



**UNIVERSIDAD
NACIONAL
DE LOJA**



Facultad de la Energía, las Industrias y los Recursos Naturales no Renovables

CARRERA DE INGENIERÍA EN SISTEMAS

“Aplicación móvil multiplataforma que brinda información de las unidades de salud públicas y privadas de la ciudad de Loja ”

*Tesis previa a la Obtención del
título de Ingeniero en Sistemas*

Autor:

- Kevin Luis – Atiencia Pogo

Director:

Ing. Roberth Gustavo Figueroa Díaz, Mg. Sc

LOJA-ECUADOR

2017

Certificación del Director

Ing. Roberth Gustavo Figueroa Díaz, Mg. Sc

DOCENTE DEL ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES NO RENOVABLES Y DIRECTOR DE TESIS.

CERTIFICA

Haber dirigido, revisado y corregido en todas sus partes el desarrollo del Trabajo de Titulación de Ingeniería en Sistemas titulado: **“Aplicación móvil multiplataforma que brinda información de las unidades de salud públicas y privadas de la ciudad de Loja.”**, con autoría de la egresada Kevin Luis Atiencia Pogo. En razón de que la misma reúne a satisfacción los requisitos de fondo y forma, exigidos para la investigación de éste nivel, autorizo su presentación, sustentación y defensa ante tribunal designado para el efecto.

Loja, 20 de Febrero del 2017



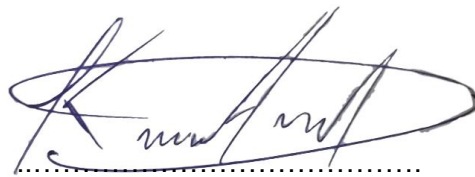
Ing. Roberth Gustavo, Figueroa Díaz, Mg. Sc.
DIRECTOR DEL TRABAJO DE TITULACIÓN.

Autoría

Yo, **KEVIN LUIS ATIENCIA POGO**, declaro ser autor del presente trabajo de tesis y eximo expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos de posibles reclamos o acciones legales por el contenido de la misma.

Adicionalmente acepto y autorizo a la Universidad Nacional de Loja, la publicación de mi tesis en el Repositorio Institucional – Biblioteca Virtual.

Firma:

A handwritten signature in blue ink, enclosed within a blue oval. The signature is stylized and appears to read 'Kevin Luis Atencia Pogo'. Below the signature is a horizontal dotted line.

Cédula: 0705738474

Fecha: 19 de abril del 2017

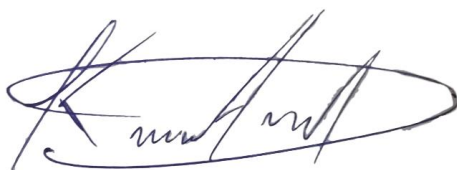
Carta de autorización de tesis por parte del autor, para la consulta, reproducción parcial o total y publicación electrónica del texto completo

Yo, **KEVIN LUIS ATIENCIA POGO**, declaro ser autor de la tesis titulada: **APLICACIÓN MÓVIL MULTIPLATAFORMA QUE BRINDA INFORMACIÓN DE LAS UNIDADES DE SALUD PÚBLICAS Y PRIVADAS DE LA CIUDAD DE LOJA**, como requisito para optar el grado de **INGENIERO EN SISTEMAS**; 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 Digital Institucional:

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

La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia de la tesis que realice un tercero.

Para constancia de esta autorización, en la ciudad de Loja a los diecinueve días del mes de abril del dos mil diecisiete.



Firma:

Autor: Kevin Luis Atencia Pogo

Cédula: 0705738474

Dirección: Marcabelí, (Cdla. 24 de junio, Av. Aurelio Apolo).

Correo Electrónico: klatenciap@unl.edu.ec

Teléfono: 072956055 **Celular:** 0969199863

DATOS COMPLEMENTARIOS

Director de Tesis: Ing. Roberth Gustavo, Figueroa Díaz, Mg. Sc.

Tribunal de Grado: Ing. Mario Andres Palma Jaramillo, Mg. Sc.

Ing. Manuel Alberto Córdova Neira, Mg. Sc.

Ing. Roberto Carlos Pineda López, Mg. Sc.

Agradecimiento

Agradezco a Dios por haberme acompañado y guiado a lo largo de toda mi carrera y a mis padres Nancy y Luis, y a mi hermana Shirley por ser la fuente de inspiración, apoyando en todo momento, sin ellos no podrían haber culminado mis estudios universitarios.

A la Universidad Nacional de Loja y a la carrera de Ingeniería en Sistemas y docentes, quienes me inculcaron los conocimientos teóricos y prácticos necesarios para mi formación académica, a mi tutor de tesis Ing. Roberth Figueroa, por haberme brindado su tiempo, paciencia y la oportunidad de acudir a sus conocimientos que me ayudaron en la realización del presente proyecto.

Y finalmente a mis amigos, por su apoyo moral y confiar en mí, además de haber hecho de esta etapa universitaria un trayecto de vivencias que nunca se olvidaran.

Dedicatoria

Este trabajo de titulación va dedicado primeramente a Dios, por darme la fortaleza necesaria para culminar esta etapa, siempre iluminando cada paso que doy en mi vida, en los momentos de dificultad y de prosperidad.

A mis padres y hermana que son mi apoyo, motivación y ejemplo para seguir adelante, por todo el cariño que día a día demuestran hacia mí, y por ser lo más valioso y preciado que tengo en este mundo.

A los docentes y tutor quienes me impartieron sus conocimientos y experiencias a lo largo de la carrera universitaria.

A.Título

“Aplicación móvil multiplataforma que brinda información de las unidades de salud públicas y privadas de la ciudad de Loja.”

B. Resumen

Hoy en día los medios de difusión que utilizan las unidades de salud públicas y privadas no son suficientes, ni los más adecuados para brindar información acerca de las especialidades y servicios, que estas ofrecen a la sociedad de la ciudad de Loja.

Es por ello que el presente trabajo de titulación se basa en el análisis y desarrollo de dos aplicaciones: una móvil denominada IUS (Información de las Unidades de Salud), la misma que se la puede utilizar en diferentes sistemas operativos, permitiendo localizar unidades de salud por medio de GPS, buscar especialidades y obtener información completa, precisa y confiable de los servicios, horarios, números telefónicos, direcciones, nivel atención, entre otros aspectos.

Además, se construyó una aplicación web dirigida al administrador, dedicada a gestionar y administrar la información de las unidades de salud, especialidades y servicios, denominadas SIUS (Sistema de Información de las Unidades de Salud).

Para el desarrollo del software, se utilizó de la metodología ágil Programación Extrema (XP), debido a que se adapta a las necesidades del proyecto, es flexible a requerimientos imprecisos y cambiantes, requiere de un grupo limitado de trabajo y su proceso de desarrollo es sencillo de utilizar.

Asimismo, el uso de métodos como el deductivo, permite llevar un proceso ordenado del desarrollo del trabajo de titulación. Partiendo desde la aplicación de técnicas como la bibliográfica, que ayuda a obtener la información más relevante de las herramientas y tecnologías necesarias para la construcción de las aplicaciones y el uso de encuestas, utilizadas para la recolección de información de las necesidades y requerimientos presentes en un tamaño muestral de la ciudad de Loja. Hasta llegar a lo específico, dando como resultado la obtención de la aplicación móvil y web funcional y libre de errores que podrían perjudicar su desempeño.

SUMMARY

Today the medias of outreach that use public and private health units are not competent, or the most appropriate to provide information about the specialties and services, which they offer to the society of the city of Loja.

Is why this final work degree is based on the analysis and development of two applications: a mobile called HUY (Health Units Information), that it can be used in different operating systems, allowing you to locate health units through GPS, also allows search specialties and get complete information of services , opening hours , telephone numbers, addresses, level attention, among other aspects.

In addition, built a web application addressed to the Manager, dedicated to manage and administer the information of health units, specialties and services. This aplication is called HUIS (Health Units Information System).

For the development of the software, the methodology was used Agile Extreme Programming (XP), due to the fact that adapts to the needs of the project, is flexible to changing requirements vague and requires a limited working group and its development process is easy to use.

Also the use of the deductive methods, allows an orderly process of the development of work of qualification. Starting from the application of techniques such as the bibliographic, which helps obtain the most relevant information of the tools and technologies needed for the construction of applications and the use of surveys, used for the collection of information on the needs and requirements present in a sample size of the city of Loja. To reach specific, resulting in the obtaining of the pone and web application functional and free of errors that could harm their performance.

ÍNDICE DE CONTENIDOS

Certificación del Director	II
Autoría	III
Carta de autorización de tesis por parte del autor, para la consulta, reproducción parcial o total y publicación electrónica del texto completo.....	IV
Agradecimiento	V
Dedicatoria.....	VI
A. Título.....	1
B. Resumen	2
Summary	3
Índice de contenidos	4
Índice de Figuras.....	10
Índice de Tablas	14
C. Introducción.....	17
D. Revisión Literaria	19
4.1. APLICACIONES MÓVILES.....	19
4.1.1. Funcionalidades de las Aplicaciones.....	19
4.1.2. Tipo de Aplicaciones Móviles	19
4.1.2.1. Aplicaciones Nativas.....	19
4.1.2.1. Aplicaciones Web	20
4.1.2.1. Aplicaciones Híbridas	20
4.2. METODOLOGÍAS ÁGILES PARA EL DESARROLLO DE SOFTWARE.....	21
4.2.1. Principios de las Metodologías Ágiles	21
4.2.2. Tipos de Metodologías Ágiles.....	22
4.2.2.1. Programación Extrema (XP).....	22
4.2.2.1.1. Fases	23

4.2.2.1.2.	Roles	24
4.2.2.1.3.	Principios	25
4.2.2.1.	SCRUM	25
4.2.2.1.1.	Características	26
4.2.2.1.2.	Artefactos	27
4.2.2.1.3.	Roles	27
4.2.2.2.	ICONIX	29
4.2.2.2.1.	Características	29
4.2.2.2.2.	Fases	29
4.2.2.3.	Mobile-D	30
4.3.	FRAMEWORKS PARA EL DESARROLLO DE APLICACIONES MÓVILES	32
4.3.1.	Ionic	32
4.3.1.1.	Características:	33
4.3.2.	Phonegap	34
4.3.2.1.	Prestaciones	35
4.3.2.2.	Ventajas	35
4.3.2.3.	Desventajas	35
4.3.3.	Titanium	35
4.3.3.1.	Prestaciones	36
4.3.3.2.	Ventajas	36
4.3.3.3.	Desventajas	36
4.4.	FRAMEWORK PARA EL DESARROLLO DE APLICACIONES WEB	37
4.4.1.	Codeigniter	37
4.4.1.1.	Características	37
4.4.1.2.	Modelo-Vista-Controlador En Codeigniter	38
4.4.1.3.	Modelo	38
4.4.1.4.	Vista	39
4.4.1.5.	Controlador	39
4.5.	SERVICIOS WEB	39
4.5.1.	REST	39

4.6.	HERRAMIENTAS PARA EL DESARROLLO DE APLICACIONES	41
4.6.1.	Node Js	41
4.6.2.	Angular Ms	41
4.6.3.	Córdova.....	41
4.6.4.	Bower	42
4.6.4.1.	Instalar Bower	42
4.6.5.	PHP	42
4.6.5.1.	Funciones de PHP	43
4.6.6.	HTML.....	43
4.6.6.1.	Características.....	44
4.6.7.	WAMP	45
4.6.7.1.	Características.....	45
4.6.7.2.	Instalación.....	45
4.6.8.	Netbeans.....	45
4.6.8.1.	Características.....	46
E.	Materiales y Métodos	47
5.1.	MÉTODOS.....	47
5.1.1.	Método Deductivo	47
5.1.2.	Método Inductivo.....	47
5.1.1.	Método Analítico	47
5.2.	TÉCNICAS.....	47
5.2.1.	Encuesta	47
5.2.2.	Bibliográfica.....	48
5.2.3.	Observación	48
5.3.	METODOLOGÍA DE DESARROLLO.....	48
F.	Resultados.....	49
6.1.	COMPARATIVA DE METODOLOGÍAS.....	49

6.2.	COMPARATIVA DE FRAMEWORKS.....	51
6.3.	FASE DE PLANEACIÓN.....	52
6.3.1.	Roles	52
6.3.2.	Historias de Usuario.....	53
6.3.2.1.	Aplicación Web	54
6.3.2.2.	Aplicación Móvil	59
6.3.2.3.	Resumen de Historias de Usuario.....	62
6.3.3.	Plan de Iteración	63
6.3.4.	Identificación de Tareas.....	64
6.4.	FASE DE DISEÑO	70
6.4.1.	Patrones de Diseño.....	70
6.4.2.	Dominio de la solución.....	71
6.4.3.	Modelo General de Casos de Uso	73
6.4.4.	Diagrama de Clases.....	74
6.4.5.	Diagrama de Componentes.....	75
6.4.6.	Diagrama de Paquetes	76
6.4.7.	Diagrama Relacional de Base de Datos.....	77
6.4.8.	Diseño de Tarjetas CRC	78
6.4.9.	Diseño de Prototipos de Pantalla	81
6.4.9.1.	Aplicación Móvil	81
6.4.9.2.	Aplicación Web	84
6.5.	FASE DE CODIFICACIÓN.....	87
6.5.1.	Codificación web	87
6.5.1.1.	Aplicación Web Final	88
6.5.2.	Codificación del Servicio Web	89
6.5.2.1.	Instalación.....	90
6.5.2.2.	Clases del API Rest.....	91

6.5.3.	Codificación de la Aplicación Móvil	92
6.5.3.1.	Controladores	92
6.5.3.2.	Aplicación Móvil Final	93
6.6.	FASE DE PRUEBAS	95
6.6.1.	Herramientas de Prueba	95
6.6.2.	Plan de Pruebas	96
6.6.2.1.	Propósito	96
6.6.2.2.	Alcance	96
6.6.2.3.	Características a probar	97
6.6.2.4.	Características que no van hacer evaluadas	97
6.6.2.5.	Actividades a Realizar	97
6.6.2.6.	Pruebas a Realizar	97
6.6.2.7.	Actividades de prueba	98
6.6.2.8.	Recursos	98
6.6.2.9.	Resultados	99
6.6.2.10.	Pruebas Unitarias de funcionalidad.....	99
6.6.2.10.1.	Historia de usuario	100
6.6.2.11.	Pruebas de Aceptación	112
6.6.2.12.	Pruebas de Carga.....	113
6.6.2.13.	Pruebas de Accesibilidad	117
G.	Discusión	118
7.1.	Evaluación del objeto de investigación	118
7.2.	Valoración Técnico – Económica – Ambiental	120
H.	Conclusiones	123
I.	Recomendaciones	125
J.	BIBLIOGRAFÍA	126
K.	Anexos	130
	Anexo I: Especificación de Requisitos de Software	130
	Anexo II: Prototipo de la Aplicación Móvil	153

Anexo III: Prototipo de la Aplicación Web	160
Anexo IV: Pruebas de Carga	164
Anexo V: Pruebas de Aceptación realizadas al Usuario.....	171
Anexo VI: Licencia Creative Commons	173

ÍNDICE DE FIGURAS

Figura 1: Ciclo de entrega de la Programación Extrema [12].....	23
Figura 2: Flujo del proceso Scrum [7].....	26
Figura 3: Reglas de SCRUM [16].....	28
Figura 4: Ciclo de desarrollo Mobile-D, Tomada de [20]	31
<i>Figura 5: Patrón MVC [49]</i>	<i>70</i>
<i>Figura 6: Arquitectura de la Solución</i>	<i>72</i>
Figura 7: Diagrama General de Casos de Uso General	73
<i>Figura 8: Diagrama de Clases</i>	<i>74</i>
Figura 9 Diagrama: de Componentes	75
Figura 10: Diagrama de Paquetes.....	76
Figura 11: Diagrama Relacional de Base de Datos	77
Figura 12: Interfaz Inicial.....	81
Figura 13: Interfaz de Google Maps	82
Figura 14: Interfaz de Información de Unidades de Salud.....	83
Figura 15: Interfaz de Login.....	84
Figura 16: Interfaz de Inicio	85
Figura 17: Interfaz de Crear Unidad de Salud.....	85
Figura 18: Interfaz de Crear Especialidad	86
Figura 19: Interfaz de Crear Servicio.....	86
Figura 20: Interfaz Final de Administración SIUS	88
Figura 21: Interfaz Final con la Lista de las Unidades de Salud.....	89
Figura 22: Interfaz Final información de una Unidades de Salud.....	89
Figura 23: Diagrama del Servicio web.....	90
Figura 24: Interfaz Final Inicio de la aplicación móvil IUS	93
Figura 25: Interfaz Final de Buscar Especialidades.....	93

Figura 26: Interfaz Final de Google Maps	94
Figura 27: Interfaz Final información de una Unidad de Salud.....	94
Figura 28: JSON de la petición Find de la clase AreasEspecialidad_SW	101
Figura 29: JSON de la petición datos_get de la clase AreasEspecialidad_SW.....	101
Figura 30: JSON de la petición datosId_get de la clase AreasEspecialidad_SW.....	102
Figura 31: JSON de la petición index_post de la clase Contribuidor_SW.....	103
Figura 32: JSON de la petición index_delete de la clase Contribuidor_SW	103
Figura 33: JSON de la petición index_get de la clase Especialidad_SW.....	104
Figura 34: JSON de la petición find_get de la clase Especialidad_SW	106
Figura 35: JSON de la petición widget de la clase Servicios_SW.....	107
Figura 36: JSON de la petición disponibilidad_get de la clase Servicios.....	108
Figura 37: JSON de la petición index_post de la clase Servicios_SW.....	108
Figura 38: JSON de la petición index_get de la clase UnidadSalud_SW	109
Figura 39: JSON de la petición find_get de la clase UnidadSalud_SW	110
Figura 40: JSON de la petición lucro_get de la clase UnidadSalud_SW	111
Figura 41: JSON de la petición disponibilidad_get de la clase UnidadSalud_SW.....	112
Figura 42: Prueba de la clase UnidadSalud_SW, función index_get.....	113
Figura 43: Resultado de la prueba realizada a la clase UnidadSalud_SW, función index_get.....	114
Figura 44: Reporte resumen de la clase UnidadSalud_SW, función index_get.....	114
Figura 45: Prueba de la clase UnidadSalud_SW, función find_get.....	115
Figura 46: Resultado de la prueba realizada a la clase UnidadSalud_SW, función find_get.....	115
Figura 47: Reporte resumen de la clase UnidadSalud_SW, función find_get	115
Figura 48: Prueba de la clase UnidadSalud_SW, función lucro_get.....	116

Figura 49: Resultado de la prueba realizada a la clase UnidadSalud_SW, función lucro_get.....	116
Figura 50: Reporte resumen de la clase UnidadSalud_SW, función lucro_get	116
Figura 51: Prueba de la clase UnidadSalud_SW, función disponibilidad _get.....	116
Figura 52: Resultado de la prueba realizada a la clase UnidadSalud_SW, función disponibilidad _get	117
Figura 53: Reporte resumen de la clase UnidadSalud_SW, función disponibilidad_get	117
Figura 54: Pruebas de accesibilidad de la aplicación web	118
Figura 55: Interfaz Inicial.....	153
Figura 56: Interfaz de Ayuda	154
Figura 57: Interfaz de Google Maps	154
Figura 58: Interfaz de Información de Unidades de Salud.....	155
Figura 59: Interfaz de Información de Unidades de Salud.....	155
Figura 60: Interfaz de las Áreas de Especialidad que dispone una Unidad de Salud	155
Figura 61: Interfaz de Información de las especialidades.....	156
Figura 62: Interfaz de Información de los Servicios	156
Figura 63: Interfaz del Formulario del Contribuidor.....	157
Figura 64: Interfaz del Formulario de Servicio	157
Figura 65: Interfaz de Buscar Especialidad.....	158
Figura 66: Interfaz de Resultado de la Búsqueda.....	158
Figura 67: Interfaz de información del Resultado de Búsqueda.....	159
Figura 68: Interfaz de Login	160
Figura 69: Interfaz de Inicio	160
Figura 70: Interfaz de Crear Unidad de Salud.....	161
Figura 71: Interfaz de Crear Especialidad	161

Figura 72: Interfaz de Crear Servicio.....	162
Figura 73: Interfaz de Listar Unidades de Salud.....	162
Figura 74: Interfaz de Información Unidades de Salud.....	163
Figura 75: Interfaz de Listar Especialidades	163
Figura 76: Interfaz de Listar Servicios	164
Figura 77: Prueba de la clase AreaEspecialidad_SW, función find_get	164
Figura 78: Reporte resumen de la clase AreaEspecialidad_SW, función find_get....	165
Figura 79: Reporte resumen de la clase AreaEspecialidad_SW, función datos_get..	165
Figura 80: Reporte resumen de la clase AreaEspecialidad_SW, función datos_get..	165
Figura 81: Reporte resumen de la clase AreaEspecialidad_SW, función datosId_get	166
Figura 82: Reporte resumen de la clase AreaEspecialidad_SW, función datosId_get	166
Figura 83: Reporte resumen de la clase Contribuidor_SW, función Index_get	166
Figura 84: Reporte resumen de la clase Contribuidor_SW, función index_get.....	166
Figura 85: Reporte resumen de la clase Contribuidor_SW, función Index_delete.....	167
Figura 86: Reporte resumen de la clase Contribuidor_SW, función index_delete.....	167
Figura 87: Reporte resumen de la clase Especialidad_SW, función index_get.....	167
Figura 88: Reporte resumen de la clase Especialidad_SW, función index_get.....	167
Figura 89: Reporte resumen de la clase Especialidad_SW, función find_get.....	168
Figura 90: Reporte resumen de la clase Especialidad_SW, función find_get.....	168
Figura 91: Reporte resumen de la clase Servicios_SW, función find_get.....	169
Figura 92: Reporte resumen de la clase Servicios_SW, función find_get.....	169
Figura 93: Reporte resumen de la clase Servicios_SW, función disponibilidad_get ..	169
Figura 94: Reporte resumen de la clase Servicios_SW, función disponibilidad_get ..	169
Figura 95: Reporte resumen de la clase Servicios_SW, función index_post.....	170

