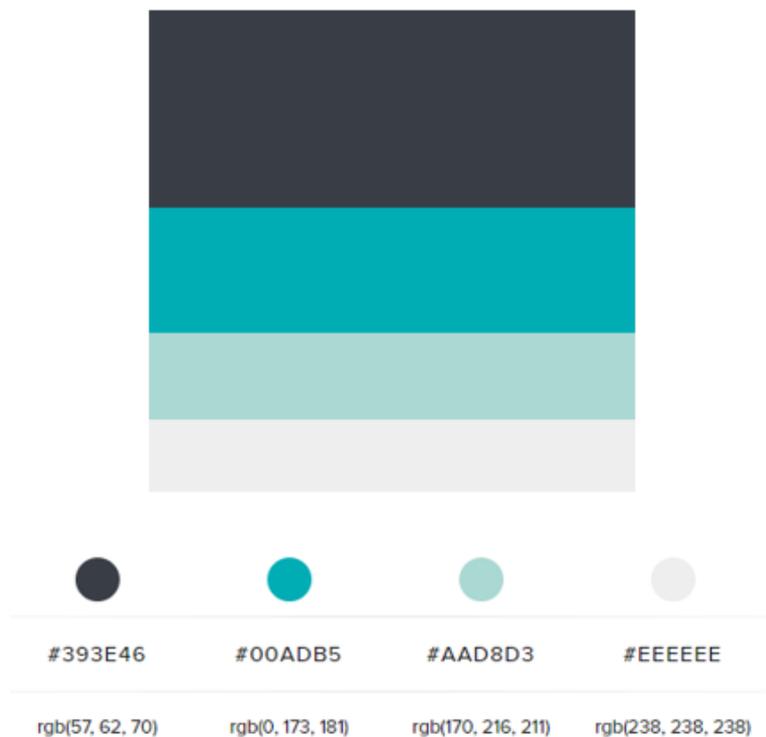
Anexo 3. Prototipo de Interfaz de Usuario

Prototipo de Interfaz de Usuario

Este documento contiene detalles sobre las interfaces desarrolladas en la herramienta Adobe XD, que fueron guía para el desarrollo del presente TT y como forma de validación de requerimientos por parte de los entrevistados.

Link para descargar el prototipo: <https://bit.ly/3NBtcZT>

1. Paleta de Colores



2. Interfaz de Usuario Final

2.1. Landing Page

Aprende LSEC Aprender Progreso Iniciar Sesión Crear Cuenta

Ahora de aprender algo nuevo

Aprende Lengua de Señas Ecuatoriana



Aprender lengua de señas, da una nueva oportunidad

[Crea tu cuenta hoy](#)




Construye un Ecuador Inclusivo

- 01 Crea tu cuenta**
Es muy sencillo, solamente ingresa tu correo electrónico y listo.
- 02 Inicia tu aprendizaje.**
Ingresa con tu cuenta y selecciona una lección que desea aprender.
- 03 Pon en práctica tus conocimientos.**
Puedes practicar por medio de preguntas de selección múltiple.

2.2. Inicio de sesión de usuario final

Aprende LSEC Aprender Progreso Iniciar Sesión Crear Cuenta

Iniciar Sesión

Correo Electrónico

Contraseña

[INICIAR SESIÓN](#)

¿No tienes una cuenta? [Regístrate](#)

2.3. Crear cuenta de usuario final

Aprende LSEC Aprender Progreso Iniciar Sesión Crear Cuenta

Crear Cuenta

Nombre

Correo Electrónico

Contraseña

[REGISTRATE](#)

¿Tienes una cuenta? [Inicia sesión](#)

2.4. Página principal de usuario final

1



Aprender

Practicar



Aprender

Practicar



Aprender

Practicar



Aprender

Practicar



Aprender

Practicar



Aprender

Practicar

2



Aprender

Practicar



Aprender

Practicar



Aprender

Practicar



Aprender

Practicar



Aprender

Practicar



Aprender

Practicar

3



Aprender

Practicar



Aprender

Practicar



Aprender

Practicar



Aprender

Practicar



Aprender

Practicar



Aprender

Practicar

2.5. Cápsula de aprendizaje

2 de 5

×

Abeja


Anterior

abeja


Siguiente

2.6. Tarjeta de aprendizaje

2 de 5

×

¿Cuál es el significado?

abeja



Abeja

Gato

AnteriorComprobar

2.7. Progreso de usuario final



3. Interfaz de Administrador

3.1. Inicio de sesión de usuario administrador

Lengua de Señas Ecuatoriano

ADMINISTRADOR

correo electrónico

contraseña

Iniciar sesión

3.2. Gestión de módulos

LSEC Crear Módulo Crear Lección Crear Contenido 

Crear Módulo

Nombre del modulo

Número

Crear

0	Nombre	Modificar	Eliminar
1	Nombre	Modificar	Eliminar
2	Nombre	Modificar	Eliminar

3.3. Gestión de lección

LSEC Crear Módulo Crear Lección Crear Contenido 

Crear Lección

Nombre

Imágen

Crear

0	Nombre	Modificar	Eliminar
1	Nombre	Modificar	Eliminar
2	Nombre	Modificar	Eliminar

3.4. Gestión de contenido

LSEC Crear Módulo Crear Lección Crear Contenido 

Crear Contenido

Nombre

Imagen

Gif

[Crear](#)

0	Nombre	Modificar Eliminar
1	Nombre	Modificar Eliminar
2	Nombre	Modificar Eliminar

Anexo 4. Descripción de Historias de Usuario

Lista de Historias de Usuario

Proyecto: Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano

Versión: 1.0

Fecha: 22/07/2022

Tabla 53.
Historia de Usuario crear cuenta usuario final

Historia de Usuario	
Número: 01	Usuario: Usuario final
Nombre de la historia: Crear cuenta usuario final	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja
Programador responsable: Jonnathan Espinoza	
Descripción: <ul style="list-style-type: none"> • Como usuario final quiero crear una cuenta de usuario para poder acceder al contenido de la aplicación web y aprender Lengua de Señas Ecuatoriana. 	
Validación: <ul style="list-style-type: none"> • Información requerida: nombre, correo electrónico y contraseña. • Usuario final registrado en la base de datos. • Mensaje de creación de cuenta exitoso. • Redirecciona a la página de iniciar sesión. • Fallo: mensaje de error si los campos del formulario están mal ingresados o si el usuario ya está registrado. 	
Observaciones: <ul style="list-style-type: none"> • Correo electrónico único. 	

Tabla 54.
Historia de Usuario iniciar sesión usuario final

Historia de Usuario	
Número: 02	Usuario: Usuario final
Nombre de la historia: Iniciar sesión usuario final	
Prioridad en negocio: Baja	Riesgo en desarrollo: Medio
Programador responsable: Jonnathan Espinoza	
Descripción: <ul style="list-style-type: none"> • Como usuario final quiero ingresar a la aplicación web mediante correo electrónico y contraseña para poder acceder al contenido. 	
Validación: <ul style="list-style-type: none"> • Información requerida: correo electrónico y contraseña. • Inicio de sesión correctamente al sistema. • Redirección a la interfaz principal. • Fallo: mensaje de error si los campos del formulario están mal ingresados. 	
Observaciones:	

Tabla 55.
Historia de Usuario cerrar sesión usuario final

Historia de Usuario	
Número: 03	Usuario: Usuario final
Nombre de la historia: Cerrar sesión usuario final	
Prioridad en negocio:	Riesgo en desarrollo:

Baja	Baja
Programador responsable: Jonnathan Espinoza	
Descripción:	
<ul style="list-style-type: none"> • Como usuario final quiero cerrar sesión para poder salir de la aplicación web. 	
Validación:	
<ul style="list-style-type: none"> • Redirección a la interfaz de principal • Mensaje de “Sesión Cerrada” 	
Observaciones:	

Tabla 56.
Historia de Usuario crear módulos

Historia de Usuario	
Número: 04	Usuario: Usuario administrador
Nombre de la historia: Crear módulos	
Prioridad en negocio:	Riesgo en desarrollo:
Alta	Alta
Programador responsable: Jonnathan Espinoza	
Descripción:	
<ul style="list-style-type: none"> • Como usuario administrador quiero crear módulos de contenido para poder seccionar el contenido de la aplicación web. 	
Validación:	
<ul style="list-style-type: none"> • Datos ingresados en el formulario correctamente validados: número y nombre. • Redirección a la interfaz de crear módulos. • Mensaje “Módulo Creado”. • Fallo: mensaje de error si los campos del formulario están mal ingresados. • Fallo: mensaje de error del backend al no poder crear el módulo. 	
Observaciones:	
<ul style="list-style-type: none"> • El número de módulo debe ser único. 	

Tabla 57.
Historia de Usuario visualizar módulo

Historia de Usuario	
Número: 05	Usuario: Usuario administrador
Nombre de la historia: Visualizar módulos	
Prioridad en negocio:	Riesgo en desarrollo:
Medio	Baja
Programador responsable: Jonnathan Espinoza	
Descripción:	
<ul style="list-style-type: none"> • Cómo usuario administrador quiero visualizar los módulos de contenido creados para poder conocer el listado de los módulos. 	
Validación:	
<ul style="list-style-type: none"> • Los módulos serán listados en orden, de acuerdo al número de módulo. • Se mostrará el número de módulo, el nombre y opciones de editar, visualizar o eliminar. • Fallo: mensaje de error en caso de no encontrar ningún módulo creado. 	
Observaciones:	

Tabla 58.
Historia de Usuario actualizar módulos

Historia de Usuario	
Número: 06	Usuario: Usuario administrador
Nombre de la historia: Actualizar módulos	
Prioridad en negocio: Medio	Riesgo en desarrollo: Medio
Programador responsable: Jonnathan Espinoza	
Descripción: <ul style="list-style-type: none"> • Como usuario administrador quiero editar los módulos de contenido creados para poder actualizar la información de éstas. 	
Validación: <ul style="list-style-type: none"> • Presentar y poder editar los datos del módulo creado en un formulario: número y nombre. • Los datos en el formulario correctamente validados. • Mensaje afirmativo. • Fallo: mensaje de error si los campos del formulario están mal ingresados o si no se ha podido realizar la actualización. 	
Observaciones: <ul style="list-style-type: none"> • El campo del número de módulo no se puede repetir con otros, este debe ser único. 	

Tabla 59.
Historia de Usuario borrar módulos

Historia de Usuario	
Número: 07	Usuario: Usuario administrador
Nombre de la historia: Borrar módulos	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja
Programador responsable: Jonnathan Espinoza	
Descripción: <ul style="list-style-type: none"> • Como usuario administrador quiero eliminar los módulos creados para poder borrar definitivamente de la base de datos. 	
Validación: <ul style="list-style-type: none"> • Pedir confirmación de borrar datos mediante el mensaje “¿Desea eliminar el módulo?” • Mensaje de éxito “Módulo eliminado”. • Fallo: mensaje de error si el módulo está siendo utilizado por un bloqueo activo. 	
Observaciones:	

Tabla 60.
Historia de Usuario crear lección

Historia de Usuario	
Número: 08	Usuario: Usuario administrador
Nombre de la historia: Crear lección	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta

Programador responsable: Jonnathan Espinoza
Descripción: <ul style="list-style-type: none"> • Como usuario administrador quiero crear una nueva lección para poder organizar los temas y agregar los microcontenidos.
Validación: <ul style="list-style-type: none"> • Datos ingresados en el formulario correctamente validados: número de módulo, número de lección, nombre de lección e icono. • Redirección a la interfaz de crear lección. • Mensaje "Lección Creada". • Fallo: mensaje de error si los campos del formulario están mal ingresados.
Observaciones:

Tabla 61.
Historia de Usuario visualizar lección

Historia de Usuario	
Número: 09	Usuario: Usuario administrador
Nombre de la historia: Visualizar lección	
Prioridad en negocio: Medio	Riesgo en desarrollo: Baja
Programador responsable: Jonnathan Espinoza	
Descripción: <ul style="list-style-type: none"> • Como usuario administrador quiero visualizar las lecciones creadas para poder conocer el listado de las lecciones. 	
Validación: <ul style="list-style-type: none"> • Las lecciones serán listadas en orden, de acuerdo al número de módulo. • Se mostrará el número de módulo, el nombre de la lección y opciones de editar, visualizar o eliminar. • Fallo: mensaje de error en caso de no encontrar ninguna lección creada. 	
Observaciones:	

Tabla 62.
Historia de Usuario actualizar lección

Historia de Usuario	
Número: 10	Usuario: Usuario administrador
Nombre de la historia: Actualizar lección	
Prioridad en negocio: Medio	Riesgo en desarrollo: Medio
Programador responsable: Jonnathan Espinoza	
Descripción: <ul style="list-style-type: none"> • Como usuario administrador quiero editar la lección creada para poder actualizar la información de estas. 	
Validación: <ul style="list-style-type: none"> • Presentar y poder editar la información de lección en un formulario: módulo, nombre e icono. • Los datos en el formulario correctamente validados. • Mensaje afirmativo. • Fallo: mensaje de error si los campos del formulario están mal ingresados o si no se ha podido realizar la actualización. 	
Observaciones:	

Tabla 63.
Historia de Usuario borrar lección

Historia de Usuario	
Número: 11	Usuario: Usuario administrador
Nombre de la historia: Borrar lección	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja
Programador responsable: Jonnathan Espinoza	
Descripción: <ul style="list-style-type: none"> Como usuario administrador quiero eliminar las lecciones creadas para poder borrar definitivamente de la base de datos. 	
Validación: <ul style="list-style-type: none"> Pedir confirmación de borrar datos mediante el mensaje "¿Desea eliminar la lección?" Mensaje de éxito "Lección eliminada". Fallo: mensaje de error si la lección está siendo utilizado por un bloqueo activo. 	
Observaciones:	

Tabla 64.
Historia de Usuario crear ejercicios

Historia de Usuario	
Número: 12	Usuario: Usuario administrador
Nombre de la historia: Crear ejercicios	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Programador responsable: Jonnathan Espinoza	
Descripción: <ul style="list-style-type: none"> Como usuario administrador quiero crear una tarjeta de aprendizaje para poder agregar en una lección. 	
Validación: <ul style="list-style-type: none"> Datos ingresados en el formulario correctamente validados: nombre de la lección, pregunta, gif, imagen y respuesta correcta e incorrecta. Redirección a la interfaz de crear tarjeta de aprendizaje. Mensaje "Tarjeta Creada". Fallo: mensaje de error si los campos del formulario están mal ingresados. 	
Observaciones:	

Tabla 65.
Historia de Usuario visualizar ejercicio

Historia de Usuario	
Número: 13	Usuario: Usuario administrador
Nombre de la historia: Visualizar ejercicio	
Prioridad en negocio: Medio	Riesgo en desarrollo: Baja
Programador responsable: Jonnathan Espinoza	
Descripción:	

<ul style="list-style-type: none"> • Como usuario administrador quiero visualizar las tarjetas de aprendizaje creadas para poder conocer el listado de las pruebas.
Validación: <ul style="list-style-type: none"> • Las tarjetas de aprendizaje serán listadas en orden, de acuerdo al número de módulo. • Se mostrará el número de módulo, el nombre de la lección y opciones de editar, visualizar o eliminar. • Fallo: mensaje de error en caso de no encontrar ninguna tarjeta creada.
Observaciones:

Tabla 66.
Historia de Usuario actualizar ejercicio

Historia de Usuario	
Número: 14	Usuario: Usuario administrador
Nombre de la historia: Actualizar ejercicio	
Prioridad en negocio: Medio	Riesgo en desarrollo: Medio
Programador responsable: Jonnathan Espinoza	
Descripción: <ul style="list-style-type: none"> • Como usuario administrador quiero editar la tarjeta de aprendizaje creada para poder actualizar la información de estas. 	
Validación: <ul style="list-style-type: none"> • Presentar y poder editar la información de la tarjeta en un formulario. • Los datos en el formulario correctamente validados. • Mensaje afirmativo. • Fallo: mensaje de error si los campos del formulario están mal ingresados o si no se ha podido realizar la actualización. 	
Observaciones:	

Tabla 67.
Historia de Usuario borrar ejercicio

Historia de Usuario	
Número: 15	Usuario: Usuario administrador
Nombre de la historia: Borrar ejercicio	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja
Programador responsable: Jonnathan Espinoza	
Descripción: <ul style="list-style-type: none"> • Como usuario administrador quiero eliminar las tarjetas de aprendizaje creadas para poder borrar definitivamente de la base de datos. 	
Validación: <ul style="list-style-type: none"> • Pedir confirmación de borrar datos mediante el mensaje "¿Desea eliminar la tarjeta?" • Mensaje de éxito "Tarjeta eliminada". • Fallo: mensaje de error si la tarjeta está siendo utilizado por un bloqueo activo. 	
Observaciones:	

Tabla 68.
Historia de Usuario visualizar lecciones completadas

Historia de Usuario	
Número: 16	Usuario: Usuario final
Nombre de la historia: Visualizar lecciones completadas.	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja
Programador responsable: Jonnathan Espinoza	
Descripción: <ul style="list-style-type: none"> Como usuario final quiero ver mis avances en las lecciones para identificar las lecciones estudiadas. 	
Validación: <ul style="list-style-type: none"> Listar las lecciones que ya estudió. 	
Observaciones:	

Tabla 69.
Historia de Usuario crear cápsulas de aprendizaje

Historia de Usuario	
Número: 17	Usuario: Usuario administrador
Nombre de la historia: Crear cápsula de aprendizaje	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Programador responsable: Jonnathan Espinoza	
Descripción: <ul style="list-style-type: none"> Como usuario administrador quiero crear una nueva cápsula de aprendizaje para poder agregar a una lección. 	
Validación: <ul style="list-style-type: none"> Datos ingresados en el formulario correctamente validados: nombre de lección, nombre de microcontenido, gif e imagen. Redirección a la interfaz de crear microcontenido. Mensaje "Microcontenido Creado". Fallo: mensaje de error si los campos del formulario están mal ingresados. 	
Observaciones:	

Tabla 70.
Historia de Usuario visualizar cápsula de aprendizaje

Historia de Usuario	
Número: 18	Usuario: Usuario administrador
Nombre de la historia: Visualizar cápsula de aprendizaje	
Prioridad en negocio: Medio	Riesgo en desarrollo: Baja
Programador responsable: Jonnathan Espinoza	
Descripción: <ul style="list-style-type: none"> Como usuario administrador quiero visualizar las cápsulas de aprendizaje creadas para poder conocer el listado de las estos. 	

Validación: <ul style="list-style-type: none"> • Los microcontenidos serán listados en orden, de acuerdo al número de módulo. • Se mostrará el número de módulo, el nombre de la lección, nombre de microcontenido y opciones de editar, visualizar o eliminar. • Fallo: mensaje de error en caso de no encontrar ningún microcontenido.
Observaciones:

Tabla 71.
Historia de Usuario actualizar cápsulas de aprendizaje

Historia de Usuario	
Número: 19	Usuario: Usuario administrador
Nombre de la historia: Actualizar cápsulas de aprendizaje	
Prioridad en negocio: Medio	Riesgo en desarrollo: Medio
Programador responsable: Jonnathan Espinoza	
Descripción: <ul style="list-style-type: none"> • Como usuario administrador quiero editar las cápsulas de aprendizaje creadas para poder actualizar la información de estos. 	
Validación: <ul style="list-style-type: none"> • Los datos en el formulario correctamente validados. • Mensaje afirmativo. • Fallo: mensaje de error si los campos del formulario están mal ingresados o si no se ha podido realizar la actualización. 	
Observaciones:	

Tabla 72.
Historia de Usuario borrar cápsulas de aprendizaje

Historia de Usuario	
Número: 20	Usuario: Usuario administrador
Nombre de la historia: Borrar cápsulas de aprendizaje	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja
Programador responsable: Jonnathan Espinoza	
Descripción: <ul style="list-style-type: none"> • Como usuario administrador quiero eliminar las cápsulas de aprendizaje creadas para poder borrar definitivamente de la base de datos. 	
Validación: <ul style="list-style-type: none"> • Pedir confirmación de borrar datos mediante el mensaje "¿Desea eliminar el microcontenido?" • Mensaje de éxito "Microcontenido eliminado". • Fallo: mensaje de error si el microcontenido está siendo utilizado por un bloqueo activo. 	
Observaciones:	

Tabla 73.

Historia de Usuario visualizar cápsulas de aprendizaje en la interfaz de usuario final

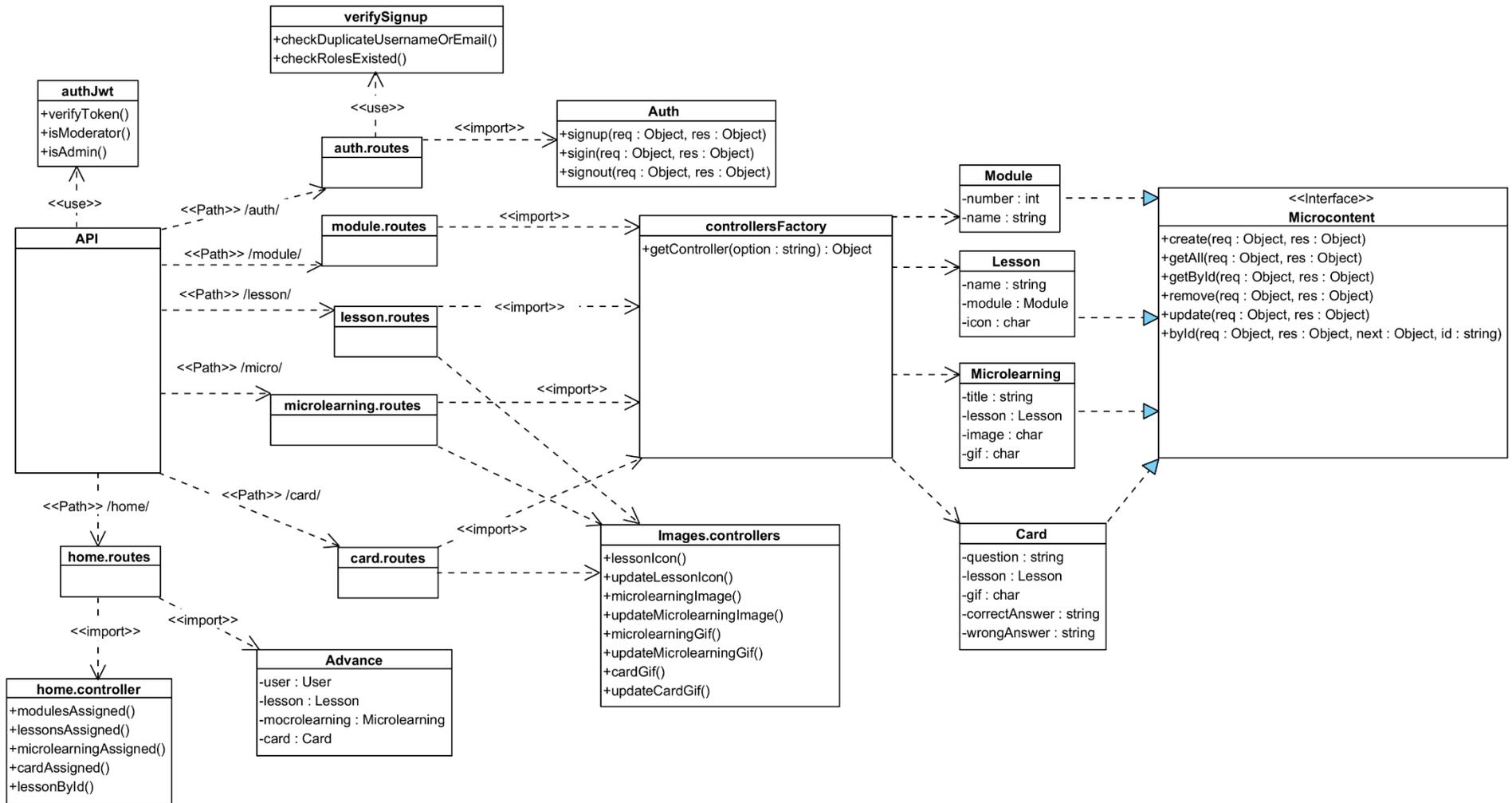
Historia de Usuario	
Número: 21	Usuario: Usuario final
Nombre de la historia: Visualizar cápsulas de aprendizaje en la interfaz de usuario final	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Programador responsable: Jonnathan Espinoza	
Descripción: <ul style="list-style-type: none"> • Como usuario final quiero visualizar los microcontenidos separados en módulos y lecciones para saber el orden en que se debe seguir cada sección y tener un aprendizaje eficaz. 	
Validación: <ul style="list-style-type: none"> • Al ingresar a la página principal del usuario final, presentará los módulos ordenados de acuerdo a su número y dentro de ellos se mostrará las lecciones correspondientes a este. • Al dar clic en la lección, se presentará los microcontenidos y se podrá pasar al siguiente microcontenido por medio de un botón. • Al terminar todos los microcontenidos, la página redirigirá al usuario final a la página principal. 	
Observaciones: <ul style="list-style-type: none"> • Para ingresar al contenido debe de iniciar sesión en la aplicación web. 	

Tabla 74.

Historia de Usuario realizar ejercicios

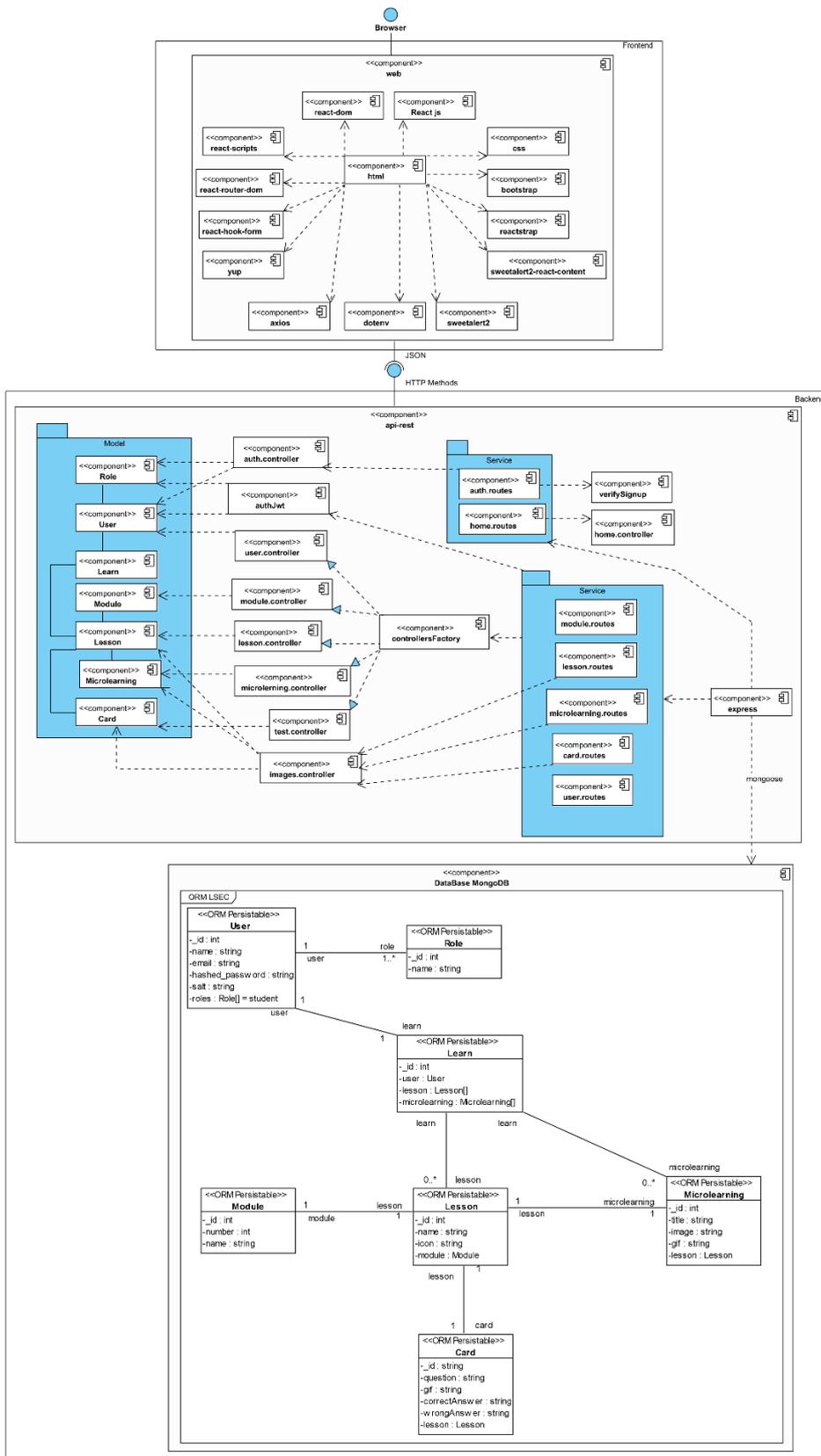
Historia de Usuario	
Número: 22	Usuario: Usuario final
Nombre de la historia: Realizar ejercicios	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Programador responsable: Jonnathan Espinoza	
Descripción: <ul style="list-style-type: none"> • Como usuario final quiero realizar ejercicios para verificar lo aprendido. 	
Validación: <ul style="list-style-type: none"> • Al ingresar a la página principal del usuario final, presentará los módulos ordenados de acuerdo a su número y dentro de ellos se mostrará las lecciones correspondientes a este con su respectivo botón para practicar. • Al dar clic en el botón "Practicar", se presentará un gif de la seña para identificar cual es el significado. • Los ejercicios no presentan calificaciones, solo sirven para la retroalimentación de lo aprendido, pero validarán la respuesta correcta. • Al terminar los ejercicios, la página redirigirá al usuario final a la página principal. 	
Observaciones: <ul style="list-style-type: none"> • Para ingresar al contenido debe de iniciar sesión en la aplicación web. 	

Anexo 5. Descripción del Diagrama de Clases



Véase la imagen en alta resolución en el siguiente link: <https://bit.ly/3RZTfvY>

Anexo 6. Descripción del Diagrama de Componentes



Véase la imagen en alta resolución en el siguiente link: <https://bit.ly/3yBPEO4>

Anexo 7. Desarrollo de la Aplicación Web

Desarrollo de la aplicación web con la metodología XP

Proyecto: Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano

Índice

Índice	115
Introducción.....	116
1. Iteración 1	118
1.1. Planificación.....	118
1.2. Diseño.....	118
1.3. Codificación	120
1.4. Pruebas.....	131
2. Iteración 2	135
2.1. Planificación.....	135
2.2. Diseño.....	136
2.3. Codificación	136
2.4. Pruebas.....	142
3. Iteración 3	145
3.1. Planificación.....	145
3.2. Diseño.....	146
3.3. Codificación	146
3.4. Pruebas.....	151

Introducción

Para el desarrollo de la aplicación web se utilizó la metodología XP y como lenguaje principal JavaScript. La planificación y seguimiento de las distintas fases e iteraciones se utilizó la herramienta Jira Software y Miro, para lo cual, se estableció la duración de dos semanas por cada iteración.

La aplicación se desarrolla bajo la arquitectura API REST y la tecnología que se utilizó en cada interfaz se presenta en la Tabla 75.

Tabla 75.
Tecnologías de la aplicación web

	Tecnología
Backend	Node js, Express, MongoDB
Frontend	React js

Para las pruebas unitarias del software se utilizó el framework de pruebas Jest de JavaScript y el módulo SuperTest para las pruebas de la API.

En la Tabla 76 se presenta la relación de los módulos, historias de usuario y requerimientos.

Tabla 76.
Módulos de la aplicación web

Módulo	Historia de usuario	Descripción	Requerimiento
Módulo de Gestión de Lecciones	HU08	Crear lección	RF05
	HU09	Visualizar lección	RF05
	HU10	Actualizar lección	RF05
	HU11	Borrar lección	RF05
Módulo de Gestión de Secciones	HU04	Crear módulos	RF04
	HU05	Visualizar módulos	RF04
	HU06	Actualizar módulos	RF04
	HU07	Borrar módulos	RF04
Módulo de Gestión de Cápsulas de aprendizaje	HU17	Crear microcontenido	RF08
	HU18	Visualizar microcontenido	RF08
	HU19	Actualizar microcontenido	RF08
	HU20	Borrar microcontenido	RF08
	HU12	Crear ejercicio	RF06
	HU13	Visualizar ejercicio	RF06

Módulo de Gestión de Tarjetas de aprendizaje	HU14	Actualizar ejercicio	RF06
	HU15	Borrar ejercicio	RF06
Módulo de Cuenta de Usuario	HU01	Crear cuenta de estudiante	RF01
	HU02	Iniciar sesión de estudiante	RF02
	HU03	Cerrar sesión de estudiante	RF03
Módulo de Interfaz de Estudiante	HU21	Visualizar cápsulas de aprendizaje en la interfaz de usuario final	RF07
	HU16	Visualizar lecciones completadas	RF07

En la Figura 42 se presenta la planeación de las versiones y requerimientos de la aplicación web. Para mejor resolución, ver: <https://bit.ly/3g43FxA>

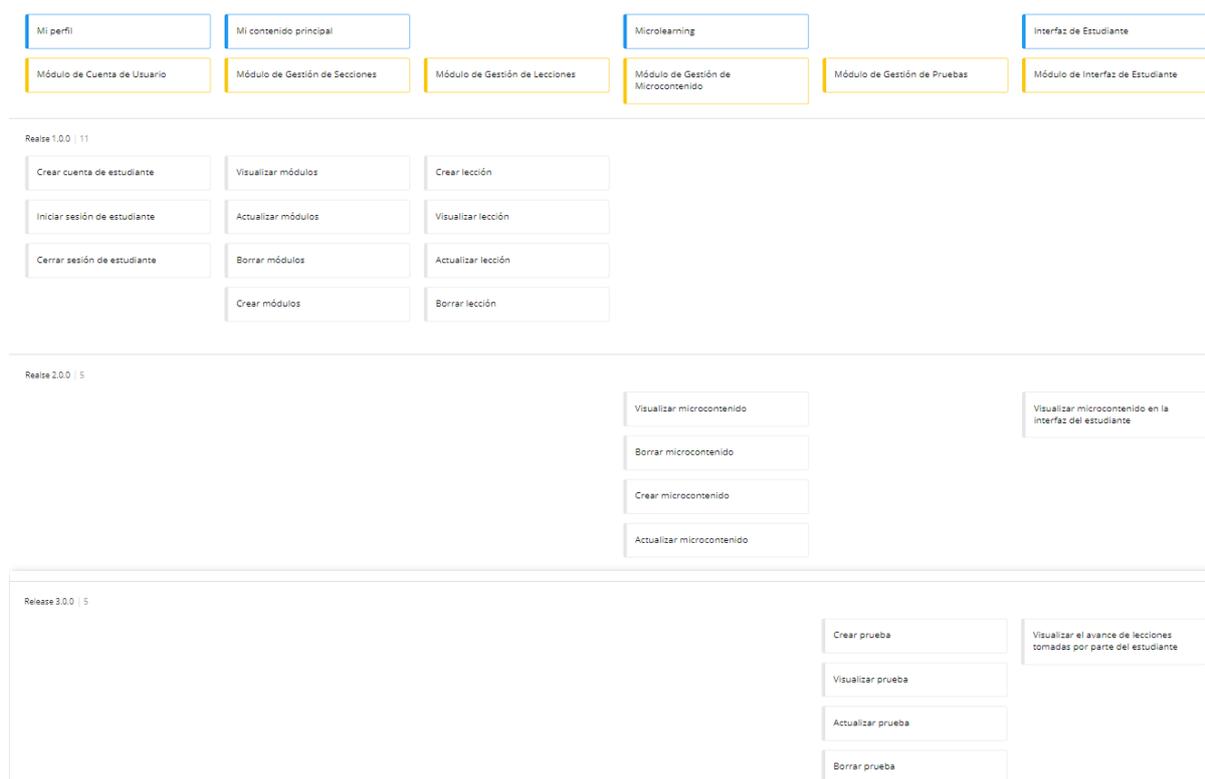


Figura 42. Planificación de los requerimientos funcionales en Miro.

A continuación, se realizan las iteraciones del proyecto. El código fuente de la aplicación web con sus respectivos comentarios se encuentra en:

- Back-end (API): <https://github.com/JonnathanE/api-microlearning-LSEC>
- Front-end: <https://github.com/JonnathanE/microlearning-LSEC>

1. Iteración 1

1.1. Planificación

En la Tabla 77 se presenta las incidencias para la primera iteración.

Tabla 77.
Tareas de la iteración 1

Clave de Incidencia	Resumen	Clave Historia de Usuario	Responsable	Estado
LSEC-24	Crear cuenta de estudiante	HU01	Jonnathan Espinoza	Completado
LSEC-25	Iniciar sesión de estudiante	HU02	Jonnathan Espinoza	Completado
LSEC-26	Cerrar sesión de estudiante	HU03	Jonnathan Espinoza	Completado
LSEC-7	Crear lección	HU08	Jonnathan Espinoza	Completado
LSEC-8	Visualizar lección	HU09	Jonnathan Espinoza	Completado
LSEC-9	Actualizar lección	HU10	Jonnathan Espinoza	Completado
LSEC-11	Borrar lección	HU11	Jonnathan Espinoza	Completado
LSEC-12	Crear módulos	HU04	Jonnathan Espinoza	Completado
LSEC-13	Visualizar módulos	HU05	Jonnathan Espinoza	Completado
LSEC-14	Actualizar módulos	HU06	Jonnathan Espinoza	Completado
LSEC-15	Borrar módulos	HU07	Jonnathan Espinoza	Completado

1.2. Diseño

El diseño dentro de la metodología XP, se lo realiza durante el tiempo de desarrollo del proyecto. En la especificación de requerimientos ya se planteó un prototipo de la interfaz de usuario para establecer las características esenciales del microlearning.

Para implementar el microlearning, la solución informática se basa en el concepto de Cápsula de Aprendizaje y Tarjeta de Aprendizaje como unidad de base para el aprendizaje. Las

Cápsulas de Aprendizaje presentan información corta de microlearning, el contenido que se eligió durante la elicitación de requerimientos fue: tema, un Formato de Intercambio de Gráficos (GIF) de la lengua de seña y una imagen representativa al patrón de la lengua de seña, como se muestra en la Figura 43. Las Tarjetas de Aprendizaje presentan una pregunta o actividad de microlearning, que requiere la entrada del usuario como respuesta, tal cual se presenta en la Figura 44. Los autores de contenidos pueden agrupar las Cápsulas de Aprendizaje y las Tarjetas de Conocimiento en Lecciones y Módulos, con el fin de abordar temas o conceptos más avanzados, como se presenta en la Figura 45.

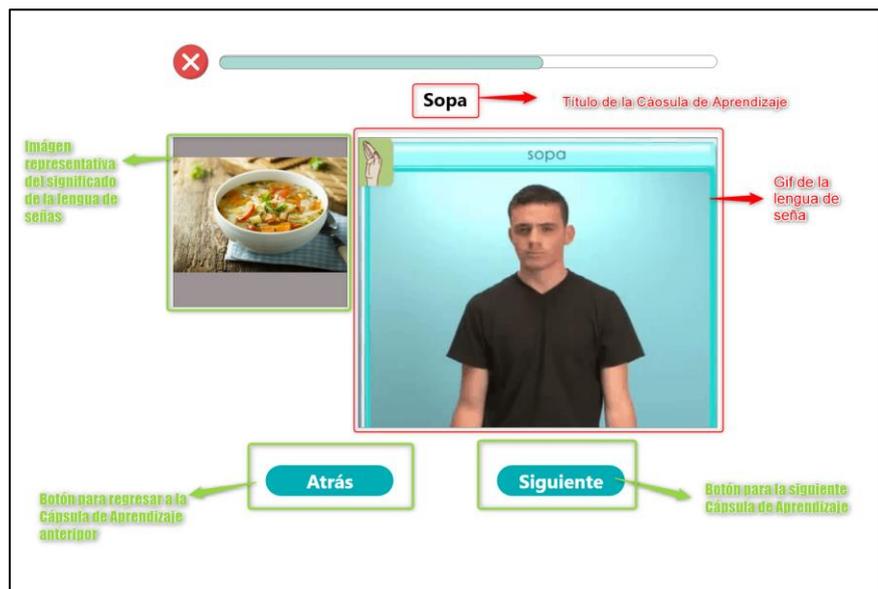


Figura 43. Cápsula de Aprendizaje.

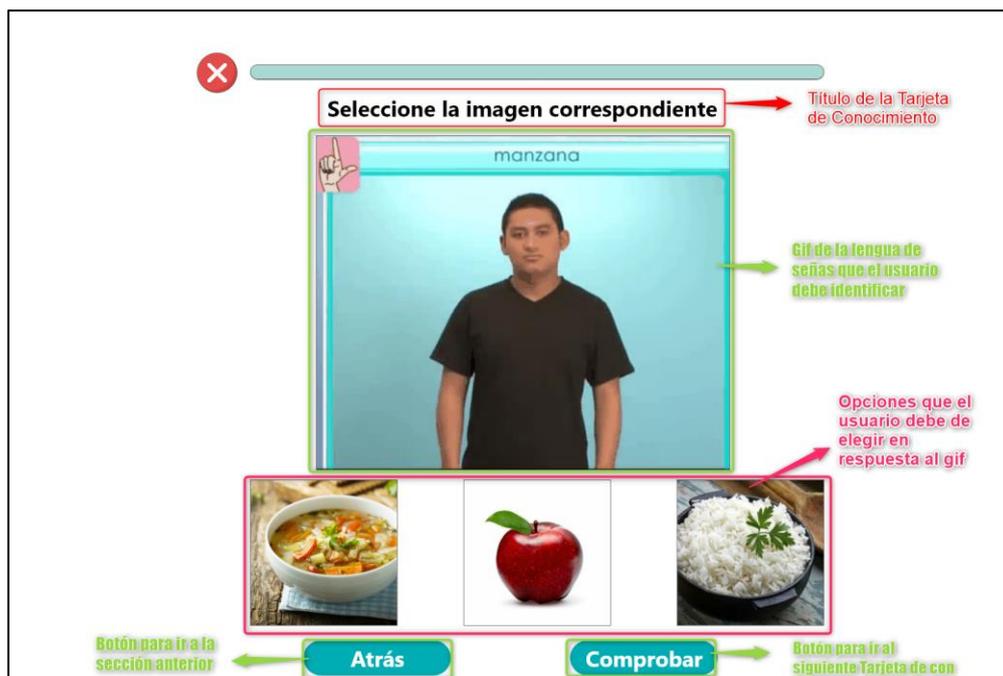


Figura 44. Tarjeta de Aprendizaje.

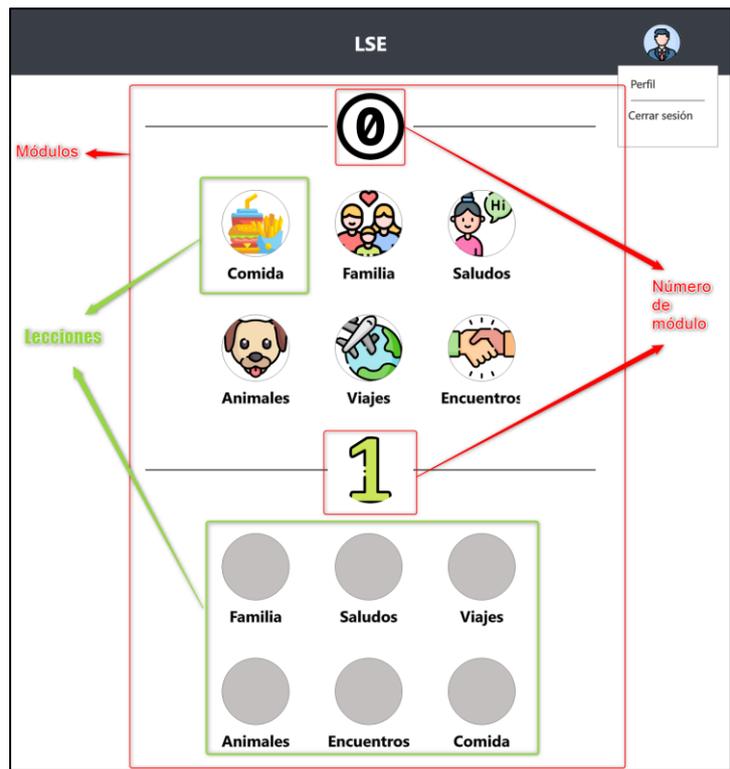


Figura 45. Prototipo, diseño de Lecciones y Módulo de la aplicación web.

Una vez definido el esquema del microlearning, se desarrolló la construcción de la autenticación de usuario, gestión de módulos y lecciones. En la Figura 46 se muestra el diagrama de clases generado para la API en esta primera iteración.

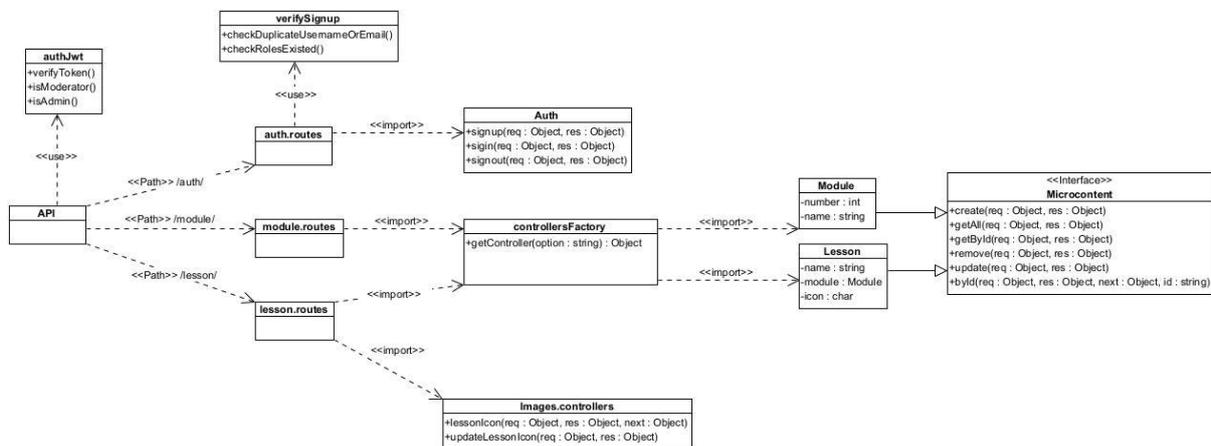


Figura 46. Diagrama de Clases de la API Iteración 1

1.3. Codificación

La primera parte del proyecto es la configuración del backend utilizando Nodejs, express para levantar el servidor, y mongoose para conectar la base de datos. Una vez configurado el entorno de desarrollo, se crearon los modelos (User, Role, Module, Lesson) y se definieron las URL's principales para acceder a la funcionalidad de la API (ver Figura 47).



```
1 // Routes
2 app.use('/api/user', require('../routes/user.routes'));
3 app.use('/api/module', require('../routes/module.routes'));
4 app.use('/api/auth', require('../routes/auth.routes'));
5 app.use('/api/lesson', require('../routes/lesson.routes'));
6 app.use('/api/micro', require('../routes/microlearning.routes'));
```

Figura 47. URL's principales de la API.