Figura 96: Reporte resumen de la clase Servicios_SW, función index_post..... 170

ÍNDICE DE TABLAS

TABLA I: COMPARATIVA DE METODOLOGÍAS AGILES.....	49
TABLA II: COMPARATIVA DE FRAMEWORKS MULTIPLATAFORMA.....	51
TABLA III: FORMATO DE LA HISTORIA DE USUARIO.....	53
TABLA IV: ACCEDER AL SISTEMA.....	54
TABLA V: EDITAR INFORMACIÓN DEL USUARIO.....	54
TABLA VI: GESTIÓN DE UNIDADES DE SALUD.....	55
TABLA VII: LISTAR UNIDADES DE SALUD.....	56
TABLA VIII: GESTIÓN DE ÁREAS DE ESPECIALIDAD.....	57
TABLA IX: GESTIÓN DE SERVICIOS.....	57
TABLA X: GESTIÓN DE CONTRIBUIDORES.....	58
TABLA XI: GESTIÓN DE MARCADORES.....	59
TABLA XII: GESTIÓN SERVICIO WEB.....	59
TABLA XIII: BUSCAR UNIDADES DE SALUD.....	60
TABLA XIV: BUSCAR ÁREAS DE ESPECIALIDAD.....	61
TABLA XV: CREAR SERVICIO.....	61
TABLA XVI: OBTENER INFORMACIÓN DE LA US.....	62
TABLA XVII: RESUMEN DE LAS HISTORIAS DE USUARIO.....	62
TABLA XVIII: PLAN DE ITERACIÓN.....	63
TABLA XIX: TAREA ACCEDER AL SISTEMA (APP WEB).....	64
TABLA XX: TAREA MODIFICAR DATOS DE USUARIO.....	64
TABLA XXI: TAREA GESTIÓN DE UNIDADES DE SALUD.....	65
TABLA XXII: TAREA ELIMINAR Y SUBIR ARCHIVO DE LAS UNIDADES DE SALUD.....	65

TABLA XLIV: PRUEBA DE ACEPTACIÓN A LA CLASE ESPECIALIDAD_SW DE LA FUNCIÓN INDEX_GET.	104
TABLA XLV: PRUEBA DE ACEPTACIÓN A LA CLASE ESPECIALIDAD_SW DE LA FUNCIÓN FIND_GET.	105
TABLA XLVI: PRUEBA DE ACEPTACIÓN A LA CLASE SERVICIOS_SW DE LA FUNCIÓN ID_GET.	106
TABLA XLVII: PRUEBA DE ACEPTACIÓN A LA CLASE SERVICIOS_SW DE LA FUNCIÓN DISPONIBILIDAD_GET.	107
TABLA XLVIII: PRUEBA DE ACEPTACIÓN A LA CLASE SERVICIOS_SW DE LA FUNCIÓN INDEX_POST.	108
TABLA XLIX: PRUEBA DE ACEPTACIÓN A LA CLASE UNIDADSALUD_SW DE LA FUNCIÓN INDEX_GET.	109
TABLA L: PRUEBA DE ACEPTACIÓN A LA CLASE UNIDADSALUD_SW DE LA FUNCIÓN FIND_GET.	110
TABLA LI: PRUEBA DE ACEPTACIÓN A LA CLASE UNIDADSALUD_SW DE LA FUNCIÓN LUCRO_GET.	110
TABLA LII: PRUEBA DE ACEPTACIÓN A LA CLASE UNIDADSALUD_SW DE LA FUNCIÓN DISPONIBILIDAD_GET.	111
TABLA LIII: RECURSOS HUMANOS.	121
TABLA LIV: RECURSOS MATERIALES.	121
TABLA LV: RECURSOS TÉCNICOS.	122
TABLA LVI: COSTO TOTAL.	122

C.Introducción

En los últimos años el aumento del uso de dispositivos móviles se ha convertido en una parte integral en la vida cotidiana de las personas, lo que conduce a la expansión del desarrollo de aplicaciones y el desafío de utilizarlas para las diferentes plataformas.

Hoy en día, las aplicaciones están dirigidas a dispositivos móviles específicos, la expectativa radica en que sea ejecutable en la mayor cantidad de dispositivos. Es decir, que sean multiplataforma, combinando lo mejor de la programación nativa y web, esto proporciona un entorno inmejorable para el desarrollo de aplicaciones capaces de usar de forma eficiente los recursos del dispositivo y esté sea utilizado en la mayor número de plataformas, reduciendo el tiempo y costo por mantenimiento o crecimiento del software desarrollado [1] [2] [3].

Actualmente las aplicaciones de información que han sido desarrolladas referente a las unidades de salud, son escasas, ineficientes, el rendimiento es sumamente lento y los resultados de su procesamiento no son los esperados, el cual provoca insatisfacción en los usuarios que las han utilizado.

Si a esto le agregamos, los medios que utilizan las unidades de salud pública y privada de la ciudad de Loja, no son los más adecuados para brindar información, por lo que no existe un centro de información, o se asume que los ciudadanos ya conocen todas las ubicaciones de los centros de salud. Esta situación genera malestar en la mayor parte de la población, al momento de suscitarse una emergencia médica, no dispondrían de un medio rápido, eficaz de información.

Considerando los antecedentes expuestos, el problema de investigación versa de la siguiente manera:

Los habitantes de la ciudad de Loja no cuentan con medios de información detallada acerca de los servicios médicos y especialistas que ofrecen las unidades de salud públicas y privadas, generando en la población malestar al sentirse desinformados

Por lo tanto, el objetivo principal del presente trabajo ha sido desarrollar una aplicación móvil multiplataforma, que cuente con información actualizada online y detallada de los servicios que disponen las unidades de salud públicas y privadas de la ciudad de Loja.

Este objetivo general puede descomponerse en los siguientes objetivos específicos: El primero parte de realizar el análisis de requerimientos de usuario para definir los

servicios y procesos que la aplicación ofrece. Seguidamente se elabora el diseño preliminar de la aplicación móvil para obtener los diferentes diagramas. Para posteriormente desarrollar la aplicación móvil, utilizando las tecnologías previamente seleccionadas e implementar un servicio web para el consumo de datos. Para finalmente probar y evaluar las funcionalidades de la aplicación móvil y de consumo de información implementadas, para determinar su correcto funcionamiento.

El presente trabajo de titulación se encuentra estructurado a lo largo de 9 secciones:

Las tres primeras secciones corresponden al: TEMA, RESUMEN e INTRODUCCIÓN. En la cuarta se encuentra la REVISIÓN LITERARIA, la misma en que se detalla la información relacionada con el proyecto: Las aplicaciones móviles y los distintos tipos que existen, una introducción a las metodologías ágiles para el desarrollo de software, considerando las más relevantes para la construcción del software. Así mismo, se describen los frameworks para el desarrollo de aplicaciones móviles tales como: Ionic, Phonegap y Titanium entre las más importantes y usadas hoy en día, y el framework Codeigniter utilizada para construir la aplicación web y el servicio REST. Finalmente se puntualizan las herramientas tecnológicas necesarias para el desarrollo del proyecto.

La quinta sección concierne a los MATERIALES Y MÉTODOS donde se detalla la metodología de desarrollo de software, los métodos y las técnicas de investigación empleadas en el transcurso de todo el proyecto.

En la sexta sección se indican los RESULTADOS obtenidos, donde se especifica la comparativa de metodologías ágiles y frameworks, muy importantes para el desarrollo de las aplicaciones. También se expone los pasos realizados para el desarrollo del software usando la metodología Programación Extrema y las pruebas llevadas a cabo para la validación de la aplicación realizada por parte del Dr. Carlos Celi.

La séptima sección engloba la DISCUSIÓN donde se evalúa los resultados obtenidos según los objetivos planteados. Se resume los inconvenientes presentados en cada etapa, y la forma en que se los resolvió. Se concluye con la octava y novena sección en que se detalla las CONCLUSIONES y RECOMENDACIONES del trabajo realizado.

D.REVISIÓN LITERARIA

4.1. APLICACIONES MÓVILES

También denominadas app, es un software que se instala en los dispositivos móviles como Smartphones y tablets, con la finalidad de proporcionar al usuario la capacidad de realizar tareas de cualquier tipo, tales como: profesionales, de ocio, educativas, salud, acceso a servicios, etc, facilitando las actividades a desarrollar, ahorrando tiempo y mejorando el estilo de vida.

4.1.1. Funcionalidades de las Aplicaciones

Las funcionalidades que ofrecen las aplicaciones son amplias y diversas, cada día surgen nuevas aplicaciones móviles que satisfacen necesidades muy diferentes, dando solución a multitud de nichos de mercado o la inclusión de nuevos segmentos, que antes no tenían cabida en el entorno digital.

La súbita expansión de las aplicaciones ha sido posible gracias al avance en la tecnología móvil. Los dispositivos actuales incorporan novedosa tecnología, que no sólo permite el acceso a internet desde una aplicación, sino desde multitud de ellas, grabación de vídeo en HD, cámara de alta calidad, geo localización, aumento de la rapidez de ejecución de varias aplicaciones, procesadores de texto o alta resolución de las pantallas. Esto incurre en una amplia oferta para los desarrolladores de aplicaciones, o mejora de las ya existentes, lo que origina que continuamente esté actualizándose el mercado de aplicaciones [4].

4.1.2. Tipo de Aplicaciones Móviles

Existen varias categorías en las cuales se podrían clasificar a las aplicaciones móviles, entre las más importante tenemos:

4.1.2.1. Aplicaciones Nativas

Las aplicaciones nativas son aquellas que se construyen para ser utilizadas en un dispositivo concreto, es decir se deben considerar aspectos como: el tipo de dispositivo, el sistema operativo a utilizar y su versión. Además posee una gran ventaja sobre las aplicaciones web, porque tiene la capacidad de aprovechar el hardware y el software específico del dispositivo, es decir, tiene la posibilidad de interactuar con todas las

capacidades del dispositivo (cámara, GPS, acelerómetro, agenda, entre otras), haciendo que la ejecución sea rápida.

Sin embargo, esto lo convierte en un arma de doble filo: mientras que la tecnología móvil es muy popular, también es constantemente cambiante. Esto hace que la tarea de mantenerse al día, al ritmo de la tecnología conlleve a mayores costos para las actualizaciones y distribuciones de nuevas versiones, especialmente en múltiples plataformas [5] [3].

4.1.2.1. Aplicaciones Web

Las aplicaciones web accedidas desde dispositivos móviles son creadas para ser ejecutadas desde el navegador, al igual que una aplicación web tradicional. Son desarrolladas utilizando las tecnologías básicas como: HTML (define el texto e imágenes estáticas), CSS (añade y define el estilo) y JavaScript (define las interacciones y animaciones), es decir, la misma tecnología que la utilizada para crear sitios web.

Una de las principales ventajas, es que las aplicaciones no necesitan instalar ningún componente en particular, ni la aprobación de la empresa suministradora de aplicaciones para que sean publicadas y utilizadas. Únicamente requiere de acceso a internet.

Además, las actualizaciones son visualizadas directamente en el dispositivo del usuario, por lo cual, los cambios son efectuados sobre el servidor y están disponibles de inmediato. Por contrapartida, utilizar el navegador disminuye la velocidad de ejecución y por ende el rendimiento disminuiría. Finalmente este tipo de aplicaciones no pueden utilizar todos los elementos de hardware del dispositivo, como por ejemplo, cámara, GPS, entre otros [5] [3].

4.1.2.1. Aplicaciones Híbridas

Las aplicaciones híbridas combinan lo mejor de los dos tipos de aplicaciones anteriores, es decir, para su desarrollo utilizan tecnologías de las aplicaciones web estándar (HTML, CSS, JavaScript) con APIs que mantienen el acceso directo a los componentes hardware del dispositivo, dando como resultado disminuir significativamente el tiempo y coste en el desarrollo, reduciendo al mínimo el trabajo de codificación personalizada.

Para el usuario, una aplicación híbrida bien diseñada es muy similar a una aplicación nativa: se descarga de una aplicación tienda, almacenado en el dispositivo móvil, y puso en marcha como una aplicación nativa.

Entre las principales ventajas se puede distribuir el software a través de las tiendas de aplicaciones, la reutilización de código para múltiples plataformas y la posibilidad de utilizar las características propias que posee el hardware del dispositivo.

Una de las desventajas es que, al utilizar la misma interfaz para todas las plataformas, la apariencia de la aplicación no será como la de una aplicación nativa. Finalmente la ejecución será más lenta que la ejecución en una aplicación nativa [2] [6] [3].

4.2. METODOLOGÍAS ÁGILES PARA EL DESARROLLO DE SOFTWARE

La ingeniería de software ágil combina una filosofía con un conjunto de lineamientos de desarrollo. La filosofía pone el énfasis en: la satisfacción del cliente y en la entrega rápida de software incremental, los equipos pequeños y muy motivados para efectuar el proyecto, los métodos informales, los productos del trabajo con mínima ingeniería de software y la sencillez general en el desarrollo [7],

Se centra más en capturar mejor los requisitos cambiantes y la gestión de los riesgos, rompiendo el proyecto en iteraciones de diferente longitud, cada una de ellas generando un producto completo y entregable; e incremental donde un producto se construye bloque a bloque durante todo el ciclo de vida de desarrollo del producto, las iteraciones individuales deben producir alguna característica completamente funcional o mejorada [8].

4.2.1. Principios de las Metodologías Ágiles

La Alianza Ágil define 12 principios de agilidad para aquellos que la quieran alcanzar [9] [7]:

1. La prioridad más alta es satisfacer al cliente a través de la entrega pronta y continua de software valioso.
2. Son bienvenidos los requerimientos cambiantes, aun en una etapa avanzada del desarrollo.

Los procesos ágiles dominan el cambio para provecho de la ventaja competitiva del cliente.

3. Entregar con frecuencia software que funcione, de dos semanas a un par de meses, de preferencia lo más pronto que se pueda.
4. Las personas de negocios y los desarrolladores deben trabajar juntos, a diario y durante todo el proyecto.
5. Hay que desarrollar los proyectos con individuos motivados. Debe darse a éstos el ambiente y el apoyo que necesiten, y confiar en que harán el trabajo.
6. El método más eficiente y eficaz para transmitir información a los integrantes de un equipo de desarrollo, y entre éstos, es la conversación cara a cara.
7. La medida principal de avance es el software que funciona.
8. Los procesos ágiles promueven el desarrollo sostenible. Los patrocinadores, desarrolladores y usuarios deben poder mantener un ritmo constante en forma indefinida.
9. La atención continua a la excelencia técnica y el buen diseño mejora la agilidad.
10. Es esencial la simplicidad: el arte de maximizar la cantidad de trabajo no realizado.
11. Las mejores arquitecturas, requerimientos y diseños surgen de los equipos con organización propia.
12. El equipo reflexiona a intervalos regulares sobre cómo ser más eficaz, para después afinar y ajustar su comportamiento en consecuencia.

4.2.2. Tipos de Metodologías Ágiles

4.2.2.1. Programación Extrema (XP)

Es una metodología ágil para pequeños y medianos equipos, desarrollando software cuando los requerimientos son ambiguos o rápidamente cambiantes, se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico [10].

El ciclo de entrega de XP se muestra en la figura 1, y todos los requerimientos se representan como escenarios (Historias de Usuario), los cuales se implementan directamente como una serie de tareas, con pequeños grupos de trabajo, en ambientes

de requerimientos imprecisos o cambiantes. XP tiene como base cinco valores: Simplicidad, Comunicación, Retroalimentación, Respeto y Coraje [11] [12].



Figura 1: Ciclo de entrega de la Programación Extrema [12]

4.2.2.1.1. Fases

Fase I Planeación: En esta fase, los clientes plantean a grandes rasgos las características y funcionalidades del software, El cliente asigna un valor o prioridad a las historias de usuario que son de interés para la primera entrega del producto. Al mismo tiempo el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. Se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema construyendo un prototipo [7] [10] [13] [14].

Fase II Diseño: El diseño XP sigue rigurosamente el principio MS (mantenlo sencillo), un diseño sencillo siempre se prefiere sobre una representación más compleja. Además, el diseño guía la implementación de una historia conforme se escribe. Se recomienda elaborar un glosario de términos y la correcta especificación de métodos y clases para facilitar posteriores modificaciones, ampliaciones o reutilización del código.

XP estimula el uso de las tarjetas CRC como un mecanismo eficaz para pensar en el software en un contexto orientado a objetos. Las tarjetas CRC (clase-responsabilidad-colaborador) identifican y organizan las clases orientadas a objetos que son relevantes para el incremento actual de software [7] [13] [14].

Fase III Codificación: En ésta fase los desarrolladores deben diseñar las pruebas de unidad que ejercitarán cada historia de usuario. Después de tener las pruebas, los desarrolladores trabajarán para concentrarse en lo que debe implementarse para pasar la prueba de unidad [13] [14].

Fase IV Pruebas: Las pruebas de unidad deben implementarse con un marco de trabajo que permita automatizarlas, con la finalidad de realizar pruebas de integración y validación diarias, esto proporcionará al equipo un indicador del progreso y revelarán a tiempo si existe alguna falla en el sistema, poseen las siguientes características [7] [13] [14]:

- **Desarrollo previamente aprobado:** Significa que primero se escriben las pruebas y luego el código. Las pruebas deben simular el envío de la entrada a probar y deben verificar que el resultado cumpla con las especificaciones de salida.
- **Desarrollo de pruebas incremental:** Los requerimientos del usuario se expresan como historias, el equipo de desarrollo evalúa cada historia y la divide en tareas. Cada una representa una característica distinta del sistema y se pueden diseñar pruebas de unidad para esa tarea.
- **Participación del usuario en el desarrollo de las pruebas:** El usuario ayuda a desarrollar las pruebas de aceptación, las cuales son pruebas que se implementan con los datos reales del cliente para verificar el cumplimiento real de sus necesidades.
- **Uso de bancos de pruebas automatizados:** Se debe usar un sistema que envíe a ejecución las pruebas automatizadas y de esta forma probar constantemente el sistema software.

4.2.2.1.2. Roles

Los roles más importantes en XP son los siguientes [13] [15]:

- **Cliente:** El cliente dirige y conoce las metas a alcanzar en el proyecto, define el proyecto y sus objetivos. Cuanto más preciso es su trabajo y cuanto mayor sea su involucración, mayores serán las oportunidades de éxito.
- **Programador:** Es el corazón de XP, el programador con base en su experiencia puede tomar decisiones que afecten el desarrollo del proyecto. Una vez que se han comprendido las historias de usuario, los desarrolladores estiman el tiempo que les va a tomar cada historia.
- **Entrenador (Coach):** Su papel es guiar y orientar al equipo, especialmente cuando un equipo comienza a trabajar siguiendo la metodología XP. Esto se debe a que no es fácil aplicar XP de forma consistente.

- **Gestor (Big Boss):** Es el gerente del proyecto, debe tener una idea general del proyecto y estar familiarizado con su estado. El cliente puede asumir este papel.

4.2.2.1.3. Principios

- **Rápida retroalimentación:** Tener una rápida retroalimentación nos permite interpretarla, aprender de ella y poner en práctica lo asimilado lo antes posible. Asumir la simplicidad [13] [12].
- **Asumir la simplicidad:** Casi siempre se planifica para el futuro y se desea para poder rehusar. En lugar de esto XP dice que hay que hacer un buen trabajo para las necesidades actuales y confiar en nuestra habilidad para solucionar problemas futuros [13] [12].
- **Cambios incrementales:** Realizar grandes cambios en una sola oportunidad no es una buena solución. Cada problema debe ser resuelto con una serie de cambios pequeños para poder atacar dicho problema mucho más en profundidad [13] [12].
- **Trabajo de calidad:** Uno de los objetivos más importantes en XP es realizar un producto de buena calidad. Si cada integrante realiza su trabajo de la mejor manera posible se puede asegurar la calidad del producto [13] [12].

4.2.2.1. SCRUM

SCRUM es un marco de trabajo basado en los métodos ágiles, que tiene como objetivo el control continuo sobre el estado actual del software [13], en el cual el cliente establece las prioridades y el equipo SCRUM se auto-organiza para determinar la mejor forma de entregar resultados, está formado por un conjunto de prácticas y reglas que resultan válidos [16].

Los principios Scrum son congruentes con el manifiesto ágil y se utilizan para guiar actividades de desarrollo dentro de un proceso de análisis que incorpora las siguientes actividades estructurales: requerimientos, análisis, diseño, evolución y entrega. Dentro de cada actividad estructural, las tareas del trabajo ocurren con un patrón del proceso llamado sprint. El trabajo realizado dentro de un sprint (el número de éstos que requiere cada actividad estructural variará en función de la complejidad y tamaño del producto)

se adapta al problema en cuestión y se define —y con frecuencia se modifica— en tiempo real por parte del equipo Scrum. El flujo general del proceso se ilustra en la figura 2.

Scrum acentúa el uso de un conjunto de patrones de proceso del software que han demostrado ser eficaces para proyectos con plazos de entrega muy apretados, requerimientos cambiantes y negocios críticos [7].

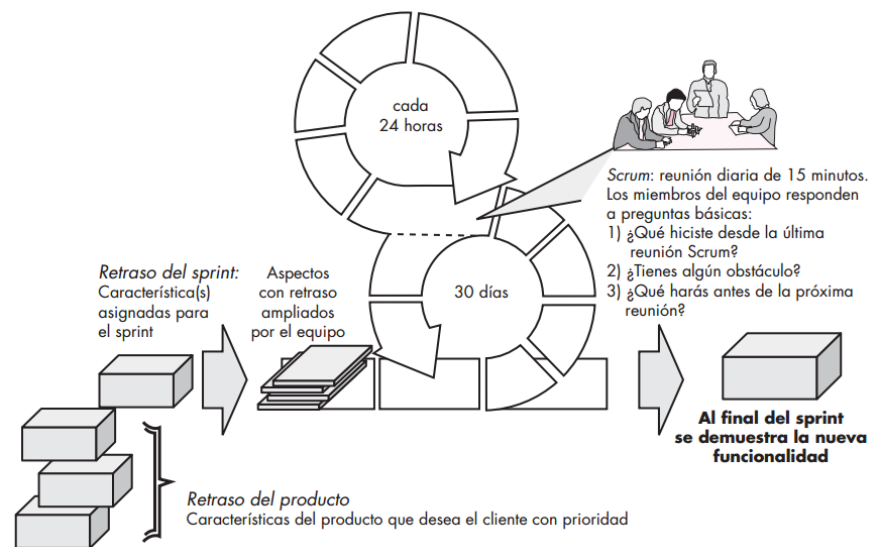


Figura 2: Flujo del proceso Scrum [7]

4.2.2.1.1. Características

- **Revisión de las Iteraciones:** Cada ciclo de desarrollo o iteración (sprint) finaliza con la entrega de una parte operativa del producto. La duración de cada sprint puede ser desde una, hasta seis semanas, aunque se recomienda que no excedan de un mes [16].
- **Desarrollo incremental:** El desarrollo incremental ofrece al final de cada iteración una parte de producto operativa, que se puede usar, inspeccionar y evaluar, Scrum resulta adecuado en proyectos con requisitos inciertos y, o inestable [16].
- **Auto organización:** Son muchos los factores impredecibles en un proyecto. La gestión predictiva asigna al rol de gestor del proyecto la responsabilidad de su gestión y resolución [16].