Se implementó el patrón de diseño Factory en los controladores del backend como se presenta en la Figura 48.



```
1 class controllersFactory {
2     getController = (option) => {
3         if (option === 'user') return new User();
4         if (option === 'module') return new Module;
5         if (option === 'lesson') return new Lesson;
6         if (option === 'microlearning') return new Microlearning;
7         return null;
8     }
9 }
```

Figura 48. Clase Factory.

Una vez establecido el patrón de diseño factory se procedió a implementar la autenticación del usuario como se indica en la Figura 49.

```

1 class Auth {
2   signup = async (req, res) => {
3     const { name, email, password, roles } = req.body;
4     const user = new User({
5       name,
6       email,
7       password
8     });
9
10    if (roles) {
11      const foundRole = await Role.find({ name: { $in: roles } });
12      user.roles = foundRole.map(role => role._id);
13    } else {
14      const role = await Role.findOne({ name: 'student' });
15      user.roles = [role._id];
16    }
17
18    await user.save((error, user) => {
19      if (error) return res.status(400).json({
20        message: 'Verifique los campos, hubo un error'
21      });
22      user.salt = undefined;
23      user.hashd_password = undefined;
24      res.status(200).json({ user });
25    });
26  }
27
28  signin = async (req, res) => {
29    const { email, password } = req.body;
30    // find the user based on email
31    await User.findOne({ email }, (error, user) => {
32      if (error || !user)
33        return res.status(400).json({ error: 'El usuario con ese correo electrónico no existe' });
34      // if the user is found, make sure the email and password match
35      // call the authentication method in the user model
36      if (!user.authenticate(password))
37        return res.status(401).json({ error: 'El correo electrónico y la contraseña no coinciden' });
38      // crear el token con vencimiento en 24 horas
39      const token = jwt.sign({ _id: user._id }, process.env.JWT_SECRET, { expiresIn: 86400 });
40      // persist the token as 't' in cookie with expiration date
41      res.cookie('t', token, { expire: new Date() + 9999 });
42      // return response with user and token to frontend client
43      const { _id, name, email, roles } = user;
44      return res.status(200).json({ token, user: { _id, email, name, roles } });
45    }).populate("roles");
46  }
47
48  signout = (req, res) => {
49    // eliminar cookie que contiene el token
50    res.clearCookie('t');
51    res.status(200).json({ message: 'Éxito de Cerrar Sesión' });
52  }
53 }

```

Figura 49. Clase Auth para la autenticación de usuario.

El siguiente proceso es crear los controladores de la sección de módulo como se presenta en la Figura 50 y se estableció las URL's referidas a este módulo, ver Figura 51.

```

1 class Module {
2   create = async (req, res) => {
3     const modules = new Modules(req.body);
4     await modules.save((err, data) => {
5       if (err) return res.status(400).json({ error: errorHandler(err) });
6       res.status(200).json({ data });
7     });
8   }
9
10  getAll = async (req, res) => {
11    await Modules.find().sort({number: 1}).exec((err, data) => {
12      if (err) return res.status(400).json({ error: errorHandler(err) });
13      res.status(200).json(data);
14    });
15  }
16
17  getById = async (req, res) => {
18    return res.status(200).json(req.module);
19  }
20
21  remove = async (req, res) => {
22    let module = req.module;
23    module.remove((err, data) => {
24      if (err) return res.status(400).json({ error: errorHandler(err) });
25      res.status(200).json({ message: 'Modulo eliminado correctamente' });
26    });
27  }
28
29  update = async (req, res) => {
30    const { number, name } = req.body;
31    let module = req.module;
32    if (number) module.number = number;
33    if (name) module.name = name;
34    module.save((err, data) => {
35      if (err) return res.status(400).json({ error: errorHandler(err) });
36      res.status(200).json({ message: 'Modulo actualizado correctamente' });
37    });
38  }
39
40  byId = async (req, res, next, id) => {
41    await Modules.findById(id).exec((err, data) => {
42      if (err || !data) return res.status(400).json({ error: 'El modulo no se encontró o no existe' });
43      req.module = data;
44      next();
45    });
46  }
47 }

```

Figura 50. Clase Module para el servicio de módulos.



```

1  const factory = new controllersFactory();
2  const ml = factory.getController('module');
3
4  // create module
5  router.post('/', [verifyToken, isAdmin], ml.create);
6  // get all modules
7  router.get('/', ml.getAll);
8  // get module by id
9  router.get('/:moduleId', ml.getById);
10 // update module
11 router.put('/:moduleId', [verifyToken, isAdmin], ml.update);
12 // delete module
13 router.delete('/:moduleId', [verifyToken, isAdmin], ml.remove);
14
15 // method to obtain the parameter
16 router.param('moduleId', ml.byId);

```

Figura 51. URL's para el servicio de módulo.

La construcción de la clase Lesson se muestra en la Figura 53 y sus URL's se presenta en la Figura 52.



```

1  const factory = new controllersFactory();
2  const lesson = factory.getController('lesson');
3
4  // create lesson
5  router.post('/', [verifyToken, isAdmin], lesson.create);
6  // get the whole lesson without the icon
7  router.get('/', lesson.getAll);
8  // get lesson by id without icon
9  router.get('/:lessonId', lesson.getById);
10 // update name and module of lesson
11 router.put('/:lessonId', [verifyToken, isAdmin], lesson.update);
12 // delete lesson
13 router.delete('/:lessonId', [verifyToken, isAdmin], lesson.remove);
14
15 // get icon
16 router.get('/icon/:lessonId', lessonIcon);
17 // update lesson icon
18 router.put('/icon/update/:lessonId', [verifyToken, isAdmin], updateLessonIcon);
19
20 // method to obtain the parameter
21 router.param('lessonId', lesson.byId);

```

Figura 52. URL's para el servicio de las Lecciones.

```

1 class Lesson {
2   create = async (req, res) => {
3     let form = new formidable.IncomingForm();
4     form.keepExtensions = true;
5     form.parse(req, async (err, fields, files) => {
6       if (err) return res.status(400).json({ error: "No se pudo cargar la imagen" });
7
8       const { name, module } = fields;
9       let lesson = new Lessons(fields);
10
11       if (files.icon) {
12         if (files.icon.size > 1000000) {
13           return res.status(400).json({ error: "La imagen debe tener un tamaño inferior a 1 MB." });
14         }
15         lesson.icon.data = fs.readFileSync(files.icon.path);
16         lesson.icon.contentType = files.icon.type;
17       }
18
19       await lesson.save((err, result) => {
20         if (err) return res.status(400).json({ error: 'No se ha creado' });
21         result.icon = undefined;
22         res.status(200).json(result);
23       });
24     });
25   }
26
27   getAll = async (req, res) => {
28     let order = req.query.order ? req.query.order : 'asc'; // variable to sort the results; ascending by default
29     let sortBy = req.query.sortBy ? req.query.sortBy : 'name'; // filter
30
31     await Lessons.find()
32       .select("-icon")
33       .populate("module")
34       .sort([[sortBy, order]])
35       .exec((err, lesson) => {
36         if (err) {
37           return res.status(400).json({ error: "Lecciones no encontrados" });
38         }
39         res.json(lesson);
40       });
41   }
42
43   getById = (req, res) => {
44     req.lesson.icon = undefined;
45     return res.status(200).json(req.lesson);
46   }
47
48   remove = (req, res) => {
49     let lesson = req.lesson;
50     lesson.remove((err, deleteLesson) => {
51       if (err) return res.status(400).json({ error: errorHandler(err) });
52       res.json({ message: "La lección se eliminó con éxito" });
53     });
54   }
55
56   update = (req, res) => {
57     const { name, module } = req.body;
58     let lesson = req.lesson;
59     if (name) lesson.name = name;
60     if (module) lesson.module = module;
61     lesson.save((err, data) => {
62       if (err) return res.status(400).json({ error: 'No se guardaron los cambios' });
63       res.status(200).json({ message: 'Lección actualizado correctamente' });
64     });
65   }
66
67   byId = async (req, res, next, id) => {
68     await Lessons.findById(id)
69       .populate("module")
70       .exec((err, lesson) => {
71         if (err || !lesson) {
72           return res.status(400).json({ error: "La lección no se encontró o no existe" });
73         }
74         req.lesson = lesson;
75         next();
76       });
77   }
78 }

```

Figura 53. Clase Lesson.

Para mostrar y actualizar las imágenes o GIF, se tomó la decisión de gestionar de forma independiente a las clases creadas, esto es para mejorar los tiempos de respuesta por parte

del servidor. En la Figura 54 se muestra las funciones para extraer y actualizar el icono de la lección.

```
1 exports.lessonIcon = (req, res, next) => {
2   if (req.lesson.icon.data) {
3     res.set('Content-Type', req.lesson.icon.contentType);
4     return res.send(req.lesson.icon.data);
5   }
6   next();
7 }
8
9 exports.updateLessonIcon = (req, res) => {
10  let form = new formidable.IncomingForm();
11  form.keepExtensions = true;
12  form.parse(req, async (err, fields, files) => {
13    if (err) return res.status(400).json({ error: "No se pudo cargar la imagen" });
14
15    let lesson = req.lesson;
16
17    if (files.icon) {
18      if (files.icon.size > 1000000) {
19        return res.status(400).json({ error: "La imagen debe tener un tamaño inferior a 1 MB." });
20      }
21      lesson.icon.data = fs.readFileSync(files.icon.path);
22      lesson.icon.contentType = files.icon.type;
23    }
24
25    await lesson.save((err, result) => {
26      if (err) return res.status(400).json({ error: errorHandler(err) });
27      res.json(result);
28    });
29  });
30 }
31
```

Figura 54. Funciones para gestionar los iconos de las lecciones.

Se crearon Middlewares (funciones que se ejecutan antes de hacer las peticiones del cliente) para verificar el token, roles del usuario (ver la Figura 56) y crear cuenta de usuario (ver Figura 55).

```
1 exports.checkDuplicateUsernameOrEmail = async (req, res, next) => {
2   const email = await User.findOne({ email: req.body.email });
3   if (email) return res.status(400).json({ message: 'El correo electrónico ya existe' });
4   next();
5 }
6
7 // check if the submitted roles exist in the database
8 exports.checkRolesExisted = (req, res, next) => {
9   const ROLES = ['student', 'admin', 'moderator'];
10  if (req.body.roles) {
11    for (let i = 0; i < req.body.roles.length; i++) {
12      if (!ROLES.includes(req.body.roles[i])) {
13        return res.status(400).json({
14          message: `El Rol ${req.body.roles[i]} no existe`
15        });
16      }
17    }
18  }
19  next();
20 }
```

Figura 55. Middleware para verificación de crear cuenta de usuario.



```

1 exports.verifyToken = async (req, res, next) => {
2   try {
3     // extract token from http header
4     const bearerHeader = req.headers['authorization'];
5     if (!bearerHeader) return res.status(403).json({ error: 'No se proporcionó token' });
6     // I separate the Bearer from the Token
7     const bearer = bearerHeader.split(' ');
8     const token = bearer[1];
9     // we verify the token and extract the id of the user
10    const decoded = jwt.verify(token, process.env.JWT_SECRET);
11    req.userId = decoded._id;
12    // we check if the user exists with the user id extracted from the token
13    const user = await User.findById(req.userId, { password: 0 });
14    if (!user) return res.status(404).json({ error: 'Ningún usuario encontrado' });
15
16    next();
17  } catch (error) {
18    res.status(401).json({ error: 'No autorizado' });
19  }
20 }
21
22 exports.isModerator = async (req, res, next) => {
23   const user = await User.findById(req.userId);
24   const roles = await Role.find({ _id: { $in: user.roles } });
25   for (let i = 0; i < roles.length; i++) {
26     if (roles[i].name === 'moderator') {
27       next();
28       return;
29     }
30   }
31   return res.status(403).json({ message: 'Requiere rol de moderador' });
32 }
33
34 exports.isAdmin = async (req, res, next) => {
35   const user = await User.findById(req.userId);
36   const roles = await Role.find({ _id: { $in: user.roles } });
37   for (let i = 0; i < roles.length; i++) {
38     if (roles[i].name === 'admin') {
39       next();
40       return;
41     }
42   }
43   return res.status(403).json({ error: 'Requiere rol de administrador' });
44 }

```

Figura 56. Middleware para verificar token y rol de usuario.

Una vez acabado la parte del back-end y realizar las pruebas respectivas, se procedió a la creación de las interfaces de usuario en el front-end. Como herramienta principal se utilizó la librería React js y la primera actividad fue configurar todo lo referido a la librería.

Como segunda actividad en el front-end, fue crear los accesos (ver Figura 57) y rutas para el administrador.

Log In

Email

Password

Enviar

Figura 57. Formulario para iniciar sesión de administrador.

La tercera actividad en el front-end es el de implementar los módulos de gestionar secciones (ver desde la Figura 58 a la Figura 60).

Lista de Modulos		
# Modulo	Nombre	Opciones
1	Introducción	Eliminar Modificar
2	Nivel 2	Eliminar Modificar
3	Nivel 3	Eliminar Modificar
4	Nivel 4	Eliminar Modificar

Figura 58. Interfaz administrador, listar todos los módulos con sus respectivas opciones.

Crear nuevo módulo

Número de módulo

Nombre

Regresar
Enviar

Figura 59. Interfaz administrador, formulario para crear módulo.

Modificar el módulo Introducción

Número de módulo

Nombre

Regresar
Enviar

Figura 60. Interfaz administrador, formulario para modificar módulo.

La cuarta actividad en el front-end, se realizó el módulo de gestión de lecciones (ver desde la Figura 61 a la Figura 64).

Lista de Lecciones

# Modulo	Nombre	Opciones		
1	Emergencia	Eliminar	Ver más	Modificar
2	Emergencia 2	Eliminar	Ver más	Modificar
1	Familia	Eliminar	Ver más	Modificar
2	Familia 2	Eliminar	Ver más	Modificar
1	Mascotas	Eliminar	Ver más	Modificar
1	Restaurante	Eliminar	Ver más	Modificar
2	Restaurante 2	Eliminar	Ver más	Modificar
1	Saludos	Eliminar	Ver más	Modificar

Figura 61. Interfaz de administrador, listar las lecciones creadas con las diferentes opciones.

Crear nueva Lección

Icono

Elegir archivo
No s...rchivo

Nombre de la lección

Módulo

Selecciona Módulo
▼

Regresar
Enviar

Figura 62. Interfaz de administrador, formulario para crear una lección

Ver información de la lección



Número de Módulo:

1

Nombre de Módulo:

Introducción

Modificar Lección

Familia

Figura 63. Interfaz de administrador, panel para ver la información de la lección.

Actualizar lección Familia

Icono

Elegir archivo
No s...rchivo

Enviar

Nombre de la lección

Módulo

Introducción
▼

Regresar
Enviar

Figura 64. Interfaz de administrador, formularios para actualizar información de la lección.

1.4. Pruebas

Para realizar las pruebas de los endpoints de la API se utilizó **Jest** y **SuperTest** durante la fase de codificación para la comprobación del correcto funcionamiento (ver desde la Figura 65 a la Figura 69).

```
1 describe('POST /api/auth/signup', () => {
2
3   test('CP01 create an end user', async () => {
4     await api
5       .post('/api/auth/signup')
6       .send(student)
7       .expect(200)
8       .expect('Content-Type', /application\/json/)
9     const response = await User.findOne({ email: student.email });
10    expect(response.email).toBe(student.email);
11  });
12
13  test('CP02 end user with repeated email cannot create the account', async () => {
14    await api
15      .post('/api/auth/signup')
16      .send(student)
17      .expect(400)
18      .expect('Content-Type', /application\/json/)
19    const response = await User.find({ email: student.email });
20    expect(response.length).toBe(1);
21  })
22
23  test('CP03 an end user cannot create the account if they do not fill in all the parameters', async () => {
24    const newEstudent = {
25      name: 'Jonnathan',
26      email: 'jonnathan@unl.edu.ec',
27    }
28    await api
29      .post('/api/auth/signup')
30      .send(newEstudent)
31      .expect(400)
32      .expect('Content-Type', /application\/json/)
33    const response = await User.find({ email: student.email });
34    expect(response.length).toBe(1);
35  })
36
37  test('CP04 create a admin', async () => {
38    await api
39      .post('/api/auth/signup')
40      .send(adminUser)
41      .expect(200)
42      .expect('Content-Type', /application\/json/)
43    const response = await User.findOne({ email: student.email });
44    expect(response.email).toBe(student.email);
45  });
46 });
```

Figura 65. Pruebas de los endpoints de la API para creación de usuario.

```

PASS test/auth.test.js
POST /api/auth/signup
  ✓ CP01 create an end user (149 ms)
  ✓ CP02 end user with repeated email cannot create the account (39 ms)
  ✓ CP03 an end user cannot create the account if they do not fill in all the parameters (45 ms)
  ✓ CP04 create a admin (65 ms)
POST /api/auth/signin
  ✓ CP05 log in end user (48 ms)
  ✓ CP06 the end user cannot enter with a different email address (30 ms)
  ✓ CP07 end user cannot login with wrong password (52 ms)
POST /api/auth/signinadmin
  ✓ CP08 login admin (32 ms)
  ✓ CP09 the admin cannot enter with a different email address (26 ms)
  ✓ CP10 admin cannot login with wrong password (28 ms)

```

Figura 66. Resultado de pruebas realizadas a los endpoints de la API para autenticación de usuario.

```

PASS test/module.test.js
POST /api/module
  ✓ CP11 create a module with an authenticated admin (350 ms)
  ✓ CP12 a user without permission cannot create a module (161 ms)
  ✓ CP13 a module cannot be created if the token is not sent (46 ms)
  ✓ CP14 a module cannot be created if the token has been modified (42 ms)
GET /api/module/
  ✓ CP15 modules are returned as array (57 ms)
  ✓ CP16 there are two modules (62 ms)
  ✓ CP17 the first module is about number 1 (51 ms)
PUT /api/module/
  ✓ CP18 update a module with authenticated admin (101 ms)
  ✓ CP19 the module cannot be updated if the user is not admin (77 ms)
  ✓ CP20 the module cannot be updated if the token has been modified (58 ms)
  ✓ CP21 update the module name (93 ms)
  ✓ CP22 update the module number (91 ms)
DELETE /api/module/
  ✓ CP23 the module cannot be removed if the id does not match (51 ms)
  ✓ CP24 the module cannot be deleted if the user is not admin (86 ms)
  ✓ CP25 the module cannot be deleted if the token has been modified (57 ms)
  ✓ CP26 delete a module with authenticated admin (82 ms)

```

Figura 67. Resultado de pruebas realizadas a los endpoints de la API para sección de módulos.

```

PASS test/lesson.test.js
  POST /api/lesson/
    ✓ CP27 create a new lesson with an authenticated admin (578 ms)
  GET /api/lesson
    ✓ CP28 get all lesson (59 ms)
    ✓ CP29 get lesson by id (90 ms)
    ✓ CP30 if an invalid id is provided it presents an error (74 ms)
  PUT /api/lesson/
    ✓ CP31 update the parameter of the lesson name (116 ms)
    ✓ CP32 update the parameter of the lesson module (135 ms)
    ✓ CP33 update name and module of the lesson (140 ms)
    ✓ CP34 update lesson icon (167 ms)
  DELETE /api/lesson/
    ✓ CP35 the lesson cannot be deleted if the id does not match (73 ms)
    ✓ CP36 the lesson cannot be deleted if the token is not sent (94 ms)
    ✓ CP37 the lesson cannot be eliminated if the token is not valid (85 ms)
    ✓ CP38 the lesson cannot be deleted if the user is not admin (186 ms)
    ✓ CP39 delete a lesson with authenticated admin (103 ms)

```

Figura 68. Resultado de pruebas realizadas a los endpoints de la API para sección de Lecciones.

```

Test Suites: 3 passed, 3 total
Tests:      39 passed, 39 total
Snapshots:  0 total
Time:       8.36 s, estimated 13 s
Ran all test suites.

```

Figura 69. Total, de pruebas realizadas a los endpoints de la API en la iteración 1.

Las pruebas del front-end se realiza a las interfaces, por ello existen librerías especializadas en hacer testing a los componentes que se ven en pantalla. React JS utiliza **Jest** por defecto porque ambas fueron creadas por Facebook y para probar los componentes se utilizó la librería **react testing library**. En la Figura 70 se muestra un ejemplo de las pruebas unitarias de una interfaz, en la Figura 71 se presentan los resultados de los mismos y en la Figura 72 está el total de pruebas.



```
1 import '@testing-library/jest-dom'
2 import { render, screen, fireEvent } from '@testing-library/react';
3 import { MemoryRouter } from 'react-router-dom';
4 import Signup from './Signup';
5
6 beforeEach(() => {
7   render(<MemoryRouter><Signup /></MemoryRouter>)
8 })
9
10 describe('<Signup />', () => {
11
12   test('CP01 render signup user', () => {
13     screen.getByText('Email')
14     screen.getByText('Password')
15     screen.getByTestId('btnSubmit')
16     screen.getByTestId('linkLogin')
17   });
18
19   test('CP02 input name change', () => {
20     const inputPassword = screen.getByTestId('name')
21     expect(inputPassword.value).toBe("");
22     fireEvent.change(inputPassword, { target: { value: 'myname' } });
23     expect(inputPassword.value).toBe('myname');
24   });
25
26   test('CP03 input email change', () => {
27     const inputEmail = screen.getByTestId('email')
28     expect(inputEmail.value).toBe("");
29     fireEvent.change(inputEmail, { target: { value: 'jonathan@test.com' } });
30     expect(inputEmail.value).toBe('jonathan@test.com');
31   });
32
33   test('CP04 input password change', () => {
34     const inputPassword = screen.getByTestId('password')
35     expect(inputPassword.value).toBe("");
36     fireEvent.change(inputPassword, { target: { value: 'mypassword' } });
37     expect(inputPassword.value).toBe('mypassword');
38   });
39
40 });
```

Figura 70. Prueba unitaria de la interfaz para crear usuario.

```

PASS src/pages/DashBoard/DashBoard.test.js (25.515 s)
PASS src/pages/UpdateModule/UpdateModule.test.js (35.617 s)
PASS src/pages/ShowUsers/ShowUsers.test.js
PASS src/pages/ShowLessons/ShowLessons.test.js (13.366 s)
PASS src/pages/ShowModules/ShowModules.test.js (39.801 s)
PASS src/pages/Home/Home.test.js (39.925 s)
PASS src/pages/UpdateLesson/UpdateLesson.test.js
PASS src/pages/AddModules/AddModules.test.js
PASS src/pages/LoginUser/LoginUser.test.js
PASS src/pages/SignUp/SignUp.test.js
PASS src/pages/AddLesson/AddLesson.test.js
PASS src/pages/LoginAdmin/LoginAdmin.test.js
PASS src/pages/ViewLesson/ViewLesson.test.js
PASS src/components/UpdateIcon/UpdateIcon.test.js
PASS src/components/NavigationAdmin/NavigationAdmin.test.js
PASS src/components/Navigation/Navigation.test.js
PASS src/pages/NotFoundPage/NotFoundPage.test.js

```

Figura 71. Resultado de las pruebas en el front-end iteración 1.

```

Test Suites: 17 passed, 17 total
Tests:       34 passed, 34 total
Snapshots:  0 total
Time:        69.037 s
Ran all test suites.

```

Figura 72. Total, de pruebas front-end iteración 1.

2. Iteración 2

2.1. Planificación

En la iteración 2 para el desarrollo de la aplicación web de acuerdo a la metodología XP, se realizaron las incidencias de la Tabla 78.

Tabla 78.
Tareas de la iteración 2

Clave de Incidencia	Resumen	Clave HU	Responsable	Estado
LSEC-17	Visualizar microcontenido	HU18	Jonnathan Espinoza	Completado
LSEC-16	Crear microcontenido	HU17	Jonnathan Espinoza	Completado
LSEC-18	Actualizar microcontenido	HU19	Jonnathan Espinoza	Completado

LSEC-19	Borrar microcontenido	HU20	Jonnathan Espinoza	Completado
LSEC-27	Visualizar cápsulas de aprendizaje en la interfaz de usuario final	HU21	Jonnathan Espinoza	Completado

2.2. Diseño

En la Figura 73 se presenta el diagrama de clases y los de color azul representan las clases sobre la cual se trabajó en la segunda iteración.

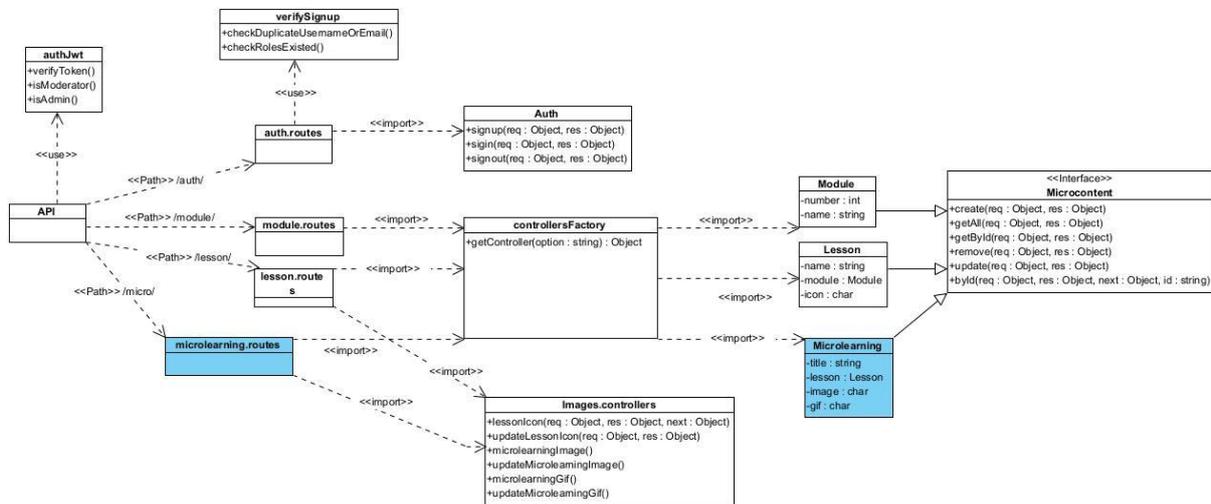


Figura 73. Diagrama de Clases de la API iteración 2.

2.3. Codificación

La elaboración de los métodos para gestionar los microcontenidos por lado del backend se muestra en la Figura 75 y las URL's para acceder a los servicios, se presenta en la Figura 74.



```
1 const factory = new controllersFactory();
2 const micro = factory.getController('microlearning');
3
4 // create microlearning
5 router.post('/', [verifyToken, isAdmin], micro.create);
6 // get the whole microlearning without the image and gif
7 router.get('/', micro.getAll);
8 // get microlearning by id without image and gif
9 router.get('/:microId', micro.getById);
10 // delete microlearning
11 router.delete('/:microId', [verifyToken, isAdmin], micro.remove);
12 // update title and lesson of microlearning
13 router.put('/:microId', [verifyToken, isAdmin], micro.update);
14
15 // get image
16 router.get('/image/:microId', microlearningImage);
17 // update image
18 router.put('/image/:microId', [verifyToken, isAdmin], updateMicrolearningImage);
19 // get gif
20 router.get('/gif/:microId', microlearningGif);
21 // update gif
22 router.put('/gif/:microId', [verifyToken, isAdmin], updateMicrolearningGif);
23
24 // method to obtain the parameter
25 router.param('microId', micro.byId);
```

Figura 74. URL's para acceder a los servicios de microcontenidos.

```

1 class Microlearning {
2   create = async (req, res) => {
3     let form = new formidable.IncomingForm();
4     form.keepExtensions = true;
5     form.parse(req, async (err, fields, files) => {
6
7       if (err) return res.status(400).json({ error: 'No se pudo cargar la imagen' });
8
9       let micro = new Micro(fields);
10
11      if (files.image) {
12        if (files.image.size > 9000000) {
13          return res.status(400).json({ error: 'La imagen debe tener un tamaño inferior a 9 MB.' });
14        }
15        micro.image.data = fs.readFileSync(files.image.path);
16        micro.image.contentType = files.image.type;
17      }
18
19      if (files.gif) {
20        if (files.gif.size > 9000000) {
21          return res.status(400).json({ error: 'El GIF debe tener un tamaño inferior a 9 MB.' });
22        }
23        micro.gif.data = fs.readFileSync(files.gif.path);
24        micro.gif.contentType = files.gif.type;
25      }
26
27      await micro.save((err, result) => {
28        if (err) return res.status(400).json({ error: 'No se pudo crear el microcontenido' });
29        res.status(200).json(result);
30      });
31    });
32  }
33
34  getAll = async (req, res) => {
35    let order = req.query.order ? req.query.order : 'asc'; // variable to sort the results; ascending by default
36    let sortBy = req.query.sortBy ? req.query.sortBy : 'title'; // filter
37
38    await Micro.find()
39      .select(['-image', '-gif'])
40      .populate('lesson')
41      .sort([[sortBy, order]])
42      .exec((err, micro) => {
43        if (err) return res.status(400).json({ error: 'Microcontenidos no encontrados' });
44        res.status(200).json(micro);
45      });
46  }
47
48  getById = (req, res) => {
49    req.microlearning.image = undefined;
50    req.microlearning.gif = undefined;
51    return res.status(200).json(req.microlearning);
52  }
53
54  remove = (req, res) => {
55    let microlearning = req.microlearning;
56    microlearning.remove((err, deleteMicro) => {
57      if (err) return res.status(400).json({ error: errorHandler(err) });
58      res.status(200).json({ message: 'El microcontenido se eliminó con éxito' });
59    });
60  }
61
62  update = (req, res) => {
63    const { title, lesson } = req.body;
64    let microlearning = req.microlearning;
65    if (title) microlearning.title = title;
66    if (lesson) microlearning.lesson = lesson;
67    microlearning.save((err, data) => {
68      if (err) return res.status(400).json({ error: errorHandler(err) });
69      res.status(200).json({ message: 'El microcontenido se actualizado correctamente' });
70    });
71  }
72
73  byId = async (req, res, next, id) => {
74    await Micro.findById(id)
75      .populate('lesson', 'name')
76      .exec((err, micro) => {
77        if (err || !micro) return res.status(400).json({ error: 'El microcontenido no se encontró o no existe' });
78        req.microlearning = micro;
79        next();
80      });
81  }
82 }

```

Figura 75. Clase Microlearning para gestionar los microcontenidos.

Para mostrar los contenidos al estudiante, se crearon funciones que retornan la información necesaria para el frontend (ver Figura 76) y las URL's para acceder a estas funciones se presenta en la Figura 77.

```

1 exports.modulesAssigned = async (req, res) => {
2   try {
3     const modules = await Modules.find().sort({ number: 1 });
4     let results = [];
5     for (const m of modules) {
6       const lessons = await Lessons.find({ module: m._id }).select(['-icon']);
7       if (lessons.length !== 0) {
8         results.push(m);
9       }
10    }
11    return res.status(200).json(results);
12  } catch (error) {
13    return res.status(400).json({ error: 'No se cargaron los datos. Recargar la página' });
14  }
15 }
16
17 exports.lessonsAssigned = async (req, res) => {
18   const module = await Modules.findById(req.params.moduleId);
19   if (!module) return res.status(400).json({ error: 'El modulo no se encontró o no está registrado' });
20   const lessons = await Lessons.find({ module: module._id }).select(['-icon']);
21   return res.status(200).json(lessons);
22 }
23
24 exports.microlearningAssigned = async (req, res) => {
25   const microlearnings = await Microlearning.find({ lesson: req.lesson_id }).select(['-image', '-gif']);
26   if (microlearnings.length === 0) return res.status(400).json({ error: 'No se ha registrado contenido para esta lección' });
27   return res.status(200).json(microlearnings);
28 }
29
30 exports.lessonById = async (req, res, next, id) => {
31   await Lessons.findById(id)
32     .populate("module")
33     .exec((err, lesson) => {
34       if (err || !lesson) {
35         return res.status(400).json({ error: "La lección no se encontró o no existe" });
36       }
37       req.lesson = lesson;
38       next();
39     });
40 }

```

Figura 76. Funciones para mostrar la información a la interfaz del estudiante.

```

1 // get the modules created if they have associated lessons
2 router.get('/modules', modulesAssigned);
3 // get the lessons from a module
4 router.get('/lessons/:moduleId', lessonsAssigned);
5 // get the micro-content of a lesson
6 router.get('/learn/:lessonId', microlearningAssigned)
7
8 // method to obtain the parameter
9 router.param('lessonId', lessonById);
10
11 module.exports = router;

```

Figura 77. URL's para proveer información al estudiante.

Una vez desarrollado la funcionalidad del backend y hacer las pruebas correspondientes, se procedió al desarrollo del frontend (ver desde la Figura 78 a la Figura 83).

Lista de Microcontenido		
Nombre	Lección	Opciones
Manzana	Restaurante	Eliminar Ver más Modificar
Sopa	Restaurante	Eliminar Ver más Modificar

Figura 78. Interfaz de administrador, mostrar los microcontenidos.

Crear nuevo Microcontenido

Título del Microcontenido

Lección
Selecciona una Lección ▼

Imágen representativa
Seleccionar archivo Ni...do

Gif
Seleccionar archivo Ni...do

[Regresar](#) [Enviar](#)

Figura 79. Interfaz de administrador, crear microcontenido.

Actualizar Microcontenido

Imágen Representativa
Seleccionar archivo Ni...do
[Enviar](#)

Gif de la lengua de señas
Seleccionar archivo Ni...do
[Enviar](#)

Título del Microcontenido
Manzana
Lección
Restaurante ▼
[Regresar](#) [Enviar](#)

Figura 80. Interfaz administrador, actualizar microcontenido.



Figura 81. Interfaz administrador, mostrar información del microcontenido.

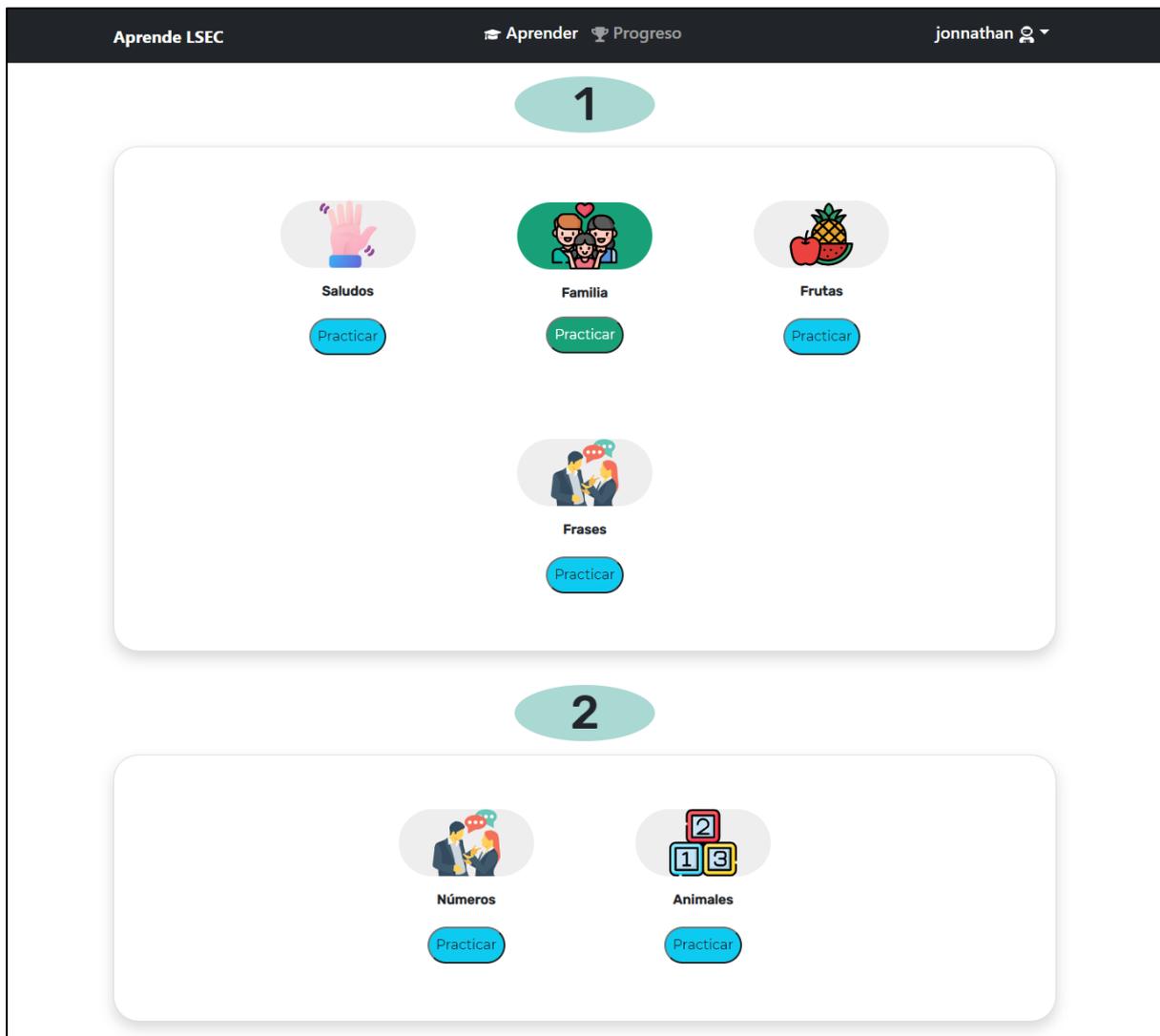


Figura 82. Interfaz estudiante, mostrar módulos y lecciones.

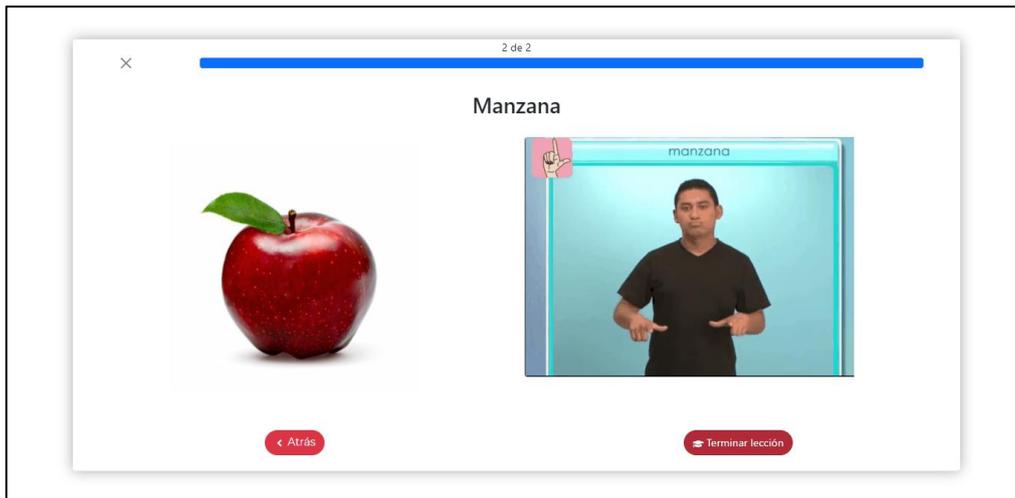


Figura 83. Interfaz estudiante, mostrar cápsula de aprendizaje.

2.4. Pruebas

Las pruebas realizadas a la API en la segunda iteración se presentan a continuación (desde la Figura 84 a la Figura 86):

```

1 test('CP46 update the parameter of the microcontent title', async () => {
2     const micros = await api
3       .get('/api/micro/')
4       .expect(200)
5       .expect('Content-Type', /application\/json/)
6     const updateMicro = {
7       title: 'Micro 137',
8     }
9     const response = await api
10      .put(`/api/micro/${micros.body[0]._id}`)
11      .send(updateMicro)
12      .set('authorization', `Bearer ${tokenAdmin}`)
13      .expect('Content-Type', /application\/json/)
14      .expect(200)
15     expect(response.body.message).toBe('El microcontenido se actualizado correctamente');
16 });

```

Figura 84. Prueba del endpoint de la API para actualizar un microcontenido.

```

PASS test/microlearning.test.js (13.904 s)
  POST /api/micro/
    ✓ CP40 create a new microcontent with an authenticated admin (2676 ms)
    ✓ CP41 the microcontent is not created if it does not send all the parameters (2273 ms)
    ✓ CP42 microcontent is not created if user is not admin (266 ms)
  GET /api/micro/
    ✓ CP43 get all microcontent (124 ms)
    ✓ CP44 get microcontent by id (112 ms)
    ✓ CP45 if an invalid id is provided it presents an error (121 ms)
  PUT /api/micro/
    ✓ CP46 update the parameter of the microcontent title (155 ms)
    ✓ CP47 update the parameter of the microcontent lesson (178 ms)
    ✓ CP48 update title and lesson of the microcontent (181 ms)
    ✓ CP49 update microcontent image (1420 ms)
    ✓ CP50 the image does not update if an image is not submitted (144 ms)
    ✓ CP51 update microcontent gif (1385 ms)
    ✓ CP52 the gif does not update if an gif is not submitted (133 ms)
  DELETE /api/micro/
    ✓ CP53 the microcontent cannot be deleted if the id does not match (135 ms)
    ✓ CP54 the microcontent cannot be deleted if the token is not sent (133 ms)
    ✓ CP55 the microcontent cannot be eliminated if the token is not valid (132 ms)
    ✓ CP56 the microcontent cannot be deleted if the user is not admin (127 ms)
    ✓ CP57 delete a microcontent with authenticated admin (1038 ms)

```

Figura 85. Resultado de pruebas realizadas a los endpoints de la API para gestión de microcontenido.

```

Test Suites: 4 passed, 4 total
Tests:       57 passed, 57 total
Snapshots:  0 total
Time:        19.85 s, estimated 26 s
Ran all test suites.

```

Figura 86. Total, de pruebas realizadas a los endpoints de la API en la iteración 2.

Las pruebas para el front-end se presentan desde la Figura 87 a la Figura 89.



```
1 import '@testing-library/jest-dom'
2 import { render, fireEvent } from '@testing-library/react';
3 import AddMicrolearning from './AddMicrolearning';
4 import { MemoryRouter } from 'react-router-dom';
5
6 let component
7
8 beforeEach(() => {
9     component = render(<MemoryRouter><AddMicrolearning /></MemoryRouter>)
10 })
11
12 describe('<AddMicrolearning />', () => {
13
14     test('CP36 renders content', () => {
15         component.getByText('Título del Microcontenido')
16         component.getByText('Lección')
17         component.getByText('Selecciona una Lección')
18         component.getByText('Imagen representativa')
19         component.getByText('Gif')
20         component.getByText('Regresar')
21         component.getByText('Crear')
22     });
23
24     test('CP37 input title name change', () => {
25         const inputName = component.getByTestId('inputTitle')
26         expect(inputName.value).toBe('');
27         fireEvent.change(inputName, { target: { value: 'Name' } });
28         expect(inputName.value).toBe('Name');
29     });
30
31     test('CP38 clicking the button regresar', () => {
32         const btnCrear = component.getByText('Regresar')
33         fireEvent.click(btnCrear)
34     });
35
36 });
```

Figura 87. Prueba unitaria de la interfaz para crear un microcontenido.

```

PASS src/pages/AddLesson/AddLesson.test.js
PASS src/pages/UpdateModule/UpdateModule.test.js
PASS src/pages/ShowMicrolearning/ShowMicrolearning.test.js
PASS src/pages/Home/Home.test.js (5.037 s)
PASS src/components/UpdateGif/UpdateGif.test.js
PASS src/components/UpdateImage/UpdateImage.test.js
PASS src/pages/UpdateLesson/UpdateLesson.test.js
PASS src/pages/ShowUsers/ShowUsers.test.js
PASS src/pages/UpdateMicrolearning/UpdateMicrolearning.test.js
PASS src/pages/AddMicrolearning/AddMicrolearning.test.js
PASS src/pages/AddModules/AddModules.test.js
PASS src/pages/ShowLessons/ShowLessons.test.js
PASS src/pages/ShowModules/ShowModules.test.js
PASS src/pages/SignUp/SignUp.test.js
PASS src/pages/LoginUser/LoginUser.test.js
PASS src/pages/LoginAdmin/LoginAdmin.test.js
PASS src/pages/ViewMicrolearning/ViewMicrolearning.test.js
PASS src/pages/DashBoard/DashBoard.test.js
PASS src/components/UpdateIcon/UpdateIcon.test.js
PASS src/pages/ViewLesson/ViewLesson.test.js
PASS src/components/NavigationAdmin/NavigationAdmin.test.js
PASS src/components/Navigation/Navigation.test.js
PASS src/pages/LandingPage/LandingPage.test.js
PASS src/components/ShowImage/ShowImage.test.js
PASS src/pages/NotFoundPage/NotFoundPage.test.js

```

Figura 88. Resultado de las pruebas en el front-end iteración 2.

```

Test Suites: 25 passed, 25 total
Tests:       46 passed, 46 total
Snapshots:  0 total
Time:        12.23 s, estimated 13 s
Ran all test suites.

```

Figura 89. Total, de pruebas front-end iteración 2.

3. Iteración 3

3.1. Planificación

En la Tabla 79 se presenta las incidencias para la tercera iteración.

Tabla 79.
Tareas de la iteración 3

Clave de Incidencia	Resumen	Clave HU	Responsable	Creada
LSEC-20	Crear ejercicio	HU12	Jonnathan Espinoza	Completado

LSEC-21	Visualizar ejercicio	HU13	Jonnathan Espinoza	Completado
LSEC-22	Actualizar ejercicio	HU14	Jonnathan Espinoza	Completado
LSEC-23	Borrar ejercicio	HU15	Jonnathan Espinoza	Completado
LSEC-28	Visualizar lecciones completadas	HU16	Jonnathan Espinoza	Completado

3.2. Diseño

El diagrama de clases referente a la iteración 3 se presenta en la Figura 90, la parte de azul, son las clases sobre la cual se trabajó en la presente iteración.

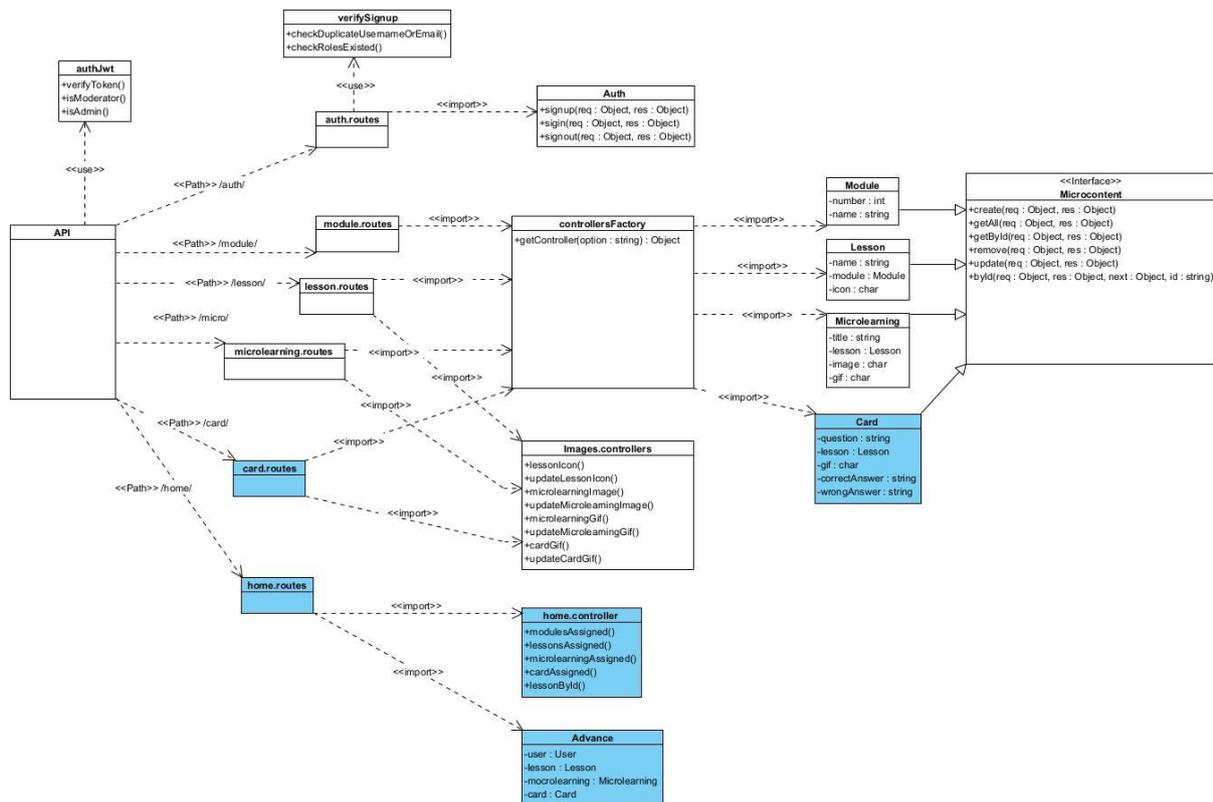


Figura 90. Diagrama de clases de la API iteración 3.

3.3. Codificación

La primera parte de la iteración 3, es el desarrollo del módulo de pruebas (Tarjeta de Conocimiento) en el backend, en donde, se creó el modelo de las Tarjetas de Conocimiento con sus respectivas funciones (ver Figura 91 y Figura 92).

```

1 class Card {
2   create = async (req, res) => {
3     let form = new formidable.IncomingForm();
4     form.keepExtensions = true;
5     form.parse(req, async (err, fields, files) => {
6       if (err) return res.status(400).json({ error: "No se pudo cargar el gif" });
7
8       let card = new Cards(fields);
9
10      if (files.gif) {
11        if (files.gif.size > 9000000) return res.status(400).json({ error: 'El gif debe de tener un tamaño inferior a 9 MB' });
12
13        card.gif.data = fs.readFileSync(files.gif.path);
14        card.gif.contentType = files.gif.type;
15      }
16
17      await card.save((err, result) => {
18        if (err) return res.status(400).json({ error: 'No se ha creado' });
19        result.gif = undefined;
20        res.status(200).json(result);
21      });
22    });
23  }
24
25  getAll = async (req, res) => {
26    let order = req.query.order ? req.query.order : 'asc';
27    let sortBy = req.query.sortBy ? req.query.sortBy : 'name';
28
29    await Cards.find()
30      .select(['-gif'])
31      .populate('lesson', '-icon')
32      .sort([[sortBy, order]])
33      .exec((err, card) => {
34        if (err) return res.status(400).json({ error: 'Tarjetas de Conocimiento no encontradas' });
35        res.status(200).json(card);
36      });
37  }
38
39  getById = (req, res) => {
40    req.card.gif = undefined;
41    res.status(200).json(req.card);
42  }
43
44  remove = async (req, res) => {
45    let card = req.card;
46    await card.remove((err, deleteCard) => {
47      if (err) return res.status(400).json({ error: 'La tarjeta de conocimiento no se eliminó' });
48      res.status(200).json({ message: 'La tarjeta de conocimiento se eliminó con éxito' });
49    });
50  }
51
52  update = async (req, res) => {
53    const { question, lesson, correctAnswer, wrongAnswer } = req.body;
54    let card = req.card;
55
56    if (question) card.question = question;
57    if (lesson) card.lesson = lesson;
58    if (correctAnswer) card.correctAnswer = correctAnswer;
59    if (wrongAnswer) card.wrongAnswer = wrongAnswer;
60
61    await card.save((err, data) => {
62      if (err) return res.status(400).json({ error: 'La tarjeta de conocimiento no se ha actualizado' });
63      res.status(200).json({ message: 'La tarjeta de conocimiento se actualizado correctamente' });
64    });
65  }
66
67  byId = async (req, res, next, id) => {
68    await Cards.findById(id)
69      .populate('lesson', '-icon')
70      .exec((err, card) => {
71        if (err || !card) return res.status(400).json({ error: 'La tarjeta de conocimiento no se encontró o no existe' });
72        req.card = card;
73        next();
74      });
75  }
76 }

```

Figura 91. Clase Card para el módulo de Pruebas LSEC.

```

1  const factory = new controllersFactory();
2  const card = factory.getController('card');
3
4  // create a new knowledge card
5  router.post('/', [verifyToken, isAdmin], card.create);
6  // get the whole card without the gif
7  router.get('/', card.getAll);
8  // get knowledge card by id without gif
9  router.get('/:cardId', card.getById);
10 // delete knowledge card
11 router.delete('/:cardId', [verifyToken, isAdmin], card.remove);
12 // update knowledge card
13 router.put('/:cardId', [verifyToken, isAdmin], card.update)
14
15 // get gif
16 router.get('/gif/:cardId', cardGif);
17 // update gif
18 router.put('/gif/:cardId', [verifyToken, isAdmin], updateCardGif);
19
20 router.param('cardId', card.byId);

```

Figura 92. URL's para acceder a los servicios de la clase Card.

Una vez creado la funcionalidad en el backend y hacer las respectivas pruebas, se realizó el desarrollo en el frontend para el módulo de Pruebas. Ver desde la Figura 93 a la Figura 97.

Pregunta	Lección	Opciones
¿Qué significa la siguiente lengua de señas?	Restaurante	Eliminar Ver más Modificar
Update question	Restaurante	Eliminar Ver más Modificar
Seleccione la respuesta correcta	Familia 2	Eliminar Ver más Modificar
Seleccione la respuesta correcta	Restaurante	Eliminar Ver más Modificar
¿Cuál es la respuesta correcta?	Saludos	Eliminar Ver más Modificar

Figura 93. Interfaz administrador, listar las pruebas creadas.

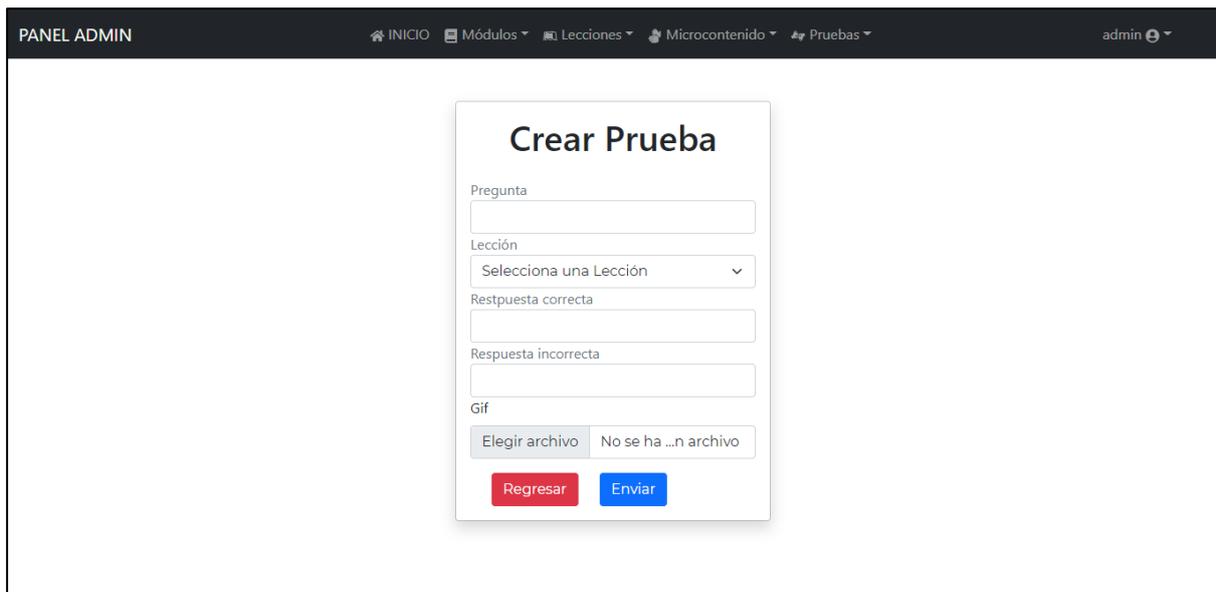


Figura 94. Interfaz administrador, crear prueba.



Figura 95. Interfaz administrador, ver información de la prueba.



Figura 96. Interfaz administrador, actualizar prueba.



Figura 97. Interfaz estudiante, tarjeta de conocimiento.

La segunda parte es el desarrollo de la historia de usuario HU16 en el backend (ver Figura 98).

```

1 exports.addCompleteLesson = async (req, res) => {
2   const { user } = req.body;
3   const learn = await Learn.find({ user }).select(['-microlearning']);
4   await Learn.updateOne(
5     { _id: learn[0]._id },
6     { $addToSet: { lesson: [req.lesson._id] } },
7     function (err, managerparent) {
8       if (err) return res.status(400).json({ error: 'no agregado la lección culminada' });
9       res.status(200).json({ message: managerparent });
10    });
11 }
12
13 exports.getCompleteLearn = async (req, res) => {
14   const userId = req.params.userId;
15   const learn = await Learn.findOne({ user: userId });
16   return res.status(200).json(learn);
17 }

```

Figura 98. Métodos para gestionar los avances de lecciones de los estudiantes.

Una vez finalizado la parte del backend, se procedió al desarrollo en el frontend de la HU16 (ver Figura 99).



Figura 99. Interfaz estudiante, ver avances de las lecciones.

3.4. Pruebas

Las pruebas realizadas a la API para la iteración 3, se presentan desde la Figura 100 a la Figura 102.

```

1 test('CP64 update the parameter of the knowledge card question', async () => {
2   const cards = await api
3     .get('/api/card')
4     .expect(200)
5     .expect('Content-Type', /application\/json/)
6   const updateCard = {
7     question: 'Card 137',
8   }
9   const response = await api
10    .put(`/api/card/${cards.body[0]._id}`)
11    .send(updateCard)
12    .set('authorization', `Bearer ${tokenAdmin}`)
13    .expect('Content-Type', /application\/json/)
14    .expect(200)
15    expect(response.body.message).toBe('La tarjeta de conocimiento se actualizado correctamente');
16  });

```

Figura 100. Prueba del endpoint de la API para actualizar la Tarjeta de Aprendizaje.

```

PASS test/card.test.js (9.342 s)
  POST /api/card
    ✓ CP58 create a knowledge card (1691 ms)
    ✓ CP59 the knowledge card is not created if it does not send all the parameters (1316 ms)
    ✓ CP60 knowledge card is not created if user is not admin (480 ms)
  GET /api/card
    ✓ CP61 get all knowledge cards (121 ms)
    ✓ CP62 get knowledge card by id (174 ms)
    ✓ CP63 if an invalid id is provided it presents an error (113 ms)
  PUT /api/card
    ✓ CP64 update the parameter of the knowledge card question (225 ms)
    ✓ CP65 update the parameter of the knowledge card lesson (356 ms)
    ✓ CP66 update correctAnswer and wrongAnswer of the knowledge card (196 ms)
    ✓ CP67 update knowledge card gif (1600 ms)
    ✓ CP68 the gif does not update if an gif is not submitted (232 ms)
  DELETE /api/card
    ✓ CP69 the knowledge cards cannot be deleted if the id does not match (143 ms)
    ✓ CP70 the knowledge cards cannot be deleted if the token is not sent (172 ms)
    ✓ CP71 the knowledge cards cannot be eliminated if the token is not valid (159 ms)
    ✓ CP72 the knowledge cards cannot be deleted if the user is not admin (145 ms)
    ✓ CP73 delete a knowledge cards with authenticated admin (625 ms)

```

Figura 101. Resultado de pruebas realizadas a los endpoints de la API para tarjetas de aprendizaje.

```

Test Suites: 5 passed, 5 total
Tests:       73 passed, 73 total
Snapshots:   0 total
Time:        36.078 s
Ran all test suites.

```

Figura 102. Total, de pruebas realizadas a los endpoints de la API en la iteración 3.

Las pruebas para el front-end se presentan desde la Figura 103 a la Figura 105.



```
1 import '@testing-library/jest-dom'
2 import { render, fireEvent } from '@testing-library/react';
3 import { MemoryRouter } from 'react-router-dom';
4 import AddCard from './AddCard';
5
6 let component
7
8 beforeEach(() => {
9     component = render(
10         <MemoryRouter>
11             <AddCard />
12         </MemoryRouter>
13     )
14 })
15
16 describe('<AddCard />', () => {
17
18     test('CP48 renders content', () => {
19         component.getByText('Pregunta')
20         component.getByText('Lección')
21         component.getByText('Selecciona una Lección')
22         component.getByText('Respuesta correcta')
23         component.getByText('Respuesta incorrecta')
24         component.getByText('Gif')
25         component.getByText('Regresar')
26         component.getByText('Crear')
27     });
28
29     test('CP49 input question change', () => {
30         const inputName = component.getByTestId('inputQuestion')
31         expect(inputName.value).toBe("");
32         fireEvent.change(inputName, { target: { value: 'Name' } });
33         expect(inputName.value).toBe('Name');
34     });
35
36     test('CP50 input CorrectAnswer change', () => {
37         const inputName = component.getByTestId('inputCorrectAnswer')
38         expect(inputName.value).toBe("");
39         fireEvent.change(inputName, { target: { value: 'Name' } });
40         expect(inputName.value).toBe('Name');
41     });
42
43     test('CP51 input WrongAnswer change', () => {
44         const inputName = component.getByTestId('inputWrongAnswer')
45         expect(inputName.value).toBe("");
46         fireEvent.change(inputName, { target: { value: 'Name' } });
47         expect(inputName.value).toBe('Name');
48     });
49
50     test('CP52 clicking the button regresar', () => {
51         const btnCrear = component.getByText('Regresar')
52         fireEvent.click(btnCrear)
53     });
54
55 });
```

Figura 103. Prueba unitaria de la interfaz para crear una Tarjeta de Aprendizaje.

```

PASS src/components/UpdateGifCard/UpdateGifCard.test.js (10.967 s)
PASS src/pages/AddCard/AddCard.test.js (11.603 s)
PASS src/pages/UpdateCard/UpdateCard.test.js (11.674 s)
PASS src/pages/ViewKnowledgeCard/ViewKnowledgeCard.test.js
PASS src/components/UpdateGifCard/UpdateGifCard.test.js
PASS src/pages/UpdateCard/UpdateCard.test.js
PASS src/pages/AddCard/AddCard.test.js
PASS src/pages/ShowCards/ShowCards.test.js (5.29 s)
PASS src/pages/UpdateModule/UpdateModule.test.js
PASS src/pages/Home/Home.test.js
PASS src/pages/AddLesson/AddLesson.test.js
PASS src/pages/ShowMicrolearning/ShowMicrolearning.test.js
PASS src/pages/ShowUsers/ShowUsers.test.js
PASS src/components/UpdateImage/UpdateImage.test.js
PASS src/pages/AddModules/AddModules.test.js
PASS src/pages/ShowLessons/ShowLessons.test.js
PASS src/pages/UpdateLesson/UpdateLesson.test.js
PASS src/pages/UpdateMicrolearning/UpdateMicrolearning.test.js
PASS src/pages/AddMicrolearning/AddMicrolearning.test.js
PASS src/pages/ShowModules/ShowModules.test.js
PASS src/pages/ViewKnowledgeCard/ViewKnowledgeCard.test.js
PASS src/pages/SignUp/SignUp.test.js
PASS src/pages/LoginUser/LoginUser.test.js
PASS src/components/UpdateIcon/UpdateIcon.test.js
PASS src/pages/LoginAdmin/LoginAdmin.test.js
PASS src/components/UpdateGif/UpdateGif.test.js
PASS src/pages/ViewMicrolearning/ViewMicrolearning.test.js
PASS src/pages/LandingPage/LandingPage.test.js
PASS src/pages/DashBoard/DashBoard.test.js
PASS src/pages/ViewLesson/ViewLesson.test.js
PASS src/components/ShowImage/ShowImage.test.js
PASS src/components/NavigationAdmin/NavigationAdmin.test.js
PASS src/pages/NotFoundPage/NotFoundPage.test.js
PASS src/components/Navigation/Navigation.test.js

```

Figura 104. Resultado de las pruebas en el front-end iteración 3.

```

Test Suites: 30 passed, 30 total
Tests:       59 passed, 59 total
Snapshots:   0 total
Time:        19.15 s, estimated 26 s
Ran all test suites.

```

Figura 105. Total, de pruebas front-end iteración 3.

Anexo 8. Plan de pruebas de la API

Pruebas Unitarias de la API

Proyecto: Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano

Versión: 2.0

Fecha: 04/07/2022

Hoja de control

Proyecto	Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano
Entregable	Plan de Pruebas de la API
Autor	Jonnathan Espinoza
Versión	2.0
Aprobado por	Edison Leonardo Coronel Romero, Mg. Sc.

Registro de cambios

Versión del documento	Descripción	Responsable	Fecha de cambio
1.0	Versión inicial de pruebas unitarias de la API.	Jonnathan Espinoza	11/07/2021
2.0	Incorporación de pruebas a los endpoints de la API	Jonnathan Espinoza	04/07/2022

Índice

Hoja de control	156
Índice	157
1. Introducción.....	158
Objetivo	158
Propósito	158
2. Definición de casos de prueba.....	159

1. Introducción

Objetivo

El objeto del presente documento es validar la funcionalidad de los endpoints de la API.

Propósito

Comprobar el correcto funcionamiento de las repuestas HTTP de la API, para asegurar que la validación de los datos sea correcta, además de probar la autenticación de los tokens del usuario final y administrador.

2. Casos de pruebas unitarias

En el presente apartado se describen los casos de prueba unitarias que se determinaron con la ayuda de las Historias de Usuario.

Código de prueba	Componente/endpoint	Descripción	Prerrequisito	Estado
CP01	POST /api/auth/signup	Crear cuenta de usuario final.	Correo electrónico. Nombre de usuario. Contraseña.	Pasado
CP02	POST /api/auth/signup	Un usuario final no puede crear una cuenta con un email repetido.	El correo electrónico único es único.	Pasado
CP03	POST /api/auth/signup	Un usuario final no puede crear la cuenta si no proporciona todos los datos del formulario.	Correo electrónico. Nombre de usuario. Contraseña.	Pasado
CP04	POST /api/auth/signup	Crear cuenta de usuario administrador.	Correo electrónico. Nombre de usuario. Contraseña. Rol de administrador.	Pasado
CP05	POST /api/auth/signin	Iniciar sesión de usuario final	Correo electrónico. Contraseña.	Pasado
CP06	POST /api/auth/signin	El usuario final no puede iniciar sesión si no proporciona un email registrado.	Correo electrónico. Contraseña.	Pasado
CP07	POST /api/auth/signin	El usuario final no puede iniciar sesión si no ingresa bien la contraseña.	Correo electrónico. Contraseña.	Pasado
CP08	POST /api/auth/signinadmin	Iniciar sesión de usuario administrador.	Correo electrónico. Contraseña. Rol administrador.	Pasado
CP09	POST /api/auth/signinadmin	El usuario administrador no puede iniciar sesión si no proporciona un email registrado.	Correo electrónico. Contraseña. Rol administrador.	Pasado
CP10	POST /api/auth/signinadmin	El usuario administrador no puede iniciar sesión si no ingresa bien la contraseña.	Correo electrónico. Contraseña. Rol administrador.	Pasado
CP11	POST /api/module	Crear un módulo con autenticación de administrador.	Nombre modulo. Numero de modulo. Token de usuario.	Pasado
CP12	POST /api/module	No se puede crear un módulo sin la autenticación de administrador.	Nombre modulo. Numero de modulo. Token de usuario.	Pasado
CP13	POST /api/module	No se puede crear un módulo si no se envía el token.	Nombre modulo. Numero de modulo.	Pasado
CP14	POST /api/module	No se puede crear un módulo si el token fue modificado.	Nombre modulo. Numero de modulo. Token de usuario.	Pasado
CP15	GET /api/module/	Se retorna la lista de módulos en un array.	N/A	Pasado
CP16	GET /api/module/	Comprobar si existe módulos en registrados en la base de datos.	N/A	Pasado

CP17	GET /api/module/	Comprobar la persistencia de datos.	N/A	Pasado
CP18	PUT /api/module/	Actualizar la información de un módulo con autenticación de administrador.	Nombre modulo. Numero de modulo. Token de usuario.	Pasado
CP19	PUT /api/module/	No se puede actualizar la información de un módulo sin autenticación de administrador.	Nombre modulo. Numero de modulo. Token de usuario.	Pasado
CP20	PUT /api/module/	No se puede actualizar la información de un módulo si el token fue modificado.	Nombre modulo. Numero de modulo. Token de usuario.	Pasado
CP21	PUT /api/module/	Actualizar el nombre del módulo.	Nombre modulo. Token de usuario.	Pasado
CP22	PUT /api/module/	Actualizar el número del módulo.	Número de modulo. Token de usuario.	Pasado
CP23	DELETE /api/module/	El módulo no se puede eliminar si el id es incorrecto.	Id del módulo. Token de usuario.	Pasado
CP24	DELETE /api/module/	El módulo no puede ser eliminado sin la autenticación de administrador.	Id del módulo. Token de usuario.	Pasado
CP25	DELETE /api/module/	El módulo no puede ser eliminado si el token es modificado.	Id del módulo. Token de usuario.	Pasado
CP26	DELETE /api/module/	Eliminar un módulo con autenticación de administrador.	Id del módulo. Token de usuario.	Pasado
CP27	POST /api/lesson/	Crear una lección con autenticación de administrador.	Nombre de la lección. Icono. Id del módulo a la que pertenece. Token de usuario.	Pasado
CP28	GET /api/lesson	Obtener todas las lecciones	N/A	Pasado
CP29	GET /api/lesson	Obtener una lección por el id.	Id de la lección.	Pasado
CP30	GET /api/lesson	Presentar error si al llamar una lección por id no se proporciona esta.	N/A	Pasado
CP31	PUT /api/lesson/	Actualizar nombre de la lección.	Nombre de la lección. Token de usuario.	Pasado
CP32	PUT /api/lesson/	Actualizar el módulo de la lección.	Módulo de la lección. Token de usuario.	Pasado
CP33	PUT /api/lesson/	Actualizar el nombre y módulo de la lección.	Nombre de la lección. Módulo de la lección. Token de usuario.	Pasado
CP34	PUT /api/lesson/	Actualizar el icono de la lección.	Icono. Token de usuario.	Pasado
CP35	DELETE /api/lesson/	No se puede eliminar una lección si el id es incorrecto.	Id de la lección. Token de usuario.	Pasado
CP36	DELETE /api/lesson/	No se puede eliminar una lección si no se envía el token.	Id de la lección.	Pasado
CP37	DELETE /api/lesson/	No se puede eliminar una lección si el token es inválido.	Id de la lección. Token de usuario.	Pasado
CP38	DELETE /api/lesson/	No se puede eliminar una lección si el usuario no es administrador.	Id de la lección. Token de usuario.	Pasado
CP39	DELETE /api/lesson/	Eliminar una lección con autenticación de administrador.	Id de la lección. Token de usuario.	Pasado

CP40	POST /api/micro/	Crear un micro contenido con autenticación de administrador.	Titulo. Imagen. Gif. Id de la lección a la que pertenece. Token de usuario.	Pasado
CP41	POST /api/micro/	No se puede crear los micro contenidos si no se envía todos los parámetros.	Titulo. Imagen. Gif. Id de la lección a la que pertenece. Token de usuario.	Pasado
CP42	POST /api/micro/	No se puede crear un micro contenido sin autenticación de administrador.	Titulo. Imagen. Gif. Id de la lección a la que pertenece. Token de usuario.	Pasado
CP43	GET /api/micro/	Obtener todos los micro contenidos.	N/A	Pasado
CP44	GET /api/micro/	Obtener micro contenido por id.	Id del micro contenido.	Pasado
CP45	GET /api/micro/	Si el id del micro contenido es incorrecto presentar un mensaje.	Id del micro contenido.	Pasado
CP46	PUT /api/micro/	Actualizar el título del micro contenido.	Titulo. Token de usuario.	Pasado
CP47	PUT /api/micro/	Actualizar la lección del micro contenido.	Id de la lección a la que pertenece. Token de usuario.	Pasado
CP48	PUT /api/micro/	Actualizar el título y la lección del micro contenido.	Titulo. Id de la lección a la que pertenece. Token de usuario.	Pasado
CP49	PUT /api/micro/	Actualizar la imagen del micro contenido.	Imagen. Token de usuario.	Pasado
CP50	PUT /api/micro/	No actualizar si no se envía la imagen del micro contenido.	Imagen. Token de usuario.	Pasado
CP51	PUT /api/micro/	Actualizar el Gif del micro contenido.	Gif. Token de usuario.	Pasado
CP52	PUT /api/micro/	No actualizar si no se envía el Gif del micro contenido.	Gif. Token de usuario.	Pasado
CP53	DELETE /api/micro/	No se elimina el micro contenido si el id es incorrecto.	Id micro contenido. Token de usuario.	Pasado
CP54	DELETE /api/micro/	No se elimina el micro contenido si no se envía el token.	Id micro contenido.	Pasado
CP55	DELETE /api/micro/	No se elimina el micro contenido si el token no es válido.	Id micro contenido. Token de usuario.	Pasado
CP56	DELETE /api/micro/	No se elimina el micro contenido si no es administrador.	Id micro contenido. Token de usuario.	Pasado
CP57	DELETE /api/micro/	Eliminar micro contenido con autenticación de administrador.	Id micro contenido. Token de usuario.	Pasado
CP58	POST /api/card	Crear una tarjeta de aprendizaje.	Pregunta. Gif.	Pasado

			Respuesta correcta. Respuesta incorrecta. Id de la lección a la que pertenece. Token de usuario.	
CP59	POST /api/card	No se crea la tarjeta de aprendizaje si no se envía todos los parámetros.	Pregunta. Gif. Respuesta correcta. Respuesta incorrecta. Id de la lección a la que pertenece. Token de usuario.	Pasado
CP60	POST /api/card	No se crea la tarjeta de aprendizaje si no es administrador.	Pregunta. Gif. Respuesta correcta. Respuesta incorrecta. Id de la lección a la que pertenece. Token de usuario.	Pasado
CP61	GET /api/card	Obtener todas las tarjetas de aprendizaje.	N/A	Pasado
CP62	GET /api/card	Obtener tarjeta de aprendizaje por id.	Id de la tarjeta de aprendizaje.	Pasado
CP63	GET /api/card	Si el id de la tarjeta de aprendizaje es incorrecta, presentar un mensaje de error.	Id de la tarjeta de aprendizaje.	Pasado
CP64	PUT /api/card	Actualizar la pregunta de la tarjeta de aprendizaje.	Pregunta. Token de usuario.	Pasado
CP65	PUT /api/card	Actualizar la lección de la tarjeta de aprendizaje.	Id de la lección a la que pertenece. Token de usuario.	Pasado
CP66	PUT /api/card	Actualizar la respuesta correcta e incorrecta de la tarjeta de aprendizaje.	Respuesta correcta. Respuesta incorrecta. Token de usuario.	Pasado
CP67	PUT /api/card	Actualizar el Gif de la tarjeta de aprendizaje.	Gif. Token de usuario.	Pasado
CP68	PUT /api/card	Si no se envía un Gif no se actualiza la tarjeta de aprendizaje.	Token de usuario.	Pasado
CP69	DELETE /api/card	No se elimina la tarjeta de aprendizaje si el id es incorrecto.	Id tarjeta de aprendizaje. Token de usuario.	Pasado
CP70	DELETE /api/card	No se elimina la tarjeta de aprendizaje si no se envía el token.	Id tarjeta de aprendizaje.	Pasado
CP71	DELETE /api/card	No se elimina la tarjeta de aprendizaje si el token no es válido.	Id tarjeta de aprendizaje. Token de usuario.	Pasado
CP72	DELETE /api/card	No se elimina la tarjeta de aprendizaje si no es administrador.	Id tarjeta de aprendizaje. Token de usuario.	Pasado
CP73	DELETE /api/card	Eliminar tarjeta de aprendizaje con autenticación de administrador.	Id tarjeta de aprendizaje. Token de usuario.	Pasado

Anexo 9. Plan de pruebas unitarias del Front-end

Plan de Pruebas Unitarias del Front-end

Proyecto: Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano

Versión: 2.0

Fecha: 28/07/2021

Hoja de control

Proyecto	Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano
Entregable	Plan de Pruebas de unitarias del Front-end
Autor	Jonnathan Espinoza
Versión	2.0
Aprobado por	Edison Leonardo Coronel Romero, Mg. Sc.

Registro de cambios

Versión del documento	Descripción	Responsable	Fecha de cambio
1.0	Versión inicial de pruebas unitarias del Front-end.	Jonnathan Espinoza	11/07/2021
2.0	Versión final del plan de pruebas unitarias del Front-end.	Jonnathan Espinoza	04/07/2022

Índice

Hoja de control	164
Índice	165
1. Introducción.....	166
Objetivo	166
Propósito	166
2. Definición de casos de prueba.....	167

3. Introducción

Objetivo

El objetivo del documento es validar la funcionalidad correcta de la interfaz de usuario y mostrar que los componentes creados se renderizan.

Propósito

Validar el funcionamiento de la interfaz de usuario, para asegurar que el renderizado de la página sea correcta, además de probar que los datos se obtienen de forma correcta de la API.

4. Casos de prueba unitarias

En el presente apartado se establece los casos de prueba unitarias, que se identificaron con la ayuda de las Historias de Usuario. En el front-end, las pruebas unitarias se basan en validar los componentes de cada interfaz, por lo que se utilizó Jest y react-testing-library, como lo sugiere la documentación oficial de React.

Código de prueba	Componente	Descripción	Prerrequisito	Estado
CP01	<Signup />	Renderizar el componente para crear cuenta.	N/A	Pasado
CP02	<Signup />	Verificar si ingresa datos en el input del nombre.	Nombre de usuario.	Pasado
CP03	<Signup />	Verificar si ingresa datos en el input del email.	Email de usuario.	Pasado
CP04	<Signup />	Verificar si ingresa datos en el input del password.	Contraseña de usuario.	Pasado
CP05	<LoginUser />	Renderizar el componente para iniciar sesión de usuario final.	N/A	Pasado
CP06	<LoginUser />	Verificar si ingresa datos en el input del email.	Email de usuario.	Pasado
CP07	<LoginUser />	Verificar si ingresa datos en el input del password.	Contraseña de usuario.	Pasado
CP08	<AddModules />	Renderizar el componente para crear un nuevo módulo.	N/A	Pasado
CP09	<AddModules />	Verificar si ingresa datos en el input del número de módulo.	Número de módulo.	Pasado
CP10	<AddModules />	Verificar si ingresa datos en el input del nombre del módulo.	Nombre del módulo.	Pasado
CP11	<AddModules />	Verificar si se puede presionar el botón regresar.	Hacer click en el botón regresar.	Pasado
CP12	<AddLesson />	Renderizar el componente para crear una nueva lección.	N/A	Pasado
CP13	<AddLesson />	Verificar si ingresa datos en el input del nombre de la lección.	Nombre de la lección.	Pasado
CP14	<AddLesson />	Verificar si se puede presionar el botón regresar.	Hacer click en el botón regresar.	Pasado
CP15	<LoginAdmin />	Renderizar el componente para iniciar sesión de usuario administrador.	N/A	Pasado
CP16	<LoginAdmin />	Verificar si ingresa datos en el input del email.	Email de usuario administrador.	Pasado
CP17	<LoginAdmin />	Verificar si ingresa datos en el input del password.	Contraseña de usuario administrador.	Pasado
CP18	<ShowModules />	Renderizar el componente para ver la lista de módulos creados.	Obtener datos de la API.	Pasado
CP19	<ShowLessons />	Renderizar el componente para ver la lista de lecciones creadas.	Obtener datos de la API.	Pasado
CP20	<AddModules />	Renderizar el componente para crear un nuevo módulo.	N/A	Pasado
CP21	<AddModules />	Verificar si ingresa datos en el input del número de módulo.	Número de módulo.	Pasado
CP22	<AddModules />	Verificar si ingresa datos en el input del nombre de módulo.	Nombre del módulo.	Pasado

CP23	<AddModules />	Verificar si se puede presionar el botón regresar.	Hacer click en el botón regresar.	Pasado
CP24	<UpdateLesson />	Renderizar el componente para actualizar una lección.	N/A	Pasado
CP25	<UpdateLesson />	Verificar si ingresa datos en el input del nombre de la lección.	Nombre de la lección.	Pasado
CP26	<UpdateLesson />	Verificar si se puede presionar el botón regresar.	Hacer click en el botón regresar.	Pasado
CP27	<ViewLesson />	Renderizar el componente para ver la información de la lección.	Obtener la lección por id de la API.	Pasado
CP28	<ShowUsers />	Renderizar el componente para ver la lista de usuarios registrados.	Obtener datos de la API.	Pasado
CP29	<NotFoundPage />	Renderizar el componente para mostrar el error 404.	N/A	Pasado
CP30	<Home />	Renderizar el componente para mostrar el contenido al usuario final.	N/A	Pasado
CP31	<DashBoard />	Renderizar el componente del panel de administrador para gestionar el contenido de la aplicación web.	N/A	Pasado
CP32	<NavigationAdmin />	Renderizar el componente Nav para el administrador.	N/A	Pasado
CP33	<Navigation />	Renderizar el componente Nav para el usuario final.	N/A	Pasado
CP34	<UpdateIcon />	Renderizar el componente para actualizar el icono de la lección.	N/A	Pasado
CP35	<ShowMicrolearning />	Renderizar el componente para ver la lista de cápsulas de aprendizaje creados.	Obtener datos de la API.	Pasado
CP36	<AddMicrolearning />	Renderizar el componente para crear una cápsula de aprendizaje.	N/A	Pasado
CP37	<AddMicrolearning />	Verificar si se ingresa datos en el input del título de la cápsula de aprendizaje.	Título de la cápsula de aprendizaje.	Pasado
CP38	<AddMicrolearning />	Verificar si se puede presionar el botón regresar.	Hacer click en el botón regresar.	Pasado
CP39	<UpdateMicrolearning />	Renderizar el componente para actualizar una cápsula de aprendizaje.	N/A	Pasado
CP40	<UpdateMicrolearning />	Verificar si se ingresa datos en el input del título de la cápsula de aprendizaje.	Título de la cápsula de aprendizaje.	Pasado
CP41	<UpdateMicrolearning />	Verificar si se puede presionar el botón regresar.	Hacer click en el botón regresar.	Pasado
CP42	<ViewMicrolearning />	Renderizar el componente para visualizar información de una cápsula de aprendizaje.	N/A	Pasado
CP43	<UpdateGif />	Renderizar el componente para actualizar el Gif de la cápsula de aprendizaje.	Gif de la seña.	Pasado
CP44	<UpdateImage />	Renderizar el componente para actualizar la imagen de la cápsula de aprendizaje.	Imagen representativa del significado de la seña.	Pasado
CP45	<ShowImage />	Renderizar el componente para visualizar la imagen de la cápsula de aprendizaje.	Imagen representativa del significado de la seña.	Pasado

CP46	<LandingPage />	Renderizar el componente de la página inicial.	N/A	Pasado
CP47	<ShowCards />	Renderizar el componente para mostrar la lista de tarjetas de aprendizaje creadas.	Obtener datos de la API.	Pasado
CP48	<AddCard />	Renderizar el componente para crear una tarjeta de aprendizaje.	N/A	Pasado
CP49	<AddCard />	Verificar si se ingresa datos en el input de pregunta de la tarjeta de aprendizaje.	Pregunta de la tarjeta de aprendizaje.	Pasado
CP50	<AddCard />	Verificar si se ingresa datos en el input de respuesta correcta de la tarjeta de aprendizaje.	Respuesta correcta de la tarjeta de aprendizaje.	Pasado
CP51	<AddCard />	Verificar si se ingresa datos en el input de respuesta incorrecta de la tarjeta de aprendizaje.	Respuesta incorrecta de la tarjeta de aprendizaje.	Pasado
CP52	<AddCard />	Verificar si se puede presionar el botón regresar.	Hacer click en el botón regresar.	Pasado
CP53	<UpdateCard />	Renderizar el componente para actualizar una tarjeta de aprendizaje.	N/A	Pasado
CP54	<UpdateCard />	Verificar si se ingresa datos en el input de pregunta de la tarjeta de aprendizaje.	Pregunta de la tarjeta de aprendizaje.	Pasado
CP55	<UpdateCard />	Verificar si se ingresa datos en el input de respuesta correcta de la tarjeta de aprendizaje.	Respuesta correcta de la tarjeta de aprendizaje.	Pasado
CP56	<UpdateCard />	Verificar si se ingresa datos en el input de respuesta incorrecta de la tarjeta de aprendizaje.	Respuesta incorrecta de la tarjeta de aprendizaje.	Pasado
CP57	<UpdateCard />	Verificar si se puede presionar el botón regresar.	Hacer click en el botón regresar.	Pasado
CP58	<ViewKnowledgeCard />	Renderizar el componente para visualizar la información de una tarjeta de aprendizaje.	Obtener datos de la API.	Pasado
CP59	<UpdateGifCard />	Renderizar el componente para actualizar el Gif de una tarjeta de aprendizaje.	Gif de la tarjeta de aprendizaje.	Pasado

Anexo 10. Plan de Pruebas de Integración

Pruebas de Integración del Sistema

Proyecto: Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano

Versión: 2.0

Fecha: 04/07/2022

Hoja de control

Proyecto	Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano
Entregable	Plan de Pruebas de Integración.
Autor	Jonnathan Espinoza
Versión	2.0
Aprobado por	Edison Leonardo Coronel Romero, Mg. Sc.

Registro de cambios

Versión del documento	Descripción	Responsable	Fecha de cambio
1.0	Versión inicial de pruebas de integración.	Jonnathan Espinoza	01/07/2021
2.0	Versión final de pruebas de integración.	Jonnathan Espinoza	04/06/2022

Índice

Hoja de control	171
Índice	172
1. Introducción.....	173
Objetivo	173
Propósito	173
2. Definición de casos de prueba.....	174

5. Introducción

Objetivo

El objetivo es identificar los casos de prueba para verificar la integración de la API con la interfaz de usuario a partir de los requerimientos funcionales.

Propósito

Validar el funcionamiento de la interacción de la interfaz de usuario con la API, para asegurar el cumplimiento de los requerimientos establecidos.

6. Casos de prueba

En el presente apartado se presenta cada caso de prueba que se determinaron con la ayuda de los requerimientos.