- **Colaboración:** Es un componente importante y necesario para que a través de la auto organización se pueda gestionar con solvencia la labor que de otra forma realizaría un gestor de proyectos. Todos los miembros del equipo colaboran de forma abierta con los demás, según sus capacidades y no según su rol o su puesto [16].

4.2.2.1.2. Artefactos

Scrum dispone de los siguientes artefactos [16] [17]:

- **Lista del producto:** (product backlog) lista de requisitos de usuario, que a partir de la visión inicial del producto crece y evoluciona durante el desarrollo.
- **Lista de Pendientes del Sprint:** (sprint backlog) lista de los trabajos que debe realizar el equipo durante el sprint para generar el incremento previsto.
- **Sprint:** nombre que recibe cada iteración de desarrollo. Es el núcleo central que genera el pulso de avance por tiempos prefijados (time boxing).
- **Incremento:** resultado de cada sprint.

4.2.2.1.3. Roles

Los roles más importantes de la metodología Scrum son los siguientes [16] [17]:

- **Propietario del producto:** El propietario del producto debe ser una persona con visión, la autoridad y la disponibilidad. El propietario del producto es responsable de comunicar continuamente la visión y las prioridades para el equipo de desarrollo.
- **Scrum Master:** Actúa como un facilitador para el propietario del producto y el equipo, su función es remover los obstáculos que obstruyen el equipo logre sus objetivos de sprint.
- **Equipo:** Un equipo de desarrollo Scrum contiene cerca de siete miembros plenamente dedicados (oficialmente, 3-9). Para proyectos de software, un equipo típico incluye una mezcla de los ingenieros de software, arquitectos, programadores, analistas, expertos en control de calidad, los probadores, y diseñadores de interfaz de usuario.

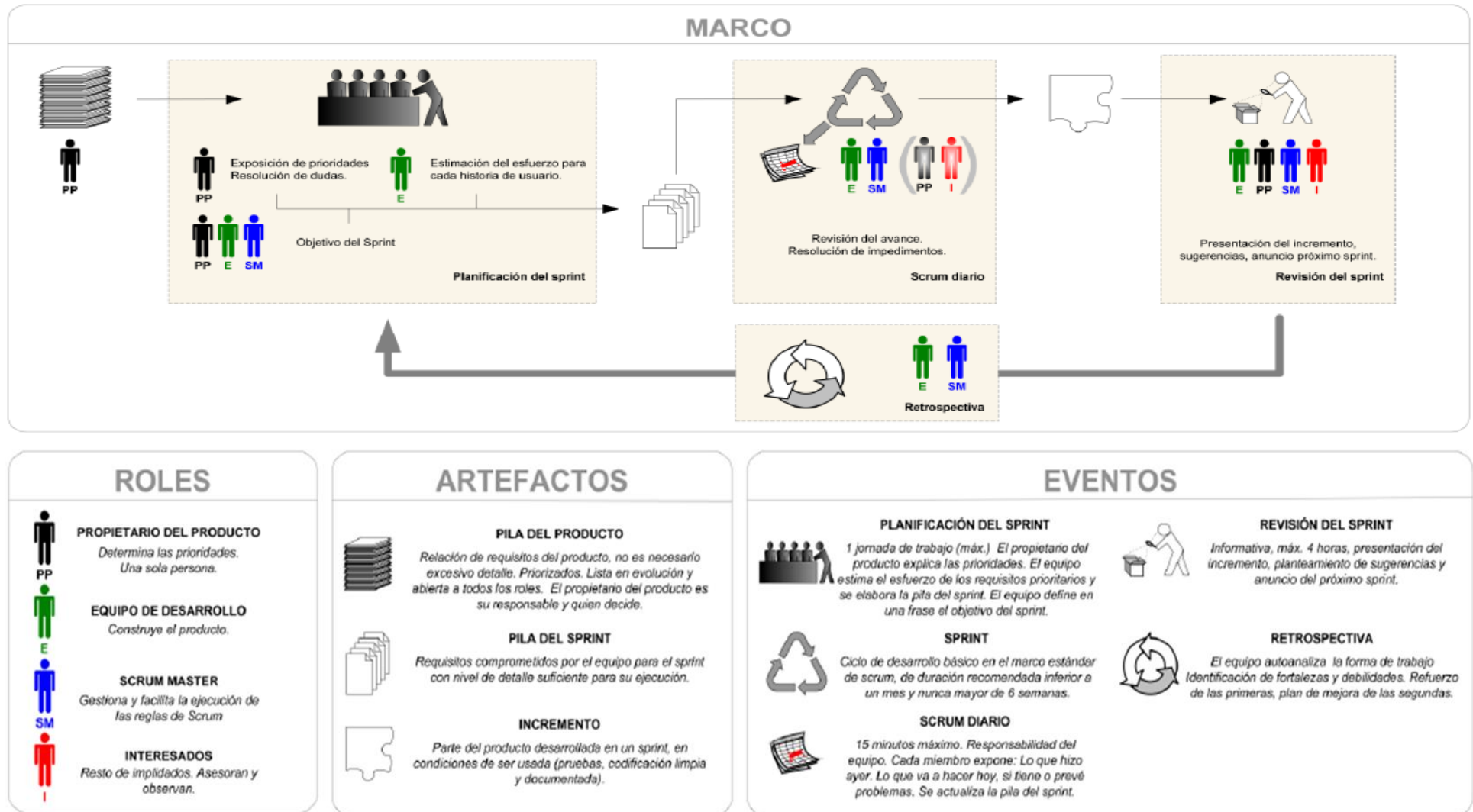


Figura 3: Reglas de SCRUM [16]

4.2.2.2. ICONIX

ICONIX se define como un “proceso” de desarrollo de software práctico. ICONIX está entre la complejidad del RUP (Rational Unified Processes) y la simplicidad y pragmatismo del XP (Extreme Programming), sin eliminar las tareas de análisis y de diseño que XP no contempla.

ICONIX es un proceso simplificado de análisis y diseño basada en casos de uso. Su enfoque principal es cómo obtener de manera fiable casos de uso para codificar en tan pocos pasos como sea posible, en comparación con otros procesos más tradicionales, que unifican un conjunto de métodos de orientación a objetos con el objetivo de abarcar todo el ciclo de vida de un proyecto. Fue elaborado por Doug Rosenberg y Kendall Scott a partir de una síntesis del proceso unificado de los “tres amigos” Booch, Rumbaugh y Jacobson y que ha dado soporte y conocimiento a la metodología ICONIX desde 1993. Presenta claramente las actividades de cada fase y exhibe una secuencia de pasos que deben ser seguidos [18] [19].

4.2.2.2.1. Características

Se caracteriza por [18]:

- **Iterativo e incremental:** varias iteraciones ocurren en el desarrollo del modelo del dominio y de la identificación de los casos de uso. El modelo estático es incremental refinado por los dos modelos dinámicos.
- **Trazabilidad:** Cada paso está referenciado por algún requisito. Siguiendo una relación entre los diferentes artefactos del producto.
- **Dinámica del UML:** Uso dinámico de UML en los diagramas de caso de uso, diagramas de secuencia y de colaboración.

4.2.2.2.2. Fases

- **Análisis de Requisitos:** En esta primera fase se realiza un Modelo de Dominio, que no es más que un Diagrama de Clases extremadamente simplificado. Este modelo contiene únicamente aquellos objetos de la vida real cuyo comportamiento o datos deban ser almacenados en el sistema [18].
- **Análisis y Diseño Preliminar:** A partir de cada caso de uso se obtienen sus correspondientes fichas de caso de uso, después será necesario realizar lo que

se conoce como Diagrama de Robustez. Cabe destacar que estas fichas no pertenecen al UML [18].

- **Diseño:** En esta fase se proceden a realizar los diagramas de secuencia, los cuales derivan directamente de las fichas de caso de uso. Esto implica que una vez finalizado el diseño, tras refinar nuevamente el diagrama de clases, podremos verificarlo directamente gracias a este factor de trazabilidad, y prepararnos para la siguiente fase [18].
- **Implementación:** De cara a poder distribuir el software correctamente, puede ser adecuado realizar un diagrama de componentes en algunos casos, pero no siempre es necesario. En cualquier caso, aquí es donde se escribe el código tal y como fue especificado en las fases anteriores y se planean las pruebas basándonos en los requisitos iniciales, al nivel que fuese necesario [18].

4.2.2.3. Mobile-D

Mobile-D TM es la metodología para el desarrollo ágil de software. Además del desarrollo de software para móviles, es conveniente para los varios contextos, por ejemplo, la seguridad, financieros, logística y aplicaciones de simulación de productos [20].

Consta de cinco fases: exploración, iniciación, producción, estabilización y prueba del sistema. Cada una de estas fases tiene un número de etapas, tareas y prácticas asociadas.

Las especificaciones completas del método están disponibles en la primera fase, Explorar, el equipo de desarrollo debe generar un plan y establecer las características del proyecto. Esto se realiza en tres etapas: establecimiento actores, definición del alcance y el establecimiento de proyectos. Las tareas asociadas a esta fase incluyen el establecimiento del cliente (los clientes que toman parte activa en el proceso de desarrollo), la planificación inicial del proyecto y los requisitos de recogida, y el establecimiento de procesos [20].

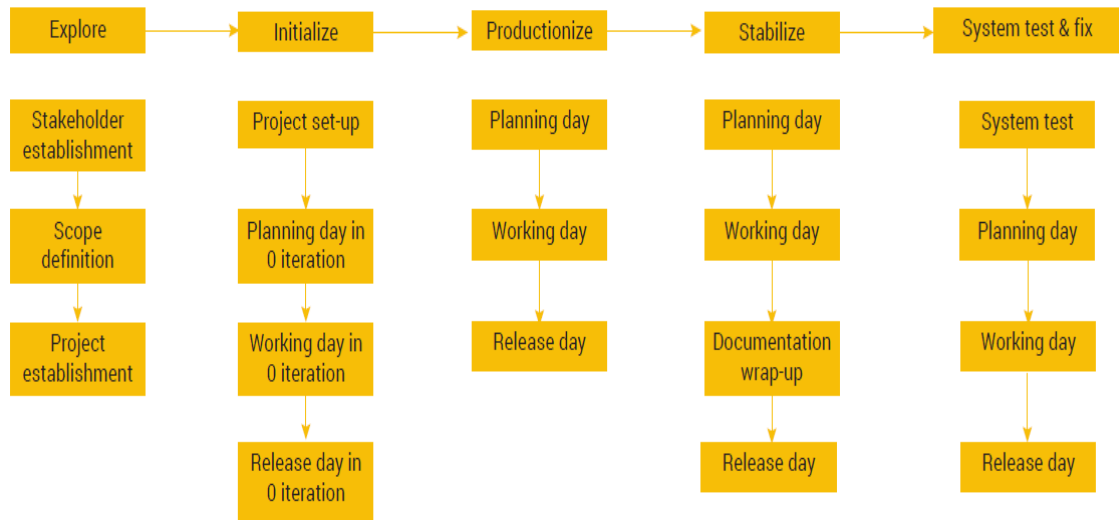


Figura 4: Ciclo de desarrollo Mobile-D, Tomada de [20]

En la siguiente fase, iniciación, los desarrolladores preparan e identifican todos los recursos necesarios. Se preparan los planes para las siguientes fases y se establece el entorno técnico como los recursos físicos, tecnológicos y de comunicaciones (incluyendo el entrenamiento del equipo de desarrollo). Esta fase se divide en cuatro etapas: la puesta en marcha del proyecto, la planificación inicial, el día de prueba y día de salida [20].

En la fase de producción se repite la programación de tres días (planificación, trabajo, liberación) se repite iterativamente hasta implementar todas las funcionalidades. Primero se planifica la iteración de trabajo en términos de requisitos y tareas a realizar. Se preparan las pruebas de la iteración de antemano. Las tareas se llevarán a cabo durante el día de trabajo, desarrollando e integrando el código con los repositorios existentes. Durante el último día se lleva a cabo la integración del sistema (en caso de que estuvieran trabajando varios equipos de forma independiente) seguida de las pruebas de aceptación [20].

En la fase de estabilización, se llevan a cabo las últimas acciones de integración para asegurar que el sistema completo funciona correctamente. Esta será la fase más importante en los proyecto multi-equipo con diferentes subsistemas desarrollados por equipos distintos. En esta fase, los desarrolladores realizarán tareas similares a las que debían desplegar en la fase de “producción”, aunque en este caso todo el esfuerzo se

dirige a la integración del sistema. Adicionalmente se puede considerar en esta fase la producción de documentación [20].

La última fase (prueba y reparación del sistema) tiene como meta la disponibilidad de una versión estable y plenamente funcional del sistema. El producto terminado e integrado se prueba con los requisitos de cliente y se eliminan todos los defectos encontrados [20].

4.3. FRAMEWORKS PARA EL DESARROLLO DE APLICACIONES MÓVILES

Con el propósito de normalizar y estructurar el código del sistema, facilitando un esquema (un patrón, un esqueleto) para el desarrollo y/o la implementación de aplicaciones. El uso de frameworks para cualquier tipo de desarrollo reduce el tiempo de elaboración e implementación y ayuda a hacer un trabajo mantenible y escalable, según las características del mismo [21].

Un framework agrega funcionalidad extendida a un lenguaje de programación, automatiza muchos de los patrones de programación para orientarlos a un determinado propósito, proporcionando una estructura al código, mejorándolo y haciéndolo más entendible y sostenible, y permite separar en capas la aplicación. En general, divide la aplicación en tres capas [22]:

- La lógica de presentación que administra las interacciones entre el usuario y el software.
- La Lógica de datos que permite el acceso a un agente de almacenamiento persistente u otros.
- La lógica de dominio o de negocio, que manipula los modelos de datos de acuerdo a los comandos recibidos desde la presentación.

4.3.1. Ionic

Ionic es un framework propiedad de la empresa Drifty. Es un marco de desarrollo en HTML5 dirigida a la creación de aplicaciones móviles híbridas, estas son esencialmente pequeños sitios web que se ejecutan en un navegador que tienen acceso a la capa de plataforma nativa. Las aplicaciones híbridas tienen muchas ventajas sobre las

aplicaciones nativas puras, específicamente en términos de soporte de la plataforma, la velocidad de desarrollo, y el acceso al código [23] [24].

Además Ionic nos va a aportar un ecosistema de herramientas que en la actualidad son completamente gratuitas y que nos van a facilitar en gran medida las labores de desarrollo. Entre ellas encontramos [23]:

- **Ionic.io Platform:** Conjunto de herramientas para gestionar nuestra aplicación, entre ellas las que nos serían más útiles para nuestro proyecto encontramos:
 - **Ionic Creator:** herramienta de desarrollo grafica que nos permite realizar las primeras fases de implementación de la aplicación de una forma gráfica, arrastrando y soltando componentes y creando páginas de forma automática.
 - **Ionic Deploy:** Ha aparecido recientemente y es la solución back-end de ionic que nos va a permitir realizar actualizaciones en nuestra aplicación.
 - **Ionic Analytics:** Servicio de analíticas para nuestra aplicación.
 - **Ionic View App:** Aplicación móvil para realizar pruebas en terminales.
 - **Ionic Lab:** Herramienta para testear nuestra aplicación en múltiples tamaños de pantalla y sistemas operativos.

Ionic funciona sobre Angular JS. Angular JS es un framework de JavaScript de código abierto propiedad de Google que proporciona a los desarrolladores web la posibilidad de escribir aplicaciones completas de forma rápida y proporciona una buena estructura de la aplicación.

4.3.1.1. Características:

Ionic Framework presenta las siguientes características [24]:

- **Alto rendimiento:** la velocidad es importante; tan importante que sólo se nota cuando no está en tu app. Ionic está construido para ser rápido, gracias a la mínima manipulación del DOM, con cero jQuery y con aceleraciones de transiciones por hardware.
- **AngularJS & Ionic:** Utiliza AngularJS con el fin de crear un marco más adecuado para desarrollar aplicaciones ricas y robustas. Ionic no sólo se ve bien, sino que su arquitectura central es robusta y seria para el desarrollo de aplicaciones.

- **Centro nativo Ionic:** se inspira en las SDK de desarrollo móviles nativos más populares, por lo que es fácil de entender para cualquier persona que ha construido una aplicación nativa para iOS o Android.
- Se desarrolla una vez, y se compila para varios.
- **Bonito diseño:** limpio, sencillo y funcional. Ionic ha sido diseñado para poder trabajar con todos los dispositivos móviles actuales. Cuenta con muchos componentes usados en móviles, tipografía, elementos interactivos, etc.
- **Un potente CLI:** con un sólo comando podrás crear, construir, probar y compilar tus aplicaciones en cualquier plataforma.
- **Ionic trabaja bajo Cordova,** por lo que podremos utilizar todos los plugins que estén desarrollados para el framework e incluso desarrollar los nuestros propios. En esta url, se pueden ver todos los que actualmente están en el repositorio de Apache Cordova: <https://cordova.apache.org/plugins/> Tan solo hay que añadirlos mediante línea de comando, tal como se instalan con Cordova, en caso de compilar de forma local.
- La licencia Ionic es de código abierto, publicada bajo una licencia MIT. Esto significa que se puede utilizar en nuestros proyectos personales o comerciales, de forma gratuita. MIT es la misma licencia que utiliza otros proyectos populares como jQuery y Ruby on Rails

4.3.2. Phonegap

PhoneGap se desarrolló originalmente por la empresa Nitobi en 2011 y fue adquirido posteriormente por Adobe, liberando el código bajo el proyecto Apache Cordova. En otras palabras, el producto que pertenece a Adobe se denomina actualmente PhoneGap y el proyecto de código abierto se denomina Cordova. Varios de los framework descritos en este documento utilizan Cordova. El objetivo de este framework es permitir desarrollar aplicaciones embebidas en código nativo utilizando exclusivamente código HTML5 y Javascript. Phone Gap Build ofrece varios tipos de plan pagos. Debido a que el código se compila a través de gitHub, el derecho a que el código sea privado depende del precio del plan. La versión gratuita permite una sola aplicación privada y la paga va desde USD \$120 a USD \$900 por año, dependiendo del número de aplicaciones [25]

4.3.2.1. Prestaciones

Phonegap brinda las siguientes prestaciones [26] [27] [28] [29]:

- Soporta las plataformas Android, iOS, BlackBerry OS, Windows Phone, WebOS, Symbian y Bada.
- Está disponible en formato de plugin para diferentes IDE's como Eclipse o Xcode. No obstante, también ofrece la opción PhoneGap Build, un servicio gratuito en la "nube".
- Proporciona la API de JavaScript que permite tener acceso a las capacidades del dispositivo como cámara, GPS, información del dispositivo, etc. Además, posee un componente nativo, el cual trabaja de forma independiente y hace el trabajo en el dispositivo.
- Permite integrarse con frameworks de desarrollo web móvil como JQuery Mobile y Sencha Touch.
- El framework está disponible para ser instalado en las tres plataformas principales que son Windows, Linux y Mac.

4.3.2.2. Ventajas

- Utiliza un lenguaje de programación fácil de aprender y de poca complejidad.
- No requiere instalación de aplicaciones complementarias
- El framework es de código abierto.
- Acceso a elementos y hardware del dispositivo mediante JavaScript, como cámara, GPS, acelerómetro, etc.

4.3.2.3. Desventajas

- No cuenta con su propio IDE.
- Acceso limitado a accesorios y hardware del dispositivo.
- Al no programarse en un lenguaje nativo del sistema operativo móvil, el rendimiento es afectado en un mínimo de porcentaje.

4.3.3. Titanium

Titanium es un framework creado por la empresa Appcelerator Inc. en el año 2008, el cual permite construir aplicaciones móviles multiplataforma mediante un único código escrito en JavaScript. Está disponible bajo la licencia Apache 2.0, por lo cual es Open

Source y gratuito para uso personal y comercial. Su oferta se completa con una gran cartera de servicios en la nube [30].

4.3.3.1. Prestaciones

Titanium brinda las siguientes prestaciones [30]:

- Permite crear aplicaciones nativas, para ello toda la aplicación debe ser programada íntegramente con JavaScript donde todos los controles (botones, listas, menús y otros) deben ser creados “a mano”, ya que cuenta con una librería JavaScript que permite acceder a los controles del sistema. Esto significa que todos los controles son nativos lo que hace más rápido renderizar y dar una respuesta al usuario.
- Permite tener un mejor rendimiento por el hecho de que el código JavaScript esta compilado y que los controles creados son nativos.
- El desarrollo de las aplicaciones multiplataforma es posible debido que se trabaja con un código común de base, el cual puede ser compilado a las diversas plataformas que el framework soporta (iOS, Android y BlackBerry).
- Al trabajar con un lenguaje común como es JavaScript se reduce los costes y el proceso de desarrollo.

4.3.3.2. Ventajas

- Es software libre y de código abierto, bajo licencia Apache 2.0.
- Basado únicamente en el lenguaje Javascript para todas las plataformas, un lenguaje muy extendido, conocido, documentado, de fácil uso y aprendizaje.
- Incluye un entorno de desarrollo basado en Eclipse.
- La API ofrece componentes independientes de la plataforma, pero también componentes específicos de cada plataforma. Por ejemplo, para la plataforma Android, proporciona acceso a Intents.

4.3.3.3. Desventajas

- Únicamente se puede desarrollar para las plataformas de Apple y Android.
- Su uso requiere una curva de aprendizaje bastante alargada.

4.4. FRAMEWORK PARA EL DESARROLLO DE APLICACIONES WEB

4.4.1. Codeigniter

Codeigniter es un framework para desarrollo de aplicaciones - un conjunto de herramientas - para gente que construye sitios web usando PHP. Su objetivo es permitirle desarrollar proyectos mucho más rápido que lo que podría hacer si escribiera el código desde cero, proveyéndole un rico conjunto de bibliotecas para tareas comunes, así como una interfaz sencilla y una estructura lógica para acceder a esas bibliotecas. Codeigniter le permite enfocarse creativamente en su proyecto al minimizar la cantidad de código necesaria para una tarea dada [31].

4.4.1.1. Características

Este framework se destaca por tener las siguientes características [31]:

- **Es Libre:** Codeigniter está liberado bajo licencias open source del estilo Apache/BSD, así que puede usarlo donde desee.
- **Es Liviano:** El núcleo del sistema sólo requiere algunas bibliotecas muy pequeñas. Esto está en marcado contraste con muchos frameworks que requieren muchos más recursos. Las bibliotecas adicionales se cargan dinámicamente bajo pedido, basado en sus necesidades para un proceso dado, por lo que el sistema base es muy ligero y bastante rápido.
- **Es Rápido:** Lo desafiamos a que encuentre un framework que tenga mejor desempeño que Codeigniter.
- **Usa M-V-C:** Codeigniter usa el enfoque Modelo-Vista-Controlador, que permite una gran separación entre la lógica y la presentación. Es particularmente bueno para proyectos en los que los diseñadores trabajan en sus archivos de plantillas, ya que el código en estos archivos será mínimo.
- **Genera URLs Claras:** Las URLs generadas por Codeigniter son claras y amigables con los motores de búsqueda. En lugar de usar el enfoque estándar "query string" característico de sistemas dinámicos, Codeigniter usa el enfoque basado en segmentos.

- **Es Extensible:** El sistema se puede extender fácilmente a través de sus propias bibliotecas, helpers, extensiones de clases o sistema de hooks.
- **Completamente Documentado:** Los programadores aman programar y odian escribir documentación. No somos diferentes, por supuesto, pero como la documentación es tan importante como el código en sí mismo, estamos comprometidos a hacerlo. Nuestro código fuente es extremadamente claro y bien comentado también.

4.4.1.2. Modelo-Vista-Controlador En Codeigniter

El Modelo, Vista, Controlador es típicamente utilizado para la creación de aplicaciones web y no sólo CodeIgniter lo implementa, sino también otra serie de frameworks de desarrollo web, en PHP u otros lenguajes. Es interesante porque separa en varios grupos las complejidades de las distintas partes que componen una página web, como la vista y la lógica, así como el acceso a la base de datos.

Quizás lo que más nos fuerce a cambiar nuestros hábitos de programación en PHP es el hecho de tener que basar nuestros scripts en este modelo de programación, porque fija un nuevo estilo de desarrollo de aplicaciones, que nos obliga a separar código fuente según su ámbito. Sin embargo, como decíamos anteriormente, estas nuevas costumbres de codificación también nos ayudarán a que nuestros programas sean mejores y disfruten de varias ventajas como ser más organizados, extendibles y entendibles por otros desarrolladores, reutilizables, con más fácil mantenimiento, etc.

Para los que no lo conocen, el Modelo - Vista - Controlador (en inglés Model - View - Controller) es un patrón de desarrollo o un estilo de arquitectura de software que separa el código fuente de las aplicaciones en tres grupos [32].

4.4.1.3. Modelo

Todo el código que tiene que ver con el acceso a base de datos. En el modelo mantendremos encapsulada la complejidad de nuestra base de datos y simplemente crearemos funciones para recibir, insertar, actualizar o borrar información de nuestras tablas. Al mantenerse todas las llamadas a la base de datos en un mismo código, desde otras partes del programa podremos invocar las funciones que necesitemos del modelo y éste se encargará de procesarlas. En el modelo nos podrán preocupar cosas como el tipo de base de datos con la que trabajamos, o las tablas y sus relaciones, pero desde las otras partes del programa simplemente llamaremos a las funciones del modelo sin

importarnos qué tiene que hacer éste para conseguir realizar las acciones invocadas [32].

4.4.1.4. Vista

La vista codifica y mantiene la presentación final de nuestra aplicación de cara al usuario. Es decir, en la vista colocaremos todo el código HTML, CSS, Javascript, etc. que se tiene que generar para producir la página tal cual queremos que la vea el usuario. En la práctica la vista no sólo sirve para producir páginas web, sino también cualquier otra salida que queramos enviar al usuario, en formatos o lenguajes distintos, como pueden ser feeds RSS, archivos JSON, XML, etc [32].

4.4.1.5. Controlador

El controlador podríamos decir que es la parte más importante, porque hace de enlace entre el modelo, la vista y cualquier otro recurso que se tenga que procesar en el servidor para generar la página web. En resumen, en el controlador guardamos la lógica de nuestras páginas y realizamos todas las acciones que sean necesarias para generarlas, ayudados del modelo o la vista [32].

4.5. SERVICIOS WEB

Los Servicios WEB son componentes de software reutilizables, modulares que pueden ser descritas, localizadas, e invocadas sobre una red que semánticamente encapsulan funcionalidades discretas que son distribuidas y accesibles a nivel de programación a través de protocolos de un intercambio estandarizado de mensajes, definidos por estándares como SOAP, WSDL, y UDDI [33] [34] [35].

Objetivo.-Su objetivo es proporcionar interoperatividad sobre una plataforma Internet entre toda clase de aplicaciones y sistemas que se integran de forma transparente siempre que se acojan a los estándares propuestos [34].

4.5.1. REST

La Transferencia de Estado Representacional (Representational State Transfer) o REST es una técnica de arquitectura software para sistemas hipermedia distribuidos como la World Wide Web. El término se originó en el año 2000, en una tesis doctoral sobre la web escrita por Roy Fielding, uno de los principales autores de la especificación del protocolo HTTP y ha pasado a ser ampliamente utilizado por la comunidad de desarrollo.

REST afirma que la web ha disfrutado de escalabilidad como resultado de una serie de diseños fundamentales clave [36] [37]:

- Un protocolo cliente/servidor sin estado: cada mensaje HTTP contiene toda la información necesaria para comprender la petición. Como resultado, ni el cliente ni el servidor necesitan recordar ningún estado de las comunicaciones entre mensajes. Sin embargo, en la práctica, muchas aplicaciones basadas en HTTP utilizan cookies y otros mecanismos para mantener el estado de la sesión.
- Un conjunto de operaciones bien definidas que se aplican a todos los recursos de información: HTTP en sí define un conjunto pequeño de operaciones, las más importantes son POST, GET, PUT y DELETE. Con frecuencia estas operaciones se equiparan a las operaciones CRUD que se requieren para la persistencia de datos, aunque POST no encaja exactamente en este esquema.
- Una sintaxis universal para identificar los recursos. En un sistema REST, cada recurso es direccionable únicamente a través de su URI.

Objetivos buscados por REST son los siguientes [36]:

- Escalabilidad en las interacciones entre componentes: La red ha ido creciendo de manera exponencial y se ha ido comportado de manera satisfactoria.
- Generalidad en las interfaces: Se proporciona acceso con distintos clientes y con distintos mecanismos de acceso.
- Desarrollo independiente de componentes: Implementaciones cliente y servidor pueden ser desarrolladas en distintos momentos.
- Existencia de componentes intermediarios con los cuales existe compatibilidad (como proxys) que permiten encapsulación e integración de sistemas no Web dentro de la misma.

Cabe destacar que REST no es un estándar, ya que es tan solo un estilo de arquitectura. Aunque REST no es un estándar, está basado en estándares [36]:

- HTTP
- URL
- Representación de los recursos: XML/HTML/GIF/JPEG/JSON...
- Tipos MIME: text/xml, text/html, text/json

4.6. HERRAMIENTAS PARA EL DESARROLLO DE APLICACIONES

Las tecnologías más importantes para el desarrollo de las aplicaciones son las siguientes:

4.6.1. Node Js

Node.js es un entorno de ejecución para JavaScript construido con el motor de JavaScript V8 de Chrome. Node.js usa un modelo de operaciones E/S sin bloqueo y orientado a eventos, que lo hace liviano y eficiente. El ecosistema de paquetes de Node.js, npm, es el ecosistema más grande de librerías de código abierto en el mundo.

4.6.2. Angular Ms

Desarrollado por Google este framework permite desarrollar de manera rápida el lado del cliente de las aplicaciones web. Uno de sus propósitos es extender la sintaxis HTML, para hacerla más funcional y ofrecer contenidos dinámicos mejorados.

AngularJS cuenta con múltiples características que facilitan el desarrollo, permitiendo crear aplicaciones web altamente escalables pero, que a la vez ofrecen una excelente experiencia de usuario [38].

AngularJS es un conjunto de herramientas para la construcción del marco más adecuado para el desarrollo de aplicaciones. Es totalmente extensible y funciona bien con otras bibliotecas. Cada función puede ser modificada o reemplazada para satisfacer sus necesidades de flujo de trabajo de desarrollo y de características únicas. Siga leyendo para averiguar cómo hacerlo [39].

4.6.3. Córdova

Las aplicaciones híbridas, pueden contar con partes de código escritos en el lenguaje nativo del sistema operativo móvil, llamados plugins, que en consecuencia se ejecutan en la plataforma nativa del móvil. Estos plugins, además de posibilitar el acceso a las características propias del móvil, también pueden ser útiles para mejorar el rendimiento de ciertas aplicaciones híbridas. Para su desarrollo, se necesita un framework que haga posible la comunicación entre las tecnologías web, con la plataforma nativa del dispositivo [38].

Estas tareas las desempeña Apache Cordova es un marco de desarrollo móvil de código abierto. Permite utilizar las tecnologías estándar web como HTML5, CSS3 y JavaScript para desarrollo multiplataforma, evitando el lenguaje de desarrollo nativo cada una de las plataformas móviles. Las aplicaciones ejecutan dentro de envolturas para cada plataforma y dependen de enlaces estándares (API) para acceder a de cada dispositivo sensores, datos y estado de la red [40].

4.6.4. Bower

Hacer un seguimiento de todos los paquetes y asegurarse de que están actualizados es complicado.

Bower puede administrar los componentes que contienen HTML, CSS, JavaScript e incluso archivos de imagen, sólo instala las versiones correctas de los paquetes que necesita y sus dependencias.

Bower realiza un seguimiento de estos paquetes en un archivo de manifiesto, bower.json. La forma de utilizar los paquetes depende de usted, proporciona ganchos para facilitar el uso de paquetes en sus herramientas y flujos de trabajo.

Bower está optimizado para el front-end. Si varios paquetes dependen de un paquete - jQuery por ejemplo - Bower descargará jQuery sólo una vez. Esto se conoce como un grafo de dependencias plana y ayuda a reducir la carga de la página [41].

4.6.4.1. Instalar Bower

Bower es una utilidad de línea de comandos. Instalarlo con la NPM.

```
$ npm install -g bower
```

4.6.5. PHP

PHP, acrónimo de "PHP: Hypertext Preprocessor", es un lenguaje de 'scripting' de propósito general y de código abierto que está especialmente pensado para el desarrollo web y que puede ser embebido en páginas HTML. Por lo tanto se puede comparar con otros lenguajes script que se ejecutan según el mismo principio: ASP (Active Server Pages), JSP (Java Server Pages) o PL/SQL Server Pages (PSP). Su sintaxis recurre a C, Java y Perl, siendo así sencillo de aprender. El objetivo principal de este lenguaje es permitir a los desarrolladores web escribir dinámica y rápidamente páginas web generadas; aunque se puede hacer mucho más con PHP [42] [43].

4.6.5.1. Funciones de PHP

PHP puede hacer cualquier cosa que se pueda hacer con un script CGI, como procesar la información de formularios, generar páginas con contenidos dinámicos, o mandar y recibir cookies. Y esto no es todo, se puede hacer mucho más. Existen tres campos en los que scripts escritos en PHP son usados [43].

- Scripts en la parte del servidor. Este es el campo más tradicional y el principal campo de trabajo. Se necesitan tres cosas para que esto funcione. El parseador PHP (CGI o módulo), un servidor web y un navegador. Se necesita correr el servidor web con PHP instalado. El resultado del programa PHP se puede obtener a través del navegador, conectando con el servidor web. Consultar la sección Instrucciones de instalación para más información.
- Scripts en línea de comandos. Se puede crear un script PHP y correrlo sin ningún servidor web o navegador. Solamente se necesita el parseador PHP para usarlo de esta manera. Este tipo de uso es ideal para scripts ejecutados regularmente desde cron (en *nix o Linux) o el Planificador de tareas (en Windows). Estos scripts también pueden ser usados para tareas simples de procesar texto.

4.6.6. HTML

HTML (HyperText Markup Language o lenguaje de marcado de hipertexto) es el lenguaje de etiquetas que funciona como una de las piedras angulares de la World Wide Web. Aunque la evolución de Internet nos ha traído muchos avances en lo que se refiere a tecnología (Web 2.0 y Web 3.0, mediante), el lenguaje de etiquetas que se popularizó en la década del noventa sigue siendo fundamental para el desarrollo web, ya que es el que comprenden e interpretan los navegadores. Claro está que por sí solo ya no es tan potente como lo fue en aquellos tiempos y, hoy por hoy, necesita combinarse con otras tecnologías y lenguajes para lograr resultados que estén a la altura de las necesidades del desarrollo web actual. Esto es lo que significa HTML, palabra por palabra [44] [45] [47]:

- El texto de hipertexto que se pulsa para saltar de un documento a otro. Esta es una referencia a la capacidad de las páginas Web de vincular entre sí.
- Marcado etiquetas que se aplican las convenciones de presentación y formato de texto plano. Literalmente, el texto sin formato está "marcado" con las etiquetas.

- Idioma Una referencia al hecho de que HTML se considera un lenguaje de programación.

4.6.6.1. Características

Entre las características más importantes de HTML, destacamos [45]:

Los elementos se representan en el código con etiquetas, las cuales se encierran mediante los símbolos menor (<) y mayor (>), a su vez, pueden tener atributos definidos en forma específica.

En su estructura básica, vemos que un documento HTML tiene una primera línea de declaración de tipo de documento, un encabezado (<head>) y posteriormente un cuerpo (<body>).

En el encabezado, encontramos los elementos que describen el documento, como por ejemplo el título, y también etiquetas de metadatos (metadata) que pueden contener descripción, nombre del autor e idioma, entre otras opciones. También, en el encabezado, se suelen especificar los archivos que se incluyen o anexan al documento, como por ejemplo hojas de estilo o scripts que se ejecutan en la página del lado cliente. En el cuerpo del documento, se definen los elementos que se representan en la pantalla del navegador.

La mayoría de estos elementos poseen una apariencia predeterminada por defecto que, en algunos casos, puede tener alguna variación según el navegador. Si bien HTML permite definir algunas características de representación, como color y tamaño, no es recomendable utilizarlas ya que, para esta labor, contamos con las hojas de estilo, más conocidas como CSS.

La función de HTML es definir la estructura del documento, dejando de lado lo que se relaciona con representación. A la hora de diseñar y desarrollar nuestros proyectos, será muy importante pensar en cómo llevaremos adelante esa estructura antes de sentarnos a escribir las líneas de código que la conformarán.

El código HTML puede ser editado con cualquier editor de texto aunque, como veremos más adelante, hay herramientas que nos permiten trabajar con mayor comodidad y nos facilitan el trabajo. En el Capítulo 2 de este libro, analizaremos con mayor detalle los elementos que forman parte de HTML.

4.6.7. WAMP

WampServer es una plataforma de desarrollo web para aplicaciones web dinámicas utilizando el servidor Apache 2, lenguaje de script PHP y base de datos MySQL. También posee phpMyAdmin para administrar fácilmente sus bases de datos [47].

4.6.7.1. Características

WampServer también tiene un "TrayIcon" que le permite gestionar y configurar sus servidores, sin tocar los archivos de configuración fácilmente [47].

Haz clic izquierdo en el icono IP Works, puede:

- Gestionar los servicios Apache y MySQL cambiar a línea / fuera de línea (accesible a todos, o se limita a localhost)
- Instalar y cambiar la versión de Apache, MySQL y PHP
- Administración de la configuración de configuración del servidor
- El acceso a los registros los archivos de configuración de acceso
- Cambiar el idioma del menú WampServer e ir directamente a esa página

4.6.7.2. Instalación

Haga doble clic en el archivo descargado y siga las instrucciones. Todo es administrado por el instalador de IP Works. Por defecto WampServer viene con las últimas versiones de Apache, MySQL y PHP [47] .

Una vez instalado, puede agregar manualmente versiones adicionales de Apache, PHP o MySQL (sólo compilado VC9, o VC10 VC11). Las explicaciones de los datos estarán en el foro.

Cada versión de Apache, MySQL y PHP tiene su propia configuración y sus propios archivos (datos para MySQL). Utilice WampServer: creación automática de un directorio "www" durante la instalación (normalmente C: \ wamp \ www).

Crear un subdirectorio para su proyecto y colocar allí sus archivos PHP Haga clic en el enlace "localhost" en el menú IP Works o abrir su navegador favorito e ir a <http://localhost>.

4.6.8. Netbeans

Ofrece analizadores de código y editores listos para usar para trabajar con Java 8 últimas tecnologías Java SE - 8, Java SE. El IDE también tiene una gama de nuevas

mejoras que mejoran aún más más su apoyo a Java EE, HTML5 nuevas herramientas, en particular para AngularJS; y mejoras al soporte PHP y C / C ++.

NetBeans IDE 8.0 está disponible en Inglés, portugués de Brasil, japonés, ruso y chino simplificado [48].

4.6.8.1. Características

JavaScript

- El apoyo RequireJS
- AngularJS la navegación a través de hipervínculos de vista de los conductores.
- Muchas mejoras en la funcionalidad AngularJS finalización de código entre los artefactos tales como finalización de código en vista de las propiedades definidas en los controladores.
- El apoyo a la finalización de código en los modelos knockout.
- El nuevo soporte para el editor para crear widgets y plugins jQuery.
- Hay soporte para la depuración de código JavaScript se ejecuta en el motor de secuencias de comandos Nashorn (JDK 8+).

HTML5

- Android 4.4 Soporte de depuración Webkit a Córdoba 3.3+.
- El nuevo soporte para el ejecutor de pruebas Karma.
- El nuevo soporte para la construcción del ronco.
- El Avatar.js está disponible en el Administrador de complementos, a través de Herramientas | Los plug-ins. Después de la instalación, usted será capaz de crear proyectos Avatar.js.

PHP

- Soporte para PHP 5.6.
- El nuevo soporte para Nette probador.
- El nuevo soporte para el CS PHP Broker.
- Mejoras en la ramita, Latte y neón.
- Apoyar el nuevo Asistente de Proyecto para Composer.

E.MATERIALES Y MÉTODOS

Para desarrollar el presente trabajo de titulación se utilizaron algunos métodos y técnicas que permitieron llevar un proceso ordenado y sistematizado de las actividades que se deben realizar para dar solución a la problemática planteada, partiendo desde la recolección de información hasta la construcción de un software de calidad y totalmente funcional. Es por ello que se describen a continuación los siguientes métodos y técnicas:

5.1. MÉTODOS

5.1.1. Método Deductivo

Se lo utilizo a lo largo de todo el proyecto, partiendo desde el análisis y determinación de los requerimientos de usuario, necesarios para definir las funciones y procesos que llevaran a cabo las aplicaciones web y móvil, hasta llegar a lo específico, con las pruebas y evaluación, que permitirá explotar el máximo rendimiento y detectar posibles errores.

5.1.2. Método Inductivo

Este método permitió conocer de manera específica los problemas de difusión de información, que se encuentran presentes en las unidades de salud pública y privada de la ciudad de Loja, y de esta forma, mediante el uso de la tecnología poder aportar con una solución innovadora a través de la creación de una aplicación móvil.

5.1.1. Método Analítico

Aporto en la determinación de las herramientas más relevantes que se emplearon para el desarrollo de las aplicaciones web y movil.

5.2. TÉCNICAS

Las técnicas que se utilizaron para llevar a cabo el desarrollo del presente trabajo de titulación son las siguientes:

5.2.1. Encuesta

Esta técnica fue utilizada para obtener información de las características más importantes que poseen las unidades de salud y, así determinar las necesidades y requerimientos es deben emplear para el desarrollo de la aplicaciones.

5.2.2. Bibliográfica

Esta técnica nos permite encontrar y recolectar información científica, extrayendo de fuentes confiables como: libros, artículos, revistas científicas y páginas web confiables, permitiendo sustentar la base teórica del presente trabajo de titulación.

5.2.3. Observación

Haciendo uso de esta técnica, se ha logrado evidenciar de forma superficial las necesidades que existen respecto a la escasa información de las unidades de salud públicas y privadas en la ciudad de Loja.

5.3. METODOLOGÍA DE DESARROLLO

Para llevar a cabo la construcción total de las aplicaciones tanto web como móvil se aplicó la Metodología de desarrollo Programación Extrema (XP). La utilización de este proceso de desarrollo se ve reflejada a lo largo del desarrollo de las aplicaciones:

- En la fase de planeación del proyecto se realizó recopilación de los requerimientos mediante el uso de historias de usuario, se estableció el plan de iteración y posteriormente la identificación de tareas.
- En la fase de diseño se realizó, los diagramas del lenguaje unificado de modelado (UML), tales como casos de uso, clases, componentes, paquetes y modelo relacional de datos, además de la creación de las tarjetas CRC (Clase-Responsabilidad-Colaboración).
- En la fase de codificación se procedió a programar las aplicaciones tomando en cuenta la instalación de herramientas e IDEs necesarios para el desarrollo del mismo.
- En la fase de pruebas se validaron cada una de las pruebas de aceptación establecidas, con el fin de determinar que las aplicaciones cumplan con los objetivos con los cuales fueron creadas.

F. RESULTADOS

6.1. COMPARATIVA DE METODOLOGÍAS

TABLA I: COMPARATIVA DE METODOLOGÍAS AGILES.