Código de prueba	Numero de Requerimiento Funcional	Descripción	Prerrequisito	Estado
CP01	RF07	La página principal se renderiza de forma correcta.	<ul style="list-style-type: none"> Ingresar a la página principal. 	Pasado
CP02	RF01	El botón para ir a la página de crear cuenta del navbar redirecciona al formulario para llenar los datos.	<ul style="list-style-type: none"> Ingresar a la página principal. 	Pasado
CP03	RF01	El botón para ir a la página de crear cuenta del landing page redirecciona al formulario para llenar los datos.	<ul style="list-style-type: none"> Ingresar a la página principal. 	Pasado
CP04	RF01	Llenar el formulario y crear una cuenta de forma exitosa.	<ul style="list-style-type: none"> Ingresar a la página de crear cuenta. Ingresar un correo electrónico válido. Ingresar un nombre de usuario. Ingresar una contraseña. 	Pasado
CP05	RF01	Presentar un error si ya existe el correo electrónico registrado.	<ul style="list-style-type: none"> Ingresar a la página de crear cuenta. Ingresar un correo electrónico que se encuentre registrado. 	Pasado
CP06	RF01	Presentar un error si el correo electrónico está mal ingresado. (debe de tener: nombre@dominio.com)	<ul style="list-style-type: none"> Ingresar a la página de crear cuenta. Ingresar un correo electrónico no válido. 	Pasado
CP07	RF02	El botón para ir a la página de iniciar sesión del navbar redirecciona al formulario para llenar los datos.	<ul style="list-style-type: none"> Ingresar a la página principal. 	Pasado
CP08	RF02	Llenar el formulario, iniciar sesión de forma correcta y redireccionar a la página de aprendizaje.	<ul style="list-style-type: none"> Ingresar a la página de iniciar sesión usuario final. Ingresar un correo electrónico registrado en la app. Ingresar la contraseña de acceso. 	Pasado
CP09	RF02	Presentar un mensaje de error si la contraseña es incorrecta.	<ul style="list-style-type: none"> Ingresar a la página de iniciar sesión usuario final. Ingresar una contraseña incorrecta. 	Pasado
CP10	RF02	Presentar un mensaje de error si el correo electrónico es incorrecto.	<ul style="list-style-type: none"> Ingresar a la página de iniciar sesión usuario final. Ingresar un correo electrónico no registrado en la app. 	Pasado
CP11	RF02	Iniciar sesión con cuenta de administrador de forma correcta.	<ul style="list-style-type: none"> Ingresar a la página de iniciar sesión para administrador. Ingresar un correo electrónico registrado en la app. Ingresar la contraseña de acceso. 	Pasado
CP12	RF02	Presentar un error al iniciar sesión de administrador si la contraseña es incorrecta.	<ul style="list-style-type: none"> Ingresar a la página de iniciar sesión para administrador. Ingresar una contraseña incorrecta. 	Pasado

CP13	RF02	Presentar un error al iniciar sesión de administrador si el correo electrónico es incorrecto.	<ul style="list-style-type: none"> • Ingresar a la página de iniciar sesión para administrador. • Ingresar un correo electrónico no registrado en la app. 	Pasado
CP14	RF04	El administrador puede crear un módulo de forma correcta.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de crear módulo. • Llenar el formulario. 	Pasado
CP15	RF04	Presentar un error si no se ingresa en número del módulo en el formulario al momento de crearlo.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de crear módulo. • No ingresar el número de módulo. 	Pasado
CP16	RF04	Presentar un error si no se ingresa el nombre del módulo en el formulario al momento de crearlo.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de crear módulo. • No ingresar el nombre del módulo. 	Pasado
CP17	RF04	Presentar un error si el número del módulo ya está creado. Este valor debe de ser único.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de crear módulo. • Ingresar un número ya asignado. 	Pasado
CP18	RF04	Listar todos los módulos creados en una tabla.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de listar módulos. 	Pasado
CP19	RF04	Actualizar la información del módulo de forma correcta.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de listar módulos. • Ingresar a la página de actualizar información de módulo. 	Pasado
CP20	RF04	Eliminar un módulo de forma correcta.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de listar módulos. • Presionar el botón de eliminar módulo. • Confirmar la eliminación del módulo. 	Pasado
CP21	RF05	El administrador puede crear una lección de forma exitosa.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de crear lección. • Llenar formulario. 	Pasado
CP22	RF05	Presentar un error si no se ingresa un icono al momento de crear una lección.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de crear lección. • No ingresar el icono de la lección. 	Pasado
CP23	RF05	Presentar un error si no se ingresa el nombre de la lección al momento de crearlo.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de crear lección. • No ingresar el nombre de la lección. 	Pasado

CP24	RF05	Presentar un error si no se selecciona el módulo al que pertenece la lección al momento de crearlo.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de crear lección. • No ingresar el módulo de la lección. 	Pasado
CP25	RF05	Listar todas las lecciones en una tabla.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de listar lecciones. 	Pasado
CP26	RF05	Ver la información de la lección en una página.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de listar lecciones. • Hacer click en el botón de ver información. 	Pasado
CP27	RF05	Actualizar la información de la lección de forma exitosa.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de listar lecciones. • Hacer click en el botón de actualizar. 	Pasado
CP28	RF05	Eliminar una lección de forma exitosa.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de listar lecciones. • Hacer click en el botón de eliminar. • Confirmar la eliminación de la lección. 	Pasado
CP29	RF08	El administrador puede crear un micro contenido de forma exitosa.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de crear micro contenido. • Llenar el formulario. 	Pasado
CP30	RF08	Presentar un error si no se ingresa una imagen al micro contenido al momento de crearlo.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de crear micro contenido. • No ingresar una imagen representativa de la seña. 	Pasado
CP31	RF08	Presentar un error si no se ingresa un gif al micro contenido al momento de crearlo.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de crear micro contenido. • No ingresar el gif de la seña. 	Pasado
CP32	RF08	Presentar un error si no se ingresa un título al micro contenido al momento de crearlo.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de crear micro contenido. • No ingresar el título o tema que se va aprender. 	Pasado
CP33	RF08	Presentar un error si no se ingresa la lección al que pertenece el micro contenido al momento de crearlo.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de crear micro contenido. • No ingresar la lección a la que pertenece. 	Pasado
CP34	RF08	Listar todos los micro contenidos en una tabla.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de listar micro contenidos. 	Pasado

CP35	RF08	Ver la información del micro contenido en una página.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de listar micro contenidos. • Hacer click en el botón de ver información. 	Pasado
CP36	RF08	Actualizar la información del micro contenido de forma exitosa.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de listar micro contenidos. • Hacer click en el botón de actualizar información. • Llenar formulario de información. 	Pasado
CP37	RF08	Actualizar el gif de la seña en el micro contenido de forma exitosa.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de listar micro contenidos. • Hacer click en el botón de actualizar información. • Llenar formulario del gif. 	Pasado
CP38	RF08	Actualizar la imagen de la seña en el micro contenido de forma exitosa.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de listar micro contenidos. • Hacer click en el botón de actualizar información. • Llenar formulario de la imagen. 	Pasado
CP39	RF08	Eliminar un micro contenido de forma exitosa.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de listar micro contenidos. • Hacer click en el botón de eliminar. 	Pasado
CP40	RF06	El administrador puede crear una tarjeta de aprendizaje de forma exitosa.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de crear tarjeta de aprendizaje. • Llenar el formulario. 	Pasado
CP41	RF06	Presentar un error si no se ingresa el gif de la seña al momento de crear una tarjeta de aprendizaje.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de crear tarjeta de aprendizaje. • No ingresar el gif de la seña. 	Pasado
CP42	RF06	Presentar un error si no se ingresa la pregunta al momento de crear una tarjeta de aprendizaje.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de crear tarjeta de aprendizaje. • No ingresar la pregunta de lo que se va a resolver. 	Pasado
CP43	RF06	Presentar un error si no se ingresa la respuesta correcta al momento de crear una tarjeta de aprendizaje.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de crear tarjeta de aprendizaje. • No ingresar la respuesta correcta. 	Pasado
CP44	RF06	Presentar un error si no se ingresa la respuesta incorrecta al momento de crear una tarjeta de aprendizaje.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de crear tarjeta de aprendizaje. • No ingresar la respuesta incorrecta. 	Pasado

CP45	RF06	Presentar un error si no se ingresa la lección a la que corresponde al momento de crear una tarjeta de aprendizaje.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de crear tarjeta de aprendizaje. • No ingresar la lección a la que pertenece. 	Pasado
CP46	RF06	Listar todas las tarjetas de aprendizaje en una tabla.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de listar tarjetas de aprendizaje. 	Pasado
CP47	RF06	Ver la información de la tarjeta de aprendizaje en una página.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de listar tarjetas de aprendizaje. • Hacer click en el botón de ver información. 	Pasado
CP48	RF06	Actualizar la información de la tarjeta de aprendizaje de forma exitosa.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de listar tarjetas de aprendizaje. • Hacer click en el botón de actualizar información. • Llenar el formulario de información. 	Pasado
CP49	RF06	Actualiza el gif de la seña en la tarjeta de aprendizaje de forma correcta.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de listar tarjetas de aprendizaje. • Hacer click en el botón de actualizar información. • Llenar el formulario del gif. 	Pasado
CP50	RF06	Eliminar una tarjeta de aprendizaje de forma exitosa.	<ul style="list-style-type: none"> • Iniciar sesión con permisos de administrador. • Ingresar a la página de listar tarjetas de aprendizaje. • Hacer click en el botón de eliminar. • Confirmar la eliminación de la tarjeta de aprendizaje. 	Pasado
CP51	RF07	Mostrar los micro contenidos en la interfaz del usuario final.	<ul style="list-style-type: none"> • Crear cuenta. • Iniciar sesión. • Ingresar a la página de aprendizaje. 	Pasado
CP52	RF07	El usuario final puede ingresar a la sección de aprender.	<ul style="list-style-type: none"> • Crear cuenta. • Iniciar sesión. • Ingresar a la página de aprendizaje. • Hacer click en el botón de aprender, que se encuentra debajo de cada lección. 	Pasado
CP53	RF07	El usuario final puede ingresar a la sección de practicar.	<ul style="list-style-type: none"> • Crear cuenta. • Iniciar sesión. • Ingresar a la página de aprendizaje. • Hacer click en el botón de practicar, que se encuentra debajo de cada lección. 	Pasado
CP54	RF07	El usuario final puede ingresar a la sección de progreso de lecciones.	<ul style="list-style-type: none"> • Crear cuenta. • Iniciar sesión. • Ingresar a la página de ver progreso. 	Pasado
CP55	RF03	Cerrar sesión de usuario.	<ul style="list-style-type: none"> • Crear cuenta. 	Pasado

			<ul style="list-style-type: none">• Iniciar sesión.• Ingresar a la página de aprendizaje.• Hacer click en el botón de cerrar sesión que se encuentra en el navbar.	
--	--	--	--	--

Anexo 11. Plan de Pruebas Funcionales

Pruebas Funcionales del Sistema

Proyecto: Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano

Versión: 1.0

Fecha: 11/07/2022

Hoja de control

Organismo	Universidad Nacional de Loja		
Proyecto	Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano		
Entregable	Plan de Pruebas Funcionales		
Autor	Jonnathan Espinoza		
Versión/Edición	1.0	Fecha Versión	09/07/2022
Aprobado por	Edison Leonardo Coronel Romero, Mg. Sc.	Fecha Aprobación	11/07/2022
		N° Total de Páginas	26

Registro de cambios

Versión doc	Causa del cambio	Responsable del Cambio	Fecha de cambio
1.0	Versión inicial de pruebas funcionales.	Jonnathan Espinoza	11/07/2022

Registro de cambios

Nombres y Apellidos
Jonnathan Damián Espinoza Erráez
Edison Leonardo Coronel Romero, Mg. Sc.

Índice

Hoja de control	181
Índice	182
1. Introducción.....	183
Objeto.....	183
Alcance.....	183
2. Trazabilidad de casos de pruebas - requerimientos.....	184
3. Definición de casos de prueba.....	185
4. Estrategia de ejecución de pruebas.....	204
5. Bibliografía y referencias	205

1. Introducción

Objeto

Identificar los casos de prueba para verificar que la aplicación web satisface los requerimientos funcionales.

Alcance

Los casos de prueba están dirigidos al desarrollador de la aplicación y al docente Ing. Edison Coronel director del Trabajo de Titulación.

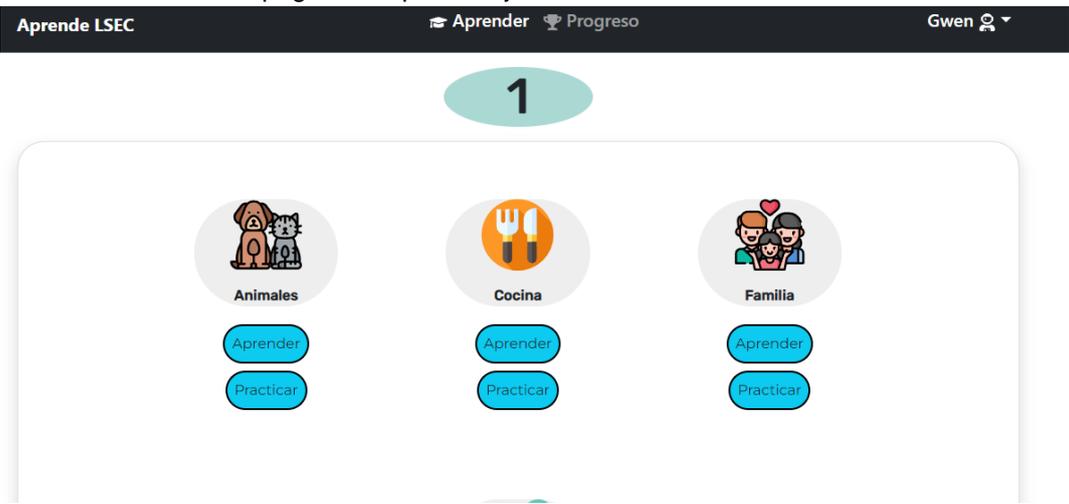
2. Trazabilidad de casos de pruebas - requerimientos

La siguiente tabla muestra la relación entre los casos de prueba y los requerimientos funcionales. Las filas son casos de prueba definidos, las columnas son requerimientos funcionales y las 'X' son relaciones existentes.

	RF01	RF02	RF03	RF04	RF05	RF06	RF07	RF08
CP01	X							
CP02	X							
CP03		X						
CP04		X						
CP05				X				
CP06				X				
CP07				X				
CP08				X				
CP09					X			
CP10					X			
CP11					X			
CP12					X			
CP13								X
CP14								X
CP15								X
CP16								X
CP17						X		
CP18						X		
CP19						X		
CP20						X		
CP21			X					
CP22							X	

3. Definición de casos de prueba

Esta sección describe cada caso de prueba identificado utilizando requerimientos funcionales e historias de usuario.

Crear cuenta de usuario exitoso	CP01	
	¿Prueba de despliegue?	Si
Descripción: Se probará la respuesta correcta al crear cuenta de usuario.		
Prerrequisitos: <ul style="list-style-type: none"> • Ingresar a la página principal. • Hacer clic en el botón “crear cuenta”. 		
Pasos: <ol style="list-style-type: none"> 1. Llenar el campo de “Nombre”. 2. Llenar el campo de “Email”. 3. Llenar el campo de “Contraseña”. 4. Hacer clic en el botón “REGISTRATE”. 5. Esperar un mensaje de confirmación. 		
Resultado esperado: <ul style="list-style-type: none"> • Un spinner. • Redireccionar a la página de aprendizaje. 		
Resultado obtenido: <ul style="list-style-type: none"> • Un spinner. • Redireccionar a la página de aprendizaje. 		

Crear cuenta de usuario fallido	CP02	
	¿Prueba de despliegue?	Si
Descripción: Se probará la respuesta al crear cuenta con los campos vacíos.		
Prerrequisitos: <ul style="list-style-type: none"> • Ingresar al landig page. • Ir al navbar y hacer clic en el botón “crear cuenta”. 		
Pasos: <ol style="list-style-type: none"> 1. No llenar los campos del formulario. 2. Hacer clic en el botón “REGISTRATE”. 		
Resultado esperado:		

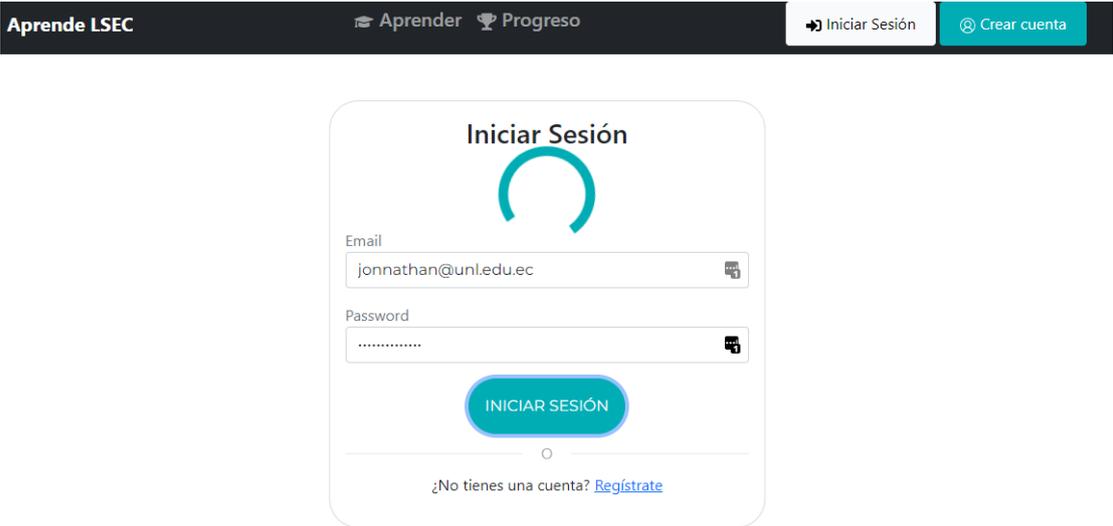
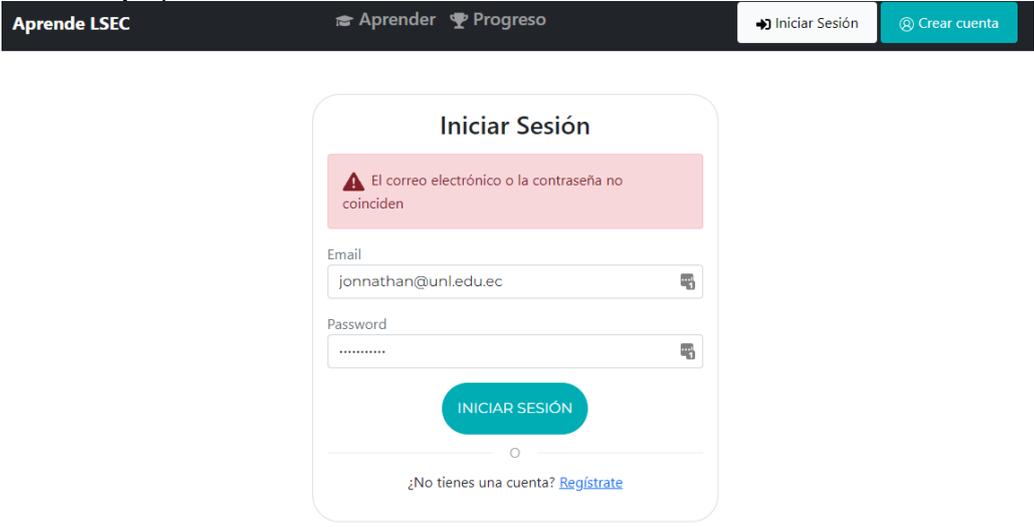
- Un mensaje en cada capo que indique el error.

Resultado obtenido:

- Un mensaje en cada capo que indique el error

The screenshot shows a web interface for 'Aprende LSEC'. At the top, there is a navigation bar with 'Aprender' and 'Progreso' icons, and buttons for 'Iniciar Sesión' and 'Crear cuenta'. Below this is a 'Crear Cuenta' form. The form contains three input fields: 'Nombre', 'Email', and 'Password'. Each field has a red error message below it: 'El nombre es un campo obligatorio', 'El correo electrónico es un campo obligatorio', and 'La contraseña es un campo requerido'. There is a 'REGISTRATE' button and a link for '¿Tienes una cuenta? Inicia sesión'.

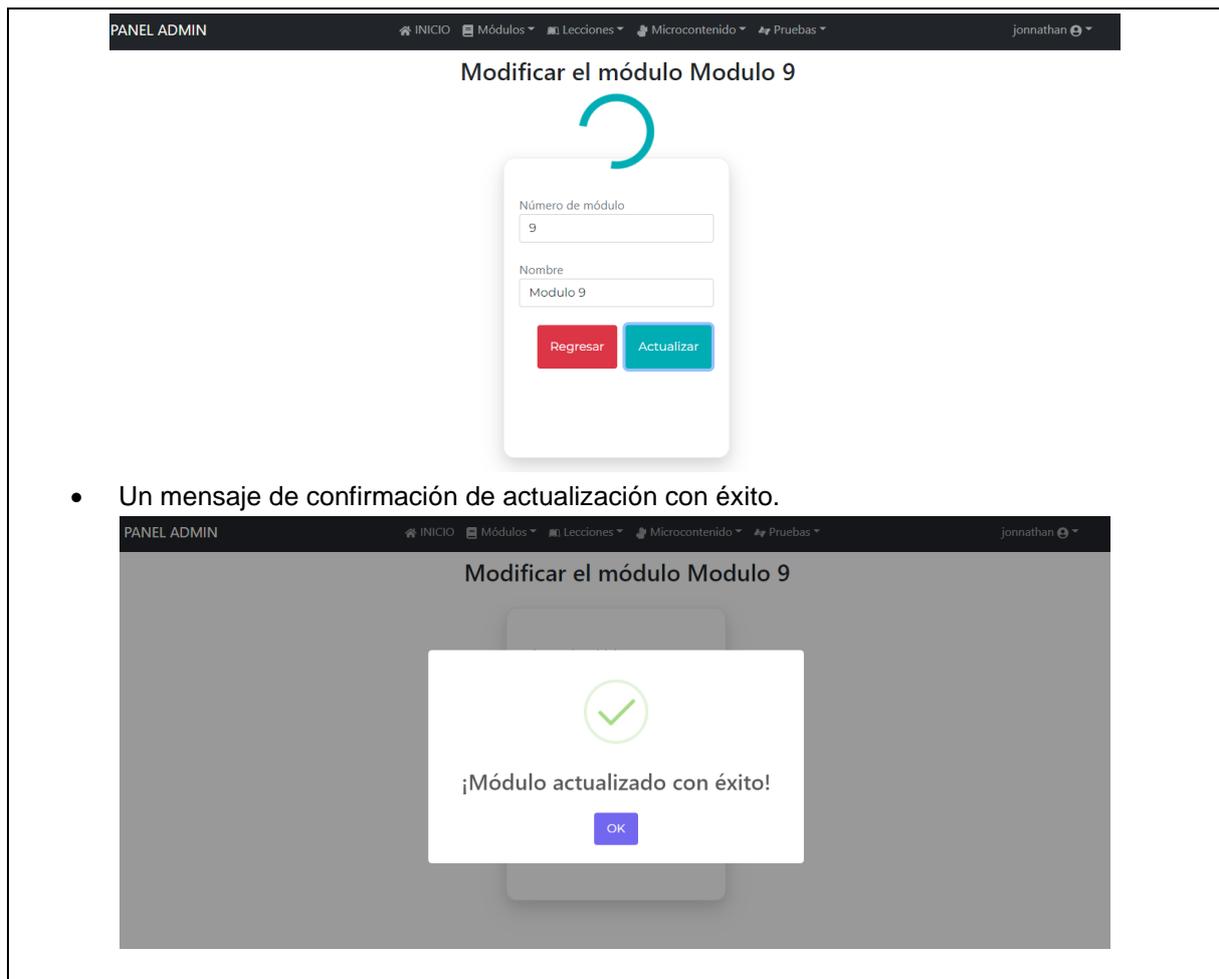
Inicio de sesión exitoso	CP03	
	¿Prueba de despliegue?	Si
Descripción: Se probará la respuesta correcta al iniciar sesión.		
Prerrequisitos: <ul style="list-style-type: none"> • Tener una cuenta registrada en la aplicación web. • Ingresar a la página principal. • Hacer clic en el botón "Iniciar Sesión". 		
Pasos: <ol style="list-style-type: none"> 1. Llenar el campo "Email". 2. Llenar el campo "Contraseña". 3. Hacer clic en el botón "INICIAR SESIÓN". 4. Esperar mensaje de confirmación. 		
Resultado esperado: <ul style="list-style-type: none"> • Un spinner de carga. • Redireccionar a la página de aprendizaje. • Almacenar el token el Local Storage del navegador. 		
Resultado obtenido: <ul style="list-style-type: none"> • Un spinner de carga. 		

Inicio de sesión fallido	CP04	
	¿Prueba de despliegue?	Si
Descripción: Se probará la respuesta al iniciar sesión con una contraseña incorrecta.		
Prerrequisitos: <ul style="list-style-type: none"> Tener una cuenta registrada en la aplicación web. Ingresar a la página principal. Hacer clic en el botón "Iniciar Sesión". 		
Pasos: <ol style="list-style-type: none"> Llenar el campo "Email". Llenar el campo "Contraseña" con una contraseña incorrecta. Hacer clic en el botón "INICIAR SESIÓN". Esperar mensaje de confirmación de error. 		
Resultado esperado: <ul style="list-style-type: none"> Un spinner de carga. Un mensaje que muestre el error. 		
Resultado obtenido: <ul style="list-style-type: none"> Un spinner de carga. 		
<ul style="list-style-type: none"> Un mensaje que muestre el error. 		

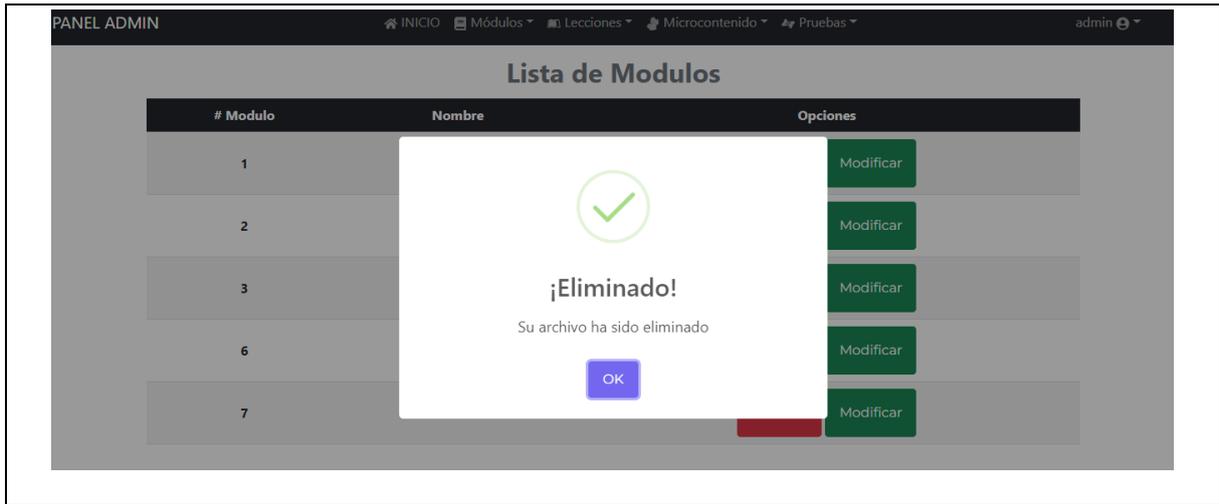
Crear un nuevo módulo	CP05	
	¿Prueba de despliegue?	Si
<p>Descripción: Se probará la respuesta correcta al crear un nuevo módulo.</p>		
<p>Prerrequisitos:</p> <ul style="list-style-type: none"> Tener registrado una cuenta con el rol administrador. Ingresar a la página de inicio de sesión para administrador. Llenar los campos del formulario y hacer clic en el botón “Enviar”. Ingresar a la página de administración. 		
<p>Pasos:</p> <ol style="list-style-type: none"> Hacer clic en “Crear módulo” en el menú. Llenar el campo “Número de módulo”. Llenar el campo “Nombre”. Hacer clic en el botón “Crear”. Hacer clic en el botón “Crear” del modal de confirmación. Esperar mensaje de confirmación. 		
<p>Resultado esperado:</p> <ul style="list-style-type: none"> Spinner de carga. Un mensaje de confirmación de registro con éxito. 		
<p>Resultado obtenido:</p> <ul style="list-style-type: none"> Spinner de carga.  <ul style="list-style-type: none"> Un mensaje de confirmación de registro con éxito. 		

Listar módulos creados	CP06	
	¿Prueba de despliegue?	Si
Descripción: Se probará la respuesta correcta al visualizar todos los módulos creados.		
Prerrequisitos: <ul style="list-style-type: none"> Tener registrado una cuenta con el rol administrador. Ingresar a la página de inicio de sesión para administrador. Llenar los campos del formulario y hacer clic en el botón “Enviar”. Ingresar a la página de administración. 		
Pasos: <ol style="list-style-type: none"> Hacer clic en la opción “Listar módulos” en el menú. Esperar que carguen los datos. 		
Resultado esperado: <ul style="list-style-type: none"> Lista de módulos creados. 		
Resultado obtenido: <ul style="list-style-type: none"> Lista de módulos creados. 		

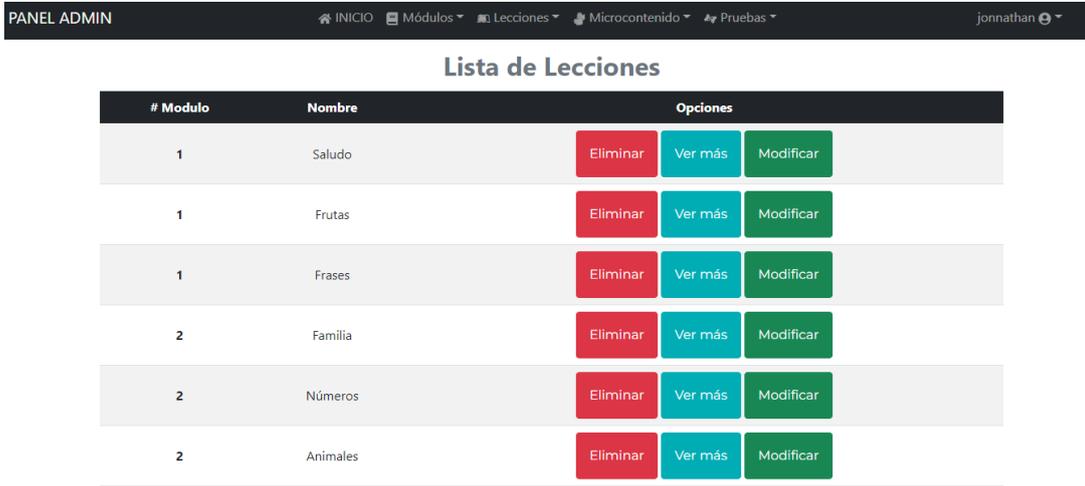
Actualizar datos del módulo	CP07	
	¿Prueba de despliegue?	Si
Descripción: Se probará la respuesta correcta al actualizar los datos de un módulo creado.		
Prerrequisitos: <ul style="list-style-type: none"> Tener registrado una cuenta con el rol administrador. Ingresar a la página de inicio de sesión para administrador. Llenar los campos del formulario y hacer clic en el botón “Enviar”. Ingresar a la página de administración. Ingresar a la lista de módulos. 		
Pasos: <ol style="list-style-type: none"> Hacer clic en el botón “Modificar” de la lista de módulos. Esperar a que la página de actualizar datos se cargue. Modificar los datos. Hacer clic en el botón “Actualizar”. Hacer clic en el botón “Guardar cambios” del modal de confirmación. 		
Resultado esperado: <ul style="list-style-type: none"> Spinner de cargando. Un mensaje de confirmación de actualización con éxito. 		
Resultado obtenido: <ul style="list-style-type: none"> Spinner de cargando. 		



Eliminar un módulo	CP08	
	¿Prueba de despliegue?	Si
Descripción: Se probará la respuesta correcta al eliminar un módulo creado.		
Prerrequisitos: <ul style="list-style-type: none"> Tener registrado una cuenta con el rol administrador. Ingresar a la página de inicio de sesión para administrador. Llenar los campos del formulario y hacer clic en el botón “Enviar”. Ingresar a la página de administración. Ingresar a la lista de módulos. 		
Pasos: <ol style="list-style-type: none"> Hacer clic en el botón “Eliminar” de la lista de módulos. Hacer clic en el botón “¡Sí, bórralo!” del modal de confirmación. 		
Resultado esperado: <ul style="list-style-type: none"> Un mensaje de confirmación de eliminado con éxito. 		
Resultado obtenido: <ul style="list-style-type: none"> Un mensaje de confirmación de eliminado con éxito. 		



Crear nueva lección	CP09	
	¿Prueba de despliegue?	Si
<p>Descripción: Se probará la respuesta correcta al crear una nueva lección.</p>		
<p>Prerrequisitos:</p> <ul style="list-style-type: none"> • Tener registrado una cuenta con el rol administrador. • Ingresar a la página de inicio de sesión para administrador. • Llenar los campos del formulario y hacer clic en el botón “Enviar”. • Ingresar a la página de administración. 		
<p>Pasos:</p> <ol style="list-style-type: none"> 1. Hacer clic en la opción “Crear lecciones” del menú. 2. Llenar el campo “Icono”. 3. Llenar el campo “Nombre de la lección”. 4. Llenar el campo “Módulo”. 5. Hacer clic en el botón “Crear”. 6. Hacer clic en el botón “Crear” del modal de confirmación. 7. Esperar mensaje de confirmación. 		
<p>Resultado esperado:</p> <ul style="list-style-type: none"> • Spinner de carga. • Un mensaje de confirmación de registro con éxito. 		
<p>Resultado obtenido:</p> <ul style="list-style-type: none"> • Spinner de carga. • Un mensaje de confirmación de registro con éxito. 		

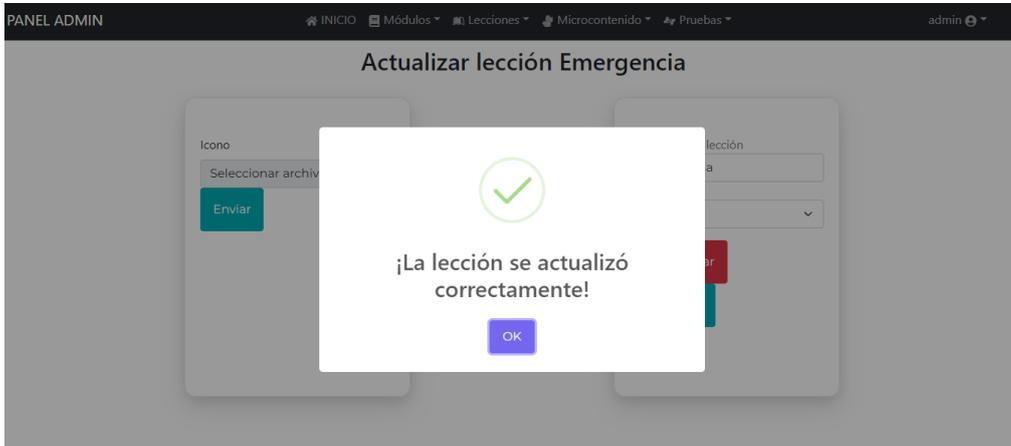
Listar lecciones creadas	CP10	
	¿Prueba de despliegue?	Si
<p>Descripción: Se probará la respuesta correcta al visualizar todas las lecciones creadas.</p>		
<p>Prerrequisitos:</p> <ul style="list-style-type: none"> Tener registrado una cuenta con el rol administrador. Ingresar a la página de inicio de sesión para administrador. Llenar los campos del formulario y hacer clic en el botón “Enviar”. Ingresar a la página de administración. 		
<p>Pasos:</p> <ol style="list-style-type: none"> Hacer clic en la opción “Listar lecciones” del menú. Esperar que carguen los datos. 		
<p>Resultado esperado:</p> <ul style="list-style-type: none"> Listar lecciones creadas. 		
<p>Resultado obtenido:</p> <ul style="list-style-type: none"> Listar lecciones creadas. 		

Actualizar datos de la lección	CP11	
	¿Prueba de despliegue?	Si
<p>Descripción: Se probará la respuesta correcta al actualizar los datos de una lección creada.</p>		
<p>Prerrequisitos:</p> <ul style="list-style-type: none"> Tener registrado una cuenta con el rol administrador. Ingresar a la página de inicio de sesión para administrador. Llenar los campos del formulario y hacer clic en el botón “Enviar”. Ingresar a la página de administración. Ingresar a la lista de lecciones. 		
<p>Pasos:</p> <ol style="list-style-type: none"> Hacer clic en el botón “Modificar” de la lista de lecciones. Esperar a que la página de actualizar datos se cargue. Modificar los datos. Hacer clic en el botón “Actualizar”. Hacer clic en el botón “Guardar cambios” del modal de confirmación. 		
<p>Resultado esperado:</p>		

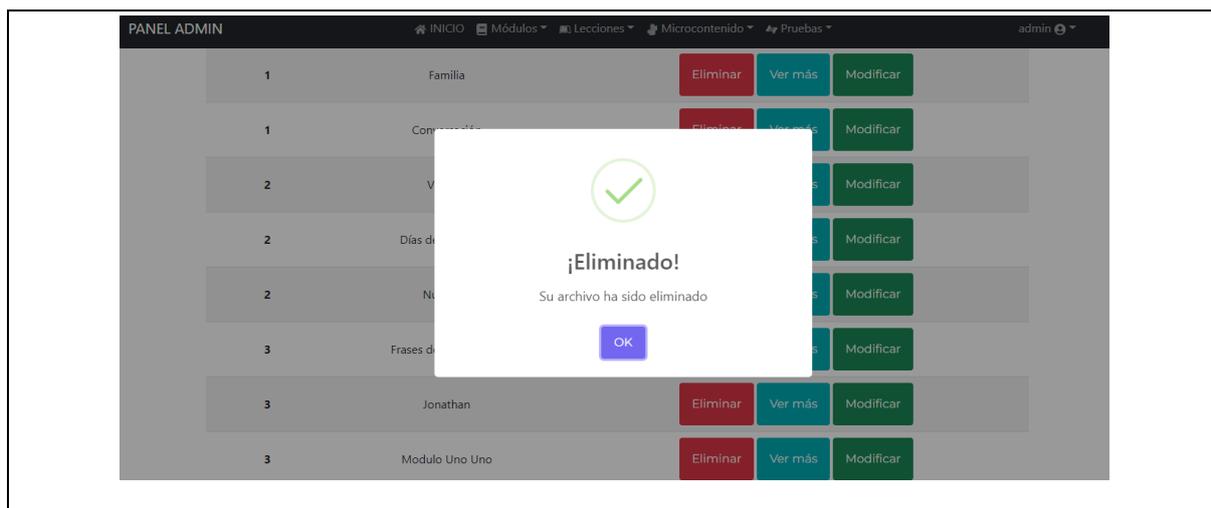
- Spinner de cargando.
- Un mensaje de confirmación de actualización con éxito.

Resultado obtenido:

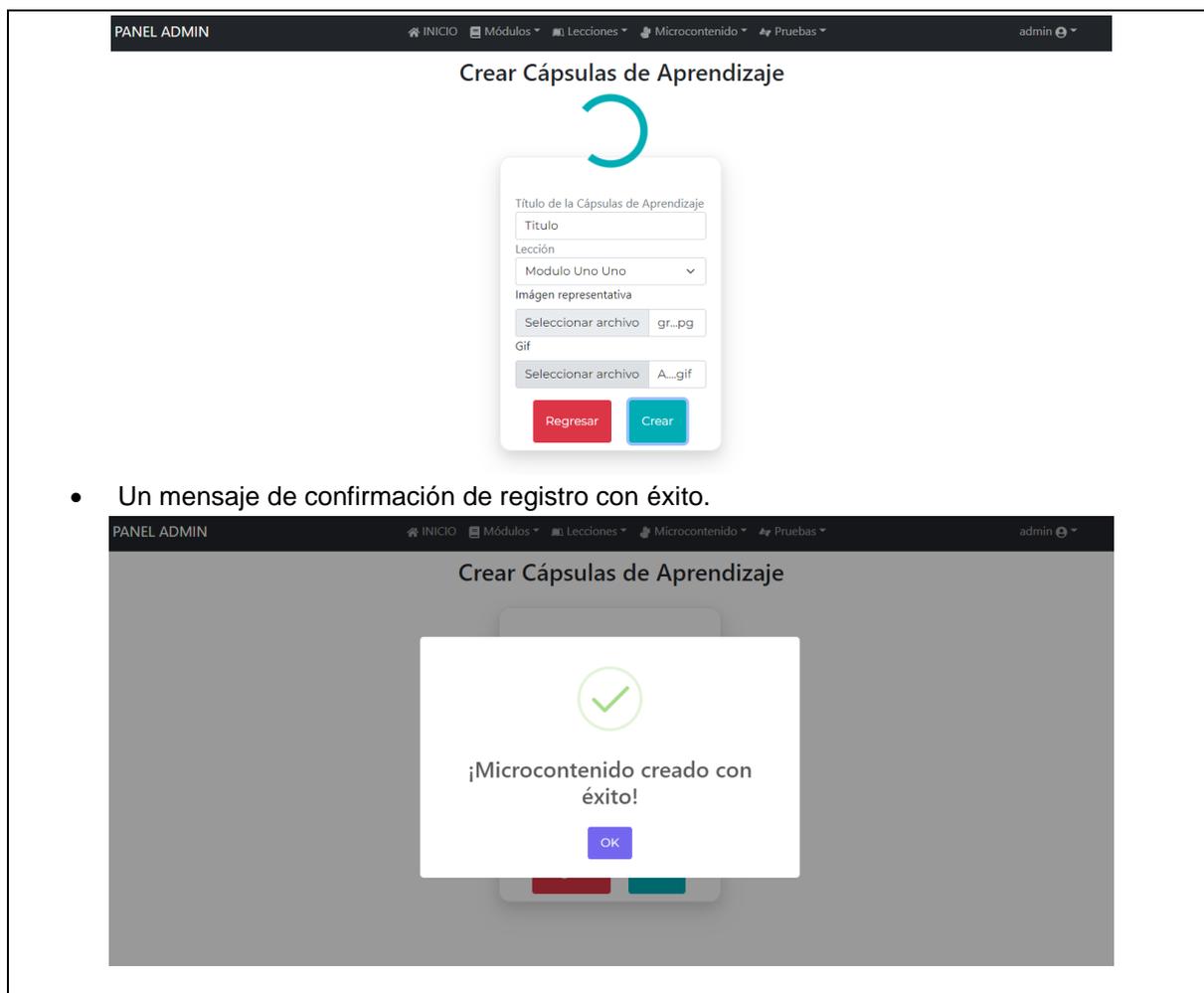
- Spinner de cargando.
- Un mensaje de confirmación de actualización con éxito.



Eliminar una lección	CP12	
	¿Prueba de despliegue?	Si
Descripción: Se probará la respuesta correcta al eliminar una lección creada.		
Prerrequisitos: <ul style="list-style-type: none"> • Tener registrado una cuenta con el rol administrador. • Ingresar a la página de inicio de sesión para administrador. • Llenar los campos del formulario y hacer clic en el botón “Enviar”. • Ingresar a la página de administración. • Ingresar a la lista de lecciones. 		
Pasos: <ol style="list-style-type: none"> 1. Hacer clic en el botón “Eliminar” de la lista de lecciones. 2. Hacer clic en el botón “¡Sí, bórralo!” del modal de confirmación. 		
Resultado esperado: <ul style="list-style-type: none"> • Spinner de cargando. • Un mensaje de confirmación de eliminado con éxito. 		
Resultado obtenido: <ul style="list-style-type: none"> • Spinner de cargando. • Un mensaje de confirmación de eliminado con éxito. 		



Crear nueva cápsula de aprendizaje	CP13	
	¿Prueba de despliegue?	Si
Descripción: Se probará la respuesta correcta al crear una cápsula de aprendizaje.		
Prerrequisitos: <ul style="list-style-type: none"> Tener registrado una cuenta con el rol administrador. Ingresar a la página de inicio de sesión para administrador. Llenar los campos del formulario y hacer clic en el botón “Enviar”. Ingresar a la página de administración. 		
Pasos: <ol style="list-style-type: none"> Hacer clic en la opción “Crear Cápsulas de Aprendizaje” del menú. Llenar el campo “Título Cápsulas de Aprendizaje”. Llenar el campo “Lección”. Llenar el campo “Imagen representativa”. Llenar el campo “Gif”. Hacer clic en el botón “Crear”. Hacer clic en el botón “Crear” del modal de confirmación. Esperar mensaje de confirmación. 		
Resultado esperado: <ul style="list-style-type: none"> Spinner de carga. Un mensaje de confirmación de registro con éxito. 		
Resultado obtenido: <ul style="list-style-type: none"> Spinner de carga. 		

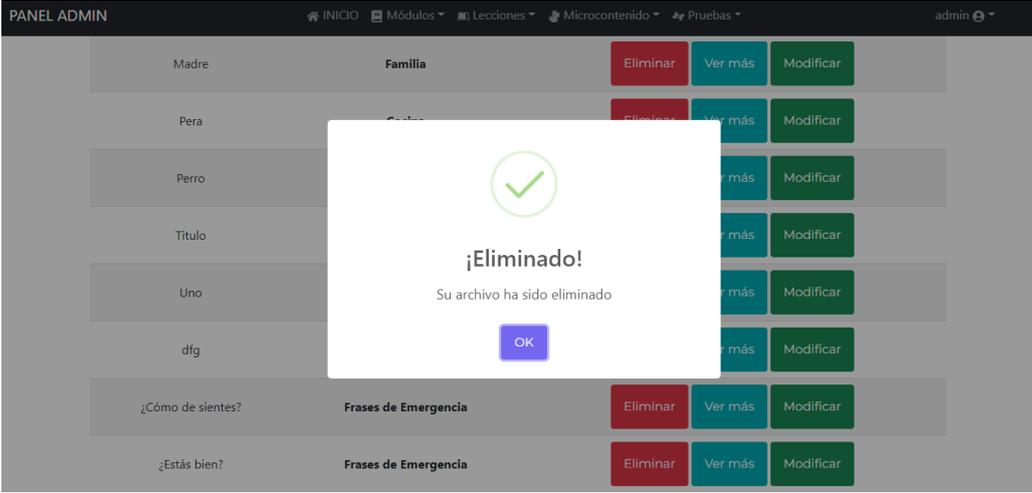


- Un mensaje de confirmación de registro con éxito.

Listar cápsulas de aprendizaje	CP14	
	¿Prueba de despliegue?	Si
Descripción: Se probará la respuesta correcta al visualizar todas las cápsulas de aprendizaje creadas.		
Prerrequisitos: <ul style="list-style-type: none"> • Tener registrado una cuenta con el rol administrador. • Ingresar a la página de inicio de sesión para administrador. • Llenar los campos del formulario y hacer clic en el botón “Enviar”. • Ingresar a la página de administración. 		
Pasos: <ol style="list-style-type: none"> 1. Hacer clic en la opción “Listar Cápsulas de Aprendizaje” del menú. 2. Esperar que carguen los datos. 		
Resultado esperado: <ul style="list-style-type: none"> • Listar Cápsulas de Aprendizaje creadas. 		
Resultado obtenido: <ul style="list-style-type: none"> • Listar Cápsulas de Aprendizaje creadas. 		

Lista de Cápsulas de Aprendizaje		
Nombre	Lección	Opciones
Cédula de identidad	Conversación	Eliminar Ver más Modificar
Dos	Números	Eliminar Ver más Modificar
Gato	Animales	Eliminar Ver más Modificar
Hermana	Familia	Eliminar Ver más Modificar
Hermano	Familia	Eliminar Ver más Modificar
Lunes	Días de la semana	Eliminar Ver más Modificar

Actualizar cápsula de aprendizaje	CP15	
	¿Prueba de despliegue?	Si
<p>Descripción: Se probará la respuesta correcta al actualizar los datos de una cápsula de aprendizaje.</p>		
<p>Prerrequisitos:</p> <ul style="list-style-type: none"> Tener registrado una cuenta con el rol administrador. Ingresar a la página de inicio de sesión para administrador. Llenar los campos del formulario y hacer clic en el botón “Enviar”. Ingresar a la página de administración. Ingresar a la lista de Cápsulas de Aprendizaje. 		
<p>Pasos:</p> <ol style="list-style-type: none"> Hacer clic en el botón “Modificar” de la lista de Cápsulas de Aprendizaje. Esperar a que la página de actualizar datos se cargue. Modificar los datos. Hacer clic en el botón “Actualizar”. Hacer clic en el botón “Guardar cambios” del modal de confirmación. 		
<p>Resultado esperado:</p> <ul style="list-style-type: none"> Spinner de cargando. Un mensaje de confirmación de actualización con éxito. 		
<p>Resultado obtenido:</p> <ul style="list-style-type: none"> Spinner de cargando. Un mensaje de confirmación de actualización con éxito. 		
		

Eliminar una cápsula de aprendizaje	CP16	
	¿Prueba de despliegue?	Si
Descripción: Se probará la respuesta correcta al eliminar una cápsula de aprendizaje creada.		
Prerrequisitos: <ul style="list-style-type: none"> Tener registrado una cuenta con el rol administrador. Ingresar a la página de inicio de sesión para administrador. Llenar los campos del formulario y hacer clic en el botón “Enviar”. Ingresar a la página de administración. Ingresar a la lista de microcontenidos. 		
Pasos: <ol style="list-style-type: none"> Hacer clic en el botón “Eliminar” de la lista de microcontenidos. Hacer clic en el botón “¡Sí, bórralo!” del modal de confirmación. 		
Resultado esperado: <ul style="list-style-type: none"> Spinner de cargando. Un mensaje de confirmación de eliminado con éxito. 		
Resultado obtenido: <ul style="list-style-type: none"> Spinner de cargando. Un mensaje de confirmación de eliminado con éxito. 		

Crear una tarjeta de aprendizaje	CP17	
	¿Prueba de despliegue?	Si
Descripción: Se probará la respuesta correcta al crear una tarjeta de aprendizaje.		
Prerrequisitos: <ul style="list-style-type: none"> Tener registrado una cuenta con el rol administrador. Ingresar a la página de inicio de sesión para administrador. Llenar los campos del formulario y hacer clic en el botón “Enviar”. Ingresar a la página de administración. 		
Pasos: <ol style="list-style-type: none"> Hacer clic en la opción “Crear Tarjeta de Aprendizaje” del menú. Llenar el campo “Pregunta”. Llenar el campo “Lección”. Llenar el campo “Respuesta correcta”. 		