	Programación Extrema	SCRUM	ICONIX	Mobile-D
Fases	Planeación		Análisis de Requisitos	Exploración
	Diseño		Análisis y Diseño Preliminar	Inicialización Producción,
	Codificación		Diseño	Estabilización
	Pruebas		Implementación	Prueba y Corrección del sistema
Estilo de desarrollo	Iterativo y rápido	Iterativo y rápido	Iterativo	Iterativo y rápido
Tamaño del equipo	Pequeño	Pequeño	Pequeño/Medio	Pequeño/Medio
Complejidad del Problema	Medio/Alto	Medio/Alto	Pequeño/Medio	Pequeño/Medio
Tamaño de los proyectos	Pequeños y medianos	Pequeños, medianos, grandes	Pequeños, medianos, grandes	Pequeños y medianos
Artefactos	Historias de usuario	Lista del producto	Modelo de Dominio	Modelo de Casos de Uso
	Tareas	Lista de Pendientes del Sprint	Modelo de Casos de Uso	Diagrama de Clases
	Tarjetas CRC	Sprint	Prototipo de Interfaz de Usuario	
			Diagrama de Robustez	

		Incremento	Diagramas de Secuencia	Diagrama entidad Relación
Documentación	Si	Si	Si	Si
Grupo de Trabajo	Programador, cliente , tester, encargado del seguimiento, coach, consultor	Scrum Master, Propietario del producto, Equipo		Grupo del cliente. Jefe del proyecto. Arquitectos del proyecto.
Flexibilidad para Cambios	Adaptable	Adaptable	Adaptable	Adaptable
Comunicación con el cliente	Continua en todas las fases	En los sprints	En todas las fases	En todas las fases

Considerando los parámetros que se tomó para evaluar cada una de las metodologías de desarrollo, se eligió Programación Extrema (XP) por ser un proceso ordenado, sistematizado, ágil, flexible a cambios de requerimientos, iterativo y rápido de llevar a cabo con un grupo limitado de trabajo. Además existe una amplia documentación con la descripción y características, roles, principios y las fases que se deben cumplir respetando los artefactos e hitos de entrega los cuales son muy pocos y sencillos de realizar. Cabe mencionar que en la fase de diseño se agregaron diagramas UML (Lenguaje de Modelado Unificado), para el modelado del sistema, permitiendo tener una mayor comprensión de los componentes y clases software que serán empleadas en las aplicaciones móviles y web.

6.2. COMPARATIVA DE FRAMEWORKS

TABLA II: COMPARATIVA DE FRAMEWORKS MULTIPLATAFORMA

	IONIC	PHONEGAP	TITANIUM
Plataformas soportadas	Android iOS Windows Phone Blackberry Os Web Os	Android ios Windows Phone Blackberry Os	Android ios Windows Phone
Acceso a los recursos del dispositivo	Acceso a todas las funcionalidades del dispositivo	Acceso a todas las funcionalidades del dispositivo	Acceso a todas las funcionalidades del dispositivo
Distribución de aplicaciones	Todas sus plataformas	Todas sus plataformas	Todas sus plataformas
Integración con otras tecnologías	Adaptable	Adaptable	Poco Adaptable
Creación de interfaces	Si	Si	No
Instalación/configuración	Sencilla	Sencilla	Sencilla
Documentación	Si	Si	Si
Comunidad	Si	Si	Si
Soporte	Si	No	Si
API y códigos para el uso del software	Existen Documentación	Existen Documentación	Insuficiente

El framework que se seleccionó para el desarrollo de la aplicación móvil fue Ionic, esta plataforma híbrida es fácil y sencilla de instalar, independientemente del sistema operativo que se vaya a desarrollar.

Ionic proporciona acceso y control a los recursos del dispositivo móvil por medio de las librerías de Cordova, además provee una completa documentación de todos sus componentes, estilos, JavaScript, demos, foros, blogs, soporte, etc.

Además permite crear interfaces intuitivas y minimalistas, por medio de la programación tradicional web (HTML5, CSS, JavaScript), asimismo cuenta con Angular js y otros

componentes software extra como Gulp, Bower que hacen que este framework sea el más idóneo para el desarrollo de la aplicación móvil.

6.3. FASE DE PLANEACIÓN

Para la creación de las historias de usuario, se realizó el documento de Especificación de Requerimientos de Software (ERS), (véase en el anexo I) el mismo que aplica características del estándar IEEE 830-1998. Este documento ayuda a describir de forma completa el comportamiento del software que se va a desarrollar, además permite obtener los requerimientos funcionales y no funcionales del sistema.

La definición de la historias de usuario de este proyecto se basa fundamentalmente en el documento de especificación de requerimientos.

6.3.1. Roles

Hay que tener en cuenta que los desarrolladores de este proyecto es realizado por una sola persona, por lo cual los roles definidos en XP fueron ocupados, y en algunos casos por el asesor del proyecto.

- **Programador:** Kevin Atiencia responsable de producción de código del sistema.
- **Cliente:** Los clientes del sistema serán en sí, los ciudadanos de la ciudad de Loja que dispongan de un Smartphone que cumpla las características mínimas requeridas.
- **Encargado de pruebas (Tester):** Al tener un personal totalmente reducido, el responsable de realizar las pruebas de carga, unitarias, funcionales será el programador.
- **Entrenador (Coach):** El Ing. Roberth Gustavo Figueroa Díaz fue el responsable del proceso global, tiene como responsabilidad proveer las guías y prácticas necesarias para la correcta aplicación de la metodología XP.

Algunos roles definidos en XP (Consultor y gestor o Big Boss, Tracker) no fueron tomados en cuenta para este proyecto, porque el tamaño tan reducido del equipo de trabajo no lo permitía.

6.3.2. Historias de Usuario

Programación extrema propone una plantilla de historia de usuario, misma que consta de los siguientes campos:

TABLA III: FORMATO DE LA HISTORIA DE USUARIO

Historia de Usuario	
Número:	Nombre Historia de Usuario:
Usuario:	Iteración Asignada:
Prioridad:	
Riesgo en Desarrollo:	Programador Responsable:
Descripción:	
Observaciones:	

- **Número:** Identificación de cada historia de usuario.
- **Nombre Historia de Usuario:** Nombre que identifica a cada historia de usuario
- **Iteración Asignada:** Iteraciones de las historias de usuario
- **Usuario:** Nombre del responsable de la actividad descrita en la historia de usuario.
- **Prioridad:** Este puede ser Alto, Medio, Bajo, de acuerdo a la descripción e importancia de la actividad.
- **Riesgo en Desarrollo:** Puede ser Alto, Medio, Bajo, de acuerdo a la complejidad de desarrollo de la actividad.
- **Programador Responsable:** Persona responsable de llevar a cabo el desarrollo de la actividad.
- **Descripción:** Detalle de la actividad a desarrollar.
- **Observaciones:** Aspectos importantes referentes a la historia.

A continuación se detallan las historias de usuarios concernientes a la aplicación web y móvil.

6.3.2.1. Aplicación Web

TABLA IV: ACCEDER AL SISTEMA

Historia de Usuario	
Número: 01	Nombre Historia de Usuario: Acceder al sistema (App Web).
Usuario: Administrador	Iteración Asignada: 1
Prioridad: Baja	
Riesgo en Desarrollo: Baja	Programador Responsable: Kevin Atiencia
<p>Descripción: El usuario tendrá que identificarse mediante un formulario en el cual ingresa el correo electrónico y su contraseña. En donde el sistema comprueba si los datos son correctos y da la apertura o denegación en caso que no lo fuera.</p> <p>La pantalla tendrá las siguientes características.</p> <ul style="list-style-type: none"> • Un formulario Login, • Un campo de texto, con una descripción de la información que debe ingresar. • Un campo password en el cual los datos ingresados serán puntos <p>Características Generales:</p> <ul style="list-style-type: none"> • Al presionar el botón "Iniciar sesión", el sistema comprobará si los campos están llenos. En caso de estarlo, el sistema da acceso. Caso contrario presentará un mensaje de error, mencionando el campo que debe ingresar bien. • Adicionalmente el login tendrá que ingresar un código Recaptcha y se llevará un registro de ingreso. 	
Observaciones: Deberá ingresarse usuarios iniciales a nivel de base de datos	

TABLA V: EDITAR INFORMACIÓN DEL USUARIO

Historia de Usuario	
Número: 02	Nombre Historia de Usuario: Editar información del usuario (App Web).
Usuario: Administrador	Iteración Asignada: 1
Prioridad: Media	
Riesgo en Desarrollo: Baja	Programador Responsable: Kevin Atiencia
<p>Descripción: Dispondrá de un formulario para editar la información de una entidad usuario, los datos que tendrá una entidad usuario son: Nombres, apellidos, correo, clave, cedula, estado, dirección, foto y teléfono.</p>	

<p>La pantalla tendrá las siguientes características.</p> <ul style="list-style-type: none"> • Cabecera: Tendrá su respectivo nombre. • Cuerpo: Tendrá un formulario con los campos antes mencionados el cual deben estar llenos con los datos requeridos, caso contrario presentará un mensaje de error. Para finalmente constar de dos botones uno para almacenar la información, y otro para cancelar. <p>Características Generales:</p> <ul style="list-style-type: none"> • Se utilizará colores sencillos para el diseño. • Los componentes de formulario estarán centrados. • Los campos de texto en un formulario tendrán una longitud adecuada para la información. • Se comprobará que se haya suministrado todos los campos obligatorios, caso contrario el sistemas mostrará un mensaje de alerta: “Este campo es requerido”, sin afectar los campos llenos. • El campo correo electrónico tendrá que ser correcto.
<p>Observaciones:</p>

TABLA VI: GESTIÓN DE UNIDADES DE SALUD

Historia de Usuario	
Número: 03	Nombre Historia de Usuario: Gestión de Unidades de Salud (App Web).
Usuario: Administrador	Iteración Asignada: 2
Prioridad: Alta	
Riesgo en Desarrollo: Alta	Programador Responsable: Kevin Atiencia
<p>Descripción: Dispondrá de un formulario para crear y editar la información de una unidad de salud. Los datos que tendrá esta entidad son: unicódigo, nombre oficial, nombre común, dirección, teléfono, provincia cantón, parroquia, tipo de parroquia, zona, zona distribución, distrito, distrito distribución, circuito, código de área, área, red de atención, lucro, institución, nivel de atención, tipología, horario de atención, foto. Todos estos campos constarán en un formulario con la interfaz de la acción (crear, editar) que se desea realizar, todos los campos obligatorios deben estar llenos y validados, caso contrario presentará un mensaje de error en el campo requerido, al final del formulario consta de un botón guardar.</p> <p>Características Generales:</p> <ul style="list-style-type: none"> • Se utilizará colores sencillos para el diseño. • Los componentes de formulario estarán centrados. • Los campos de texto en un formulario tendrán una longitud adecuada para la información que será introducirá en ellos. 	

- Se comprobará que se haya suministrado todos los campos obligatorios, caso contrario el sistemas mostrará un mensaje de alerta: “Este campo es requerido”, sin afectar los campos llenos.
- El campo correo electrónico tendrá que ser correcto.

Observaciones:

TABLA VII: LISTAR UNIDADES DE SALUD.

Historia de Usuario	
Número: 04	Nombre Historia de Usuario: Listar Unidades de Salud (App Web).
Usuario: Administrador	Iteración Asignada: 2
Prioridad: Media	
Riesgo en Desarrollo: Baja	Programador Responsable: Kevin Atiencia
<p>Descripción: Contará con una interfaz donde se listarán todas las unidades de salud almacenadas en la bases de datos, en donde se mostrará en una tabla con los datos más importantes y la opción de eliminar.</p> <p>Para optimizar tiempo en la creación de Unidades de Salud, se implementará la opción de subir archivo, este debe ser una hoja de cálculo de Excel en un formato preestablecido para garantizar que esta funcionalidad opera sin ningún problema.</p> <p>Características Generales:</p> <ul style="list-style-type: none"> • Se utilizará colores sencillos para el diseño. • Los componentes de formulario estarán centrados. • Los campos de texto en un formulario tendrán una longitud adecuada para la información que será introducirá en ellos. • Se comprobará que se haya suministrado todos los campos obligatorios, caso contrario el sistemas mostrará un mensaje de alerta: “Este campo es requerido”, sin afectar los campos llenos. • El campo correo electrónico tendrá que ser correcto. 	
Observaciones:	

TABLA VIII: GESTIÓN DE ÁREAS DE ESPECIALIDAD

Historia de Usuario	
Número: 05	Nombre Historia de Usuario: Gestión de Áreas de Especialidad (App Web).
Usuario: Administrador	Iteración Asignada: 3
Prioridad: Alta	
Riesgo en Desarrollo: Alta	Programador Responsable: Kevin Atencia
<p>Descripción: Los datos que tendrá esta entidad áreas de especialidad son: nombre del servicio, nombre de la especialidad, disponibilidad, teléfono, nombre del profesional responsable, horario de atención, días de atención. El gestor contará con formularios para las acciones de crear, editar en una interfaz donde se listarán, en una tabla todos los campos que poseen esta entidad y el botón eliminar, cabe recalcar que estos datos pertenecen a la interfaz de unidad de salud en concreto. Para optimizar tiempo en la creación de áreas de especialidad, se implementará la opción de subir archivo, el cual debe ser una hoja de cálculo de Excel en un formato preestablecido para garantizar que esta funcionalidad opera sin ningún problema.</p> <p>Características Generales:</p> <ul style="list-style-type: none"> • Se utilizará colores sencillos para el diseño. • Los componentes de formulario estarán centrados. • Los campos de texto en un formulario tendrán una longitud adecuada para la información que será introducirá en ellos. • Se comprobará que se haya suministrado todos los campos obligatorios, caso contrario el sistemas mostrará un mensaje de alerta: "Este campo es requerido", sin afectar los campos llenos. • El campo correo electrónico tendrá que ser correcto. 	
Observaciones: Para crear una área de especialidad, primero debe existir una unidad de salud	

TABLA IX: GESTIÓN DE SERVICIOS.

Historia de Usuario	
Número: 06	Nombre Historia de Usuario: Gestión de Servicios (App Web).
Usuario: Administrador	Iteración Asignada: 3
Prioridad: Alta	
Riesgo en Desarrollo: Alta	Programador Responsable: Kevin Atencia

Descripción: Cada unidad de salud dispondrá de un formulario para crear o editar la información de una entidad servicios. Los campos que tendrá esta entidad son: descripción, disponibilidad, teléfono, nombre del profesional responsable, horario de atención.

El gestor contará con formularios para las acciones de crear, editar en una interfaz donde se listarán en una tabla los servicios que dispone una unidad de salud con sus respectivos campos, y se añade un botón de eliminar.

Para optimizar tiempo en la creación de áreas de especialidad, se implementará la opción de subir archivo, el cual debe ser una hoja de cálculo de Excel en un formato preestablecido para garantizar que esta funcionalidad opera sin ningún problema.

Características Generales:

- Se utilizará colores sencillos para el diseño.
- Los componentes de formulario estarán centrados.
- Los campos de texto en un formulario tendrán una longitud adecuada para la información que será introducirá en ellos.
- Se comprobará que se haya suministrado todos los campos obligatorios, caso contrario el sistemas mostrará un mensaje de alerta: "Este campo es requerido", sin afectar los campos llenos.
- El campo correo electrónico tendrá que ser correcto.

Observaciones: Para crear una área de especialidad, primero debe existir una unidad de salud

TABLA X: GESTIÓN DE CONTRIBUIDORES

Historia de Usuario	
Número: 07	Nombre Historia de Usuario: Gestión de Contribuidores (App Web).
Usuario: Administrador	Iteración Asignada: 4
Prioridad: Alta	
Riesgo en Desarrollo: Alta	Programador Responsable: Kevin Atiencia
<p>Descripción: El sistema contará con notificaciones las cuales permitirán conocer la información detallada del servicio y el contribuidor, que ha sido ingresada desde la aplicación móvil dicha información.</p> <p>Es por ello que debe existir una entidad contribuidor la misma que dispondrá con los campos como nombre, apellido, correo, teléfono, fecha de registro.</p> <p>El gestor además contará con las opciones de eliminar el contribuidor con la información ingresada o aceptarla como valedera.</p> <p>Características Generales:</p> <ul style="list-style-type: none"> • Se utilizará colores sencillos para el diseño. • Los componentes de contenedores estarán centrados. • Los campos de texto tendrán una longitud adecuada para la información que será introducirá en ellos. 	

Observaciones:

TABLA XI: GESTIÓN DE MARCADORES

Historia de Usuario	
Número: 08	Nombre Historia de Usuario: Gestión de Marcadores (App Web).
Usuario: Administrador	Iteración Asignada: 4
Prioridad: Media	
Riesgo en Desarrollo: Media	Programador Responsable: Kevin Atiencia
<p>Descripción: Se podrá modificar la información de latitud y longitud mediante la selección de la unidad de salud a la que deseemos modificar. También podrá cambiar la información de georreferencia mediante una interfaz gráfica la cual dispondrá del google map para seleccionar directamente la ubicación de la unidad de salud.</p> <p>Características Generales:</p> <ul style="list-style-type: none">• Se utilizará colores sencillos para el diseño.• Los componentes de formulario estarán centrados.• Los campos de texto en un formulario tendrán una longitud adecuada para la información que será introducirá en ellos.• Se comprobará que se haya suministrado todos los campos obligatorios, caso contrario el sistemas mostrará un mensaje de alerta: "Este campo es requerido", sin afectar los campos llenos.• El campo correo electrónico tendrá que ser correcto.	
Observaciones: Debe existir unidades de Salud	

6.3.2.2. Aplicación Móvil

TABLA XII: GESTIÓN SERVICIO WEB

Historia de Usuario	
Número: 09	Nombre Historia de Usuario: Gestión de Servicio web.
Usuario: Cliente	Iteración Asignada: 5
Prioridad: Alta	

Riesgo en Desarrollo: Alto	Programador Responsable: Kevin Atiencia
<p>Descripción: El servicio web debe permitir el intercambio de información entre la aplicación móvil y web de forma rápida, segura, eficiente, libre de errores y que ofrezca la interoperatividad.</p> <p>Para ello se debe crear las clases necesarias que permitan el intercambio de los mensajes bajo un estándar específico como lo es JSON.</p> <p>Las funciones más comunes que existen en los servicio web son:</p> <ul style="list-style-type: none"> • GET • FIND • PUT • POST • DELETE 	
Observaciones:	

TABLA XIII: BUSCAR UNIDADES DE SALUD

Historia de Usuario	
Número: 10	Nombre Historia de Usuario: Buscar Unidades de Salud (App Móvil).
Usuario: Cliente	Iteración Asignada: 5
Prioridad: Alta	
Riesgo en Desarrollo: Alto	Programador Responsable: Kevin Atiencia
<p>Descripción: El usuario dispondrá de un menú, que al presionar el botón de “Buscar Unidad de Salud” accede a google maps, Se carga el mapa del lugar donde se encontró por última vez en el caso que tenga desactivado el sensor de GPS, caso contrario, desde su ubicación actual, aparecen en el mapa marcadores de la posición de las unidades de salud que se encuentran en esa zona.</p> <p>Características Generales:</p> <ul style="list-style-type: none"> • Se utilizará colores sencillos para el diseño. • Los componentes de formulario estarán centrados. 	
Observaciones: El dispositivo móvil debe contar con conexión a internet	

TABLA XIV: BUSCAR ÁREAS DE ESPECIALIDAD

Historia de Usuario	
Número: 11	Nombre Historia de Usuario: Buscar Áreas de Especialidad (App Móvil).
Usuario: Cliente	Iteración Asignada: 5
Prioridad: Alta	
Riesgo en Desarrollo: Alto	Programador Responsable: Kevin Atencia
<p>Descripción: El usuario dispondrá de un menú en el cual al presionar el botón de “Buscar Áreas de Especialidad” accede a una interfaz nueva, con un input para introducir el campo a buscar y un botón “buscar”, que al pulsarlo, cargará en una lista el nombre de las unidades de salud que disponen del área que el usuario está buscando, caso contrario aparece un mensaje “No se ha encontrado resultado a su búsqueda”.</p> <p>Características Generales:</p> <ul style="list-style-type: none"> • Se utilizará colores sencillos para el diseño. • Los componentes de formulario estarán centrados. 	
Observaciones: El dispositivo móvil debe contar con conexión a internet	

TABLA XV: CREAR SERVICIO

Historia de Usuario	
Número: 12	Nombre Historia de Usuario: Crear Servicio (App Móvil).
Usuario: Cliente	Iteración Asignada: 6
Prioridad: Alta	
Riesgo en Desarrollo: Alto	Programador Responsable: Kevin Atencia
<p>Descripción: El usuario dispondrá de un botón “Crear Servicio” en la interfaz donde se presenta los datos del mismo, en el cual al dar clic, accederá a una interfaz nueva, con un formulario en el cual se le solicitará la información del servicio que desea ingresar al sistema.</p> <p>Posteriormente al llenar el formulario del servicio se solicitará información básica de la persona, que posteriormente pasa a la aplicación web para su validación.</p> <p>Características Generales:</p> <ul style="list-style-type: none"> • Se utilizará colores sencillos para el diseño. • Los componentes de formulario estarán centrados. 	

Observaciones: El dispositivo móvil debe contar con conexión a internet

TABLA XVI: OBTENER INFORMACIÓN DE LA US

Historia de Usuario	
Número: 13	Nombre Historia de Usuario: Obtener información de la Unidad de Salud (App Móvil).
Usuario: Cliente	Iteración Asignada: 6
Prioridad: Alta	
Riesgo en Desarrollo: Alto	Programador Responsable: Kevin Atiencia
<p>Descripción: Al seleccionar en un marcador del mapa o en una lista el nombre de la Unidad de Salud, automáticamente realizará una petición al servicio web, el cual, le responderá con la información completa de la unidad de salud y lo presentará en un diseño sencillo y amigable con el usuario.</p> <p>Características Generales:</p> <ul style="list-style-type: none"> ○ Se utilizará colores sencillos para el diseño. ○ Los componentes de formulario estarán centrados. 	
Observaciones: El dispositivo móvil debe contar con conexión a internet	

6.3.2.3. Resumen de Historias de Usuario

TABLA XVII: RESUMEN DE LAS HISTORIAS DE USUARIO

N°	Nombre	Prioridad	Riesgo	Iteración
1	Acceder al sistema (App Web).	Baja	Baja	1,2
2	Editar de usuario (App Web).	Media	Media	1,2
3	Gestión de Unidades de Salud (App Web).	Alta	Alta	2,3
4	Listar Unidades de Salud (App Web).	Media	Media	2,3
5	Gestión de Áreas de Especialidad (App Web).	Alta	Alta	3,4
6	Gestión de Servicios (App Web).	Alta	Alta	3,4
7	Gestión de Contribuidores (App Web).	Alta	Alta	4,5
8	Gestión de Marcadores (App Web)	Media	Media	4,5
9	Gestión de Servicio Web	Media	Media	5,6
10	Buscar Unidades de Salud (App Móvil).	Alta	Alta	5,6
11	Buscar Áreas de Especialidad (App Móvil).	Alta	Alta	5,6

12	Crear Servicio (App Móvil).	Alta	Alta	6
13	Obtener información de la Unidad de Salud (App Móvil).	Alta	Alta	6

6.3.3. Plan de Iteración

TABLA XVIII: PLAN DE ITERACIÓN

Modulo	Historia de Usuario	Requerimientos	Iteración	Días Ideales
1	Acceder al sistema (App Web).	Autenticar Usuario	1,2	4
	Editar información del usuario (App Web).	Actualizar información de Usuario	1,2	2
2	Gestión de Unidades de Salud (App Web).	Crear Unidades de Salud	2,3	3
		Modificar Unidades de Salud	2,3	3
		Eliminar Unidades de Salud	2,3	2
	Listar Unidades de Salud (App Web).	Listar Unidades de Salud	2,3	3
		Subir Archivo de Unidades de Salud	2,3	2
3	Gestión de Áreas de Especialidad (App Web).	Crear Áreas de Especialidad	1,3,4	3
		Modificar Áreas de Especialidad	1,3,4	3
		Eliminar Áreas de Especialidad	1,3,4	2
		Subir Archivo de Áreas de Especialidad	1,3,4	2
	Gestión de Servicios (App Web).	Crear Servicios	3,4	3
		Modificar Servicios	3,4	3
		Eliminar Servicios	3,4	2
4	Gestión de Contribuidores (App Web).	Notificación de Servicio	4,5	3
		Validar Servicio	4,5	3
		Eliminar Contribuidor	4,5	3
		Eliminar Servicio	4,5	2

	Gestión de Marcadores (App Web)	Modificar Información de Georreferencia	4,5	2
5	Gestión de Servicio Web		5.6	7
	Buscar Unidades de Salud (App Móvil).		5,6	8
	Buscar Áreas de Especialidad (App Móvil).		6	8
6	Crear Servicio (App Móvil).		6	5
	Obtener información de la Unidad de Salud (App Móvil).		6	7

6.3.4. Identificación de Tareas

TABLA XIX: TAREA ACCEDER AL SISTEMA (APP WEB).

Tarea	
Número Tarea: 1	Historia de Usuario: 1
Nombre Tarea: Autenticar Usuario (App Web).	
Tipo de Tarea : Desarrollo	
Programador Responsable: Kevin Atencia	
Descripción: El usuario web de la aplicación, debe ingresar su correo, su contraseña, llenar el ReCaptcha, para ingresar al sistema.	

TABLA XX: TAREA MODIFICAR DATOS DE USUARIO.

Tarea	
Número Tarea: 2	Historia de Usuario: 2
Nombre Tarea: Actualizar información del de Usuario.	
Tipo de Tarea : Desarrollo	
Programador Responsable: Kevin Atencia	

Descripción: El usuario web podrá modificar los datos como nombres, apellidos, contraseña, número de teléfono, dirección, foto, pregunta de seguridad, etc.

TABLA XXI: TAREA GESTIÓN DE UNIDADES DE SALUD.

Tarea	
Número Tarea: 3	Historia de Usuario: 3
Nombre Tarea: <ul style="list-style-type: none"> • Crear Unidades de Salud. • Modificar Unidades de Salud. • Eliminar Unidades de Salud. 	
Tipo de Tarea : Desarrollo	
Programador Responsable: Kevin Atencia	
Descripción: <ul style="list-style-type: none"> • El administrador podrá crear Unidades de Salud, a través de un formulario, en el cual debe ingresar la información. • El administrador podrá modificar la información de las Unidades de Salud. • El usuario Web podrá eliminar una Unidad de Salud. 	

TABLA XXII: TAREA ELIMINAR Y SUBIR ARCHIVO DE LAS UNIDADES DE SALUD.

Tarea	
Número Tarea: 4	Historia de Usuario: 4
Nombre Tarea: Listar y subir archivo de las Unidades de Salud.	
Tipo de Tarea : Desarrollo	
Programador Responsable: Kevin Atencia	

Descripción:

- El usuario Web podrá visualizar en una lista todas las Unidades de Salud.
- El usuario Web podrá subir Archivo con todos los datos de las Unidades de Salud.

TABLA XXIII: TAREA GESTIONAR ÁREAS DE ESPECIALIDAD.

Tarea	
Número Tarea: 5	Historia de Usuario: 5
Nombre Tarea: Gestionar Áreas de Especialidad	
Tipo de Tarea : Desarrollo	
Programador Responsable: Kevin Atencia	
Descripción: <ul style="list-style-type: none">• El usuario Web podrá crear Áreas de Especialidad en cada Unidad de Salud• El usuario Web podrá modificar la información de las Áreas de Especialidad• El usuario Web podrá eliminar la información de las Áreas de Especialidad.• El Administrador podrá subir un archivo con la información de Áreas de Especialidad.	

TABLA XXIV: TAREA GESTIONAR SERVICIOS

Tarea	
Número Tarea: 6	Historia de Usuario: 6
Nombre Tarea: Gestionar Servicios	
Tipo de Tarea : Desarrollo	
Programador Responsable: Kevin Atencia	
Descripción:	

- El Administrador podrá crear Servicios de cada Unidad de Salud.
- El Administrador podrá modificar la información de Servicios.
- El Administrador podrá eliminar la información de Servicios.
- El Administrador podrá subir un archivo de información sobre los Servicios

TABLA XXV: GESTIONAR SERVICIO

Tarea	
Número Tarea: 7	Historia de Usuario: 7
Nombre Tarea: Gestionar de Servicio	
Tipo de Tarea : Desarrollo	
Programador Responsable: Kevin Atencia	
Descripción: <ul style="list-style-type: none"> • El Administrador podrá disponer de un contador que indica cuántos y cuáles son los servicios ingresados desde la aplicación móvil. • El Administrador podrá visualizar la información del nuevo servicio registrado en el sistema el cual contará con las opciones de: dar de alta y eliminar. • El Administrador podrá eliminar un contribuidor. • El Administrador podrá eliminar el servicio creado 	

TABLA XXVI: TAREA MODIFICAR INFORMACIÓN DE GEORREFERENCIA

Tarea	
Número Tarea: 8	Historia de Usuario: 8
Nombre Tarea: Modificar Información de Georreferencia	
Tipo de Tarea : Desarrollo	

Programador Responsable: Kevin Atencia
Descripción: El Administrador podrá modificar Información de Georreferencia

TABLA XXVII: TAREA GESTIONAR SERVICIO WEB.

Tarea	
Número Tarea: 9	Historia de Usuario: 9
Nombre Tarea: Gestionar Servicio Web	
Tipo de Tarea : Desarrollo	
Programador Responsable: Kevin Atencia	
<p>Descripción: El servicio debe estar disponible en todo momento, y accedido por el administrador, además de estar estandarizado en formato JSON enviando los recursos solicitados.</p> <p>Los métodos esenciales y dependiendo de las necesidades de la aplicación móvil deben ser GET, FIND, POST, PUT, DELETE.</p>	

TABLA XXVIII: BUSCAR UNIDADES DE SALUD

Tarea	
Número Tarea: 10	Historia de Usuario: 10
Nombre Tarea: Buscar Unidades de Salud (App Móvil).	
Tipo de Tarea : Desarrollo	
Programador Responsable: Kevin Atencia	
<p>Descripción: El usuario de la aplicación móvil podrá buscar Unidades de Salud (App Móvil) mediante el uso de google maps.</p>	

TABLA XXIX: TAREA BUSCAR ÁREAS DE ESPECIALIDAD (APP MÓVIL).

Tarea	
Número Tarea: 11	Historia de Usuario: 11
Nombre Tarea: Buscar Áreas de Especialidad (App Móvil).	
Tipo de Tarea : Desarrollo	
Programador Responsable: Kevin Atencia	
Descripción: El usuario de la aplicación móvil podrá buscar Áreas de Especialidad.	

TABLA XXX: TAREA CREAR SERVICIO (APP MÓVIL).

Tarea	
Número Tarea: 12	Historia de Usuario: 12
Nombre Tarea: Crear Servicio (App Móvil).	
Tipo de Tarea : Desarrollo	
Programador Responsable: Kevin Atencia	
Descripción: El usuario de la aplicación móvil dispondrá de un formulario el cual pondrá ingresar la información del servicio.	

TABLA XXXI: TAREA OBTENER INFORMACIÓN DE LAS UNIDADES DE SALUD (APP MÓVIL).

Tarea	
Número Tarea: 24	Historia de Usuario: 13
Nombre Tarea: Obtener información de las Unidades de Salud (App Móvil).	

Tipo de Tarea : Desarrollo
Programador Responsable: Kevin Atencia
Descripción: El usuario de la aplicación móvil podrá obtener información de la unidad de salud, Área de Especialidad, Médico, según requiera.

6.4. FASE DE DISEÑO

En esta fase se modela las aplicaciones móvil y web mediante el uso de UML, permitiendo controlar el comportamiento dinámico del sistema, reduciendo la complejidad de los componentes que lo integran. Además, nos sirve para comunicarnos con el usuario, obteniendo y validando los requerimientos establecidos.

6.4.1. Patrones de Diseño

Con el objetivo de buscar dar solución a problemas comunes en el desarrollo de software. Un patrón de diseño permitiría asegurar la calidad del software, aprovechar al máximo el tiempo disponible para el desarrollo del proyecto, separar el código en función de sus responsabilidades o conceptos, es por ello, que se utiliza el patrón Modelo-Vista-Controlador (MVC), puesto que las capas que lo componen se adaptan a la arquitectura cliente-servidor.

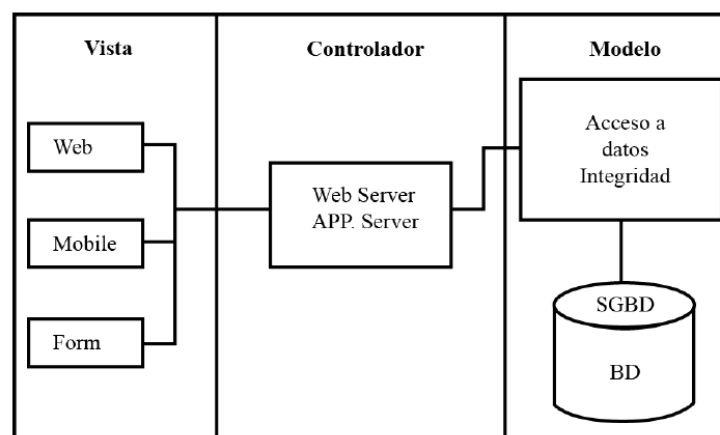


Figura 5: Patrón MVC [49]

6.4.2. Dominio de la solución

Todas las herramientas que se usaron para el desarrollo de la aplicación son Open Source y tienen licencia de libre distribución, además es importante mencionar que el aplicativo web, será el encargado de administrar y gestionar toda la información de las áreas de especialidad, servicios y unidades de salud, ya que esta posee la base de datos central, siendo la responsable de proporcionar por medio del Servicio web (API Rest) la información en formato JSON a la aplicación móvil (IUS), la cual, necesita para almacenar, transferir y presentar la información en las interfaces gráficas de usuario.

A continuación se presenta el diagrama de la solución.

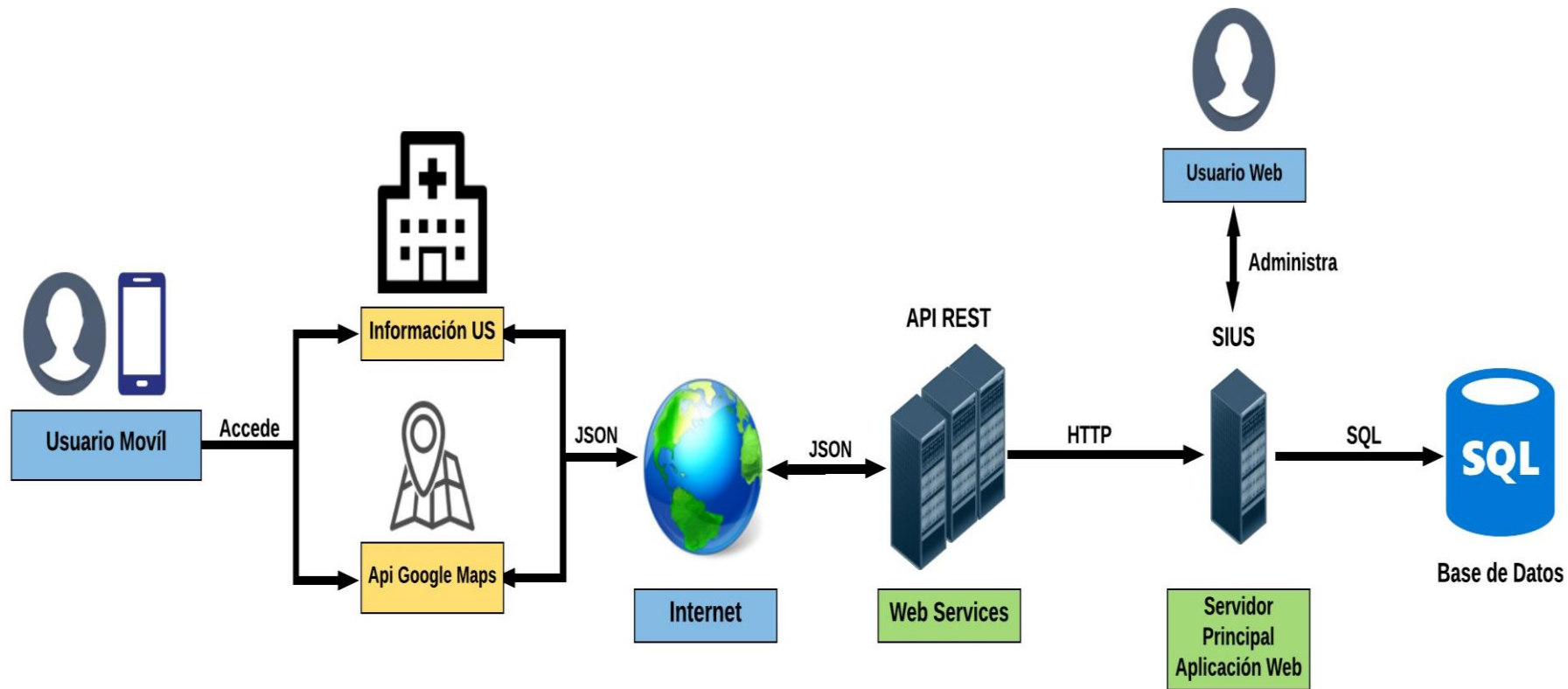


Figura 6: Arquitectura de la Solución

6.4.3. Modelo General de Casos de Uso

El diagrama de casos de uso nos permite especificar la comunicación y el comportamiento de los procesos que lleva a cabo el sistema mediante su interacción con los usuarios, los personajes o entidades que participarán en un caso de uso se denominan actores.

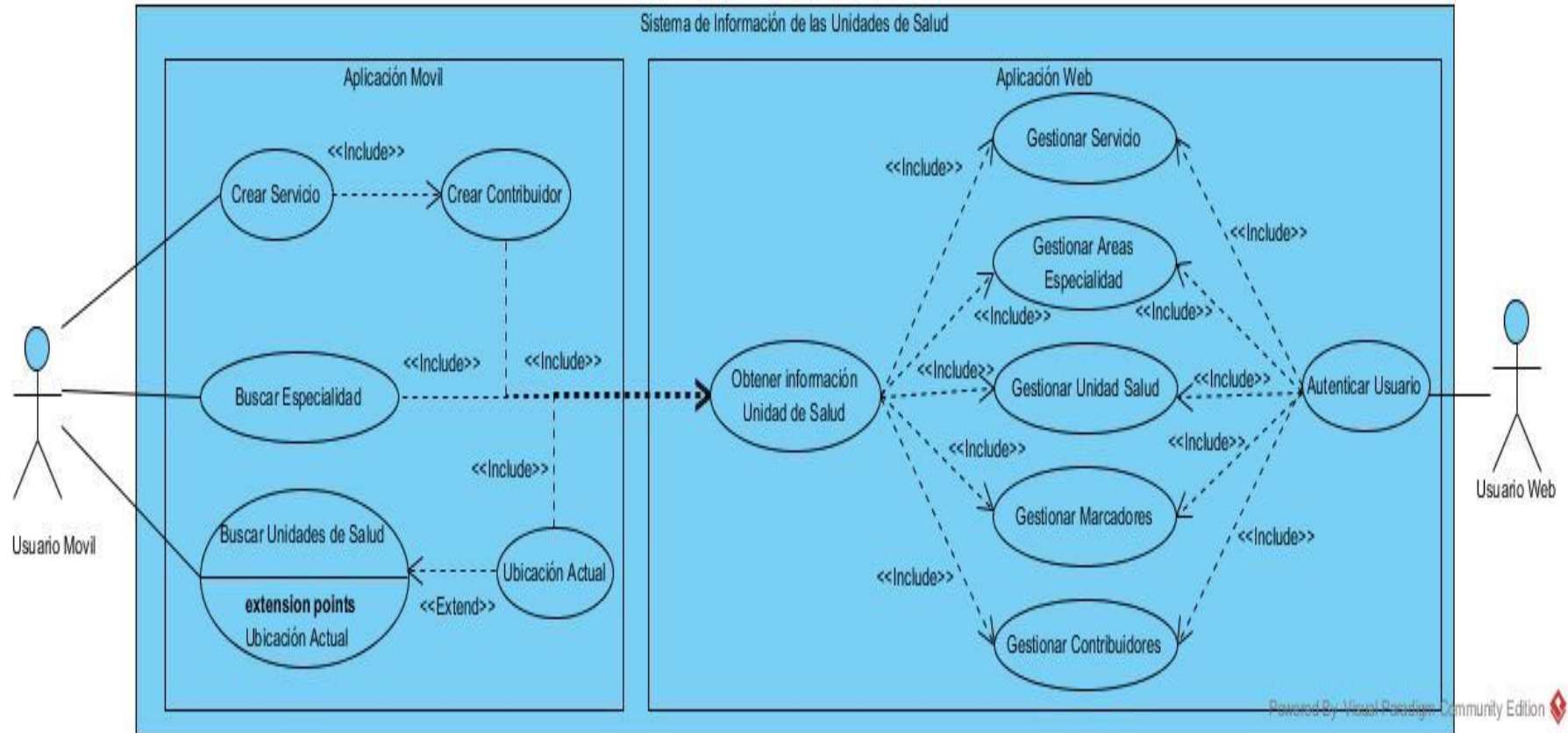


Figura 7: Diagrama General de Casos de Uso General

6.4.4. Diagrama de Clases

El diagrama de clases sirve para identificar las relaciones entre las clases que están involucradas en el sistema. Las clases pueden tener información como los atributos y funciones, mientras las relaciones pueden ser asociativas, herencia, agregación o asociación.

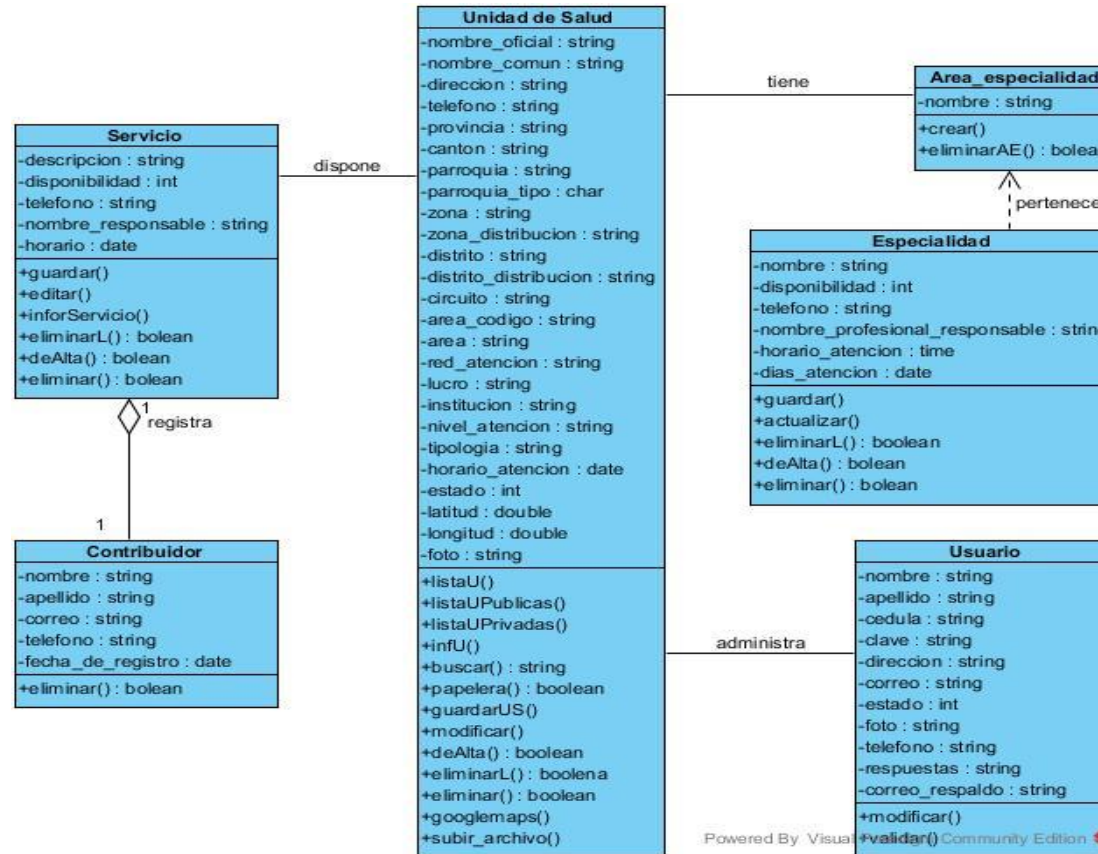


Figura 8: Diagrama de Clases

6.4.5. Diagrama de Componentes

El diagrama de componentes representa cómo el sistema está dividido en módulos y muestra las dependencias entre estos componentes. En el presente diagrama están involucrados los elementos del sistema web, el servicio web y la aplicación móvil.