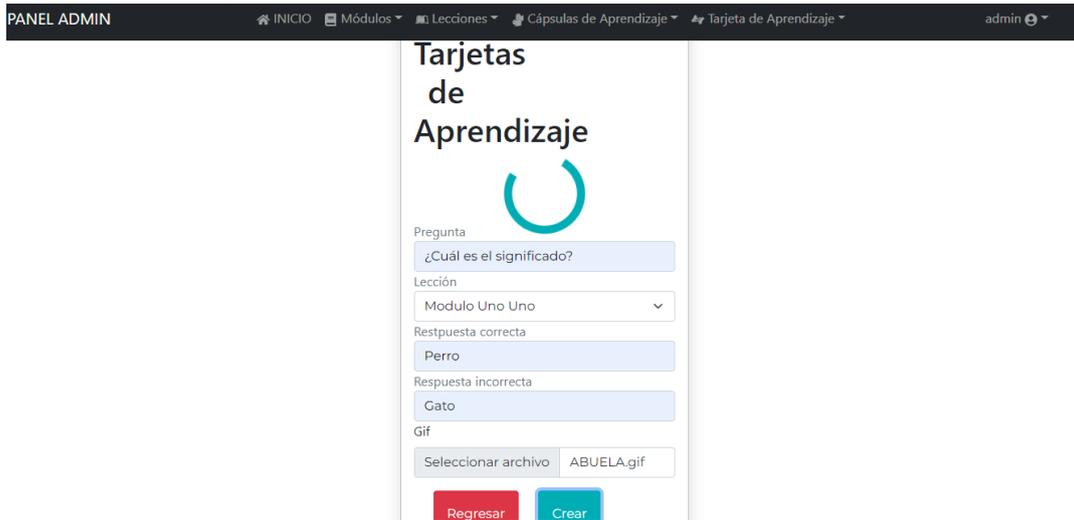
5. Llenar el campo "Respuesta incorrecta".
6. Llenar el campo "Gif".
7. Hacer clic en el botón "Crear".
8. Hacer clic en el botón "Crear" del modal de confirmación.
9. Esperar mensaje de confirmación.

Resultado esperado:

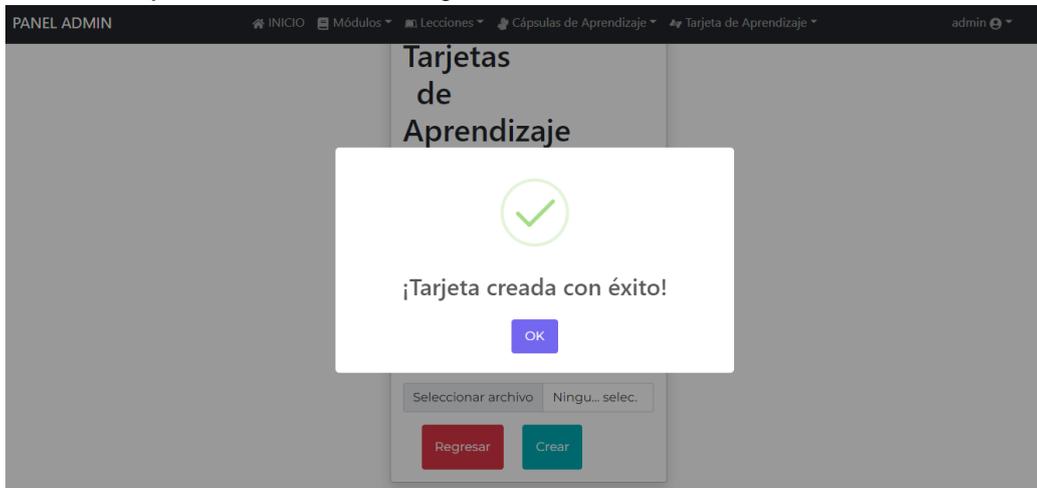
- Spinner de carga.
- Un mensaje de confirmación de registro con éxito.

Resultado obtenido:

- Spinner de carga.



- Un mensaje de confirmación de registro con éxito.



Listar tarjetas de aprendizaje	CP18	
	¿Prueba de despliegue?	Si
Descripción: Se probará la respuesta correcta al visualizar todas las tarjetas de aprendizaje creadas.		
Prerrequisitos: <ul style="list-style-type: none"> • Tener registrado una cuenta con el rol administrador. • Ingresar a la página de inicio de sesión para administrador. • Llenar los campos del formulario y hacer clic en el botón "Enviar". • Ingresar a la página de administración. 		
Pasos:		

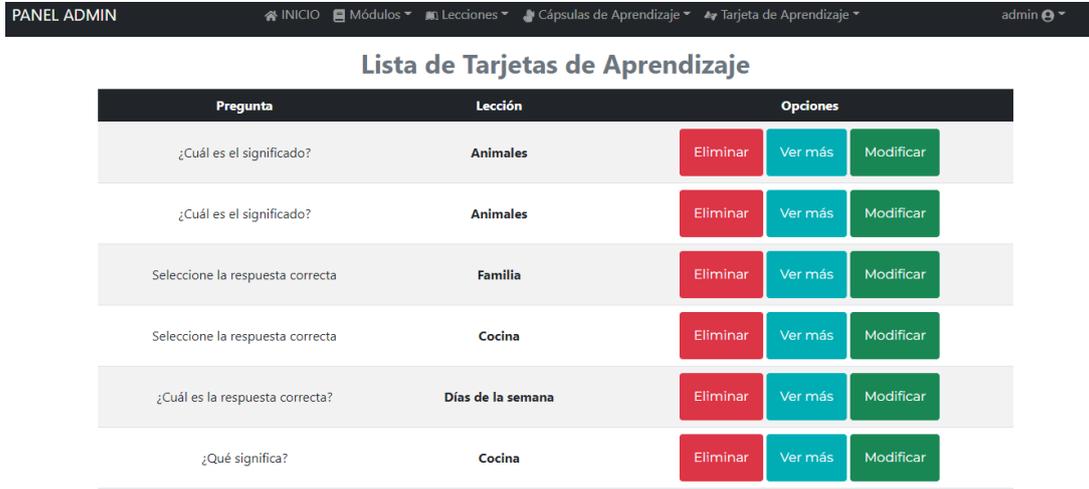
1. Hacer clic en la opción “Listar Tarjeta de Aprendizaje” del menú.
2. Esperar que carguen los datos.

Resultado esperado:

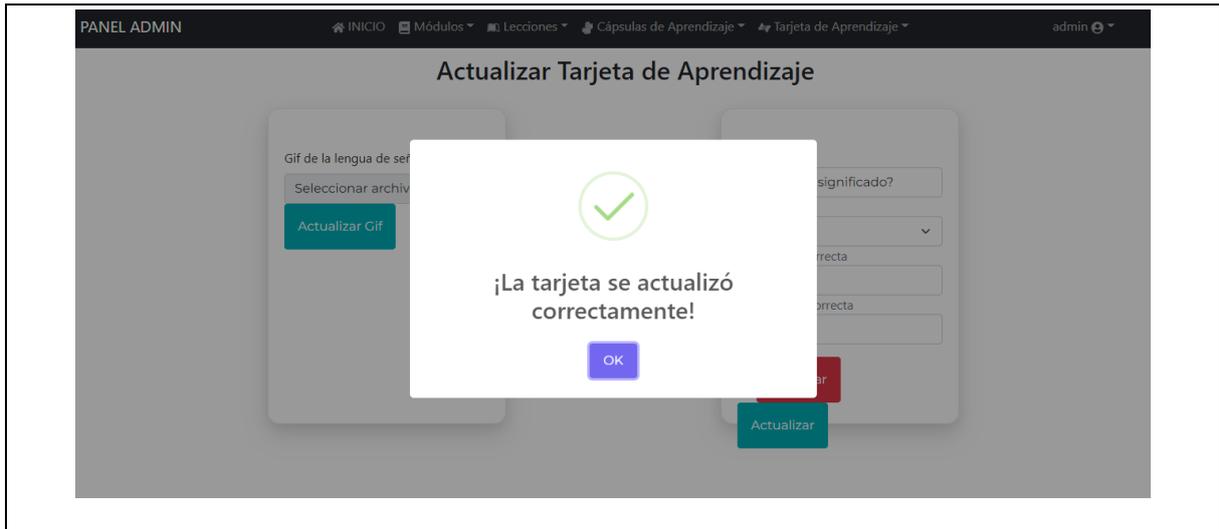
- Listar tarjetas de aprendizaje

Resultado obtenido:

- Listar tarjetas de aprendizaje



Actualizar tarjeta de aprendizaje	CP19	
	¿Prueba de despliegue?	Si
<p>Descripción: Se probará la respuesta correcta al actualizar los datos de una tarjeta de aprendizaje.</p>		
<p>Prerrequisitos:</p> <ul style="list-style-type: none"> • Tener registrado una cuenta con el rol administrador. • Ingresar a la página de inicio de sesión para administrador. • Llenar los campos del formulario y hacer clic en el botón “Enviar”. • Ingresar a la página de administración. • Ingresar a la lista de tarjetas de aprendizaje. 		
<p>Pasos:</p> <ol style="list-style-type: none"> 1. Hacer clic en el botón “Modificar” de la lista de tarjetas de aprendizaje. 2. Esperar a que la página de actualizar datos se cargue. 3. Modificar los datos. 4. Hacer clic en el botón “Actualizar”. 5. Hacer clic en el botón “Guardar cambios” del modal de confirmación. 		
<p>Resultado esperado:</p> <ul style="list-style-type: none"> • Spinner de cargando. • Un mensaje de confirmación de actualización con éxito. 		
<p>Resultado obtenido:</p> <ul style="list-style-type: none"> • Spinner de cargando. • Un mensaje de confirmación de actualización con éxito. 		



Eliminar tarjeta de aprendizaje	CP20	
	¿Prueba de despliegue?	Si
<p>Descripción: Se probará la respuesta correcta al eliminar una tarjeta de aprendizaje creada.</p>		
<p>Prerrequisitos:</p> <ul style="list-style-type: none"> • Tener registrado una cuenta con el rol administrador. • Ingresar a la página de inicio de sesión para administrador. • Llenar los campos del formulario y hacer clic en el botón “Enviar”. • Ingresar a la página de administración. • Ingresar a la lista de tarjetas de aprendizaje. 		
<p>Pasos:</p> <ol style="list-style-type: none"> 1. Hacer clic en el botón “Eliminar” de la lista de tarjetas de aprendizaje. 2. Hacer clic en el botón “¡Sí, bórralo!” del modal de confirmación. 		
<p>Resultado esperado:</p> <ul style="list-style-type: none"> • Spinner de cargando. • Un mensaje de confirmación de eliminado con éxito. 		
<p>Resultado obtenido:</p> <ul style="list-style-type: none"> • Spinner de cargando. • Un mensaje de confirmación de eliminado con éxito. 		

Cerrar sesión	CP21	
	¿Prueba de despliegue?	Si

Descripción:

Se probará la respuesta correcta al cerrar sesión.

Prerrequisitos:

- Tener una cuenta registrada en la aplicación web.
- Ingresar a la página principal.
- Iniciar sesión en la aplicación.

Pasos:

1. Hacer clic en select del menú de usuario.
2. Hacer clic en el botón “Cerrar sesión”.
3. Esperar a que la página se redireccione a la página principal.

Resultado esperado:

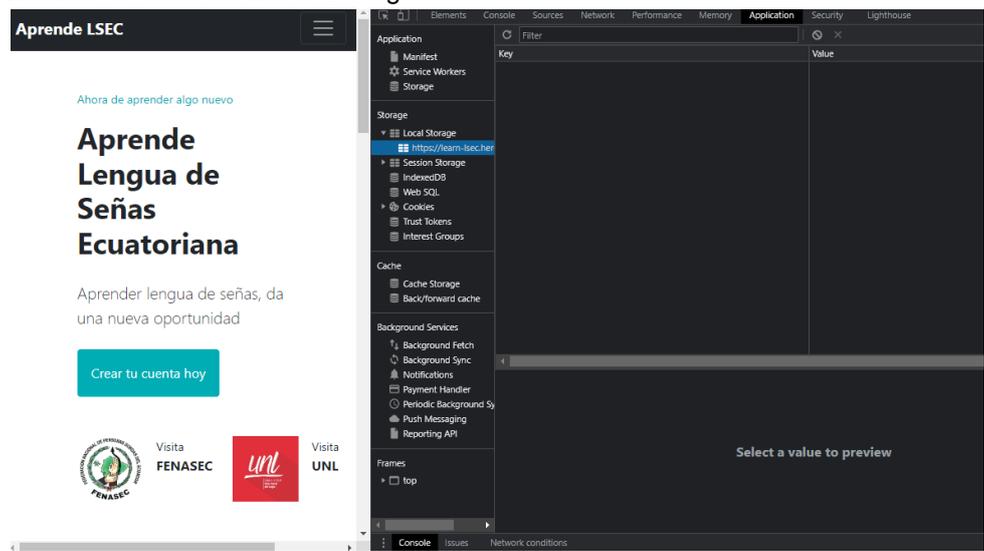
- Redireccionar a la página principal.
- Eliminar el token del Local Storage.

Resultado obtenido:

- Redireccionar a la página principal.



- Eliminar el token del Local Storage.



Visualizar lecciones completadas	CP22	
	¿Prueba de despliegue?	Si

Descripción:

Se probará la repuesta al listar todas las lecciones completadas por el usuario.

Prerrequisitos:

- Tener una cuenta registrada en la aplicación web.
- Ingresar a la página principal.
- Iniciar sesión en la aplicación.
- Completar una lección.

Pasos:

1. Hacer clic en la opción "Progreso" del menú de usuario.
2. Esperar a que cargue la página de Progreso.

Resultado esperado:

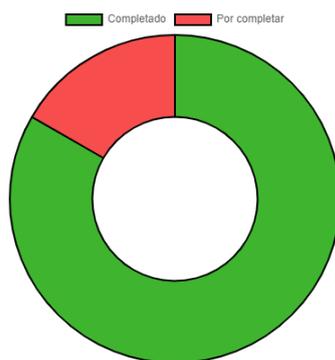
- Visualizar un gráfico de las lecciones completadas.
- Visualizar una tabla de las lecciones.

Resultado obtenido:

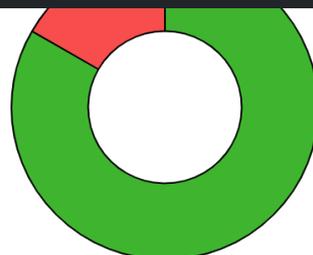
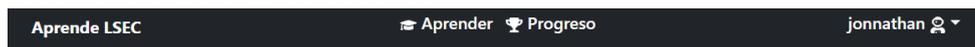
- Visualizar un gráfico de las lecciones completadas.



Avance de las Lecciones



- Visualizar una tabla de las lecciones.



# de Módulo	Lecciones	Completado
1	Saludo	✓
1	Frutas	✓
1	Frases	✓
2	Familia	✗
2	Números	✓
2	Animales	✓

4. Estrategia de ejecución de pruebas

En la siguiente matriz se indica los ciclos realizados para la ejecución de los casos de prueba.

	Ciclo 1	Ciclo 2	Ciclo 2
CP01			X
CP02			X
CP03			X
CP04			X
CP05	X		
CP06	X		
CP07	X		
CP08	X		
CP09	X		
CP10	X		
CP11	X		
CP12	X		
CP13		X	
CP14		X	
CP15		X	
CP16		X	
CP17		X	
CP18		X	
CP19		X	
CP20		X	
CP21			X
CP22			X

5. Bibliografía y referencias

Referencia	Título
Ref. 1	Plantilla Plan de Pruebas Funcionales del portal del marco de desarrollo de software de la Junta de Andalucía (MADEJA)
Ref. 2	Documento de Especificación de Requerimientos de Software IEEE 830 (Anexo 2)

Anexo 12. Plan de Pruebas de Aceptación

Pruebas de Aceptación

Proyecto: Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano

Versión: 1.0

Fecha: 18/07/2022

Hoja de control

Proyecto	Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano
Entregable	Pruebas de Aceptación.
Autor	Jonnathan Espinoza
Versión	1.0
Aprobado por	Edison Leonardo Coronel Romero, Mg. Sc.

Registro de cambios

Versión del documento	Descripción	Responsable	Fecha de cambio
1.0	Versión inicial de pruebas de aceptación.	Jonnathan Espinoza	18/07/2022

Índice

Hoja de control	207
Índice	208
1. Introducción.....	209
Objetivo	209
Propósito	209
Hipótesis.....	209
2. Diseño de encuestas	210
3. Ejecución de encuestas.....	211
4. Análisis de resultados.....	216
4.1. Evaluación de la hipótesis H1	216
4.2. Sugerencias al investigador	216
5. Amenazas a la validez de los resultados obtenidos.....	219

1. Introducción

Objetivo

Determinar el nivel de aceptación de la solución informática por medio de una muestra de personas de la Universidad Nacional de Loja.

Propósito

Validar la aplicación web con una muestra de personas de la Universidad Nacional de Loja mediante la técnica de las encuestas.

Hipótesis

La hipótesis planteada es:

- H1: La aplicación web para enseñar lengua de señas ecuatoriana mediante el micro-learning tiene una buena aceptación por parte de los usuarios finales.

2. Diseño de encuestas

A continuación, se presenta el cuestionario, el cual consta de 12 preguntas:

Tabla 80.
Preguntas de las encuestas.

N°	Pregunta
P1	¿Considera que la interfaz de la aplicación es intuitiva?
P2	¿Se entiende el contenido dentro de la aplicación web?
P3	¿Resulta fácil entender el resultado de una acción?
P4	¿Son apropiados los mensajes presentados por el sistema?
P5	¿La aplicación presenta claramente los errores?
P6	¿Permite una cómoda navegación dentro de la aplicación?
P7	¿Es fácil identificar un objeto o acción?
P8	¿La interfaz está diseñada para facilitar la realización de las tareas de la mejor forma posible?
P9	¿El uso de GIFs (imágenes animadas) ayuda en la comprensión de la lengua de señas?
P10	¿Los repasos o prácticas por medio de la selección múltiple en la aplicación web, sirven como un medio de practicar la lengua de señas?
P11	¿La aplicación web garantiza su operabilidad al no presentar fallas?
P12	¿La funcionalidad de la aplicación web, permite el aprendizaje de la lengua de señas?

En la siguiente tabla se determina los criterios de las pruebas de aceptación:

Tabla 81.
Criterio para el estado de las Pruebas de Aceptación

Estado de las Pruebas de Aceptación	Criterio		
	Si	Parcialmente	No
Positivo	$\geq 70\%$	$\leq 30\%$	$< 1\%$
Negativo	$< 70\%$	$> 30\%$	$> 1\%$

3. Ejecución de encuestas

La investigación realizada tiene como finalidad conocer si las aplicaciones web desarrolladas en el trabajo de investigación cumplen con los requisitos de aceptación operativa y de usuario final. Se tomó una muestra de 23 estudiantes de la Universidad Nacional de Loja específicamente de la carrera de ingeniería en sistemas. La muestra poblacional fue seleccionada convenientemente de acuerdo a su accesibilidad a los investigadores por métodos no probabilísticos.

La encuesta se realizó por medio de los formularios de Google Form, como se presenta en la Figura 106.



Encuesta LSEC ☆ Se han guardado todos los cambios en Drive

Preguntas Respuestas Configuración

UNL Universidad Nacional de Loja

Pruebas de aceptación de la aplicación web

La presente encuesta tiene el objetivo de determinar la aceptación de la aplicación web para la enseñanza de la lengua de señas ecuatoriana.

Estimado estudiante, la encuesta es totalmente de carácter voluntaria, por lo cual tiene plena facultad de decisión a participar o no en el proceso; la recolección de los datos, se realizará de forma anónima y confidencial.

Este formulario registra automáticamente los correos de los usuarios de Universidad Nacional de Loja. [Cambiar configuración](#)

Nombres y apellidos *

Texto de respuesta corta

Cédula *

Figura 106. Encuestas en Google Form.

Además, se solicitó al encuestado, nombre, apellido, numero de cedula, correo institucional y sugerencias con el fin de dar fiabilidad a las respuestas. A continuación, se presentan los resultados obtenidos en las encuestas.

1. ¿Considera que la interfaz de la aplicación es intuitiva?

24 respuestas

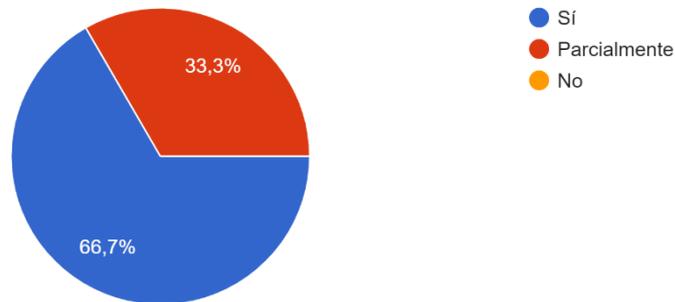


Figura 107. Respuesta a la pregunta 1.

2. ¿Se entiende el contenido dentro de la aplicación web?

24 respuestas

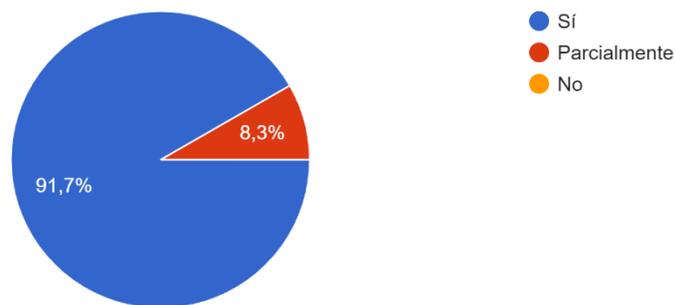


Figura 108. Respuesta a la pregunta 2.

3. ¿Resulta fácil entender el resultado de una acción?

24 respuestas

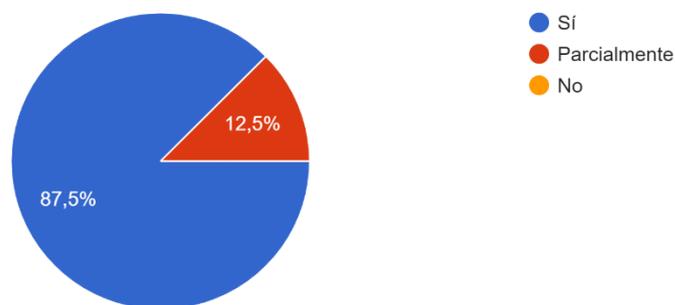


Figura 109. Respuesta a la pregunta 3

4. ¿Son apropiados los mensajes presentados por el sistema?

24 respuestas

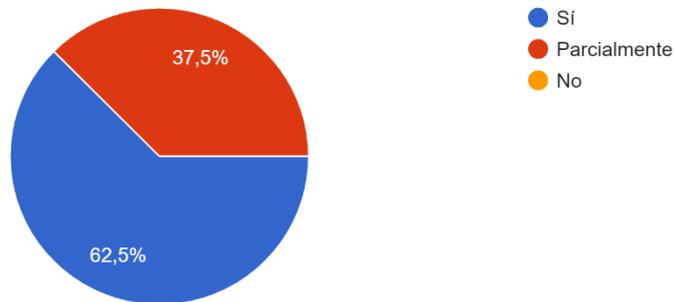


Figura 110. Respuesta a la pregunta 4.

5. ¿La aplicación presenta claramente los errores?

24 respuestas

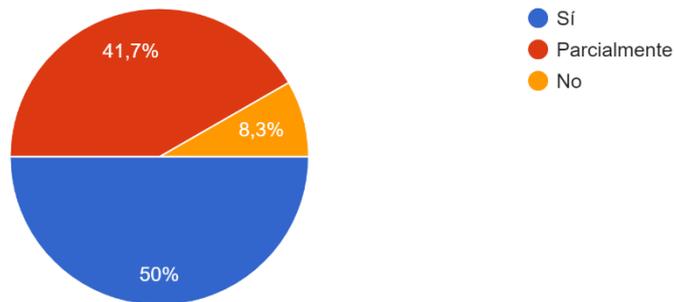


Figura 111. Respuesta a la pregunta 5.

6. ¿Permite una cómoda navegación dentro de la aplicación?

24 respuestas

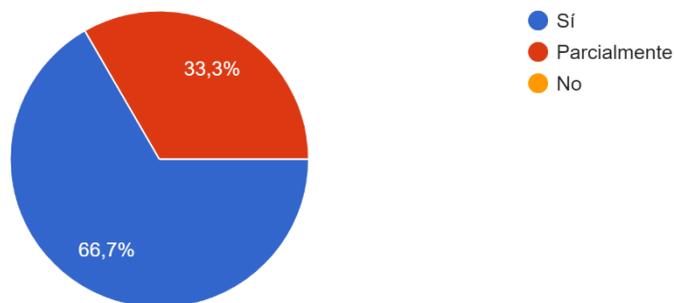


Figura 112. Respuesta a la pregunta 6.

7. ¿Es fácil identificar un objeto o acción?

24 respuestas

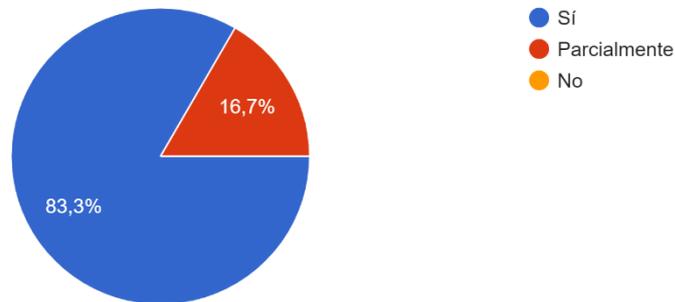


Figura 113. Respuesta a la pregunta 7.

8. ¿La interfaz está diseñada para facilitar la realización de las tareas de la mejor forma posible?

24 respuestas

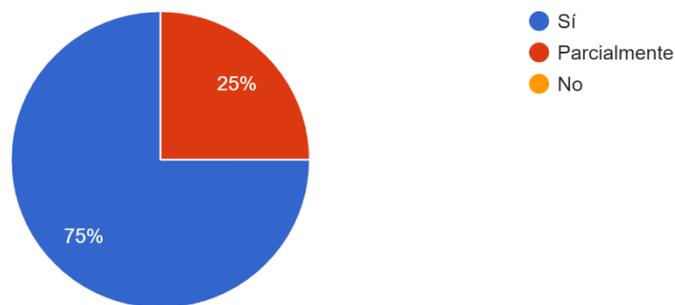


Figura 114. Respuesta a la pregunta 8.

9. ¿El uso de GIFs (imágenes animadas) ayuda en la comprensión de la lengua de señas?

24 respuestas

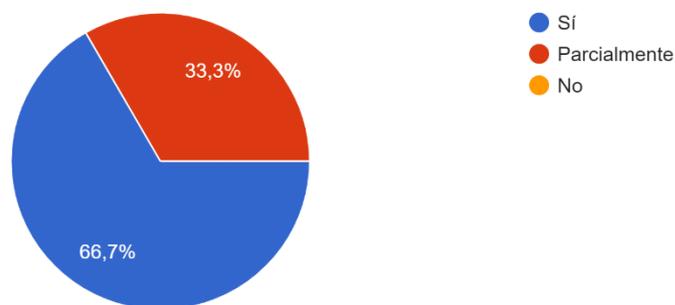


Figura 115. Respuesta a la pregunta 9.

10. ¿Los repasos o prácticas por medio de la selección múltiple en la aplicación web, sirven como un medio de practicar la lengua de señas?

24 respuestas

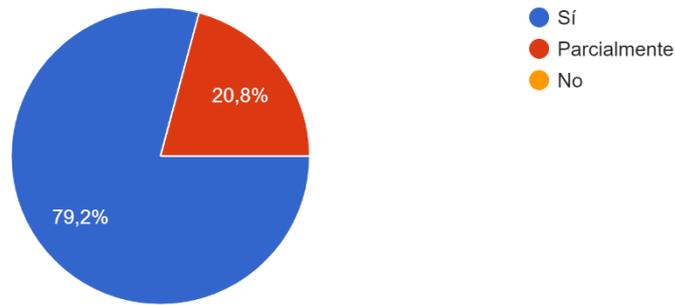


Figura 116. Respuesta a la pregunta 10.

11. ¿La aplicación web garantiza su operabilidad al no presentar fallas?

24 respuestas

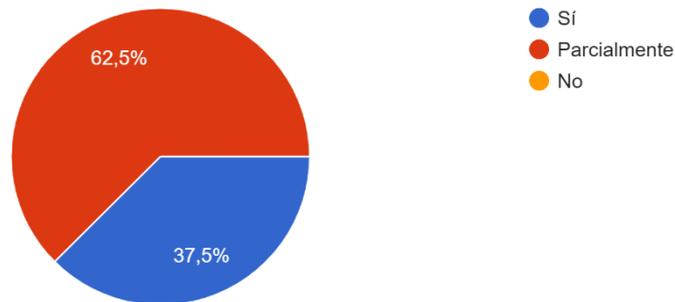


Figura 117. Respuesta a la pregunta 11.

12. ¿La funcionalidad de la aplicación web, permite el aprendizaje de la lengua de señas?

24 respuestas

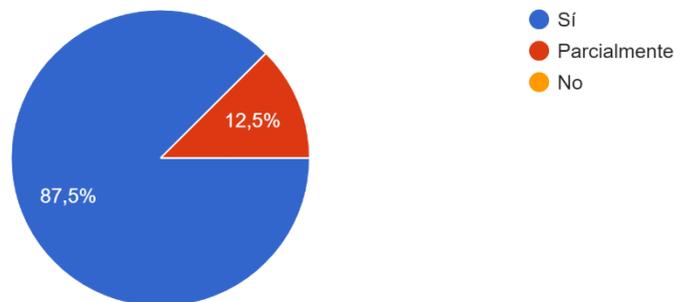


Figura 118. Respuesta a la pregunta 12.

4. Análisis de resultados

La Tabla 82 presenta el resumen de los resultados a las encuestas realizadas.

Tabla 82.
Resumen de las respuestas a las encuestas

No.	Parámetros	Porcentaje (%)		
		Sí	Parcialmente	No
1	¿Considera que la interfaz de la aplicación es intuitiva?	66,7	33,3	0
2	¿Se entiende el contenido dentro de la aplicación web?	91,7	8,3	0
3	¿Resulta fácil entender el resultado de una acción?	87,5	12,5	0
4	¿Son apropiados los mensajes presentados por el sistema?	62,5	37,5	0
5	¿La aplicación presenta claramente los errores?	50	41,7	8,3
6	¿Permite una cómoda navegación dentro de la aplicación?	66,7	33,3	0
7	¿Es fácil identificar un objeto o acción?	83,3	16,7	0
8	¿La interfaz está diseñada para facilitar la realización de las tareas de la mejor forma posible?	75	25	0
9	¿El uso de GIFs (imágenes animadas) ayuda en la comprensión de la lengua de señas?	66,7	33,3	0
10	¿Los repasos o prácticas por medio de la selección múltiple en la aplicación web, sirven como un medio de practicar la lengua de señas?	79,2	20,8	0
11	¿La aplicación web garantiza su operabilidad al no presentar fallas?	37,5	62,5	0
12	¿La funcionalidad de la aplicación web, permite el aprendizaje de la lengua de señas?	87,5	12,5	0
PROMEDIO		71,19	28,12	0,69

Como se puede observar en la Tabla 82, los promedios para los criterios **Sí**, **Parcial** y **No** son 71,19%, 28,12% y 0,69%, respectivamente. Según los criterios de aceptación y las respuestas promedio de la Tabla 81, el estado de la prueba de aceptación es **Positivo**.

a. Evaluación de la hipótesis H1

El 71,19% de los estudiantes indicaron un criterio de **Sí**, considerando que la interfaz de la aplicación es intuitiva, se entiende el contenido, aceptación de los GIFs y que la aplicación web permite el aprendizaje de la lengua de señas. Como resultado a este análisis se puede aceptar la hipótesis H1.

b. Sugerencias al investigador

Adicionalmente se agregó una pregunta para obtener sugerencias por parte de los encuestados, entre las cuales se encuentran las siguientes:

- Mejorar los tiempos de respuesta de la aplicación al seleccionar una opción o pasar de un ítem a otro.
- En las practicas ya te muestra las respuestas. Cuando seleccionas una correcta la pantalla de correcto aparece en un extremo de la pantalla. Los gifs cargan lento, por lo que al pasar a la siguiente prueba no debe presentar el gif por que confunde si es que se cambió o no. El menú o el panel superior donde se encuentra aprender y progreso no debe aparecer cuando no se inicia sesión o en la pantalla principal. Los botones de registrarse y crear cuenta en el menú veo innecesario ya que también hay otra manera de hacer eso.
- En la práctica de frutas usar más ejemplos, además el mensaje de confirmación en ocasiones aparece en el lado derecho y no al centro.
- En el apartado de frases, específicamente el módulo de practicar los gifs está intercambiados referente a las respuestas a colocar. En el apartado de familia, en el módulo de aprender el lenguaje de señas "hermana" tienes dos gifs, puedes colocar unas fotos de "dos niñas abrazándose". Sería buena idea quitar el texto de los gifs en el módulo de practicar, para lograr un mejor resultado en el aprendizaje de la lengua de señas.
- Tratar de mejorar la respuesta por parte de los gifs a veces se tardan en actualizar, evitar los subtítulos en algunas respuestas de vez en cuando estamos en el segmento de practica suelen aparecer los gifs con el subtítulo por lo que se nos está entregando la respuesta de la misma pregunta.
- Deberían quitarse los subtítulos en los gifs, ya que continúan al momento de realizar las prácticas.
- Que la práctica puede ser de todas las palabras aprendidas o de forma aleatoria en cada intento, modales de las lecciones estén centradas cuando ocurren por primera vez.
- Que en las prácticas tiene que ser todas las palabras de la lección o, en cambio, que sean aleatorias.
- La opción de PRACTICAR (considerando el apartado de "números") debería ser más aleatorio con respecto a la selección de opciones de práctica, puesto que al volver a pulsar aprender el sistema muestra las mismas 2 opciones anteriores, finalmente recomendando revisar las respuestas de algunas preguntas a la hora de practicar, puesto que las respuestas no tienen la respuesta correcta a lo que el GIF corresponde.
- Mejorar el contenido. Agregar contenido. Mejorar rendimiento del servidor, ya que, al cargar las nuevas secciones se genera un delay entre los gifs.

- El error más evidente sería que al momento de cargar la siguiente lección el gif no se actualiza al mismo tiempo, lo que provoca confusión.
- Mejorar la interacción con el usuario, y algunos errores al hacer clic en botones.
- El tiempo de cambio de las imágenes al realizar la práctica.
- Creo que la aplicación cumple con las funcionalidades.
- En las prácticas para poder verificar la respuesta se tiene que hacerlo una doble verificación, solo esa parte encontré de incomoda.
- Tal vez mejorar la navegación por los cursos con una barra de navegación que permita visualizar el contenido del curso.
- Arreglar textos en el apartado de números como por ejemplos "cinco mil" debería ser solo 5, revisar cada número respectivamente, verificar también que los gifs cambien al mismo tiempo que las demás viñetas ya que cuando ocurre un problema de internet lento se puede llegar a confundir con lo que el texto dice y con lo que el gif muestra.
- En primera, hay recargos en la página inesperados. La creación de cuenta puede contar con el inicio fácil que ofrecen los servicios como Google, Facebook y de más, Además se puede incluir la autenticación de 2 pasos.
- Se podría incluir una forma de medir el progreso en relación a todo lo que se tiene que estudiar dentro del aplicativo. Por otra parte, de ser posible mejorar la interfaz de las tarjetas, creo que se puede mejorar el espacio que se ocupan, principalmente hay una desproporción, cuando los botones ocupan demasiado espacio que las imágenes podrían aprovechar.
- Es genial como se marcan las secciones que han sido realizadas, pero no funciona correctamente, debido a que solo se marca una sección cuando fue realizada y la otra sección, aunque se complete no lo hace.
- ¿Has usado Typing Club? - Si la respuesta es no, esta es una página web para aprender a mecanografiar, la forma de aprender es muy buena, puedes analizar su forma de enseñar, pero en relación a esto puedes implementar una etapa inicial donde se enseñe cómo funciona el aplicativo tal como en Typing Club en un comienzo.
- Finalmente, la parte de practica tiene un problema, se ven las respuestas. Por otra parte, si pudieras incluir un modo noche sería genial.
- La aplicación se ve bastante bien hasta ahora entiendo que ya hace falta pulir detalles, animo puede quedar aún mejor.
- Cuando se cambia de palabra en la misma actividad las imágenes tardan en corresponder con la palabra señalada en pantalla.
- GUI, mejorar ciertos aspectos.
- El inicio de la aplicación en mi dispositivo móvil no es responsive.

5. Amenazas a la validez de los resultados obtenidos

En esta sección se describen los aspectos que pueden haber afectado la validez del experimento y los métodos que se intentaron mitigar.

- **Validez externa:** En relación con la generalización de los resultados, es posible que se haya visto afectado por el uso de estudiantes como usuarios finales, pero en este caso, los estudiantes eran idóneos para las tareas que debían realizar porque están entre la edad de 18 a 30 años. Otro de los problemas externos, son los tiempos de respuesta de los servidores, puesto que estos son gratuitos y dan límite de procesamiento.
- **Validez interna:** se estableció un procedimiento, materiales y el cuestionario para los estudiantes; además el investigador estuvo presente para resolver cualquier duda.
- **Validez constructo:** En este caso se utilizó una selección de tres parámetros (sí, parcial, no) muy utilizados en otros estudios.

Anexo 13. Encuestas a maestros en lengua de señas ecuatoriana

Encuestas a personas inmersas en el LSEC

Proyecto: Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano

Versión: 1.0

Fecha: 18/07/2022

Hoja de control

Proyecto	Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano
Entregable	Pruebas de Aceptación.
Autor	Jonnathan Espinoza
Versión	1.0
Aprobado por	Edison Leonardo Coronel Romero, Mg. Sc.

Registro de cambios

Versión del documento	Descripción	Responsable	Fecha de cambio
1.0	Versión inicial de pruebas de aceptación a personas involucradas en LSEC.	Jonnathan Espinoza	18/07/2022

Índice

Hoja de control	221
Índice	222
1. Introducción.....	223
Objetivo	223
Propósito	223
Hipótesis.....	223
2. Diseño de encuestas	224
3. Ejecución de encuestas.....	225
4. Análisis de resultados.....	232
4.1. Evaluación de la hipótesis H1	233
4.2. Evaluación de la hipótesis H2	233
4.3. Sugerencias al investigador	233
5. Amenazas a la validez de los resultados obtenidos.....	233
6. Anexos	235

1. Introducción

Objetivo

Determinar el nivel de aceptación de la solución informática por medio de una muestra de maestros en lengua de señas ecuatoriana (LSEC) de la Asociación de Personas Sordas Virgen del Cisne.

Propósito

Validar la aplicación web con una muestra de maestros de lengua de señas ecuatoriana (LSEC) mediante la técnica de las encuestas.

Hipótesis

La hipótesis planteada es:

- H1: La aplicación web es fácil de usar e intuitivo con el usuario.
- H2: La aplicación web fomenta el aprendizaje de la lengua de señas ecuatoriana.

2. Diseño de encuestas

A continuación, se presenta el cuestionario, el cual consta de 11 preguntas:

Tabla 83.
Preguntas de las encuestas.

N°	Pregunta
P1	¿La aplicación web al momento de usarla fue?
P2	¿Considera que la aplicación es atractiva e intuitiva?
P3	¿Considera que la aplicación web es fácil de usar para cualquier persona con capacidad de manejar un navegador web?
P4	¿La funcionalidad de la aplicación web, permite el aprendizaje de la lengua de señas?
P5	¿Presentar las lecciones en contenidos cortos (Microlearning), ayuda a la enseñanza de la lengua de señas?
P6	¿El uso de GIFs (imágenes animadas) ayuda en la comprensión de la lengua de señas?
P7	¿Los repasos o prácticas por medio de la selección múltiple en la aplicación web, sirven como un medio de practicar la lengua de señas?
P8	¿Considera que la aplicación web fomenta el aprendizaje de la lengua de señas?
P9	¿La aplicación web garantiza su operabilidad al no presentar fallas?
P10	¿Haría uso cotidiano de la aplicación?
P11	¿Recomendaría a otros usuarios que usen la aplicación?

En la siguiente tabla se muestra las respuestas confirmadas según el tipo de pregunta, lo que permite a los encuestados elegir entre una serie de opciones utilizando diferentes escalas de Likert, rangos numéricos y sugiriendo preguntas al investigador.

Tabla 84.
Respuestas a las preguntas de la encuesta

Pregunta	Respuesta
P1	a. Muy fácil b. Fácil c. Normal d. Difícil e. Muy Difícil
P2, P3, P4, P5, P6, P7, P8, P9, P10, P11	a. Totalmente de acuerdo b. De acuerdo c. Indeciso d. En desacuerdo e. Totalmente en desacuerdo
Sugerencias al investigador	Abierta

3. Ejecución de encuestas

La investigación realizada tiene como finalidad conocer si las aplicaciones web desarrolladas en el trabajo de investigación cumplen con los requisitos de aceptación operativa y de usuario final. Participaron 4 docentes en lengua de señas ecuatoriana de la Asociación de Personas Sordas Virgen del Cisne, de los cuales 3 eran personas con discapacidad auditiva, lo que da mayor validez a los resultados. Las muestras seleccionadas fueron seleccionadas convenientemente de acuerdo a su accesibilidad a los investigadores por métodos no probabilísticos.

Se realizó una demostración de la aplicación web desarrollada; después de probar y evaluar el contenido del software, los encuestados llenaron las encuestas, como se presenta en la Figura 106.



Figura 119. Aplicación de las encuestas.

Además, se solicitó al encuestado, nombre, apellido, número de cédula y sugerencias, con el fin de dar fiabilidad a las respuestas. A continuación, se presentan los resultados obtenidos en las encuestas.

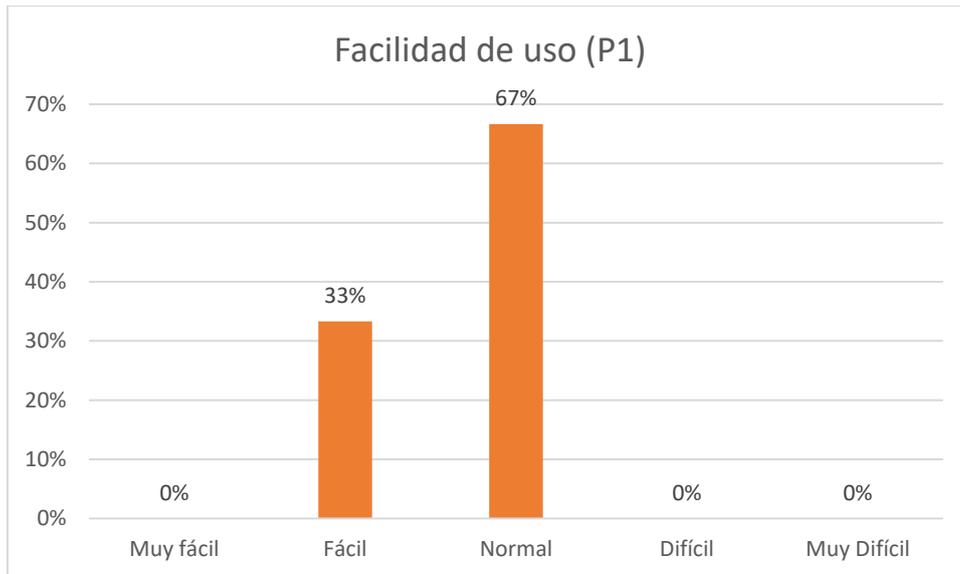


Figura 120. Respuesta a la pregunta 1.

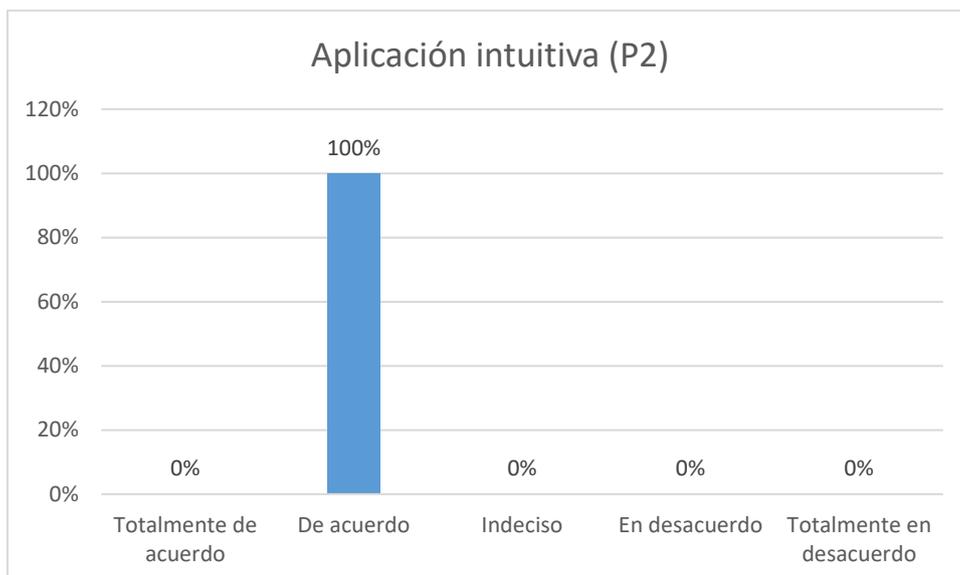


Figura 121. Respuesta a la pregunta 2.