Partiendo inicialmente con la aplicación web, está dividido en las capas del MVC y la base de datos central; el servicio web adquiere la información de la base de datos por medio de los modelos y este proporciona el servicio en formato JSON, para que sea consumido por la aplicación móvil, la misma que está representada con el paquete www y el conjunto de componentes que viene integrada en el framework de Ionic.

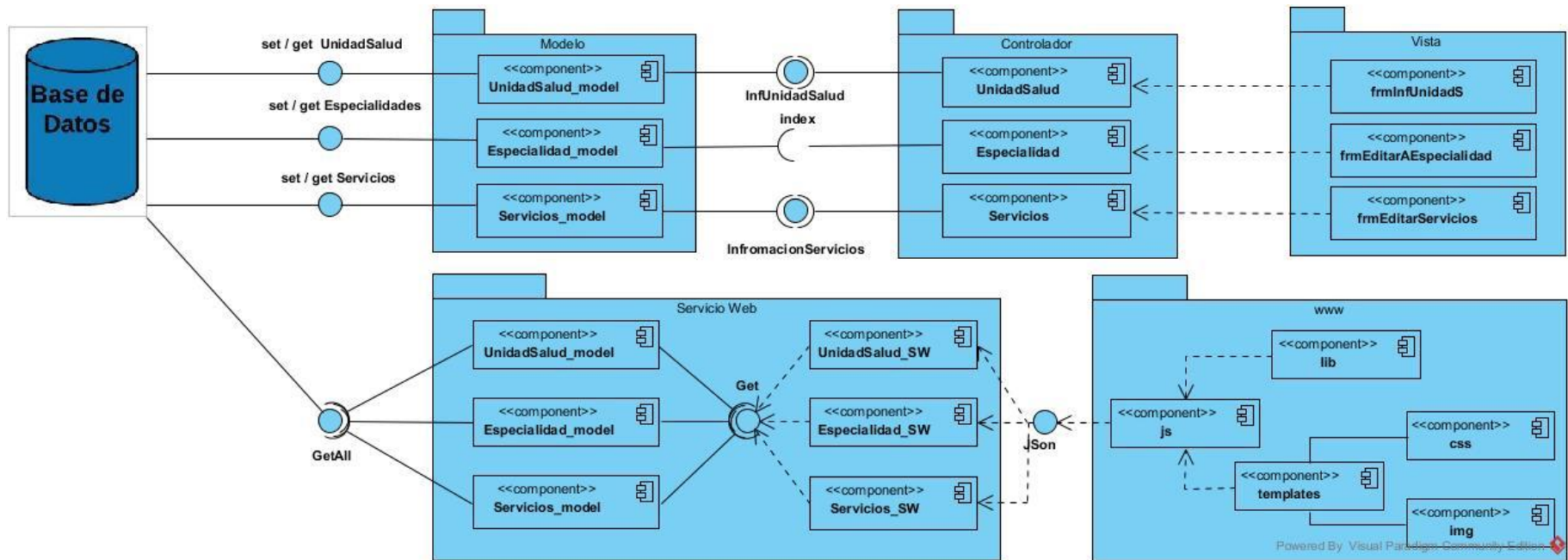


Figura 9 Diagrama: de Componentes

6.4.6. Diagrama de Paquetes

Representa las dependencias entre los distintos paquetes que componen el sistema. Es decir, muestra cómo está estructurada lógicamente y las dependencias entre estas agrupaciones. Las dependencias que se encuentran en la aplicación web están basadas en la arquitectura MVC y los paquetes que integran la aplicación móvil son proporcionados por el framework Ionic.

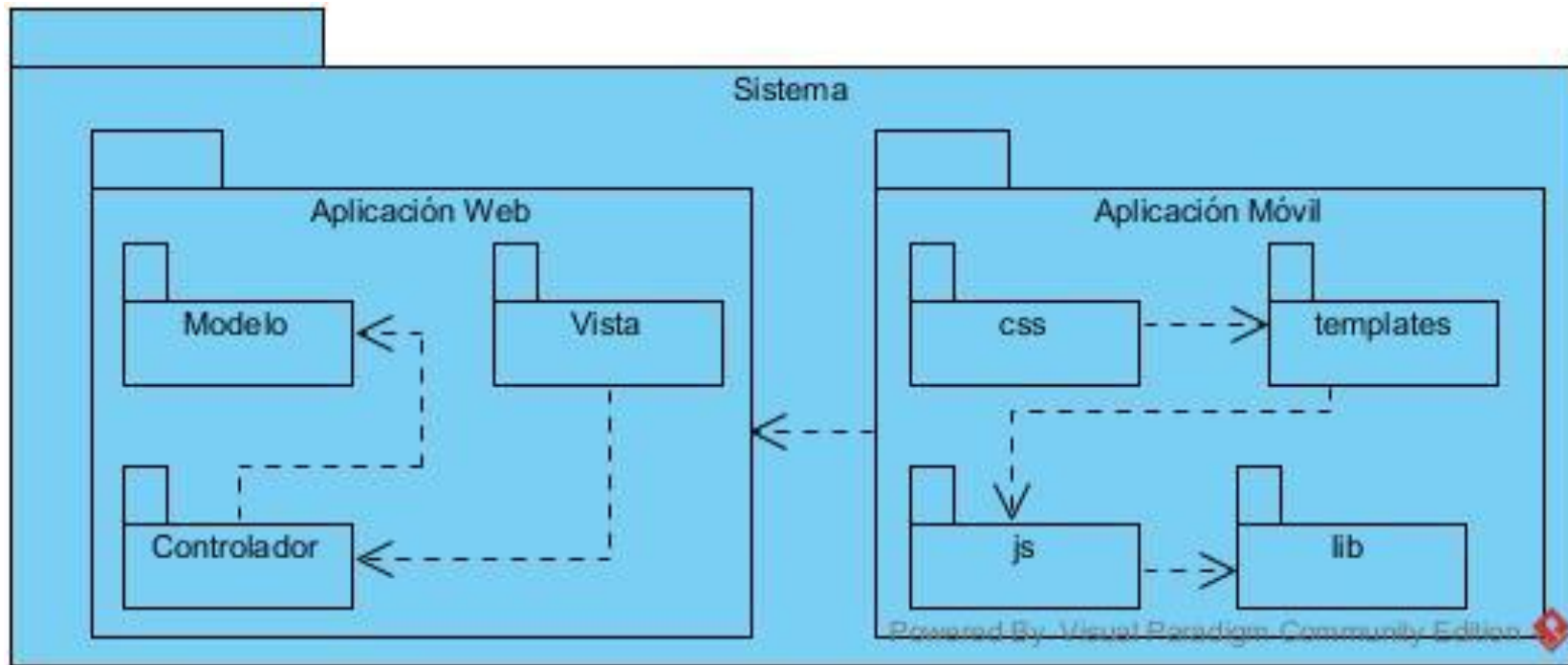


Figura 10: Diagrama de Paquetes

6.4.7. Diagrama Relacional de Base de Datos

El modelo relacional permite realizar un análisis de la gestión de la base de datos para determinar y definir las tablas que serán utilizadas para el desarrollo del sistema.

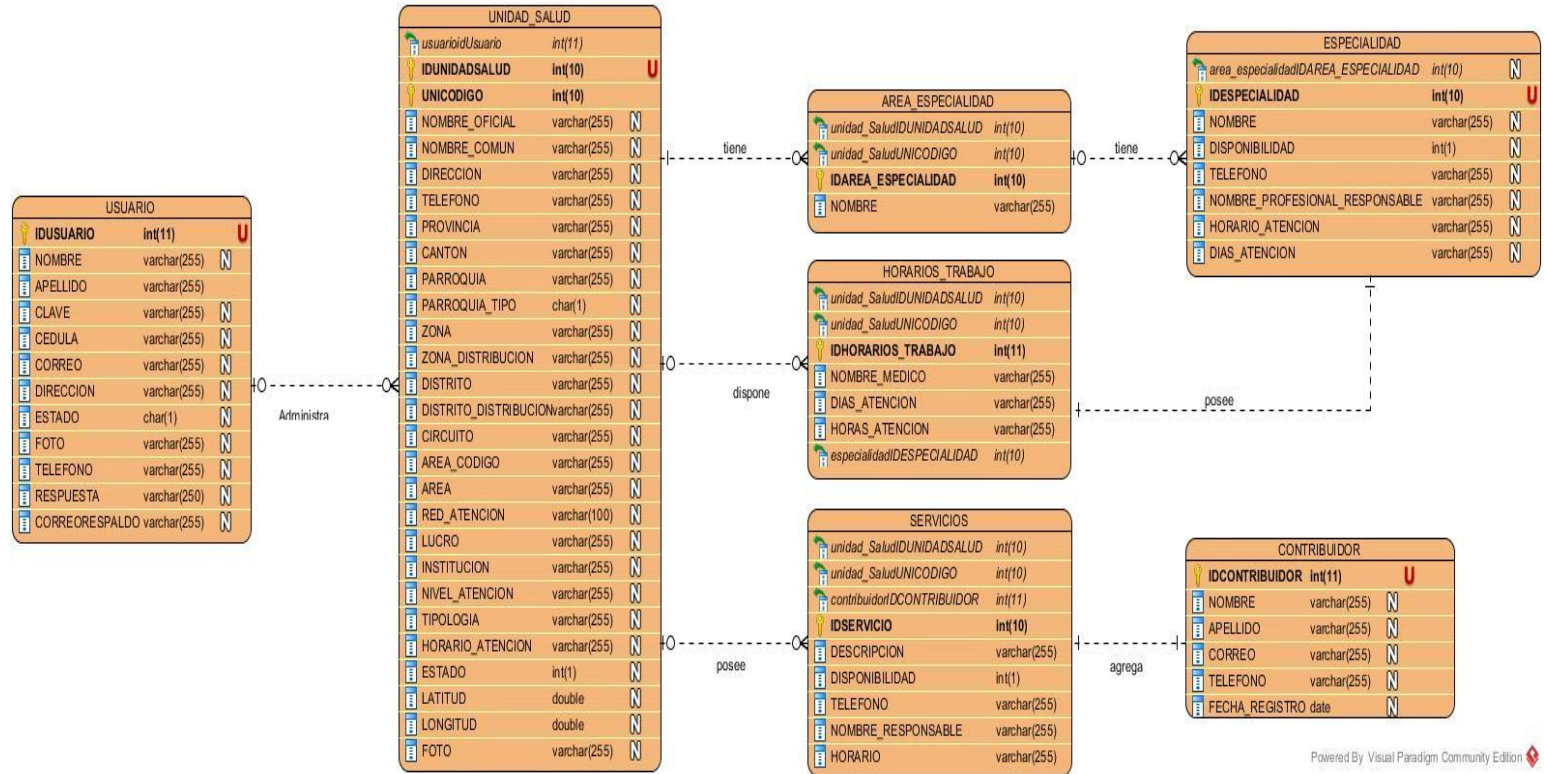


Figura 11: Diagrama Relacional de Base de Datos

6.4.8. Diseño de Tarjetas CRC

TABLA XXXII: TARJETA CRC DE ACCEDER AL SISTEMA WEB.

Nombre CRC: Acceder al Sistema Web		
Responsabilidades	Colaboradores	Métodos
<p>Esta clase permite:</p> <ul style="list-style-type: none"> ○ Ingresar a la página de login, para identificar el usuario web. ○ Validar la información del login. ○ Cargar el mapa de google maps. ○ Cerrar sesión 	<p>Usuario</p> <p>Unidades Salud</p>	<p>Index()</p> <p>page_Admin()</p> <p>validar()</p> <p>validarPregunta()</p> <p>logout()</p> <p>googleMaps()</p> <p>actualizar()</p>

TABLA XXXIII: TARJETA CRC DE UNIDAD DE SALUD.

Nombre CRC: Unidad Salud		
Responsabilidades	Colaboradores	Métodos
<p>Esta clase permite:</p> <ul style="list-style-type: none"> ○ Listar las todas las Unidades de Salud, o se las puede filtrar para listar solo las públicas o privadas. 	<p>Especialidades</p> <p>Servicios</p> <p>Usuario</p>	<p>listaU()</p> <p>listaUPublicas()</p> <p>listaUPrivadas()</p> <p>infU(\$id)</p> <p>buscar()</p>

<ul style="list-style-type: none"> ○ Carga la información de una Unidad de Salud por medio de su Id. ○ Obtener la información de una unidad de salud en concreto. ○ Guardar nuevas centros de salud. ○ Modificar su información. ○ Eliminar unidad de salud ya sea física o lógica. 		<p>papelera()</p> <p>guardarUS()</p> <p>modificar(\$id)</p> <p>guardar()</p> <p>deAlta(\$idUS)</p> <p>eliminarL(\$idUS)</p> <p>eliminar(\$idUS)</p>
--	--	---

TABLA XXXIV: TARJETA CRC DE ESPECIALIDAD.

Nombre CRC: Especialidad		
Responsabilidades	Colaboradores	Métodos
<p>Esta clase permite:</p> <ul style="list-style-type: none"> ○ Crear, modificar y eliminar de forma lógica o física una especialidad. ○ Permite recuperar una especialidad eliminada de forma lógica. 	<p>Unidad de Salud</p> <p>Servicios</p>	<p>guardar(\$idUS)</p> <p>actualizar(\$idUS)</p> <p>eliminarL(\$idUS, \$idE)</p> <p>deAlta(\$idUS, \$idE)</p> <p>eliminar(\$idUS, \$idE)</p> <p>eliminarAE(\$idUS, \$idAE)</p> <p>crear(\$idUS, \$idAE)</p> <p>guardarE(\$idUS)</p>

<ul style="list-style-type: none"> ○ Permite eliminar una Área de Especialidad. 		
--	--	--

TABLA XXXV: TARJETA CRC DE SERVICIOS.

Nombre CRC: Servicios		
Responsabilidades	Colaboradores	Métodos
<p>Esta clase permite:</p> <ul style="list-style-type: none"> ○ Crear, modificar y eliminar de forma lógica o física un servicio. ○ Permite recuperar un servicio eliminada de forma lógica. 	Unidad de Salud Especialidad	guardar(\$idUS) editar(\$idUS) inforServicio(\$idC) eliminarL(\$idUS, \$idS) deAlta(\$idUS, \$idS) eliminar(\$idUS, \$idS)

6.4.9. Diseño de Prototipos de Pantalla

Para tener un mejor comprensión de las funcionalidades, características que el sistema ofrece, se ha realizado el prototipo de las aplicaciones móvil y web, las misma que facilitan una visión clara de los componentes a ser desarrollados.

6.4.9.1. Aplicación Móvil

Para la creación del prototipo de la aplicación se utilizó la plataforma que el framework Ionic proporciona, como es “Ionic Creator” permite generar código, crear interfaces robustas, exportarlas para usarlas directamente con el proyecto.

Las interfaces que se presentan a continuación son las más esenciales de la aplicación móvil como son la de inicio, google maps y la de información de unidades de salud, para conocer todas las interfaces revisar el anexo 4.

- **Interfaz de inicio**



Figura 12: Interfaz Inicial

El usuario puede acceder por medio de los botones a las funcionalidades que ofrece la aplicación:

- Buscar Unidades de Salud
- Buscar Especialidades
- Ayuda

- **Interfaz de Google Maps**

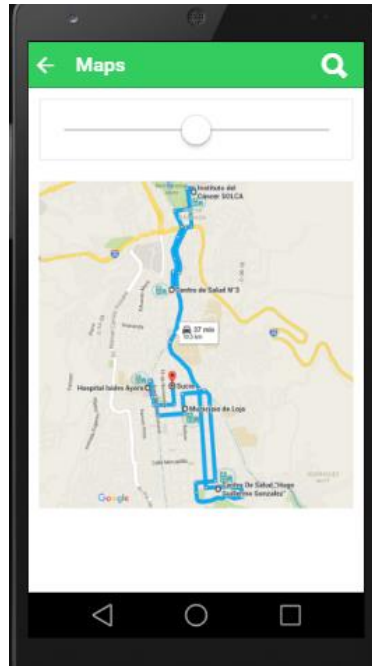


Figura 13: Interfaz de Google Maps

Esta interfaz permite localizar la unidad de salud más cercana a la posición del usuario, además, permite filtrar la búsqueda por medio de una barra personalizada, si este está a la izquierda, mostrará centros de salud públicos y si está a la derecha presentará centros de salud privados. En la parte superior izquierda se encuentra el botón de buscar, que permite cargar la interfaz para buscar unidades de salud por nombre.

- **Interfaz de Información Unidad de Salud**



Figura 14: Interfaz de Información de Unidades de Salud

En la presente interfaz se muestra la información de un centro de salud que se haya elegido, en ella se presenta la imagen e información adicional como tipología, sector, número de teléfono, dirección referente al establecimiento de salud, además posee dos botones, que cargan los datos dependiendo de su función:

- Áreas de Especialidad.
- Servicios.

6.4.9.2. Aplicación Web

El prototipo de la aplicación web fue creada en el software Axure RP, una herramienta que permite simular el comportamiento de la aplicación, como la generación de código, a continuación se presentan las interfaces más importantes de la aplicación SIUS, las cuales son: login, crear Unidad de Salud, crear especialidad y crear servicio.

La descripción de las interfaces de la aplicación web se encuentra en el anexo 5.

- **Interfaz de Login**

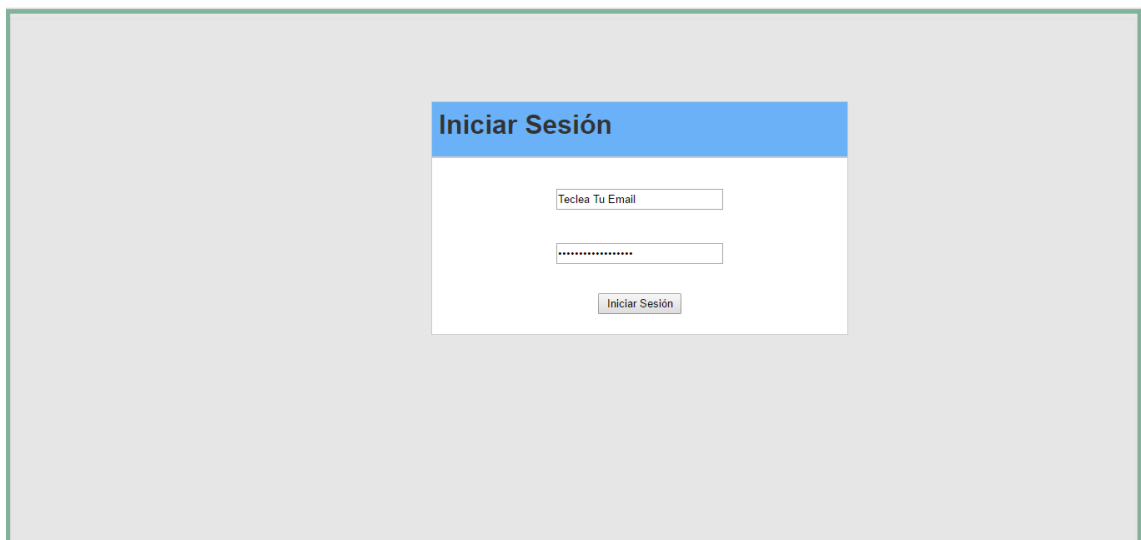
El prototipo de la interfaz de login muestra un formulario centrado en un fondo gris. El formulario tiene un encabezado azul con el texto "Iniciar Sesión". Debajo del encabezado, hay un campo de texto con el placeholder "Teclea Tu Email", un campo de contraseña con caracteres ocultos por puntos, y un botón que dice "Iniciar Sesión".

Figura 15: Interfaz de Login

El usuario debe ingresar su correo y contraseña, para tener acceso al sistema, en el caso que esta información sea correcta, caso contrario no podrá tener acceso.

- **Interfaz de Inicio**



Figura 16: Interfaz de Inicio

La interfaz de inicio permite tener acceso a todas las funcionalidades que ofrece la aplicación web, como son crear y modificar unidades de salud, especialidades, servicios.

- **Interfaz de Crear Unidad de Salud**



Figura 17: Interfaz de Crear Unidad de Salud

La interfaz permite al usuario por medio de un formulario crear una unidad de salud.

- **Interfaz de Crear Especialidad**

Sistema de Información Aplicación Móvil Bienvenido: admin | Tipo Usuario: ADMINISTRADOR [[Mis Datos] [Salir]

Menú Principal

- Unidades de Salud
 - Publicas
 - ISIDRO AYORA
 - Especialidad
 - Servicios
 - DANIEL ALVAREZ
 - ...
 - Privadas

Formulario de Especialidad

Nombre de la Especialidad ✓

Nombre del Medico Responsable ✓

Disponibilidad ✓

Horario de Atención ✓

Telefono ✓

Dias de Atención ✓

Figura 18: Interfaz de Crear Especialidad

La interfaz permite al usuario crear una especialidad por medio de un formulario.

- **Interfaz de Crear Servicio**

Sistema de Información Aplicación Móvil Bienvenido: admin | Tipo Usuario: ADMINISTRADOR [[Mis Datos] [Salir]

Menú Principal

- Unidades de Salud
 - Publicas
 - ISIDRO AYORA
 - Especialidad
 - Servicios
 - DANIEL ALVAREZ
 - ...
 - Privadas

Formulario de Servicio

Descripcion del Servicio ✓

Nombre del Responsable ✓

Disponibilidad ✓

Horario de Atención ✓

Numero Telefonico ✓

Figura 19: Interfaz de Crear Servicio

La interfaz permite al usuario crear un servicio por medio de un formulario.

6.5. FASE DE CODIFICACIÓN

Esta fase consiste en desarrollar la aplicación, tomando en cuenta el diseño y los requerimientos de software determinados en la fase previa, mediante el uso de lenguajes de programación, que dan como resultado un software presto para ser usado por los usuarios.