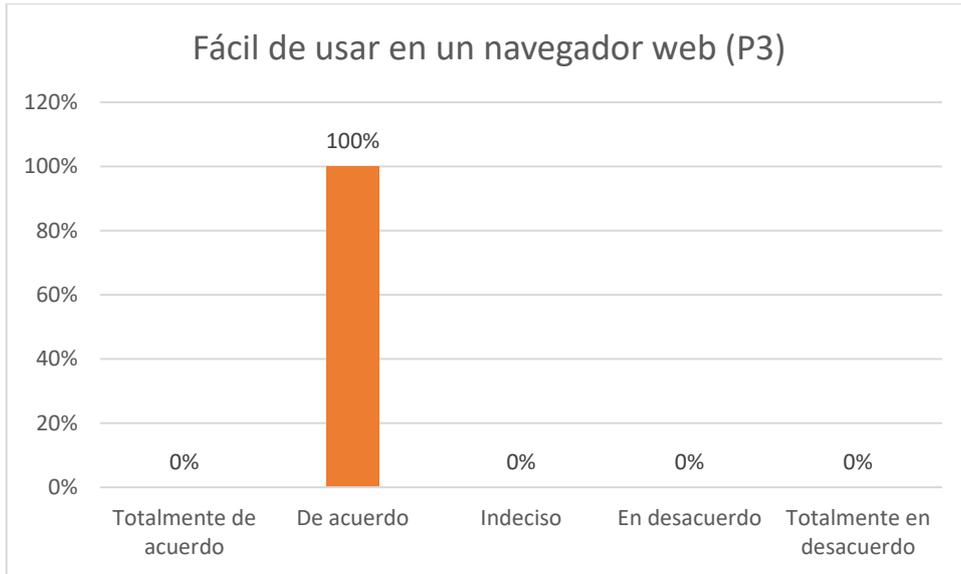


Figura 122. Respuesta a la pregunta 3

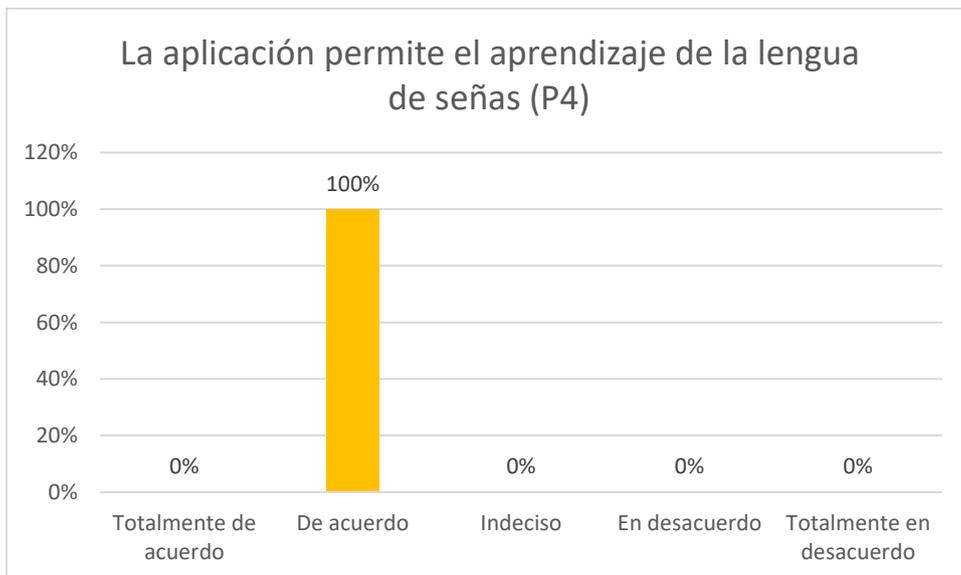


Figura 123. Respuesta a la pregunta 4.

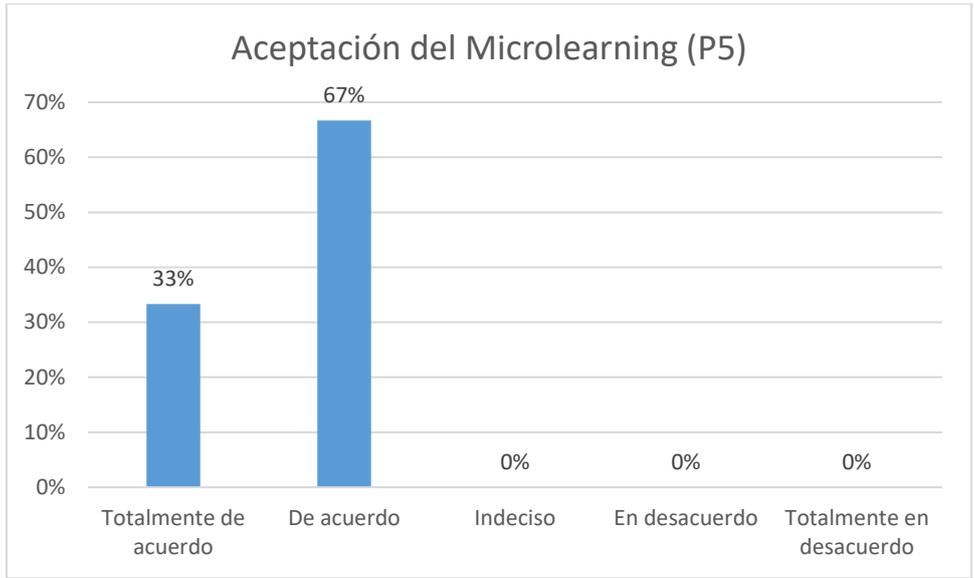


Figura 124. Respuesta a la pregunta 5.

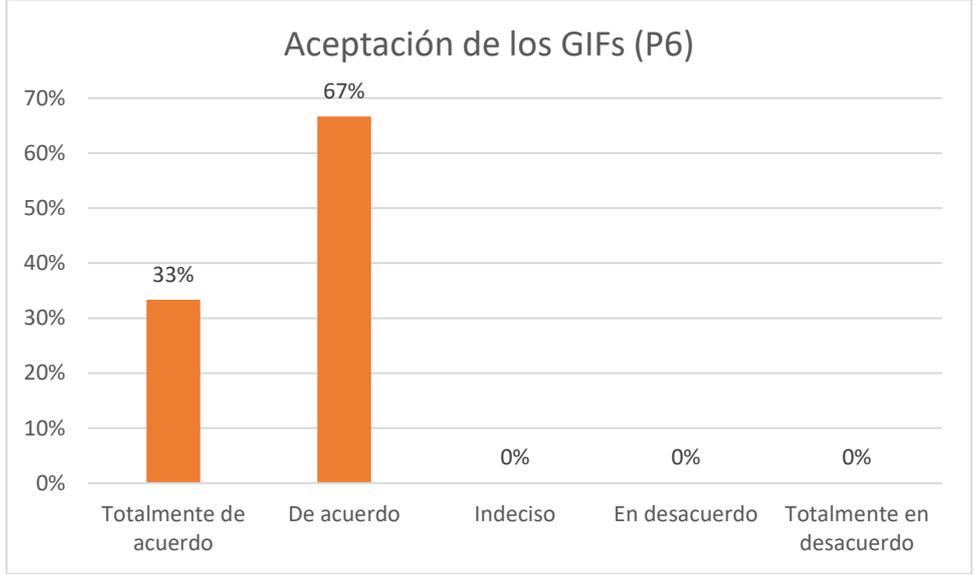


Figura 125. Respuesta a la pregunta 6.

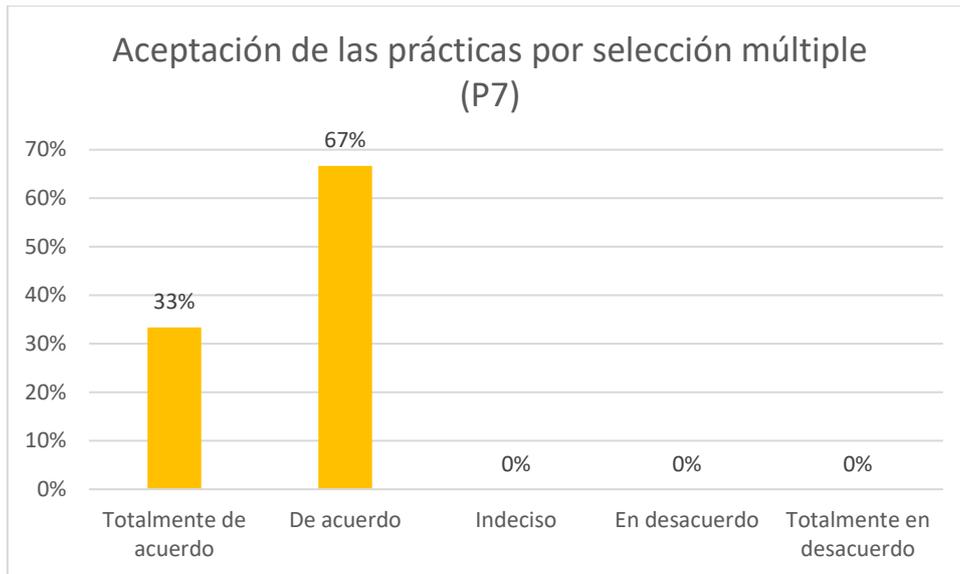


Figura 126. Respuesta a la pregunta 7.

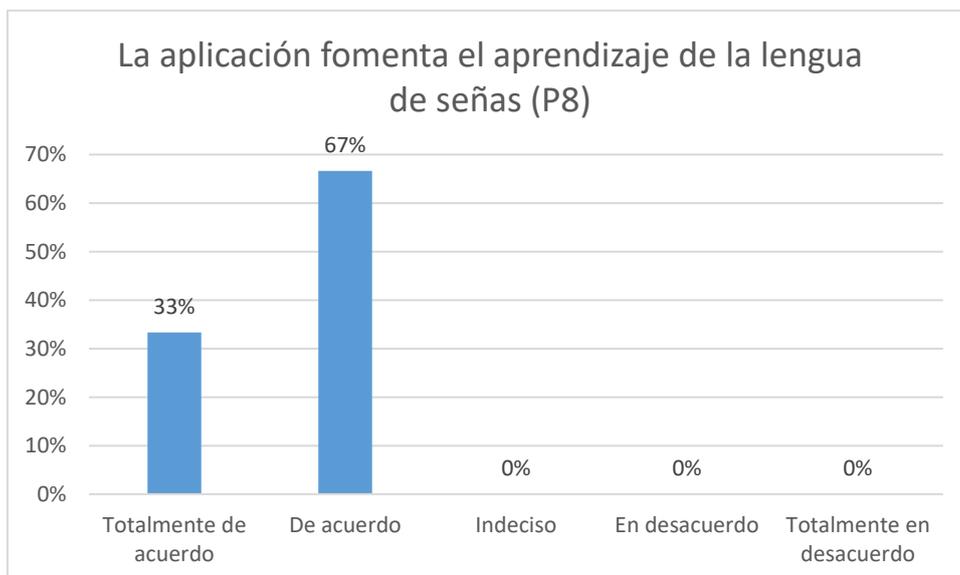


Figura 127. Respuesta a la pregunta 8.

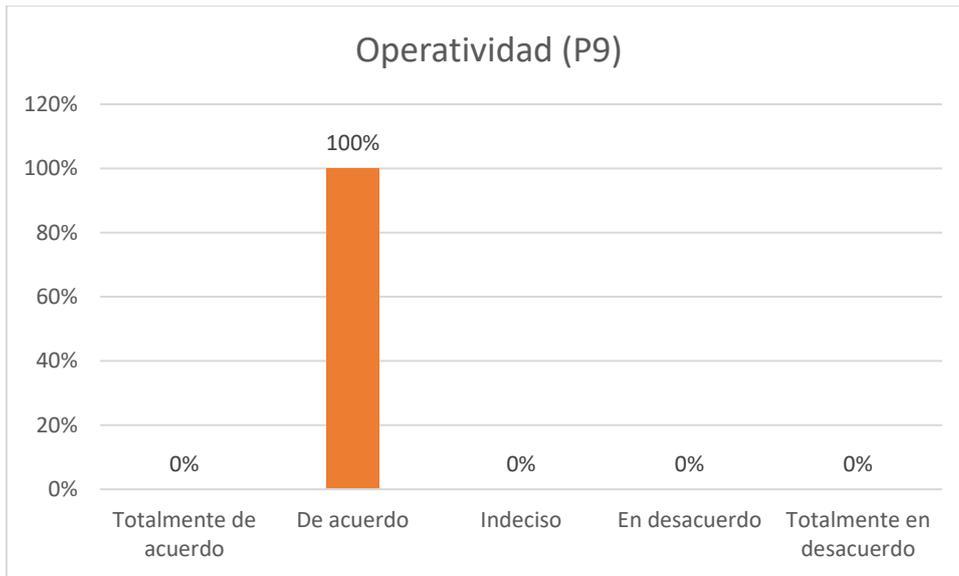


Figura 128. Respuesta a la pregunta 9.

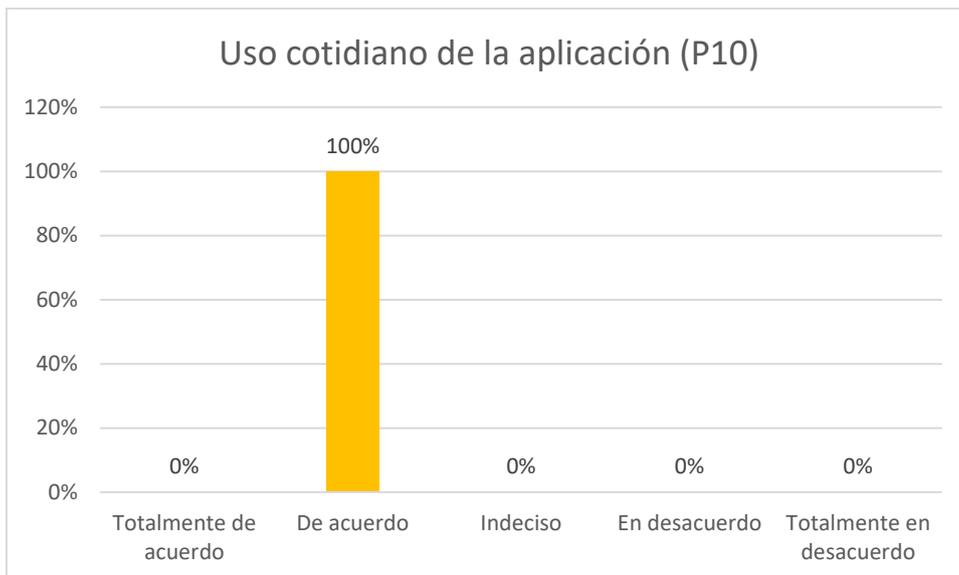


Figura 129. Respuesta a la pregunta 10.

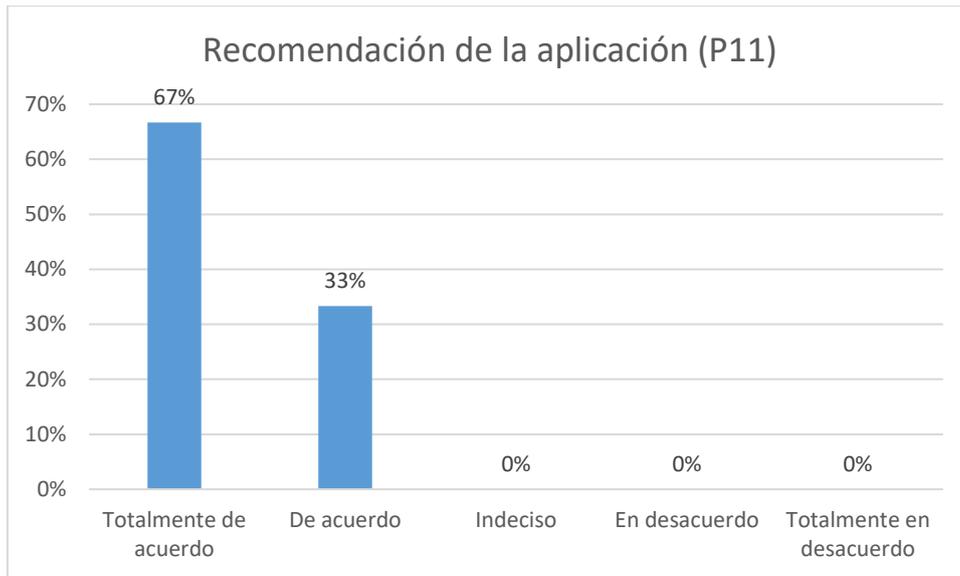


Figura 130. Respuesta a la pregunta 11.

4. Análisis de resultados

La Tabla 82 presenta el resumen de los resultados a las encuestas realizadas.

Tabla 85.
Resumen de las respuestas a las encuestas

No.	Parámetros	Porcentaje (%)				
		Totalmente de acuerdo	De acuerdo	Indeciso	En desacuerdo	Totalmente en desacuerdo
1	¿La aplicación web al momento de usarla fue?	0	33	67	0	0
2	¿Considera que la aplicación es atractiva e intuitiva?	0	100	0	0	0
3	¿Considera que la aplicación web es fácil de usar para cualquier persona con capacidad de manejar un navegador web?	0	100	0	0	0
4	¿La funcionalidad de la aplicación web, permite el aprendizaje de la lengua de señas?	0	100	0	0	0
5	¿Presentar las lecciones en contenidos cortos (Microlearning), ayuda a la enseñanza de la lengua de señas?	33	67	0	0	0
6	¿El uso de GIFs (imágenes animadas) ayuda en la comprensión de la lengua de señas?	33	67	0	0	0
7	¿Los repaos o prácticas por medio de la selección múltiple en la aplicación web, sirven como un medio de practicar la lengua de señas?	33	67	0	0	0
8	¿Considera que la aplicación web fomenta el aprendizaje de la lengua de señas?	33	67	0	0	0
9	¿La aplicación web garantiza su operabilidad al no presentar fallas?	0	100	0	0	0
10	¿Haría uso cotidiano de la aplicación?	0	100	0	0	0
11	¿Recomendaría a otros usuarios que usen la aplicación?	67	33	0	0	0
PROMEDIO		18,09	75,82	6,09	0	0

a. Evaluación de la hipótesis H1

Con las preguntas P1, P2, P3 y P9, se determina que, el 33% de los expertos están considerando que la aplicación web es Fácil de usar y el 67% no tuvo complicaciones en usar la aplicación. El 100% de los expertos están De Acuerdo en que la aplicación es atractiva e intuitiva, cualquier persona con capacidad de manejar un navegador web podría utilizar sin problema la aplicación web y que la aplicación opera sin fallas. Como resultado de este análisis, se acepta la hipótesis H1.

b. Evaluación de la hipótesis H2

Con las preguntas P4, P5, P6, P7, P8, P10 y P11, se determina que, el 100% de los expertos indicaron que están De Acuerdo en que la aplicación web permite el aprendizaje de la lengua de señas ecuatoriana y la ocuparían cotidianamente. El 33% indicaron que están Totalmente de Acuerdo que el uso de la metodología del microlearning ayudaría en la enseñanza de la lengua de señas ecuatoriana y el 67% están De Acuerdo; estos mismos porcentajes se repiten en la aceptación de los GIFs como medio visual para presentar el patrón de la lengua de señas, en la implementación de prácticas de retroalimentación por medio de selección múltiple, y consideran que la aplicación fomenta el aprendizaje de la lengua de señas ecuatoriana. El 67% están Totalmente de Acuerdo en recomendar la aplicación web y el 33% están De Acuerdo. Como resultado a este análisis se puede aceptar la hipótesis H2.

c. Sugerencias al investigador

Adicionalmente se agregó una pregunta para obtener sugerencias por parte de los encuestados, entre las cuales se encuentran las siguientes:

- La aplicación está bien, solo que sería necesario de que se les enseñara a utilizar la parte administrativa.
- El contenido básico que muestra la aplicación se considera suficiente para iniciar en el aprendizaje de la lengua de señas, no obstante, se puede ir generando más contenido.
- Como primera muestra de la aplicación y su contenido, es fácil de entender y se recomendaría mejorar la calidad de las imágenes de las señas.

5. Amenazas a la validez de los resultados obtenidos

En esta sección se describen los aspectos que pueden haber afectado la validez del experimento y los métodos que se intentaron mitigar.

- **Validez externa:** En relación con la generalización de los resultados, es posible que se haya visto afectado por una muestra pequeña de encuestados, pero en este caso,

los docentes de LSEC que participaron eran idóneos para evaluar el contenido de la aplicación y dar una crítica sobre la enseñanza mediante el micro-learning. Otro de los problemas externos, son los tiempos de respuesta de los servidores, puestos que estos son gratuitos y dan límite de procesamiento.

- **Validez interna:** se estableció un procedimiento, materiales y el cuestionario para los encuestados; además el investigador estuvo presente para resolver cualquier duda.
- **Validez constructo:** En este caso se utilizó las escalas de Likert.

6. Anexos



Figura 131. Profesores de LSEC e investigador del TT

Nombre	Teléfono	Cédula	Firma
Eyvin Abad.	0992882817	110560731-9	
Hugo Ch Ramírez	0983752821	910025106-7	
Luis A. Quichimbo	0995780149	0701952883	

Figura 132. Firmas de los expertos

Anexo 14. Certificado de la Asociación de Personas Sordas Virgen del Cisne



Loja, 30 de agosto de 2022

Darwin Alfredo Jumbo Quichimbo.

Presidente de la Asociación de Personas Sordas “Virgen del Cisne”

Certifica:

Que la aplicación web perteneciente al trabajo de titulación denominado “Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano”; desarrollado por el Sr. Jonnathan Damián Espinoza Erráez con cédula 0107022006 egresado de la carrera de Ingeniería en Sistemas de la Facultad de Energía, las Industrias y los Recursos Naturales no Renovables de la Universidad Nacional de Loja, ha sido evaluada en los aspectos relacionados a la funcionalidad y usabilidad, por lo que se da fe de la aceptación satisfactoria de las funcionalidades que compone la aplicación y de la ayuda que brinda al proceso de enseñanza de la lengua de señas.

Es todo cuanto puedo certificar en honor a la verdad

Darwin Alfredo Jumbo Quichimbo
CI: 1900400639

**PRESIDENTE DE LA ASOCIACIÓN DE
PERSONAS SORDAS “VIRGEN DEL CISNE”**

Manual para el manejo de la aplicación web Aprende Lengua de Señas Ecuatoriana

Versión 2.0

Elaborado por:

Jonnathan Damián Espinoza Erráez

Agosto 2022

Loja – Ecuador

Historial de versiones

Revisión	Fecha	Responsable	Descripción de la modificación
1.0	25/08/2021	Jonnathan Damián Espinoza Erráez	Documento Inicial
2.0	01/08/2022	Jonnathan Damián Espinoza Erráez	Se rediseño la interfaz de usuario, se cambió el framework de css Bootstrap por Tailwind css.

Índice de contenido

Historial de versiones	238
Índice de contenido	239
Introducción.....	240
Objetivo	240
Alcance.....	240
Aplicación de Aprender Lengua de Señas Ecuatoriana	240
Prerrequisitos	240
Instrucciones	241
1. Creación de Cuenta de usuario	242
2. Inicio de sesión en la plataforma.....	243
3. Iniciar una lección	244
4. Iniciar una práctica.....	247
5. Ver tus avances de lecciones	249

Introducción

Objetivo

El objetivo del presente documento es proporcionar a los usuarios una guía para el correcto uso de la aplicación Aprende Lengua de Señas Ecuatoriana (LSEC).

Alcance

El presente documento y sus indicaciones van dirigidas a las personas que desean aprender Lengua de Señas Ecuatoriana (LSEC), con el propósito de poner a disposición una aplicación web interactiva que ayude al aprendizaje de la LSEC, a través del micro-learning.

Aplicación de Aprender Lengua de Señas Ecuatoriana

Aplicación web que permite el aprendizaje de la Lengua de Señas Ecuatoriana de una manera interactiva a través de imágenes animadas, con el concepto del micro-learning.

Prerrequisitos

- Conexión a internet

Instrucciones

El presente manual de usuario indica paso a paso el procedimiento para el correcto uso de la aplicación web. Esta aplicación funciona a través de un computador con sistema Windows, GNU Linux o MacOS que tengan instalado un navegador de Internet, se sugiere el uso de navegadores como: Google Chrome, Brave, Opera, Firefox o Microsoft Edge.

Ingresa al siguiente enlace: <https://learn-lsec.herokuapp.com/> donde se mostrará la siguiente ventana principal.

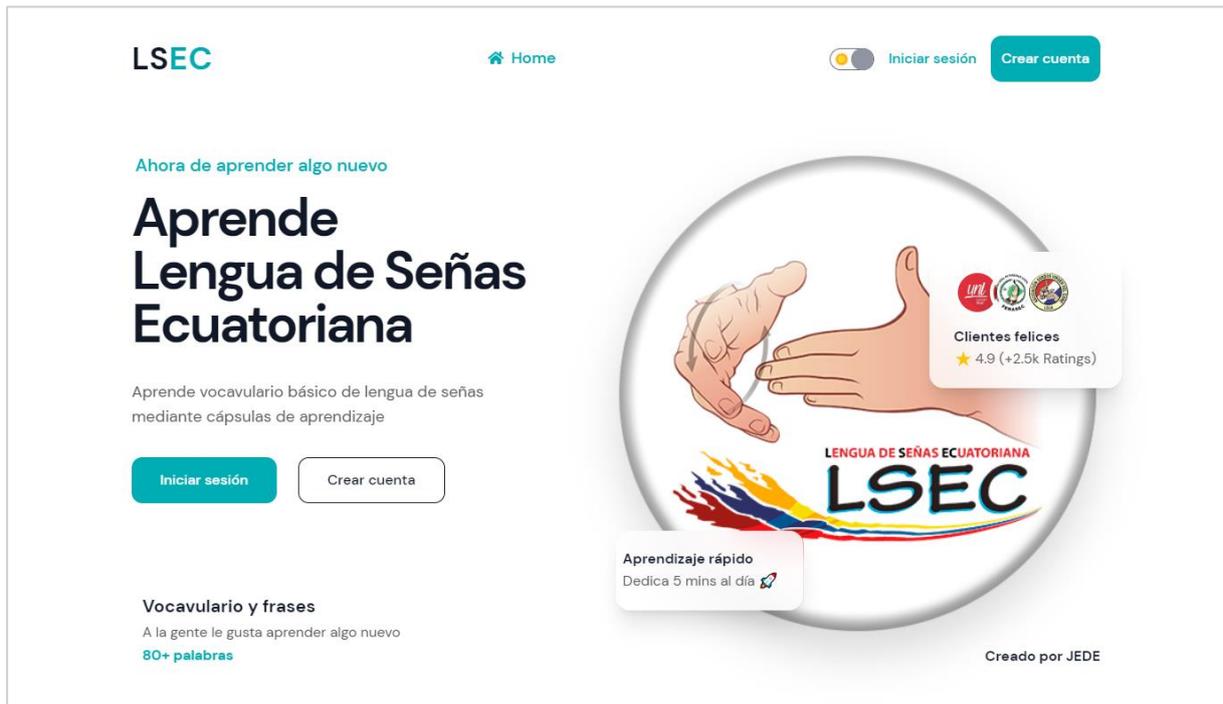


Imagen 1. Página principal de la aplicación

1. Creación de Cuenta de usuario

Debe de ingresar a la pantalla de Crear Cuenta que contiene el formulario registrarse y se mostrará la siguiente pantalla:

The image shows a user registration interface for LSEC (Lengua de Señas Ecuatoriana). On the left, an illustration depicts a woman at a laptop, with a large smartphone overlay showing a registration form with fields for 'username@gmail.com' and a masked password. To the right, the actual registration form is displayed with the title 'A registrarnos!' and the subtitle 'Regístrate para aprender Lengua de Señas Ecuatoriana.' The form contains three input fields: 'Nombre' (Name) with the value 'jonnathan', 'Correo electrónico' (Email) with the value 'jonnathan123@un.edu.ec', and 'Contraseña' (Password) with masked characters. A teal 'Regístrate' button is at the bottom of the form, with a link '¿Tienes una cuenta? Inicia sesión' below it. The LSEC logo is in the top right, and a copyright notice '© 2021 JEDE. All rights reserved.' is at the bottom right.

Imagen 2. Pantalla de Creación de Cuenta

En el formulario se deberá hacer:

ETIQUETA	DESCRIPCIÓN
1	Ingresar el nombre con el que desea que aparezca en la aplicación.
2	Ingresar un correo electrónico válido.
3	Ingresar una contraseña.
4	Botón para registrarse.

2. Inicio de sesión en la plataforma

Para iniciar sesión se debe de ingresar a la página de iniciar sesión y se mostrará la siguiente pantalla:

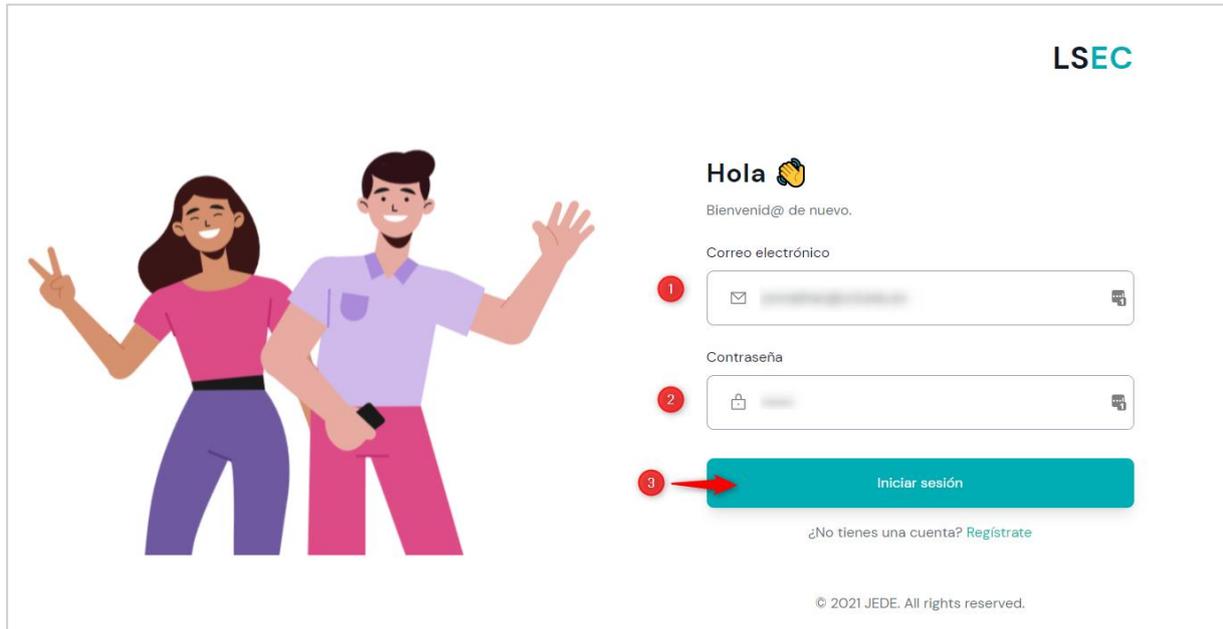


Imagen 3. Pantalla de Iniciar Sesión

En el formulario se deberá hacer:

ETIQUETA	DESCRIPCIÓN
1	Ingresar el correo electrónico registrado.
2	Ingresar la contraseña.
3	Botón para Iniciar Sesión.

Nota: De no contar el usuario con una cuenta, se procede a registrar un nuevo usuario en crear cuenta.

3. Iniciar una lección

Una vez iniciado sesión, se presentará la siguiente pantalla:

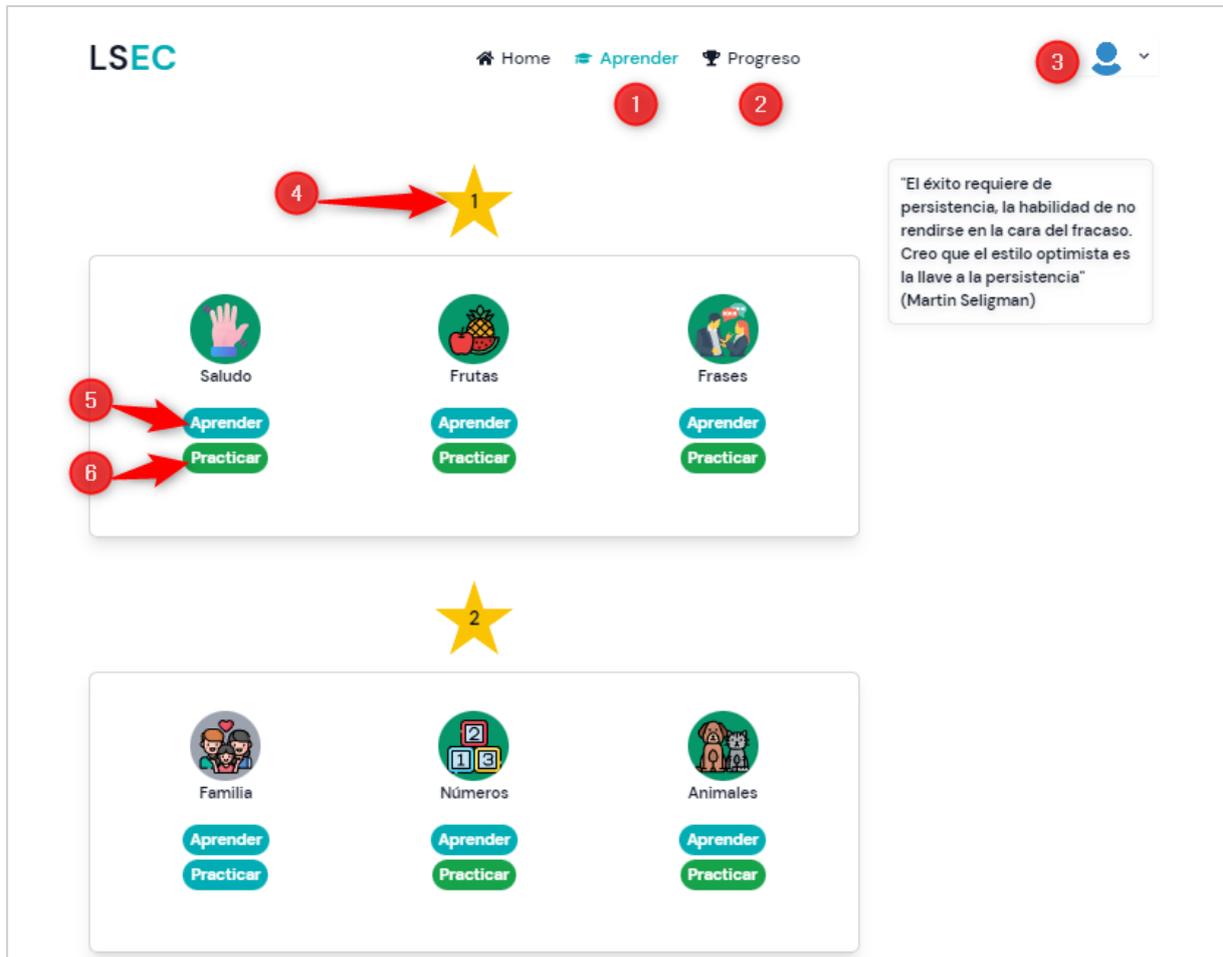


Imagen 4. Pantalla de Aprendizaje

En la pantalla se puede diferenciar los siguientes elementos:

ETIQUETA	DESCRIPCIÓN
1	Botón para acceder a todas las lecciones.
2	Botón para acceder a los avances del usuario.
3	Nombre del usuario y opciones de usuario.
4	Número del Módulo.
5	Lección para aprender lengua de señas.
6	Botón para practicar lengua de señas de acuerdo a la lección correspondiente.

Para dar inicio al aprendizaje, debe dar clic sobre el icono o el nombre de la lección de la etiqueta **5** en la **Imagen 4**. Una vez ingresado a la lección, se presentará Cápsulas de Aprendizaje, como se indica a continuación:

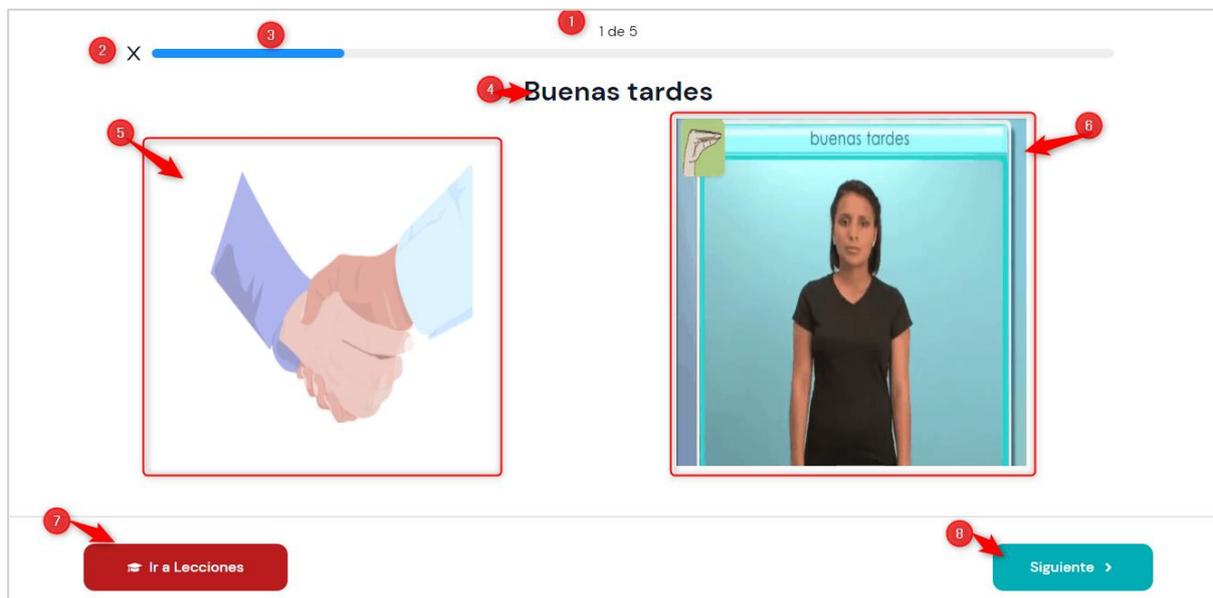


Imagen 5. Cápsulas de Aprendizaje

En la pantalla se identifica:

ETIQUETA	DESCRIPCIÓN
1	Número de cápsulas de aprendizajes.
2	Botón para salir a la página principal.
3	Barra de progreso.
4	Título de la cápsula de aprendizaje.
5	Imagen representativa a la lengua de señas presentada.
6	Imagen dinámica (GIF) de la lengua de señas para aprender.
7	Botón para volver a la sección de lecciones. Cuando siga con la siguiente cápsula, este botón se cambia para regresar a la cápsula anterior.
8	Botón para ir a la siguiente cápsula. Cuando esté en la cápsula final, este botón cambia para terminar la lección correspondiente.

Cuando haya terminado una lección, el fondo del icono se vuelve de color verde, como se muestra en la siguiente imagen:

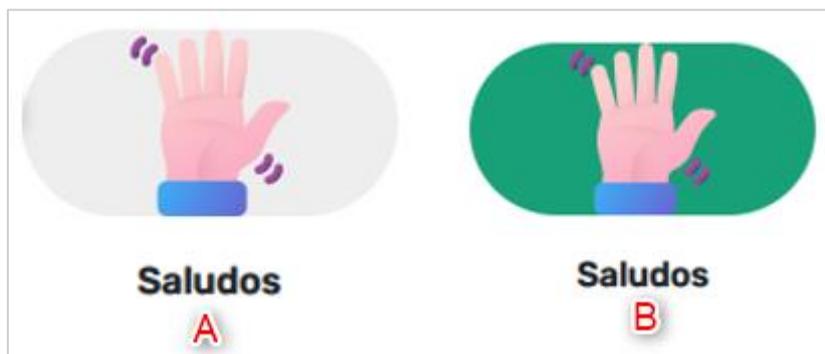


Imagen 6. Icono de lecciones. A) Lección sin comenzar. B) Lección terminada

4. Iniciar una práctica

Una vez iniciado sesión, se presentará la siguiente pantalla:

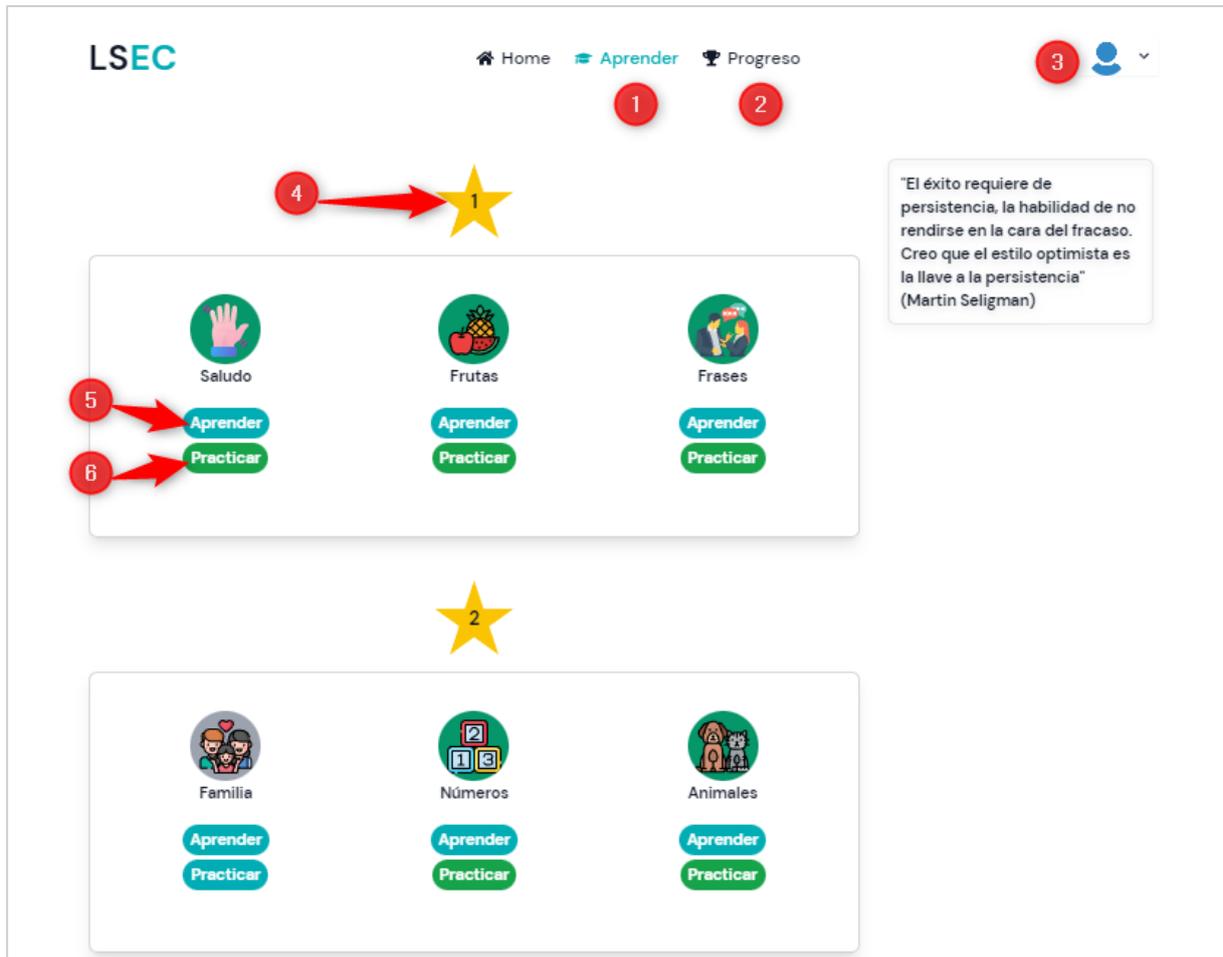


Imagen 7. Pantalla de Aprendizaje

En la pantalla se puede diferenciar los siguientes elementos:

ETIQUETA	DESCRIPCIÓN
1	Botón para acceder a todas las lecciones.
2	Botón para acceder a los avances del usuario.
3	Nombre del usuario y opciones de usuario.
4	Número del Módulo.
5	Lección para aprender lengua de señas.
6	Botón para practicar lengua de señas de acuerdo a la lección correspondiente.

Para practicar y poner a prueba los conocimientos, debe dar clic sobre el botón “Practicar” que se encuentra debajo de cada lección de la etiqueta **6** en la **Imagen 7**. Una vez ingresado a la práctica, se presentará las Tarjetas de Aprendizaje, como se indica a continuación:

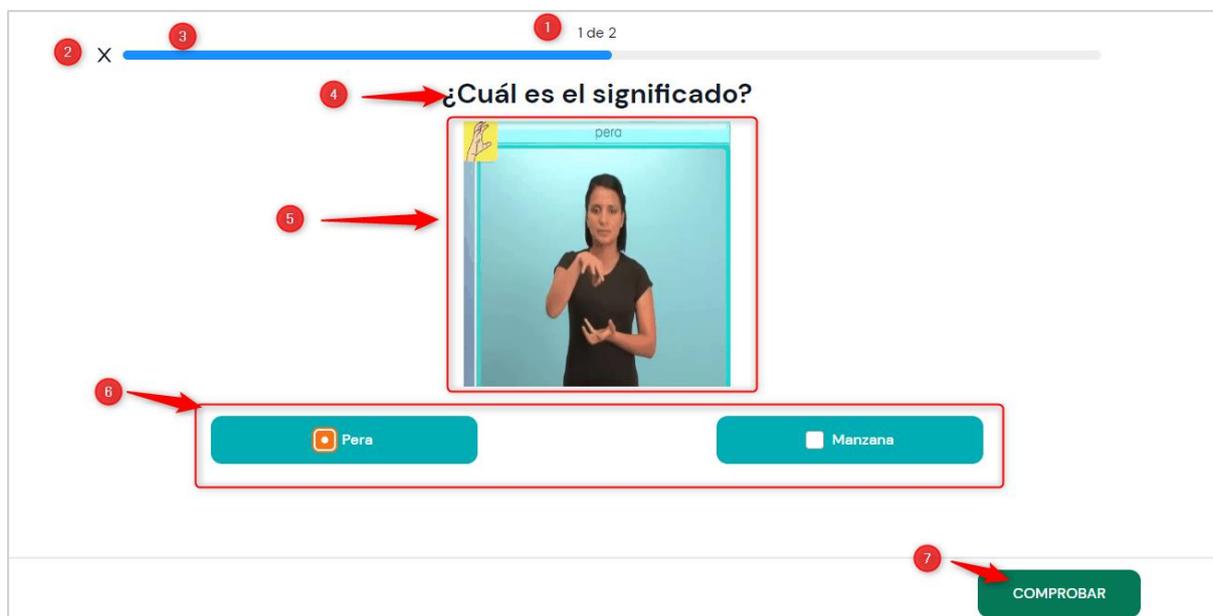


Imagen 8. Tarjeta de Aprendizaje

En la pantalla se identifica:

ETIQUETA	DESCRIPCIÓN
1	Número de Tarjetas de Aprendizaje.
2	Botón para salir a la página principal.
3	Barra de progreso.
4	Pregunta sobre la actividad que debe hacer el usuario.
5	Imagen dinámica (GIF) de la lengua de señas que el usuario debe de adivinar.
6	Opciones que el usuario debe de seleccionar.
7	Botón para comprobar la respuesta correcta. Cuando esté en la cápsula final, este botón cambia para terminar la práctica correspondiente.

5. Ver tus avances de lecciones

Una vez iniciado sesión, se presentará el siguiente menú:

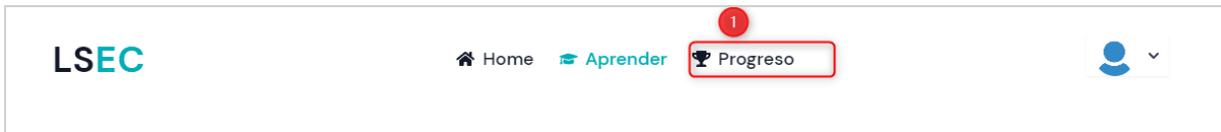


Imagen 9. Menú de la aplicación

Debe de seleccionar la etiqueta **1**, y se presentará la pantalla de progreso de las lecciones que haya terminado:



Imagen 10. Pantalla de avance de lecciones

Manual para el manejo de la aplicación web Aprende Lengua de Señas Ecuatoriana

Administrador

Versión 2.0

Elaborado por:

Jonnathan Damián Espinoza Erráez

Agosto 2022

Loja – Ecuador

Historial de versiones

Revisión	Fecha	Responsable	Descripción de la modificación
1.0	27/08/2021	Jonnathan Damián Espinoza Erráez	Documento Inicial
2.0	01/08/2022	Jonnathan Damián Espinoza Erráez	Se rediseño la interfaz de usuario administrador, se cambió el framework de css Bootstrap por Tailwind css.

Índice de contenido

Historial de versiones	251
Índice de contenido	252
Introducción.....	254
Objetivo	254
Alcance.....	254
Aplicación de Aprender Lengua de Señas Ecuatoriana	254
Prerrequisitos	254
Instrucciones.....	255
1. Creación de Cuenta de usuario	256
2. Inicio de sesión en la plataforma como administrador.....	257
3. Gestión de módulos.....	258
a. Crear un nuevo módulo.....	258
b. Listar todos los módulos creados	259
c. Actualizar datos del módulo	259
d. Eliminar un módulo	260
4. Gestionar lecciones	261
a. Crear nueva lección	261
b. Listar todas las lecciones creadas.....	262
c. Actualizar datos una lección.....	262
d. Eliminar una lección	263
e. Visualizar una lección.....	264
5. Gestionar Cápsula de Aprendizaje.....	265
a. Crear nueva cápsula de aprendizaje	265
b. Listar todas las cápsulas de aprendizaje	266
c. Actualizar datos de una cápsula de aprendizaje.....	266
d. Eliminar una cápsula de aprendizaje.....	267
e. Visualizar información cápsula de aprendizaje.....	268
6. Gestionar Tarjetas de Aprendizaje.....	269

a.	Crear una tarjeta de aprendizaje	269
b.	Listar todas las tarjetas de aprendizaje	270
c.	Actualizar datos de una tarjeta de aprendizaje	270
d.	Eliminar una tarjeta de aprendizaje	271
e.	Visualizar contenido de una tarjeta de aprendizaje	272
7.	Listar usuarios	273

Introducción

Objetivo

El objetivo del presente documento es proporcionar al usuario administrador una guía para el correcto uso de la aplicación Aprende Lengua de Señas Ecuatoriana (LSEC) en la gestión de contenido.

Alcance

El presente documento y sus indicaciones van dirigidas a los administradores de la aplicación web, con el propósito de poner a disposición una aplicación web interactiva que ayude a la enseñanza de la LSEC, a través del micro-learning

Aplicación de Aprender Lengua de Señas Ecuatoriana

Aplicación web que permite el aprendizaje de la Lengua de Señas Ecuatoriana de una manera interactiva a través de imágenes animadas, con el concepto del micro-learning.

Prerrequisitos

- Conexión a internet

Instrucciones

El presente manual de usuario indica paso a paso el procedimiento para el correcto uso de la aplicación web. Esta aplicación funciona a través de un computador con sistema Windows, GNU Linux o MacOS que tengan instalado un navegador de Internet, se sugiere el uso de navegadores como: Google Chrome, Brave, Opera, Firefox o Microsoft Edge.

Ingresa al siguiente enlace: <https://learn-lsec.herokuapp.com/> donde se mostrará la siguiente ventana principal.

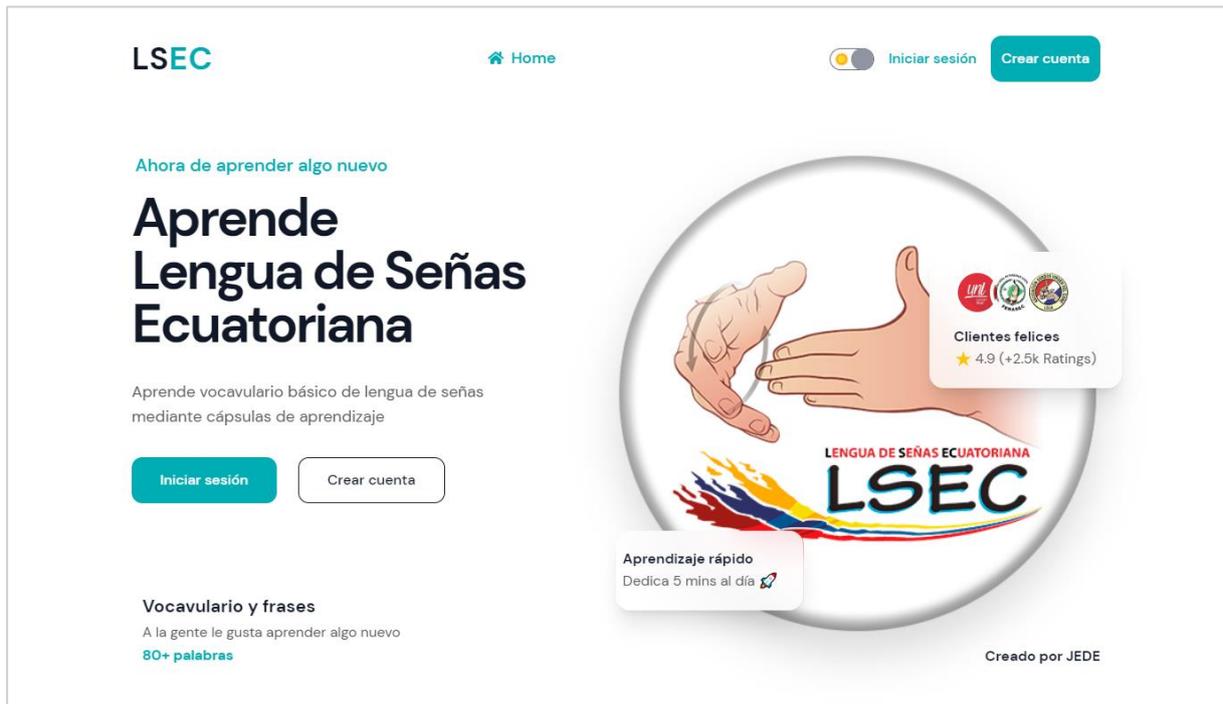


Imagen 11. Página principal de la aplicación

6. Creación de Cuenta de usuario

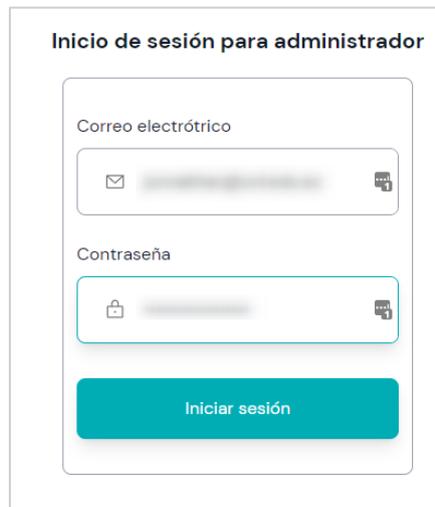
La creación de cuentas de administrador, se debe de enviar los datos de la **Imagen 12** en formato JSON al siguiente enlace <http://<ip-del-servidor>/api/auth/signup>

```
"name": "tu_name",
"email": "tu_email@gmail.com",
"password": "my_contraseña",
"roles": [
  "student",
  "admin",
  "moderator"
]
```

Imagen 12. Datos para la creación de usuario administrador

7. Inicio de sesión en la plataforma como administrador

Ingresar al siguiente enlace: <https://<ip-del-servidor>/admin/signin>, donde se mostrará el formulario de iniciar sesión:



Inicio de sesión para administrador

Correo electrónico

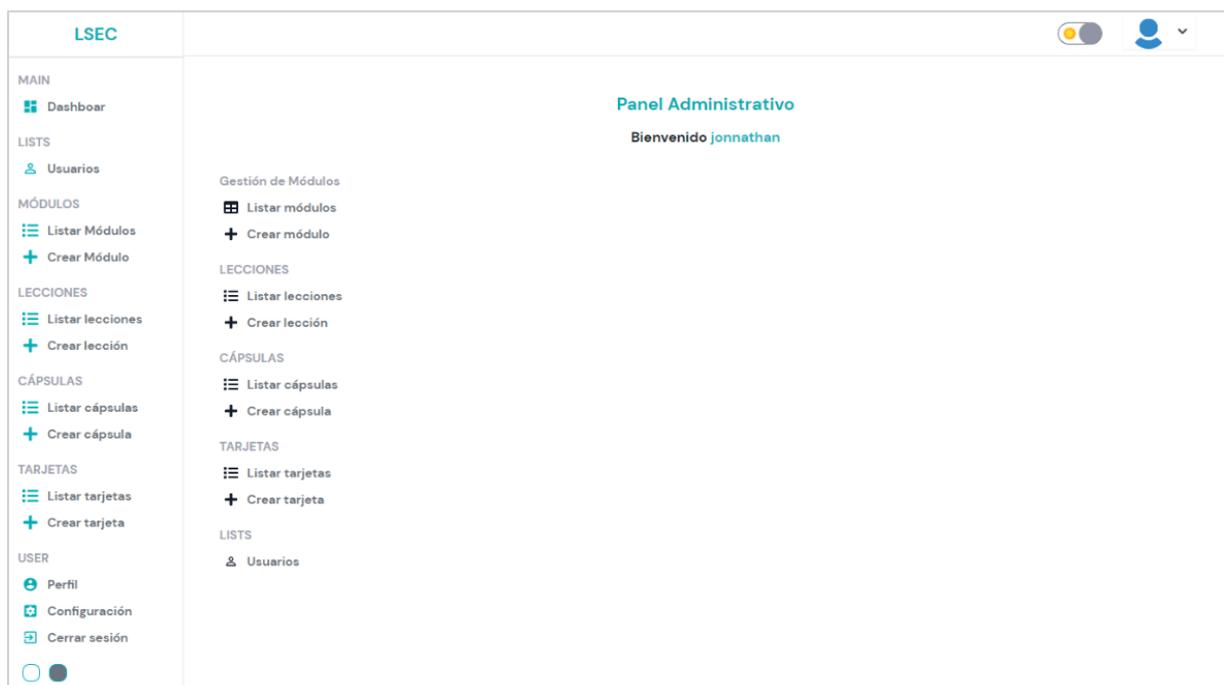
Contraseña

Iniciar sesión

Imagen 13. Pantalla de Iniciar Sesión

En el formulario deberá de ingresar el correo electrónico y su contraseña.

Una vez iniciado sesión, se presentará el dashboard de administrador o puede ingresar directamente al siguiente enlace: <https://<ip-del-servidor>/admin/dashboard>, recuerde que debe de iniciar sesión. A continuación, se presenta el dashboard:



LSEC

Panel Administrativo

Bienvenido jonnathan

MAIN

- Dashboard

LISTS

- Usuarios

MÓDULOS

- Listar Módulos
- Crear Módulo

LECCIONES

- Listar lecciones
- Crear lección

CÁPSULAS

- Listar cápsulas
- Crear cápsula

TARJETAS

- Listar tarjetas
- Crear tarjeta

USER

- Perfil
- Configuración
- Cerrar sesión

Gestión de Módulos

- Listar módulos
- Crear módulo

LECCIONES

- Listar lecciones
- Crear lección

CÁPSULAS

- Listar cápsulas
- Crear cápsula

TARJETAS

- Listar tarjetas
- Crear tarjeta

LISTS

- Usuarios

Imagen 14. Dashboard de administrador

8. Gestión de módulos

En el dashboard se puede acceder a gestionar los módulos a través de las etiquetas de la **Imagen 4**.

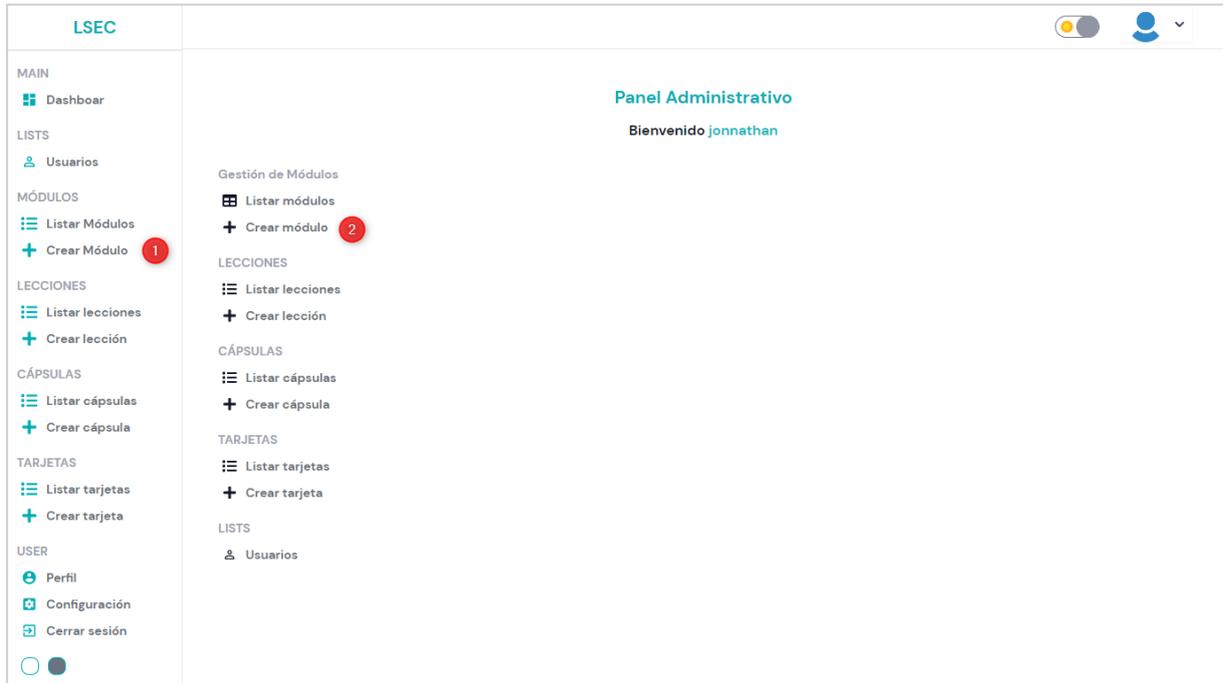


Imagen 15. Ingresar a gestionar módulos

a. Crear un nuevo módulo

En el dashboard de administrador, debe de ingresar a “Crear módulo” señalado en las etiquetas de la **Imagen 4** y se presentará el formulario para crear un nuevo módulo:

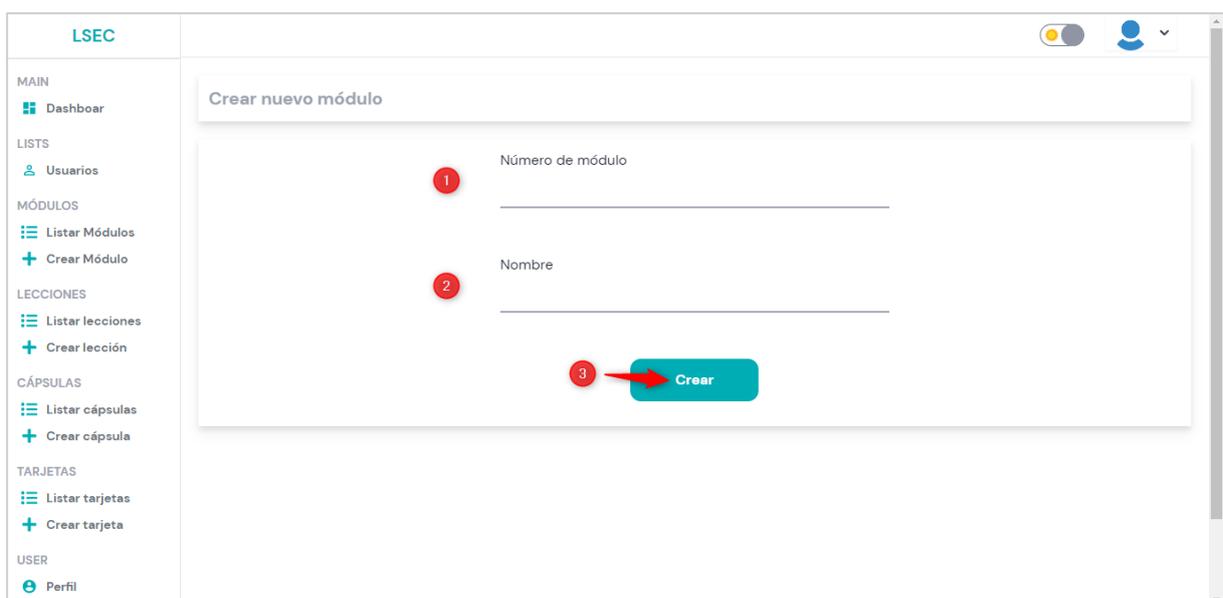


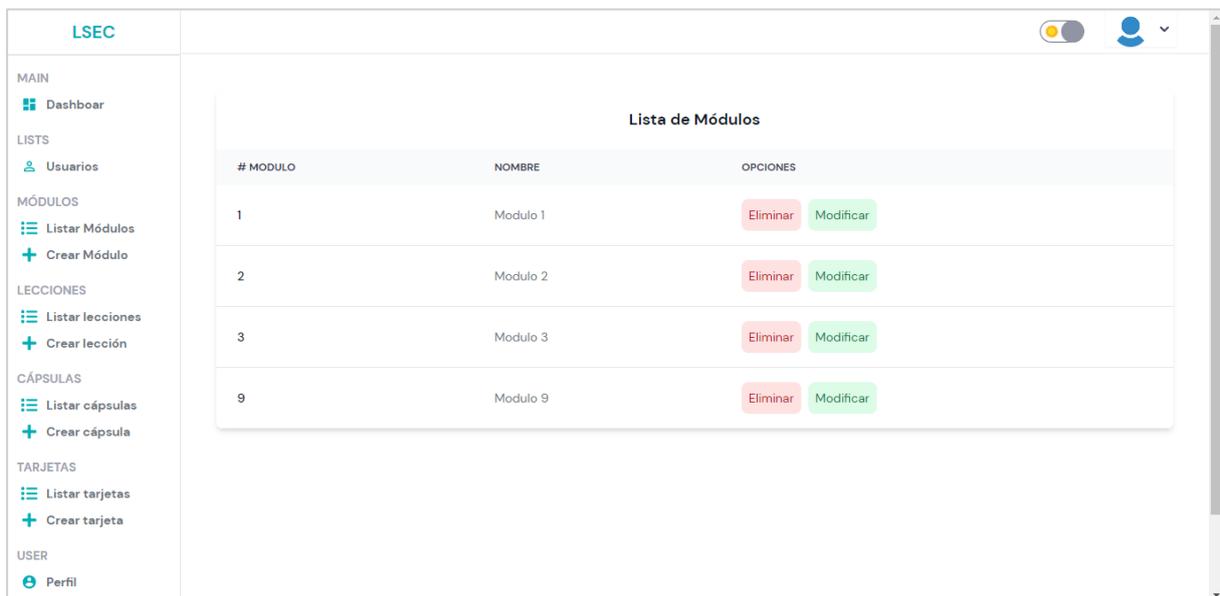
Imagen 16. Formulario para crear un nuevo módulo

En el formulario podrá hacer:

ETIQUETA	DESCRIPCIÓN
1	Ingresar el número del módulo.
2	Ingresar el nombre del módulo.
3	Botón de crear módulo.

b. Listar todos los módulos creados

En el dashboard de administrador, debe de ingresar a “Listar módulo” señalado en las etiquetas de la **Imagen 4** y se presentará una lista de todos los módulos creados:



The screenshot shows the LSEC administrator dashboard. On the left is a sidebar menu with categories: MAIN (Dashboard), LISTS (Usuarios), MÓDULOS (Listar Módulos, Crear Módulo), LECCIONES (Listar lecciones, Crear lección), CÁPSULAS (Listar cápsulas, Crear cápsula), TARJETAS (Listar tarjetas, Crear tarjeta), and USER (Perfil). The main content area is titled 'Lista de Módulos' and contains a table with the following data:

# MODULO	NOMBRE	OPCIONES
1	Modulo 1	Eliminar Modificar
2	Modulo 2	Eliminar Modificar
3	Modulo 3	Eliminar Modificar
9	Modulo 9	Eliminar Modificar

Imagen 17. Listar todos los módulos creados

c. Actualizar datos del módulo

Primero debe de listar todos los módulos creados, elegir un módulo y seleccionar la opción de “Modificar”.

Una vez seleccionado, se presentará el siguiente formulario:

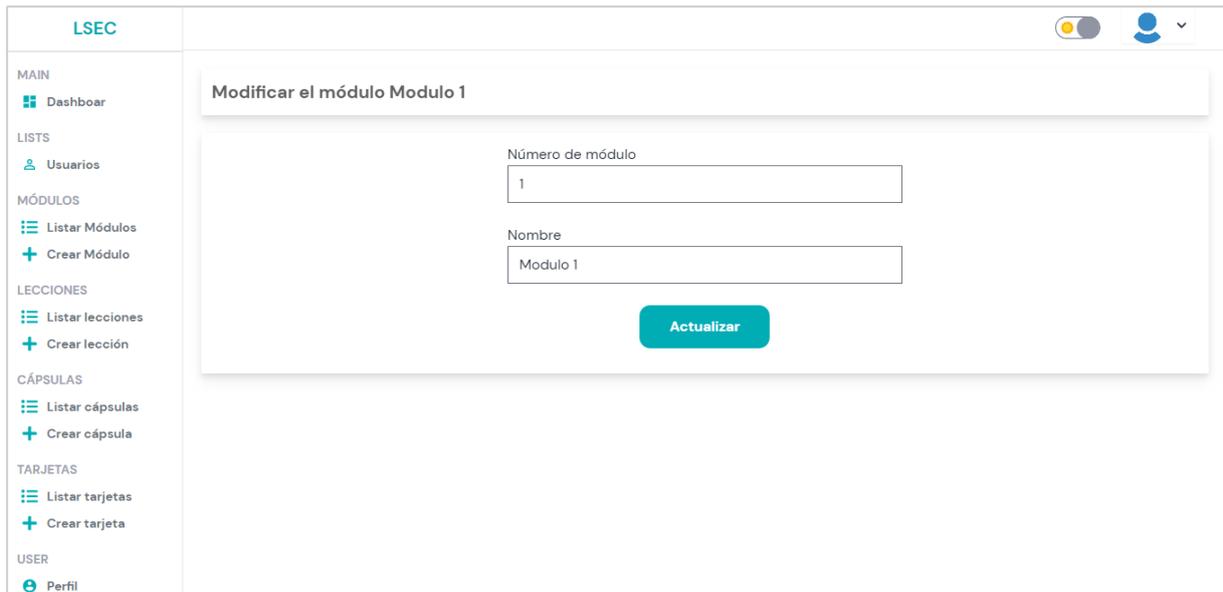


Imagen 18. Actualizar los datos del módulo.

En el formulario se podrá editar los datos como: número de módulo y el nombre del módulo.

d. Eliminar un módulo

Primero debe de listar todos los módulos creados, elegir un módulo y seleccionar la opción de “Eliminar”.

Una vez seleccionado, se presentará la siguiente ventana de confirmación:

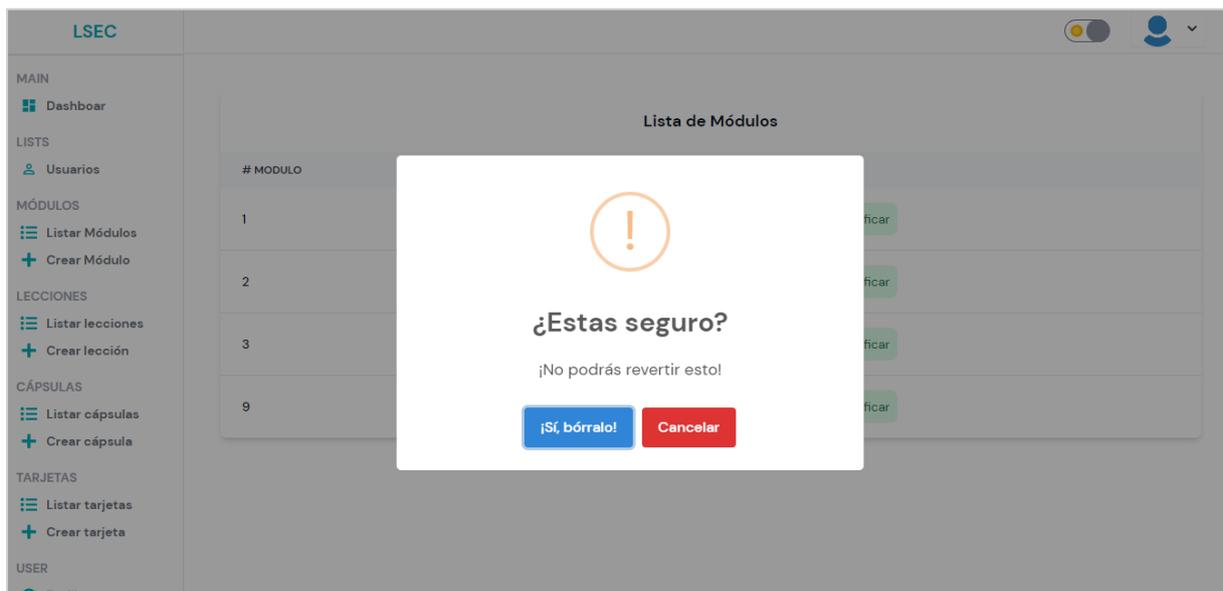


Imagen 19. Panel de confirmación para eliminar un módulo

9. Gestionar lecciones

En el dashboard se puede acceder a gestionar las lecciones a través de las etiquetas de la **Imagen 7**.

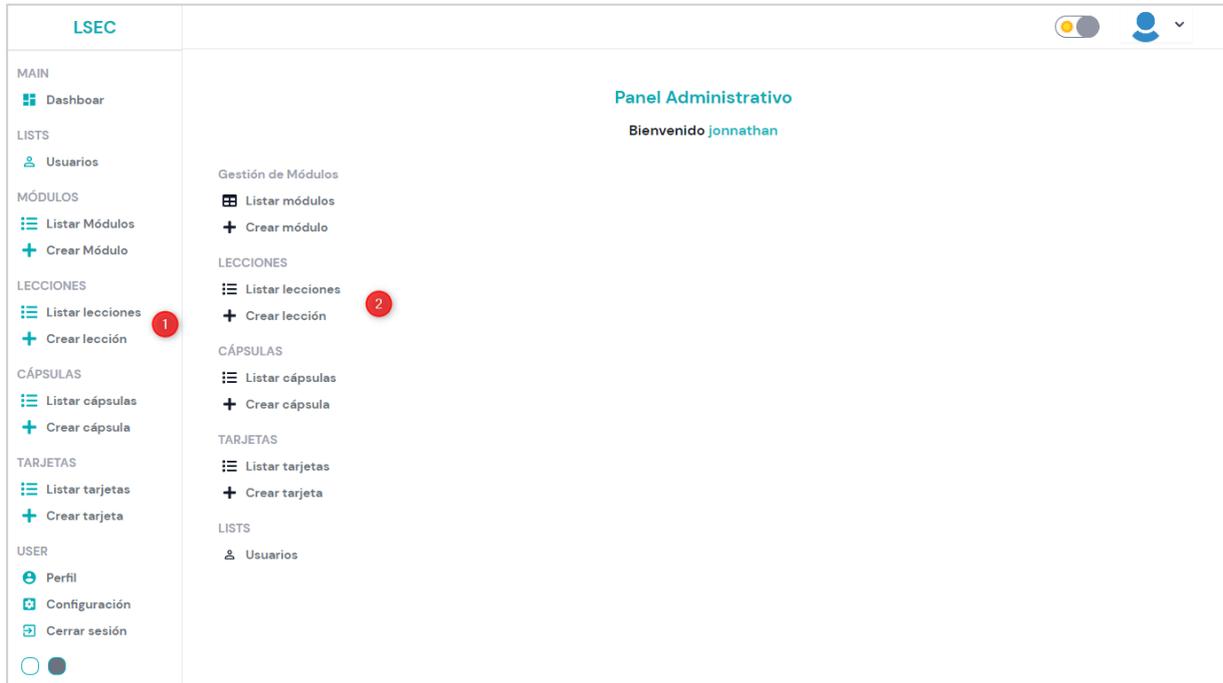


Imagen 20. Ingresar a gestionar lecciones

a. Crear nueva lección

En el dashboard de administrador, debe de ingresar a “Crear Lección” señalado en las etiquetas de la Imagen 9 y se presentará el formulario para crear una nueva lección:

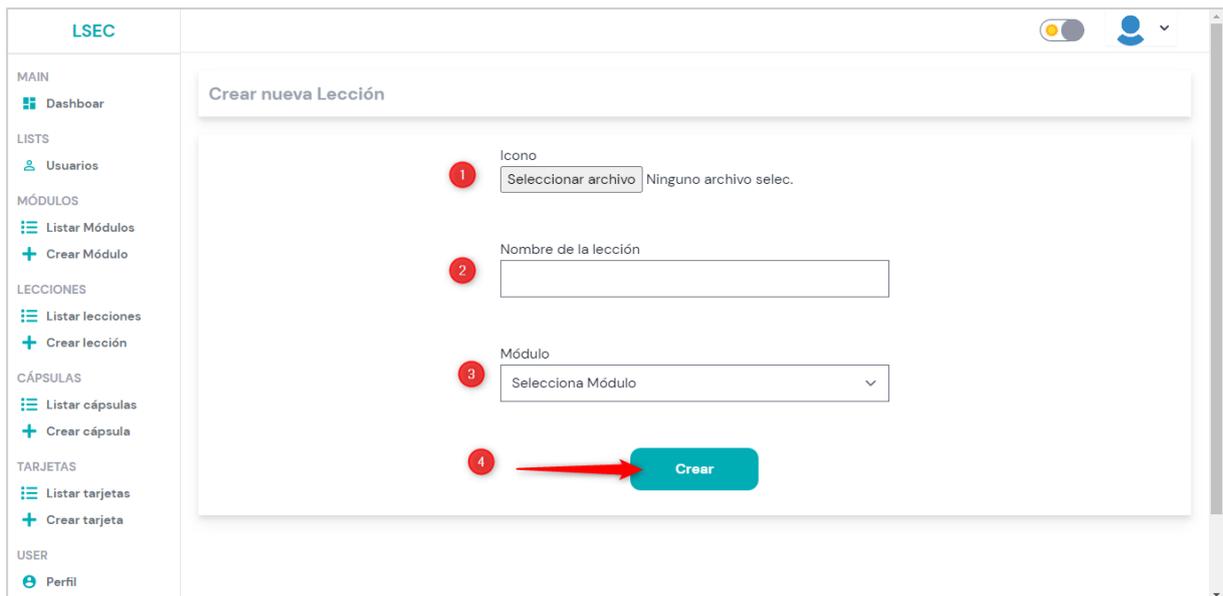


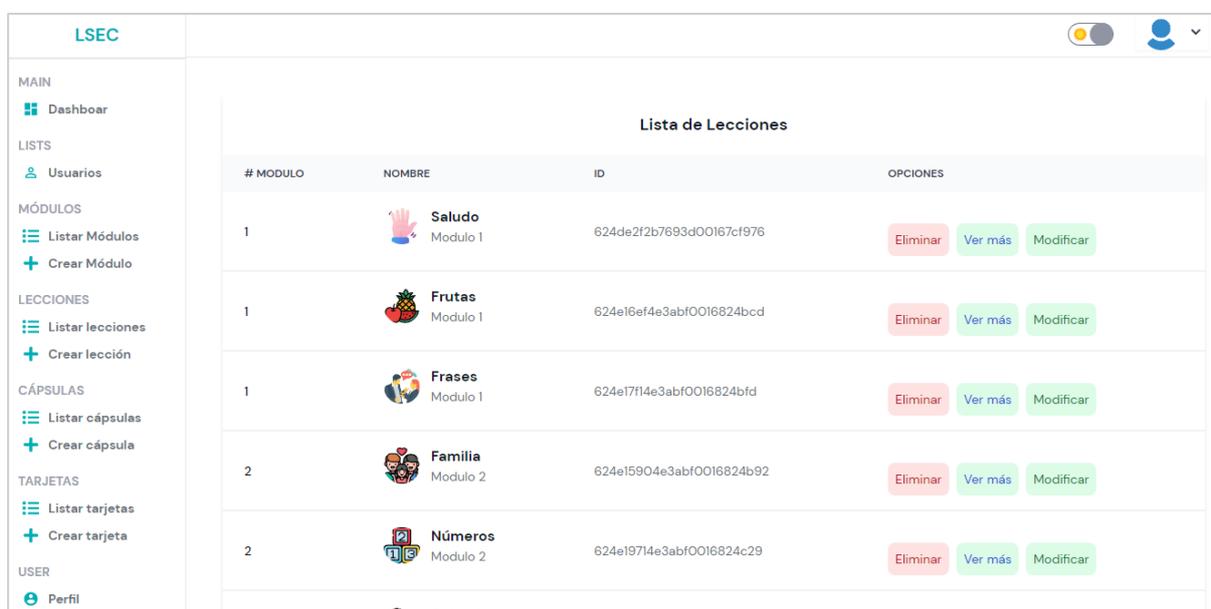
Imagen 21. Formulario para crear nueva lección

En el formulario podrá hacer:

ETIQUETA	DESCRIPCIÓN
1	Ingresar un icono.
2	Ingresar el nombre de la lección.
3	Asociar a un módulo.
4	Botón para crear una nueva lección.

b. Listar todas las lecciones creadas

En el dashboard de administrador, debe de ingresar a “Listar lecciones” señalado en las etiquetas de la **Imagen 7** y se presentará una lista de todas las lecciones creadas:



# MODULO	NOMBRE	ID	OPCIONES
1	 Saludo Modulo 1	624de2f2b7693d00167cf976	Eliminar Ver más Modificar
1	 Frutas Modulo 1	624e16ef4e3abf0016824bcd	Eliminar Ver más Modificar
1	 Frases Modulo 1	624e17f14e3abf0016824bfd	Eliminar Ver más Modificar
2	 Familia Modulo 2	624e15904e3abf0016824b92	Eliminar Ver más Modificar
2	 Números Modulo 2	624e19714e3abf0016824c29	Eliminar Ver más Modificar

Imagen 22. Listar todas las lecciones creadas

c. Actualizar datos una lección

Primero debe de listar todas las lecciones creadas, elegir una lección y seleccionar la opción de “Modificar”.

Una vez seleccionado, se presentará el siguiente formulario:

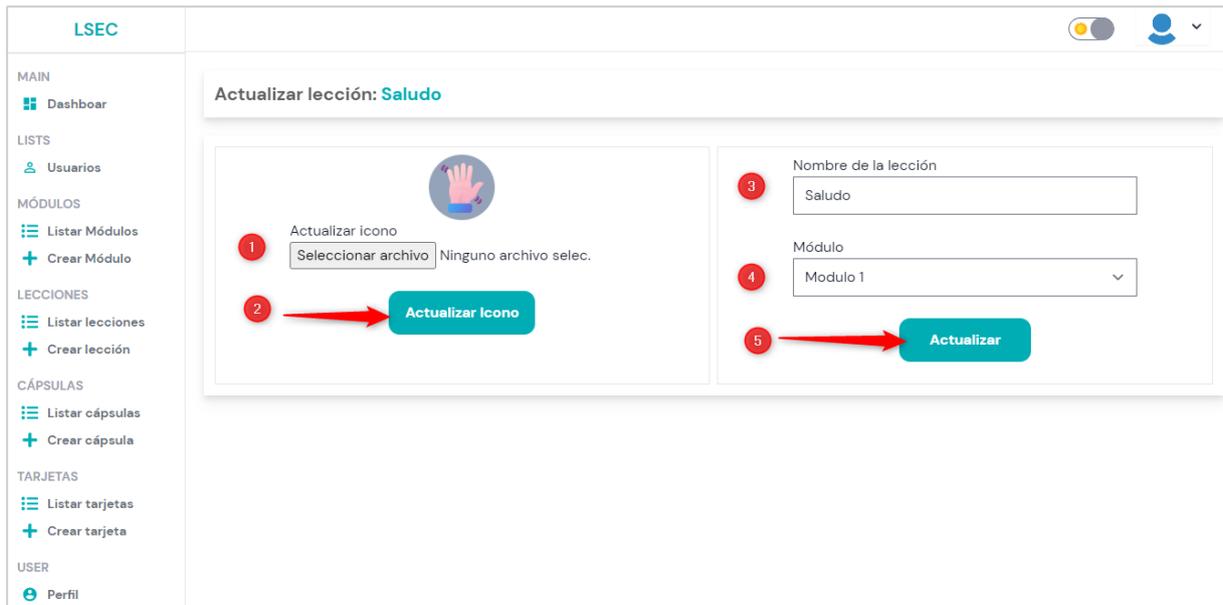


Imagen 23. Actualizar los datos una lección

Existen dos formularios, en el primer formulario podrá actualizar el icono de la lección y en el segundo formulario podrá actualizar: nombre de la lección y el módulo al que pertenece.

d. Eliminar una lección

Primero debe de listar todas las lecciones creadas, elegir una lección y seleccionar la opción de “Eliminar”.

Una vez seleccionado, se presentará la siguiente ventana de confirmación:

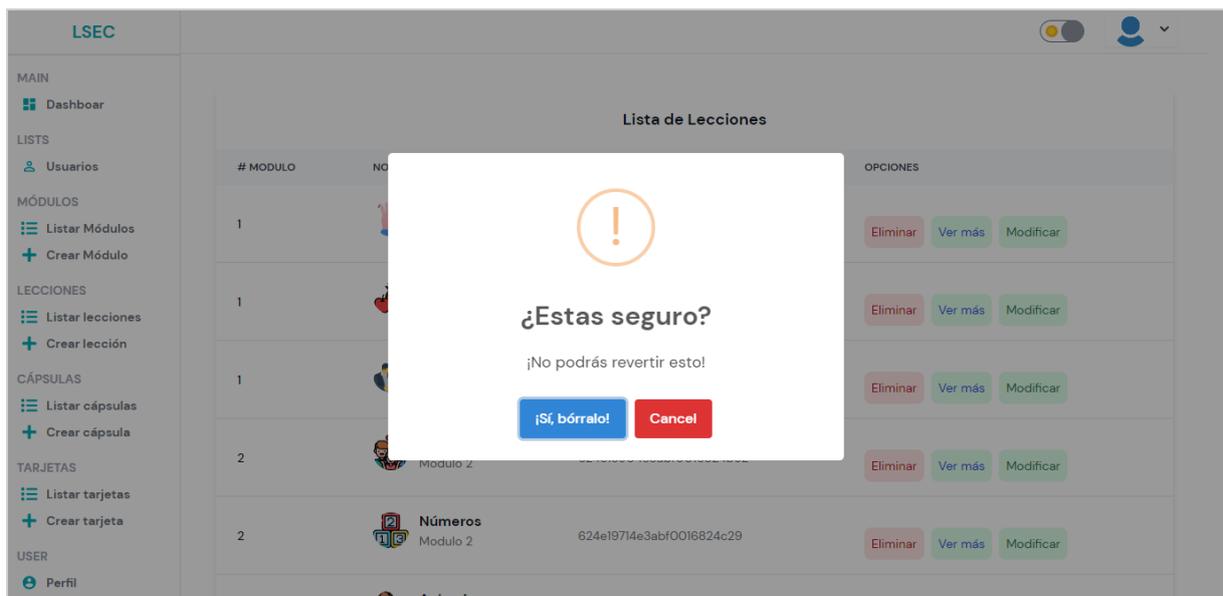


Imagen 24. Panel de confirmación para eliminar una lección

e. Visualizar una lección

Primero debe de listar todas las lecciones creadas, elegir una lección y seleccionar la opción de “Ver más”.

Una vez seleccionado, se presentará la siguiente pantalla con toda la información de la lección:

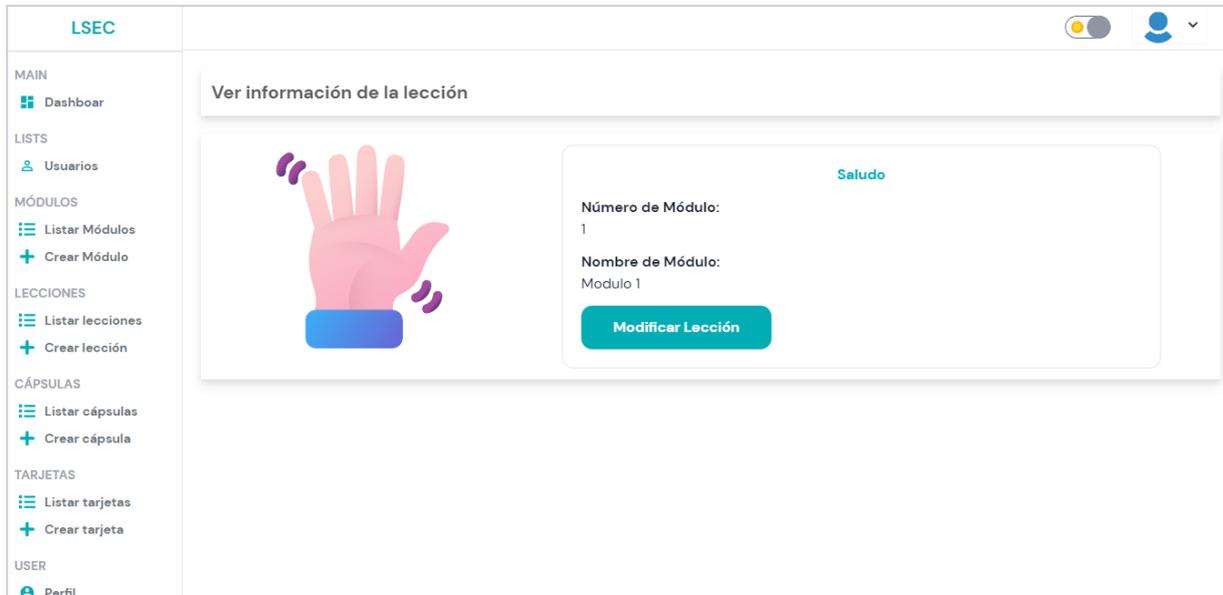


Imagen 25. Visualizar información de una lección

10. Gestionar Cápsula de Aprendizaje

En el dashboard se puede acceder a gestionar las cápsulas de aprendizaje a través de las etiquetas de la **Imagen 26**.

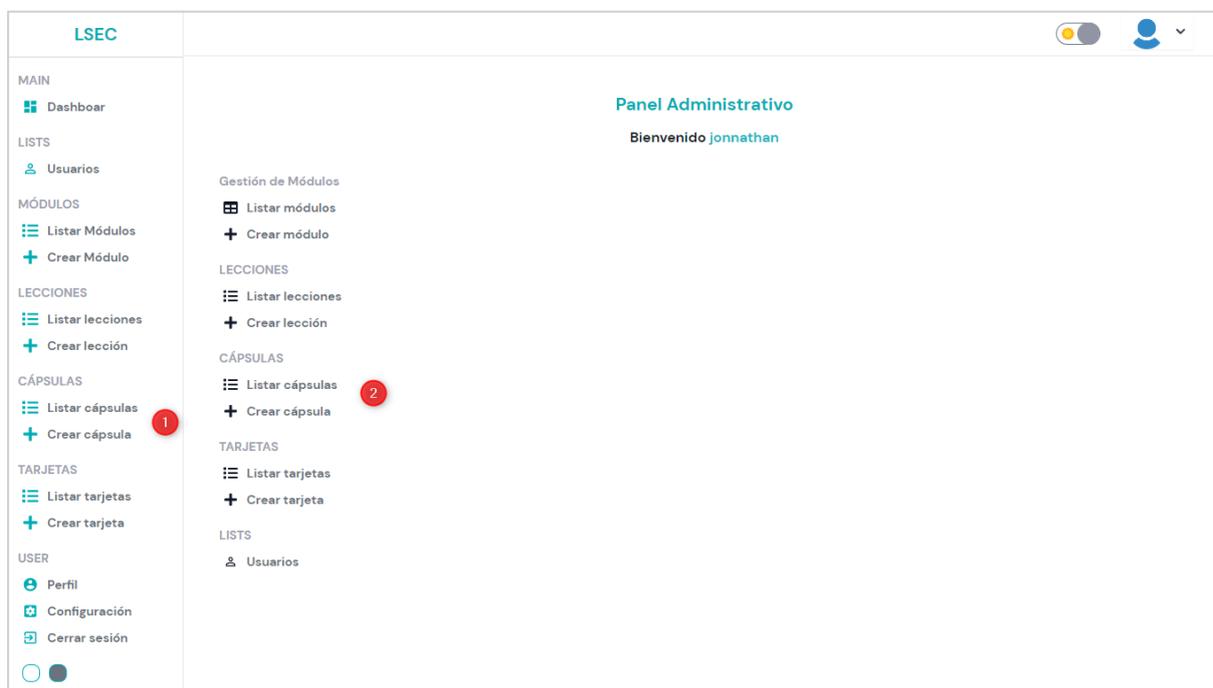


Imagen 26. Ingresar a gestionar cápsula de aprendizaje

a. Crear nueva cápsula de aprendizaje

En el dashboard de administrador, debe de ingresar a “Crear Cápsula” señalado en las etiquetas de la **Imagen 26** y se presentará el formulario para crear una cápsula de aprendizaje:

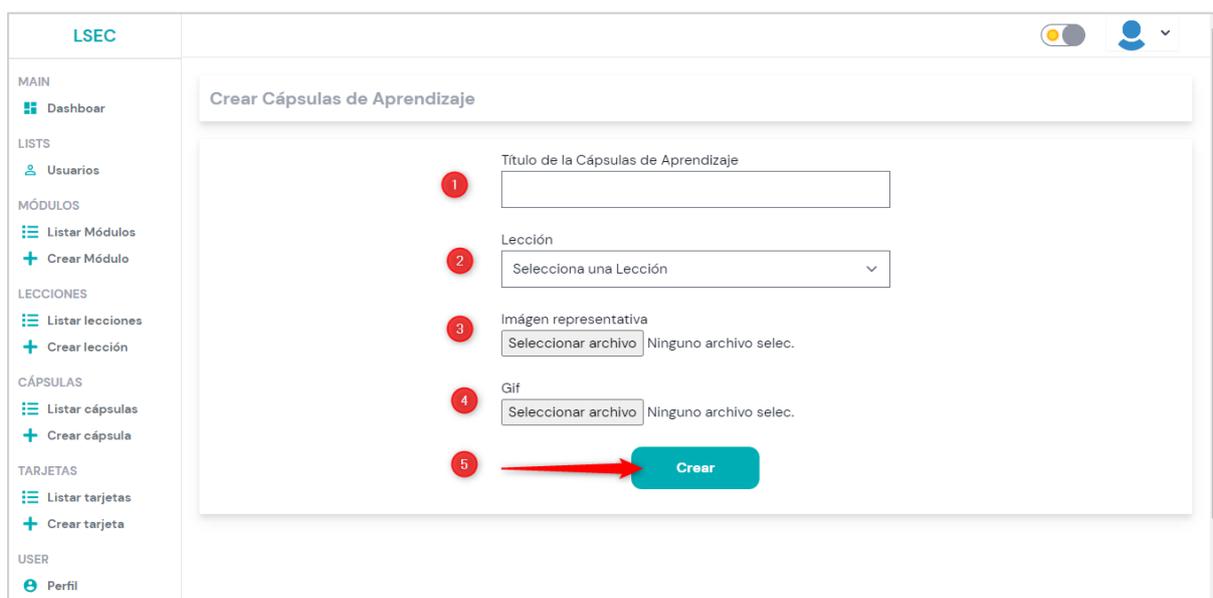


Imagen 27. Formulario para crear una cápsula de aprendizaje

En el formulario podrá hacer:

ETIQUETA	DESCRIPCIÓN
1	Ingresar el título de la cápsula de aprendizaje.
2	Seleccionar una lección.
3	Ingresar una imagen.
4	Ingresar un gif (imagen animada).
5	Botón para crear una cápsula de aprendizaje.

b. Listar todas las cápsulas de aprendizaje

En el dashboard de administrador, debe de ingresar a “Listar Cápsulas” señalado en las etiquetas de la **Imagen 26** y se presentará una lista de todas las cápsulas de aprendizaje creadas:

ID	NOMBRE	LECCIÓN	OPCIONES
624e1bed4e3abf0016824c91	Abeja Animales	Animales	Eliminar Ver más Modificar
624e16554e3abf0016824bb1	Abuela Familia	Familia	Eliminar Ver más Modificar
624de485b7693d00167cf9ac	Adios Saludo	Saludo	Eliminar Ver más Modificar
624de3ddb7693d00167cf999	Buenas Noches Saludo	Saludo	Eliminar Ver más Modificar
624de340b7693d00167cf982	Buenas tardes Saludo	Saludo	Eliminar Ver más Modificar
	Buenos días		

Imagen 28. Listar todas las cápsulas de aprendizaje creadas

c. Actualizar datos de una cápsula de aprendizaje

Primero debe de listar todas las cápsulas, elegir una cápsula y seleccionar la opción de “Modificar”.

Una vez seleccionado, se presentará el siguiente formulario:

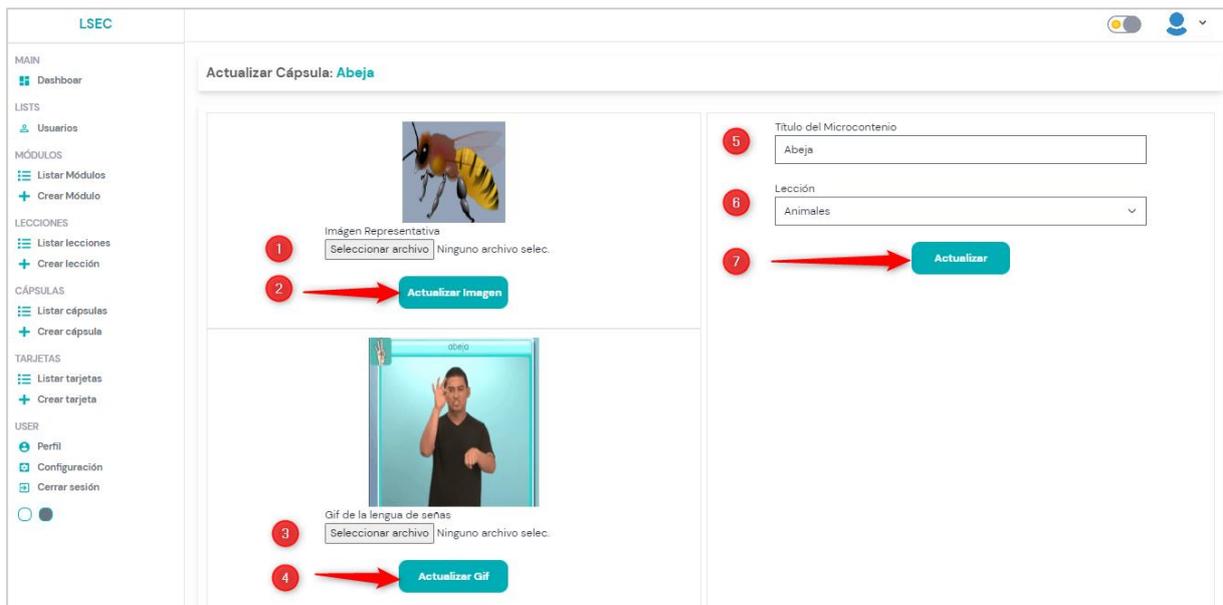


Imagen 29. Actualizar los datos de una cápsula de aprendizaje

Existen tres formularios, en el primer formulario podrá actualizar la imagen representativa, en el segundo formulario se puede actualizar el gif de la lengua de señas y en el tercer formulario se podrá actualizar: título de la cápsula de aprendizaje y la lección asociada.

d. Eliminar una cápsula de aprendizaje

Primero debe de listar todas las cápsulas de aprendizaje, elegir una cápsula y seleccionar la opción de “Eliminar”.

Una vez seleccionado, se presentará la siguiente ventana de confirmación:

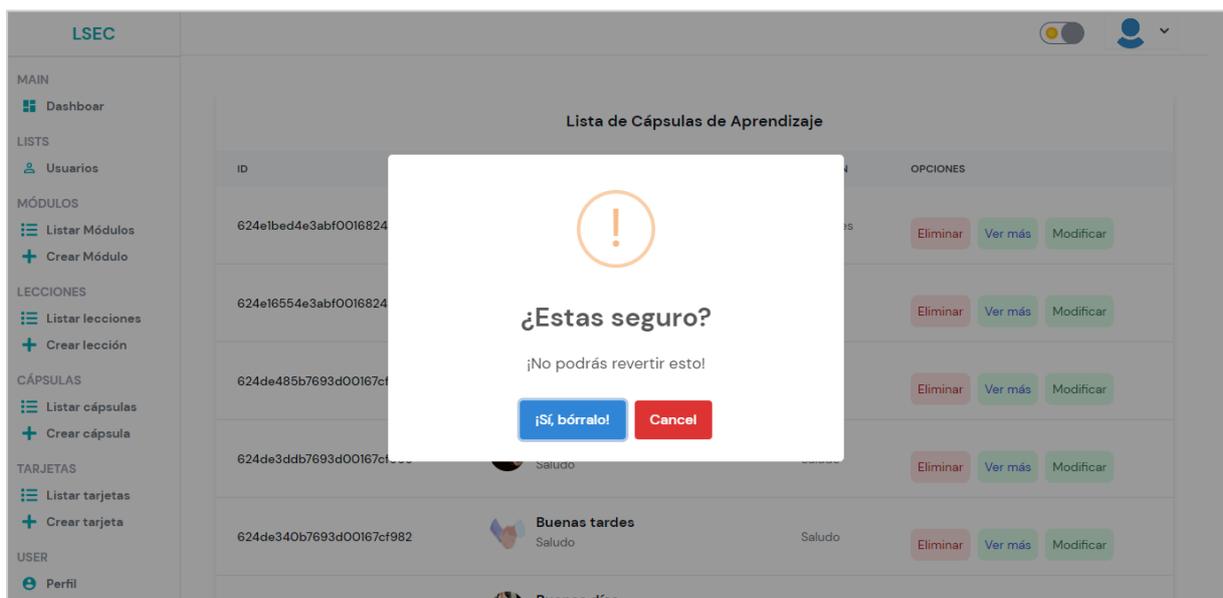


Imagen 30. Panel de confirmación para eliminar una cápsula de aprendizaje

e. Visualizar información cápsula de aprendizaje

Primero debe de listar todas las cápsulas de aprendizaje, elegir una cápsula y seleccionar la opción de “Ver más”.

Una vez seleccionado, se presentará la siguiente pantalla con toda la información de la cápsula de aprendizaje.

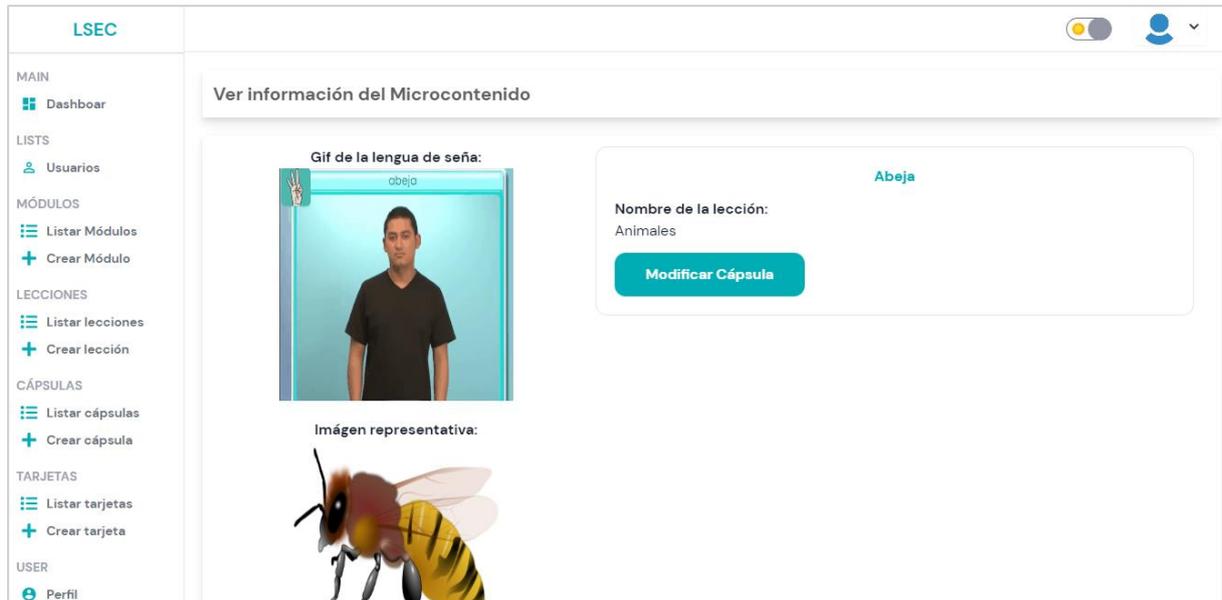


Imagen 31. Visualizar información de una cápsula de aprendizaje

11. Gestionar Tarjetas de Aprendizaje

En el dashboard se puede acceder a gestionar las tarjetas de aprendizaje a través de las etiquetas de la **Imagen 32**.

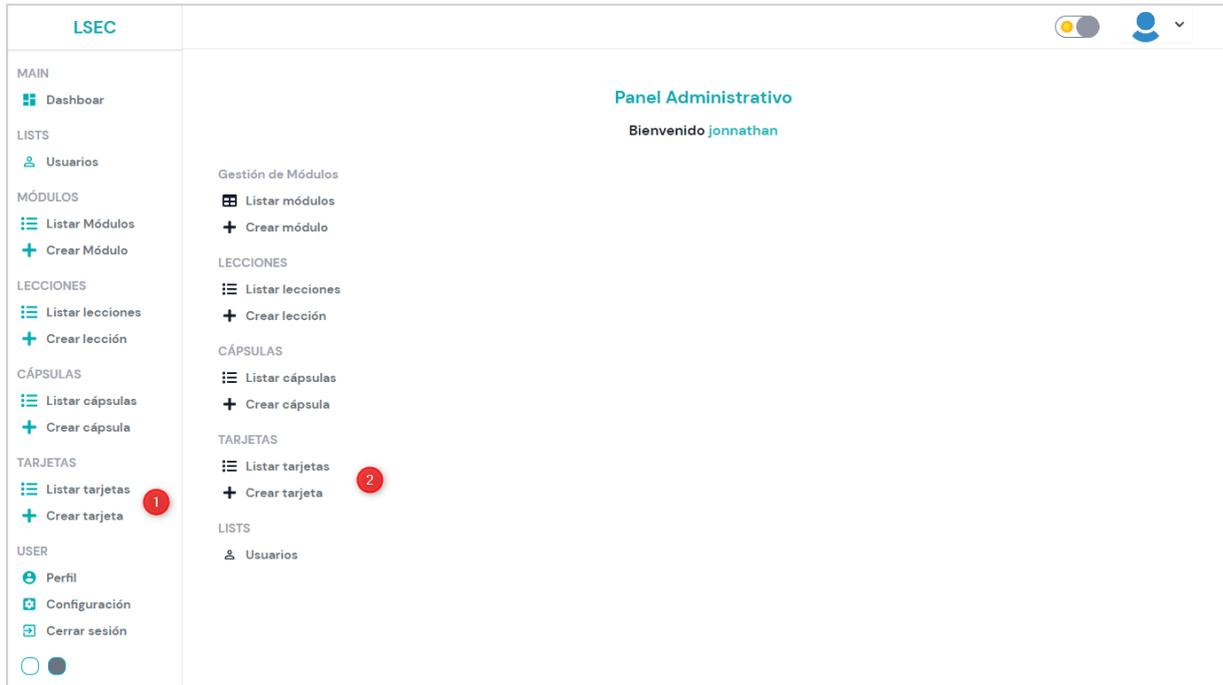


Imagen 32. Ingresar a gestionar tarjetas de aprendizaje

a. Crear una tarjeta de aprendizaje

En el dashboard de administrador, debe de ingresar a “Crear Tarjeta” señalado en las etiquetas de la **Imagen 32** y se presentará el formulario para crear una nueva tarjeta:

The image shows the 'Crear Tarjetas de Aprendizaje' form. It has a title bar 'Crear Tarjetas de Aprendizaje'. The form contains the following fields: 'Pregunta' (text input), 'Lección' (dropdown menu with 'Selecciona una Lección'), 'Respuesta correcta' (text input), 'Respuesta incorrecta' (text input), and 'Gif' (file selection button with 'Seleccionar archivo' and 'Ninguno archivo selec.'). At the bottom right is a blue 'Crear' button. Red circles with numbers 1 through 6 highlight the following elements: 1. Pregunta input field; 2. Lección dropdown menu; 3. Respuesta correcta input field; 4. Respuesta incorrecta input field; 4. Gif selection button; 6. The 'Crear' button, which is also pointed to by a red arrow.

Imagen 33. Formulario para crear una tarjeta de aprendizaje

En el formulario podrá hacer:

ETIQUETA	DESCRIPCIÓN
1	Ingresar la pregunta.
2	Seleccionar una lección.
3	Ingresar la respuesta correcta en texto.
4	Ingresar la respuesta incorrecta en texto.
5	Ingresar un gif (imagen animada).
6	Botón para crear una tarjeta de aprendizaje.

b. Listar todas las tarjetas de aprendizaje

En el dashboard de administrador, debe de ingresar a “Listar tarjetas” señalado en las etiquetas de la **Imagen 32** y se presentará una lista de todas las tarjetas creadas:

PREGUNTA	LECCIÓN	OPCIONES
¿Cuál es el significado?	Saludo	Eliminar Ver más Modificar
¿Cuál es el significado?	Saludo	Eliminar Ver más Modificar
¿Cuál es el significado?	Familia	Eliminar Ver más Modificar
¿Cuál es el significado?	Familia	Eliminar Ver más Modificar
¿Cuál es el significado?	Frutas	Eliminar Ver más Modificar
¿Cuál es el significado?	Frutas	Eliminar Ver más Modificar
¿Cuál es el significado?	Frases	Eliminar Ver más Modificar

Imagen 34. Listar todas las tarjetas de aprendizaje creadas

c. Actualizar datos de una tarjeta de aprendizaje

Primero debe de listar todas las tarjetas de aprendizaje, elegir una tarjeta y seleccionar la opción de “Modificar”.

Una vez seleccionado, se presentará el siguiente formulario:

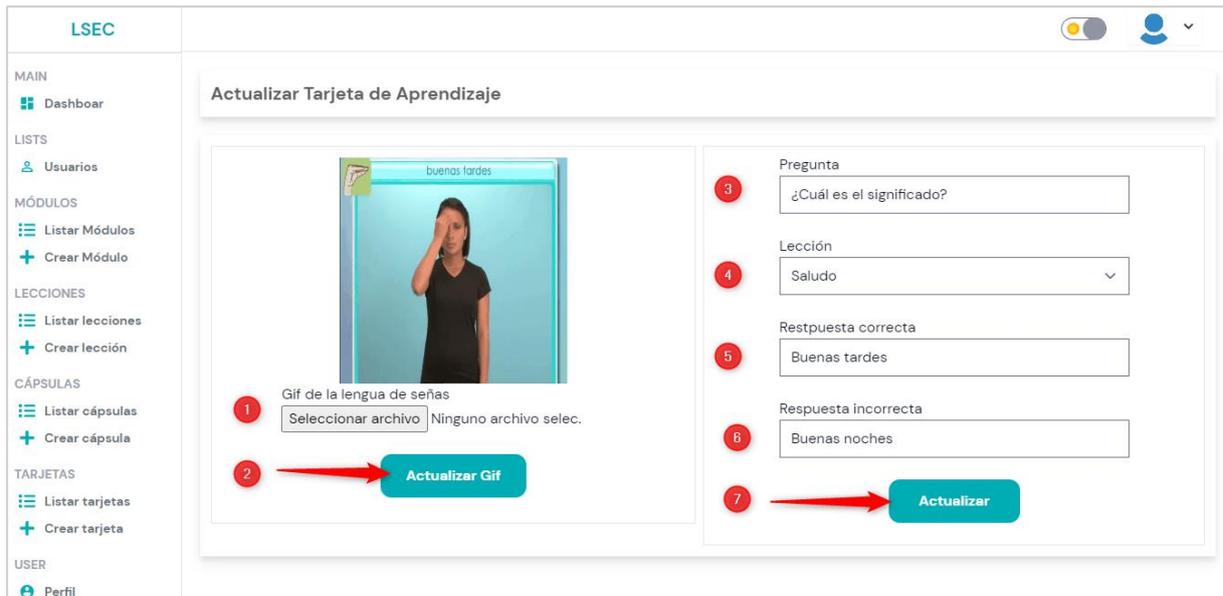


Imagen 35. Actualizar los datos de una tarjeta de aprendizaje

Existen dos formularios, en el primer formulario podrá actualizar el gif representativo y en el segundo formulario se podrá actualizar: la pregunta, lección a la que pertenece, la respuesta correcta y la respuesta incorrecta.

d. Eliminar una tarjeta de aprendizaje

Primero debe de listar todas las tarjetas de aprendizaje, elegir una prueba y seleccionar la opción de “Eliminar”.

Una vez seleccionado, se presentará la siguiente ventana de confirmación:

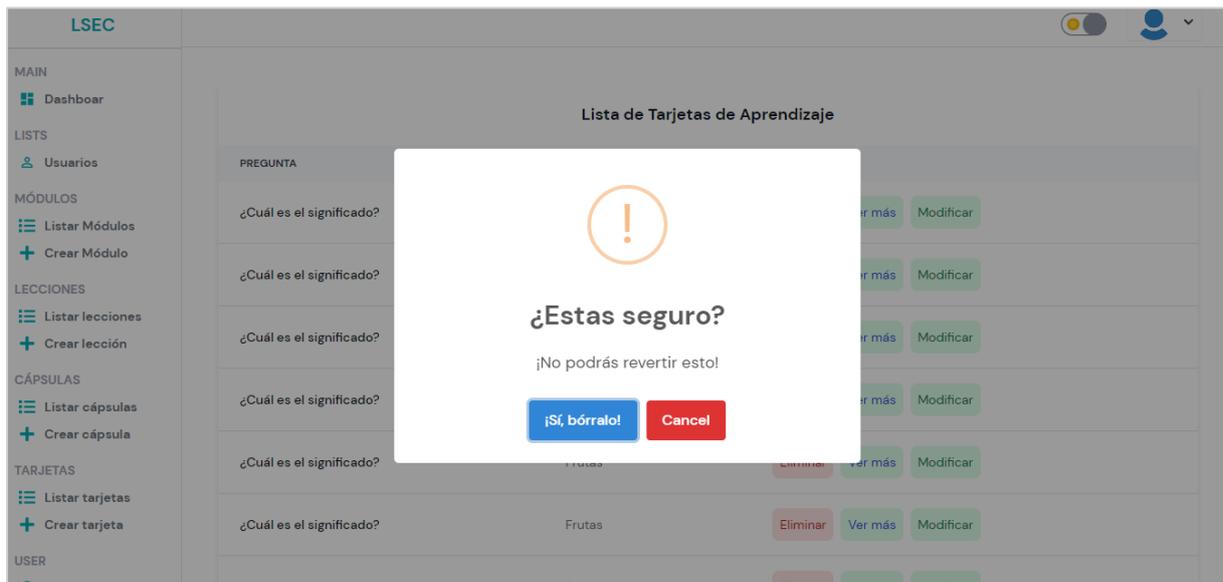


Imagen 36. Panel de confirmación para eliminar una tarjeta de aprendizaje

e. Visualizar contenido de una tarjeta de aprendizaje

Primero debe de listar todas las tarjetas de aprendizaje, elegir una tarjeta y seleccionar la opción de “Ver más”.

Una vez seleccionado, se presentará la siguiente pantalla con toda la información de la tarjeta.

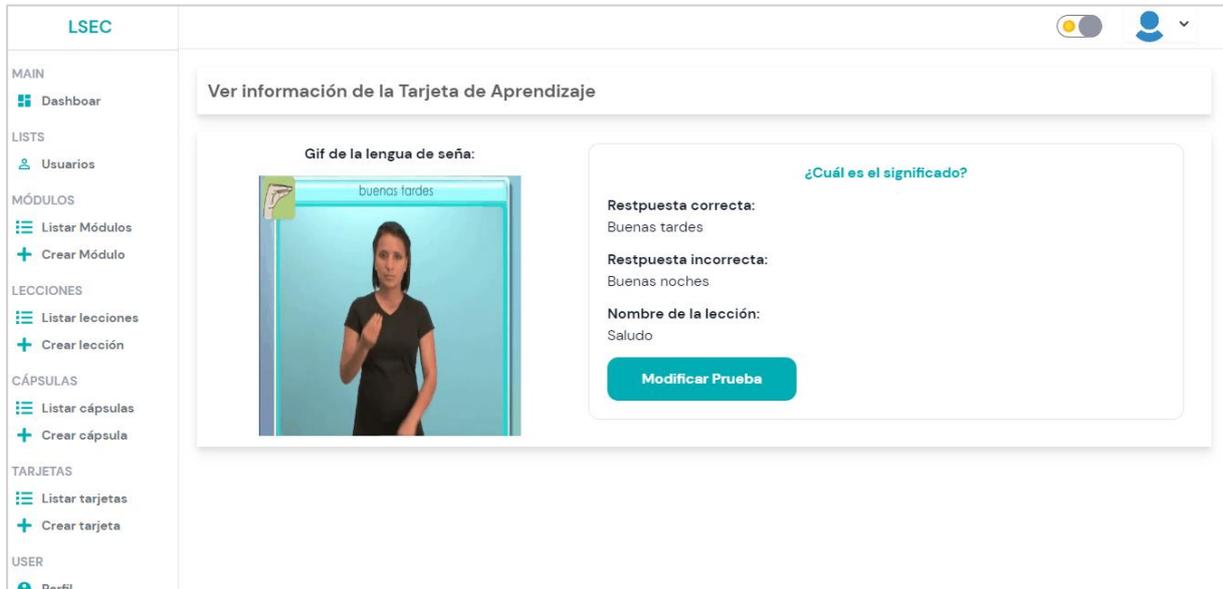


Imagen 37. Visualizar información de una tarjeta de aprendizaje

12. Listar usuarios

En el dashboard se puede acceder a la lista de usuarios creados a través de las etiquetas de la **Imagen 38**.

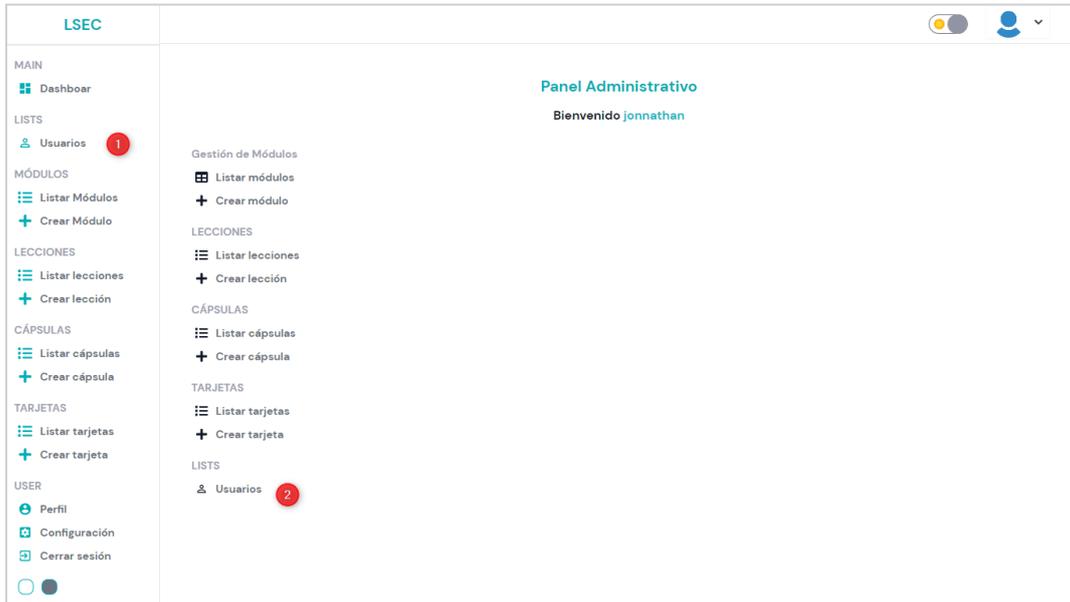


Imagen 38. Ingresar a la lista de usuarios

Al ingresar se mostrará una lista de los usuarios registrados en la aplicación, en donde se puede ver el id, nombre, correo electrónico y el rol.

ID	NOMBRE	EMAIL	ROL
[REDACTED]	jonnathan	[REDACTED]	student admin
[REDACTED]	jonathan	[REDACTED]@gmail.com	student
[REDACTED]	Genoveva Suing	[REDACTED]@gmail.com	student
[REDACTED]	Vladimir Celi	[REDACTED]@unl.edu.ec	student
[REDACTED]	victor	[REDACTED]@unl.edu.ec	student
[REDACTED]	Xavier	[REDACTED]@unl.edu.ec	student

Imagen 39. Lista de usuarios registrados

Anexo 17. Prueba de aceptación a profesional de la educación

Prueba de aceptación a profesional de la educación

Proyecto: Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano

Versión: 1.0

Fecha: 23/11/2022

Hoja de control

Proyecto	Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano
Entregable	Pruebas de Aceptación.
Autor	Jonnathan Espinoza
Versión	1.0
Aprobado por	Edison Leonardo Coronel Romero, Mg. Sc.

Registro de cambios

Versión del documento	Descripción	Responsable	Fecha de cambio
1.0	Versión inicial de prueba de aceptación a profesional de la educación.	Jonnathan Espinoza	23/11/2022

1. Introducción

Objetivo

Obtener la opinión del diseño de la aplicación web por parte de profesionales en la educación para verificar si la aplicación es apta para enseñar LSEC por medio del micro-learning.

Propósito

Obtener la opinión del diseño de la aplicación web por parte de profesionales en la educación y aceptar el micro-learning como método de enseñanza de la LSEC.

2. Diseño de encuestas

A continuación, se presenta el cuestionario, el cual consta de 11 preguntas:

Tabla 86.
Preguntas de las encuestas.

N°	Pregunta
P1	¿La aplicación web al momento de usarla fue?
P2	¿Considera que la aplicación es atractiva e intuitiva?
P3	¿Considera que la aplicación web es fácil de usar para cualquier persona con capacidad de manejar un navegador web?
P4	¿La funcionalidad de la aplicación web, permite el aprendizaje de la lengua de señas?
P5	¿Presentar las lecciones en contenidos cortos (Microlearning), ayuda a la enseñanza de la lengua de señas?
P6	¿El uso de GIFs (imágenes animadas) ayuda en la comprensión de la lengua de señas?
P7	¿Los repasos o prácticas por medio de la selección múltiple en la aplicación web, sirven como un medio de practicar la lengua de señas?
P8	¿Considera que la aplicación web fomenta el aprendizaje de la lengua de señas?
P9	¿La aplicación web garantiza su operabilidad al no presentar fallas?
P10	¿Haría uso cotidiano de la aplicación?
P11	¿Recomendaría a otros usuarios que usen la aplicación?

En la siguiente tabla se muestra las respuestas confirmadas según el tipo de pregunta, lo que permite a los encuestados elegir entre una serie de opciones utilizando diferentes escalas de Likert, rangos numéricos y sugiriendo preguntas al investigador.

Tabla 87.
Respuestas a las preguntas de la encuesta

Pregunta	Respuesta
P1	a. Muy fácil b. Fácil c. Normal d. Difícil e. Muy Difícil
P2, P3, P4, P5, P6, P7, P8, P9, P10, P11	a. Totalmente de acuerdo b. De acuerdo c. Indeciso d. En desacuerdo e. Totalmente en desacuerdo
Sugerencias al investigador	Abierta

3. Ejecución de encuestas

La investigación realizada tiene como finalidad conocer si la aplicación web desarrollada en el trabajo de investigación es apta para enseñar lengua de señas ecuatoriana por medio del micro-learning. Participo de forma voluntaria un docente en Pedagogía de las Ciencias Experimentales - Informática de la Universidad Nacional de Loja. Las muestras fueron seleccionadas convenientemente de acuerdo a su accesibilidad al investigador por el método no probabilístico.

Se realizó una demostración de la aplicación web desarrollada; después de probar y evaluar el contenido del software, el participante lleno una encuesta en la herramienta de Google Forms, como se presenta en la Figura 106.



The image shows a screenshot of a Google Forms survey. The title is "Encuesta LSEC Profesionales". At the top, there is a notification: "Se han guardado todos los cambios en Drive". The survey is divided into three tabs: "Preguntas", "Respuestas" (with a count of 1), and "Configuración". The main content area features the logo of the Universidad Nacional de Loja (UNL) and the text "Universidad Nacional de Loja". Below this, the survey title "Cuestionario de Funcionalidad" is displayed. The introductory text reads: "La presente encuesta forma parte del Trabajo de Titulación denominado 'Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano'. Por lo cual se desarrolló el siguiente cuestionario con el objetivo de 'Determinar el nivel de la facilidad de uso y aceptación de la aplicación web'." A second paragraph states: "La encuesta es totalmente de carácter voluntaria, por lo cual tiene plena facultad de decisión a participar o no en el proceso; la recolección de los datos, se realizará de forma anónima y confidencial." There is a required field for "Correo" (Email) with a red asterisk and a "Correo válido" label below it. On the right side, there is a vertical toolbar with icons for adding, deleting, and other form actions.

Figura 133. Aplicación de las encuestas.

Además, se solicitó al encuestado, nombre, apellido, numero de cedula y sugerencias, con el fin de dar fiabilidad a las respuestas. A continuación, se presentan los resultados obtenidos en las encuestas.

1. ¿La aplicación web al momento de usarla fue?

1 respuesta

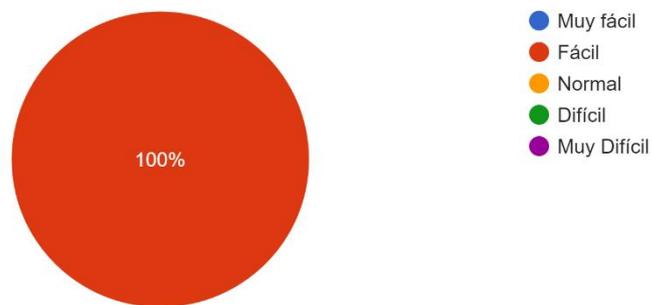


Figura 134. Respuesta a la pregunta 1.

2. ¿Considera que la aplicación web es atractiva e intuitiva?

1 respuesta

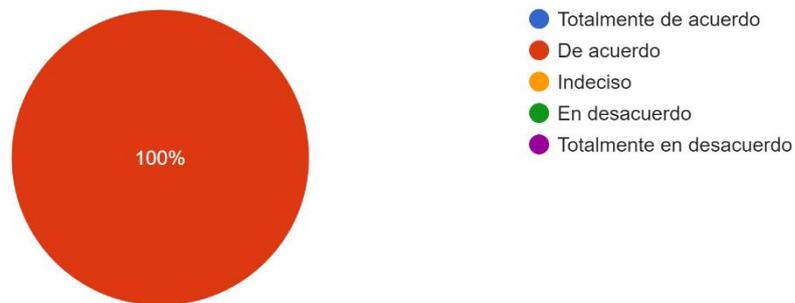


Figura 135. Respuesta a la pregunta 2.

3. ¿Considera que la aplicación web es fácil de usar para cualquier persona con capacidad de manejar un navegador web?

1 respuesta

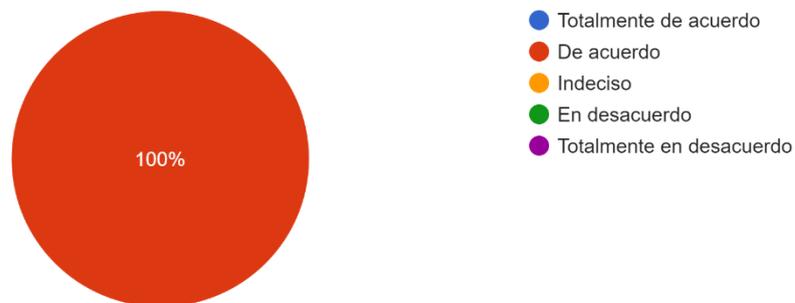


Figura 136. Respuesta a la pregunta 3

4. ¿La funcionalidad de la aplicación web, permite el aprendizaje de la lengua de señas?

1 respuesta

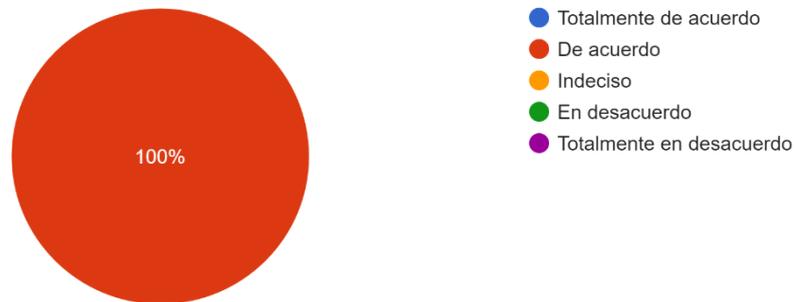


Figura 137. Respuesta a la pregunta 4.

5. ¿Presentar las lecciones en contenidos cortos (Microlearning), ayuda a la enseñanza de la lengua de señas?

1 respuesta

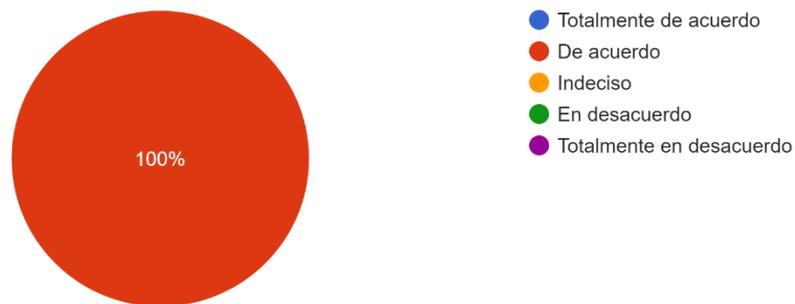


Figura 138. Respuesta a la pregunta 5.

6. ¿El uso de GIFs (imágenes animadas) ayuda en la comprensión de la lengua de señas?

1 respuesta

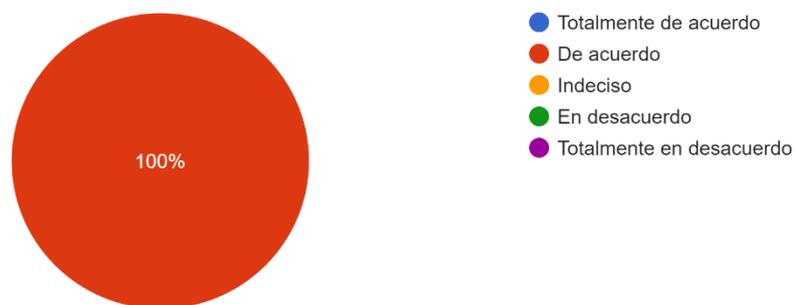


Figura 139. Respuesta a la pregunta 6.

7. ¿Los repasos o prácticas por medio de la selección múltiple en la aplicación web, sirven como un medio de practicar la lengua de señas?

1 respuesta

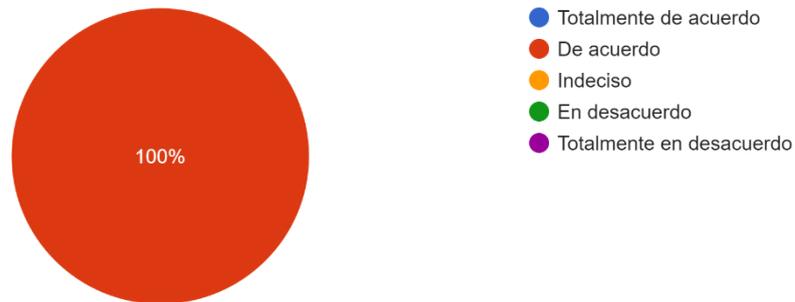


Figura 140. Respuesta a la pregunta 7.

8. ¿Considera que la aplicación web fomenta el aprendizaje de la lengua de señas?

1 respuesta

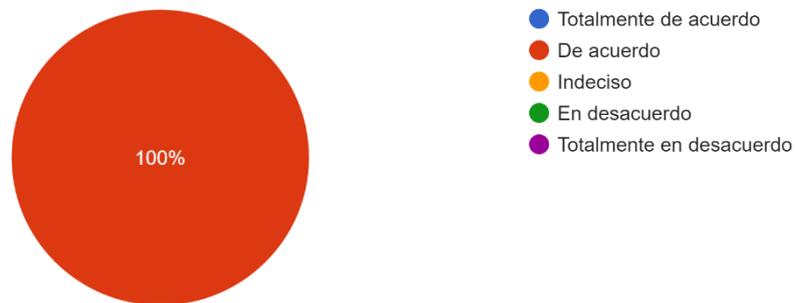


Figura 141. Respuesta a la pregunta 8.

9. ¿La aplicación web garantiza su operabilidad al no presentar fallas?

1 respuesta

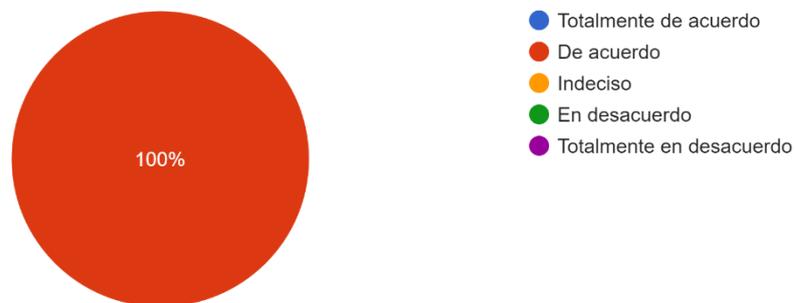


Figura 142. Respuesta a la pregunta 9.

10. ¿Haría uso cotidiano de la aplicación?

1 respuesta

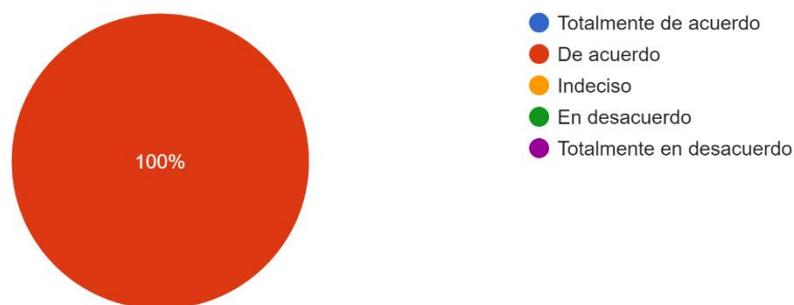


Figura 143. Respuesta a la pregunta 10.

11. ¿Recomendaría a otros usuarios que usen la aplicación?

1 respuesta

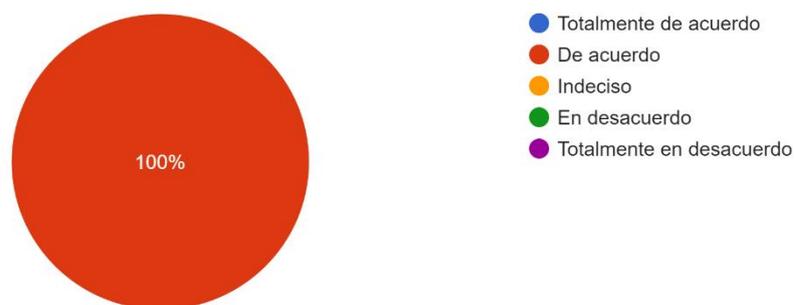


Figura 144. Respuesta a la pregunta 11.

a. Sugerencias al investigador

Adicionalmente se agregó una pregunta para obtener sugerencias por parte del encuestado, entre las cuales se encuentran las siguientes:

- Es importante sugerir que la imagen debe estar en relación a la Señá, considerando que eso facilita el proceso de enseñanza-aprendizaje. También sugiero que el recuadro de la persona que realiza la Señá, sea más grande y claro, para su ejercicio y comprensión.

4. Anexos

Usuarios que han respondido
<p>Correo electrónico</p> <p>ana.arciniega@unl.edu.ec</p>
<p>Nombres y Apellidos</p> <p>1 respuesta</p> <p>Ana Cristina Arciniega Carrión</p>
<p>Cédula</p> <p>1 respuesta</p> <p>1104579634</p>

Figura 145. Ejemplo de datos del encuestado

Anexo 18. Certificado de traducción del resumen



Mg. Yanina Quizhpe Espinoza
Licenciada en Ciencias de Educación mención
Inglés
Magister en Traducción y mediación cultural

Celular: +593989805087
Email: yaniges@icloud.com
Loja, Ecuador 110104

Loja, 6 de diciembre de 2022

Yo, Lic. Yanina Quizhpe Espinoza, con cédula de identidad 1104337553, docente del Instituto de Idiomas de la Universidad Nacional de Loja, y certificada como traductora e interprete en la Senescyt y en el Ministerio de trabajo del Ecuador con registro **MDT-3104-CCL-252640**, certifico:

Que tengo el conocimiento y dominio de los idiomas español e inglés y que la traducción del trabajo de titulación, **Diseño de aplicación web mediante el patrón de diseño Factory para Micro-Learning del lenguaje de señas ecuatoriano**, de autoría del estudiante, Jonnathan Damián Espinoza Erráez, con cédula 0107022006, es verdadero y correcto a mi mejor saber y entender.

Atentamente

YANINA BELEN QUIZHPE ESPINOZA
Firmado digitalmente por YANINA BELEN QUIZHPE ESPINOZA
Fecha: 2022.12.06 16:36:55 -05'00'

Yanina Quizhpe Espinoza.
Traductora

Full text translator: servicios de traducción