6.5.1. Codificación web

Se realizó la división por módulos, los cuales ayudaron a la abstracción de los componentes a desarrollar, tomando en cuenta el modelo-vista-controlador en la cual se basa el framework Codeigniter, este permitió llevar de manera ordenada la lógica de negocio y las interfaces, la división que se realizó es la siguiente:

Módulos:

- **Gestión de Unidades de Salud**
 - Ingresar datos de una nueva Unidad de Salud
 - Modificar información de una Unidad de Salud existente
 - Buscar información de unidades de salud
 - Eliminar de manera lógica y física el centro de salud
 - Listar los datos más importantes de los centros de salud
- **Gestión de Especialidad**
 - Ingresar los datos de una nueva especialidad
 - Actualizar la información de una especialidad
 - Eliminar de manera Lógica y física una especialidad
- **Gestión de Servicios**
 - Ingresar los datos de un nuevo servicio
 - Modificar la información de los servicios
 - Eliminar de manera lógica y física un servicio
- **Gestión de Contribuidor**
 - Eliminar el contribuidor con la información ingresada sobre un servicio específico perteneciente a una unidad de salud.
 - Listar los contribuidores con su información completa.
- **Gestión de Usuario**
 - Modificar la información del usuario

- **Google Maps**

- Mapa de Google con los marcadores de todas las unidades de salud
- Actualizar la posición de las Unidades de Salud representado en marcadores.

6.5.1.1. Aplicación Web Final

En la presente imagen se muestra la aplicación web SIUS, ya en su etapa final de desarrollo, donde dispone en la parte superior de un buscador de Unidades de Salud y en la parte derecha las notificaciones de los servicios ingresados por los contribuidores, las unidades eliminadas y opciones para salir del sistema, modificar información del usuario.



Figura 20: Interfaz Final de Administración SIUS



Figura 21: Interfaz Final con la Lista de las Unidades de Salud



Figura 22: Interfaz Final información de una Unidades de Salud

6.5.2. Codificación del Servicio Web

Para la creación del servicio web se usó una librería que fue desarrollada por Phil Sturgeon bajo el lenguaje PHP, cuyo nombre es CodeIgniter Rest Server, esta librería permite realizar la implementación del servicio de forma rápida, segura, sencilla, usando simplemente los archivos de la biblioteca, un archivo de configuración y un controlador, para su instalación.

La implantación de un servicio web permite el intercambio de datos entre aplicaciones, mediante el formato de envío de mensajes estandarizados, el cual es JSON, de forma

segura, rápida y eficiente, añadiendo funcionalidad a la aplicación web al ser integrada en un solo proyecto.

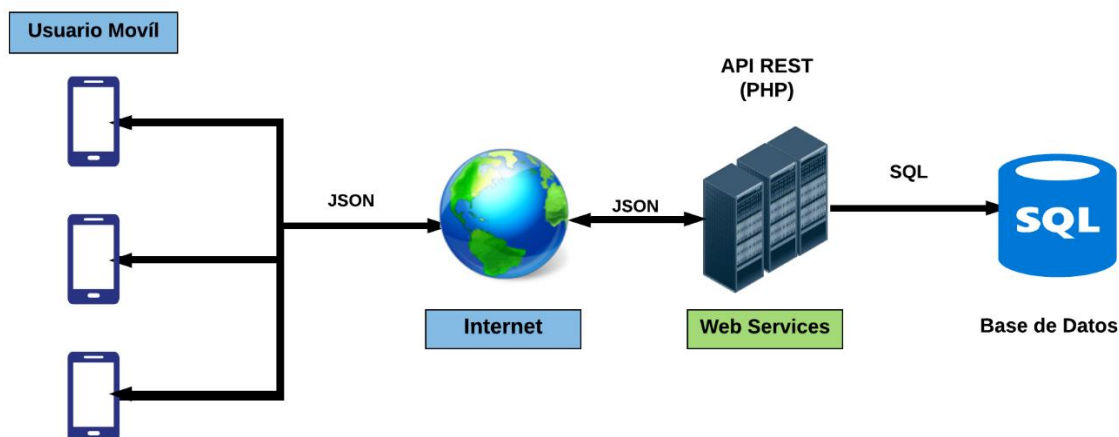


Figura 23: Diagrama del Servicio web

Su funcionamiento radica en las solicitudes (Get, Post, Put, Delete) que realice la aplicación móvil al servicio web a través de internet, este atiende la solicitud, realiza la consulta en la base de datos y responde en el formato Json.

6.5.2.1. Instalación

Arrastre y suelte los archivos `application/libraries/Format.php` y `application/libraries/REST_Controller.php` en los directorios de su aplicación. Para llamar las clases de la librería de escribe `require_once` en la parte superior de los controladores para cargarlo en el ámbito. Además, copie el archivo `rest.php` desde `application/config` en el directorio de configuración de su aplicación, y es todo para iniciar el desarrollo de los métodos del servicio web.

Cuando su controlador se extiende desde `REST_Controller`, los nombres de métodos se anexarán con el método HTTP utilizado para acceder a la solicitud.

Los métodos que soporta la librería `REST_Controller` son:

- `public function index_get()`
- `public function index_post()`
- `public function index_put()`
- `public function index_delete()`

Además es posible crear sus propios métodos, siempre y cuando estos tengan al final la operación a realizar, ejemplo `lucro_get()`.

6.5.2.2. Clases del API Rest

Las clases creadas y funciones del API REST utilizadas por la aplicación móvil para el consumo de datos son:

- **AreaEspecialidad_SW:**
 - **find_get(\$id):** Permite obtener los datos de una área de especialidad por medio del ID de las Unidad de Salud
 - **datos_get(\$nombreP):** Esta función permite obtener los datos relevantes de las unidades de salud, áreas de especialidad y especialidad por medio del nombre de la especialidad.
 - **datosId_get(\$id):** Permite obtener los datos de las especialidades por medio del ID de la Especialidad.
- **Contribuidor_SW:**
 - **index_post():** Permite crear un nuevo contribuidor.
 - **index_delete(\$id):** Permite eliminar un contribuidor por medio de su ID.
- **Especialidad_SW:**
 - **index_get():** Permite obtener la información de todas las especialidades.
 - **find_get(\$id):** Obtiene la información de las especialidades por medio del ID del área de especialidad.
- **Servicios_SW:**
 - **id_get(\$id):** Obtiene la información de todos los servicios por medio de la ID de una Unidad de Salud.
 - **disponibilidad_get():** Permite obtener los servicios que estén pendientes de validación.
 - **index_post():** Función que permite crear un nuevo servicio.
- **UnidadSalud_SW**
 - **find_get(\$id):** Obtiene la información de un centro de salud por medio del ID de la unidad de salud.
 - **lucro_get (\$id):** Obtiene información de los centros de salud de acure al ID enviado por parámetro, si es 1 la información respondida será las

unidades de salud sin fines de lucro, si el ID es 2 serán todas las unidades de salud, si el ID es 3 serán los centros de salud con fines de lucro.

- **disponibilidad_get():** Obtiene la información de los centros de salud cuyo estado sea 0.

6.5.3. Codificación de la Aplicación Móvil

La aplicación móvil se la construyo con el framework multiplataforma Ionic, usando como entorno de desarrollo Atom.

6.5.3.1. Controladores

Los controladores se los utiliza para realizar las funciones que debe desempeñar la aplicación móvil en una interfaz determinada. Hace uso de los servicios web, funciones de Angular Js, estos se denotan con “Ctrl” al final del nombre, los controladores usados en la aplicación móvil son:

- buscarUnidadSaludCtrl
- mapsCtrl
- informacionUnidadDeSaludCtrl
- areasDeEspecialidadCtrl
- serviciosCtrl
- buscarEspecialidadesCtrl
- resultadoBusquedaECtrl
- informacionResultadoBECtrl
- ayudaCtrl
- inicioCtrl

6.5.3.2. Aplicación Móvil Final

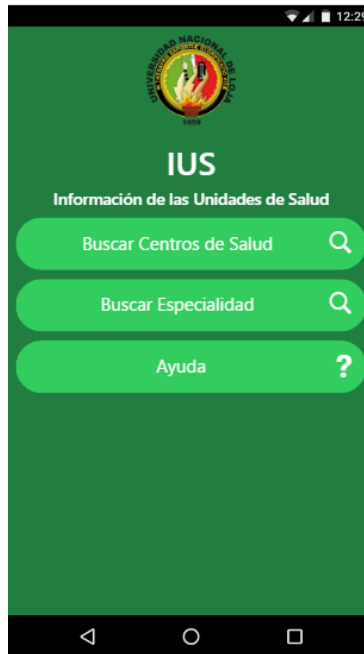


Figura 24: Interfaz Final Inicio de la aplicación móvil IUS



Figura 25: Interfaz Final de Buscar Especialidades

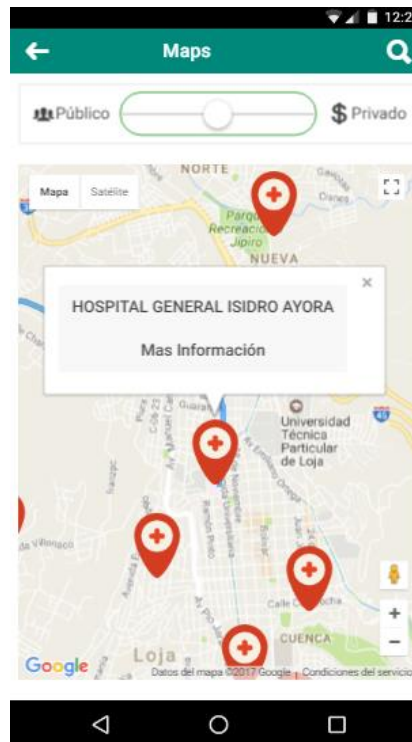


Figura 26: Interfaz Final de Google Maps



Figura 27: Interfaz Final información de una Unidad de Salud

6.6. FASE DE PRUEBAS

6.6.1. Herramientas de Prueba

Para analizar el tiempo de respuesta del servidor se utilizó la herramienta Jmeter, una aplicación 100% Java diseñada para cargar el comportamiento funcional y medir el rendimiento.

Jmeter se puede utilizar para probar el rendimiento tanto en recursos estáticos y dinámicos, aplicaciones dinámicas Web.

Se puede utilizar para simular una carga pesada en un servidor, grupo de servidores, la red o el objeto a probar su resistencia o para analizar el rendimiento general bajo diferentes tipos de carga [50], [51].

- **Características**

Presenta las siguientes características [51]:

- Capacidad de cargar y pruebas de rendimiento muchos tipos diferentes aplicaciones / servidor / protocolo:
 - Web - HTTP, HTTPS (Java, NodeJS, PHP, ASP.NET, ...)
 - SOAP / REST Servicios Web
 - FTP
 - Base de datos a través de JDBC
 - LDAP
 - Middleware orientado a mensajes (MOM) a través de JMS
 - Comandos o scripts de shell nativa
 - TCP
 - Objetos Java
- IDE con todas las funciones de prueba que permite Plan de prueba rápida de grabación (de navegadores o aplicaciones nativas), construcción y depuración.
- El modo de línea de comandos para cargar de prueba desde cualquier sistema operativo compatible con Java.
- Una completa y el informe listo para presentar HTML dinámico
- Extracción datos de la mayoría de los formatos de respuesta populares, HTML , JSON , XML o formato de cualquier texto

- La portabilidad completa y 100% de pureza de Java.
- Completa multi-threading marco permite el muestreo simultáneo de muchos hilos y muestreo simultáneo de funciones diferentes por grupos de hilos separados.
- El almacenamiento en caché y fuera de línea de análisis / Reproducción de resultados de la prueba.
- Análisis de datos y visualización plugins permiten una gran extensibilidad, así como la personalización.
- Las funciones pueden ser utilizadas para proporcionar la entrada dinámica a una prueba o proporcionar la manipulación de datos.

6.6.2. Plan de Pruebas

El presente plan de pruebas, se ha desarrollado para documentar el comportamiento y funcionamiento del sistema a través de diferentes pruebas de evaluación, para el cual se ha adoptado el modelo de pruebas especificado en el estándar IEEE 829 [52], que es el estándar de Documentación de pruebas de Software, adaptado a las características de proyecto que se está desarrollando.

6.6.2.1. Propósito

Este plan de pruebas tiene el propósito de realizar una evaluación de las funcionalidades del sistema, además son requeridas para brindar mayor calidad al producto de este proyecto.

El siguiente plan de pruebas tiene como propósito los siguientes objetivos:

- Identificar la información existente en el proyecto y los componentes que deben ser ponerse a prueba.
- Enumerar los principales requisitos a probar.
- Definir las estrategias de prueba que deben emplearse.
- Identificar los recursos necesarios que pueden requerirse.
- Presentar los resultados de las pruebas.

6.6.2.2. Alcance

El principal propósito de la evaluación es encontrar errores que puedan existir durante el testeo y uso del sistema a fin de corregirlos, las pruebas a realizarse se encuentran

dentro del proceso o marco de desarrollo de la metodología que se ha definido, XP (Extreme Programming): Fase de Pruebas.

6.6.2.3. Características a probar

Las pruebas a desarrollarse son para verificar la funcionalidad de la aplicación móvil además del servicio web con sus respectivos métodos get, post, delete, usados por la app móvil para el consumo de datos.

6.6.2.4. Características que no van a ser evaluadas

- Herramientas no desarrolladas (Sistema operativo, navegador web, IDEs)
- Entorno de trabajo (disponibilidad de internet)

6.6.2.5. Actividades a Realizar

Las pruebas a desarrollar son las siguientes:

- Pruebas unitarias
- Pruebas de Aceptación
- Pruebas de Carga
- Prueba de accesibilidad

6.6.2.6. Pruebas a Realizar

El plan de prueba está dividido en:

Pruebas unitarias. Para las pruebas unitarias, se hará uso de un software para automatizar y agilizar el testeo, esta herramienta además debe hacer posible la obtención de un reporte total de las funciones probadas y su estado.

La herramienta usada para la realización de las pruebas unitarias de los servicios web es Advanced Rest Client.

Este software gratuito nos permite evaluar el desempeño de cada función que es utilizada por la aplicación móvil, brindándonos el resultado de las pruebas los parámetros necesarios que nos permite observar si la función está realizando las operaciones requeridas correctamente.

Pruebas funcionales. Estas pruebas se basan en comprobar el funcionamiento de los procedimientos requeridos por el sistema, las mismas que se encuentran especificados

en las historias de usuario (HU). En base a estas se ha desarrollado un documento de especificación de los casos de prueba para las HU que están siendo evaluadas.

Pruebas de carga. Este tipo de prueba se las realiza para evaluar el comportamiento de las aplicaciones bajo una cantidad específica de peticiones, para llevar a cabo esto ha sido utilizada la herramienta JMeter, la misma que nos permite realizar las pruebas y generar un reporte detallado con los resultados de las mismas.

Pruebas de Accesibilidad. Para llevar a cabo las pruebas de accesibilidad se utilizó el siguiente URL: <http://examinator.ws/>, que es un servicio en línea para evaluar de modo automático la accesibilidad de una página web, y usa como referencia algunas técnicas recomendadas por las pautas de accesibilidad para el Contenido Web 2.0 (WCAG 2.0).

6.6.2.7. Actividades de prueba

Las actividades que se llevan a cabo en el desarrollo de las pruebas son las siguientes:

- Crear casos para evaluar las pruebas
- Ejecutar los casos de pruebas, realizar el análisis, evaluación e informe de las mismas.
- Llevar a cabo las pruebas de aceptación.

6.6.2.8. Recursos

En la presente tabla se detallan los recursos que son necesarios para llevar a cabo la realización de las pruebas.

TABLA XXXVI: RECURSOS HUMANOS- PLAN DE PRUEBAS.

Recursos Humanos- Plan de Pruebas		
Nombre del Recurso	Mínimo de recursos recomendados	Especificación de responsabilidades
Jefe de Pruebas	1	<ul style="list-style-type: none"> • Dirige el flujo de trabajo y asigna responsabilidades. • Gestiona informes y administra recursos
Diseñador de Pruebas	1	<ul style="list-style-type: none"> • Identifica, prioriza e implementa casos de prueba. • Evaluar efectividad de las pruebas realizadas

Probador	1	<ul style="list-style-type: none"> • Realizar las pruebas. • Registrar y detallar los resultados en un informe
----------	---	--

TABLA XXXVII: RECURSOS TECNOLÓGICOS – PLAN DE PRUEBAS.

Recursos Tecnológicos – Plan de Pruebas		
Recursos	Mínimo de recursos recomendados	Nombre – Tipo
Servidor	1	Apache/PHP/MySql
Herramientas: Jmeter	1	Automatización de pruebas unitarias de servicios web

6.6.2.9. Resultados

Los resultados obtenidos de las pruebas están representados en formularios, en los que se definen los aspectos importantes en la ejecución y una descripción detallada de los resultados obtenidos.

Los estados del resultado de las pruebas son:

- **Exitoso:** El resultado de la prueba ha sido satisfactorio.
- **Detenido:** Presencia de dificultades para la correcta ejecución de la prueba.
- **Fallido:** El resultado de la prueba no ha sido el esperado.

6.6.2.10. Pruebas Unitarias de funcionalidad

Las pruebas son muy importantes para verificar el correcto funcionamiento de una unidad del código, éste debe estar libre de errores y en caso de tenerlos, deben ser corregidos.

Las pruebas unitarias se centran en tomar la menor unidad posible para realizar las pruebas, sean métodos de una clase, detectando así que la porción de código que estamos evaluando se encuentre sin errores para su posterior integración con el resto del código [53].

La herramienta Advance Rest Client fue necesaria para realizar las pruebas de funcionalidad de los servicios web que son utilizados para el consumo de la aplicación móvil.

A continuación se muestra las pruebas realizadas al servicio web, tomando en cuenta las clases y funciones creadas en la fase de codificación, las cuales son:

6.6.2.10.1. Historia de usuario

TABLA XXXVIII: PRUEBA DE ACEPTACIÓN.

Prueba De Aceptación	
Título Historia de Usuario:	Gestión de Servicio Web
Descripción:	Evaluar el rendimiento, funcionamiento, respuesta de las funciones del API Rest que son utilizadas por la aplicación móvil para el consumo de datos.

- **Clase AreaEspecialidad_SW**
 - **Función AreaEspecialidad_get**

TABLA XXXIX: PRUEBA DE ACEPTACIÓN A LA CLASE AREAESPECIALIDAD_SW
FUNCIÓN FIND_GET.

Nombre De La Clase De Prueba: AreaEspecialidad_SW

Nombre Del Caso De Prueba:	find_get(\$id)
Tipo:	Prueba Funcional –Aceptación
Descripción:	Esta función permite obtener los datos de una área de especialidad por medio del ID de una Unidad de Salud.
Condiciones De Ejecución	Que se haya accedido al servicio web
Entrada	Id de la Unidad de Salud como parámetro
Resultado Esperado	La información de las áreas de Especialidad pertenecientes a la Unidad de Salud enviada por parámetro.
Estado Del Resultado	Exitoso

Resultado obtenido de realizar la petición find (Buscar) al servicio web.

```
{
  -"response": [Array[2]
    -0: {
      "unidad_SaludIDUNIDADSalud": "165",
      "unidad_SaludUNICODIGO": "1145",
      "IDAREA_ESPECIALIDAD": "1",
      "NOMBRE": "EMERGENCIA"
    },
    -1: {
      "unidad_SaludIDUNIDADSalud": "165",
      "unidad_SaludUNICODIGO": "1145",
      "IDAREA_ESPECIALIDAD": "2",
      "NOMBRE": "CONSULTA EXTERNA"
    }
  ],
}
```

Figura 28: JSON de la petición Find de la clase AreasEspecialidad_SW

○ **Función datos_get**

TABLA XL: PRUEBA DE ACEPTACIÓN A LA CLASE AREAESPECIALIDAD_SW DE LA FUNCIÓN DATOS_GET.

Nombre De La Clase De Prueba: AreaEspecialidad_SW	
Nombre Del Caso De Prueba:	datos_get(\$nombreP)
Tipo:	Prueba Funcional –Aceptación
Descripción:	Esta función permite obtener los datos relevantes de las unidades de salud, áreas de especialidad y especialidad por medio del nombre de la especialidad.
Condiciones De Ejecución	Que se haya accedido al servicio web
Entrada	Nombre de la especialidad
Resultado Esperado	La información de las áreas de especialidad, unidad de salud y especialidad.
Estado Del Resultado	Exitoso

Resultado obtenido de realizar le petición datos_get al servicio web.

```
{
  - "response": [Array[1]
    - 0: {
      "IDESPECIALIDAD": "21",
      "TELEFONO": "",
      "NOMBRE_OFICIAL": "HOSPITAL GENERAL ISIDRO AYORA",
      "LUCRO": "NO DEFINIDO",
      "FOTO": "uploads/HIA.jpg",
      "NOMBRE_AE": "EMERGENCIA"
    }
  ],
}
```

Figura 29: JSON de la petición datos_get de la clase AreasEspecialidad_SW

○ **Función datosId_get**

TABLA XLI: PRUEBA DE ACEPTACIÓN A LA CLASE AREAESPECIALIDAD_SW DE LA FUNCIÓN DATOSID_GET.

Nombre De La Clase De Prueba: AreaEspecialidad_SW	
Nombre Del Caso De Prueba:	datosId_get(\$id)
Tipo:	Prueba Funcional –Aceptación
Descripción:	Esta función permite obtener los datos relevantes de las unidades de salud, áreas de especialidad y especialidad por medio del ID de la Especialidad.

Condiciones De Ejecución	Que se haya accedido al servicio web
Entrada	ID de la especialidad
Resultado Esperado	La información de las áreas de especialidad, unidad de salud y especialidad.
Estado Del Resultado	Exitoso

Resultado obtenido de realizar le petición datos_get al servicio web.

```

{
  -"response": [Array[1]
    -0: {
      "NOMBRE_OFICIAL": "HOSPITAL GENERAL ISIDRO AYORA",
      "FOTO": "uploads/HIA.jpg",
      "TELEFONO": "072570540",
      "PROVINCIA": "LOJA",
      "TIPOLOGIA": "HOSPITAL GENERAL",
      "NIVEL_ATENCION": "NIVEL 2",
      "DIRECCION": "J. J. SAMANIEGO E IBEROAMERICA",
      "LUCRO": "NO DEFINIDO",
      "NOMBRE_AE": "EMERGENCIA",
      "NOMBRE_ESPECIALIDAD": "Sala de Emergencia ",
      "TELEFONO_ESPECIALIDAD": "",
      "HORARIO_ATENCION": "24H00",
      "DIAS_ATENCION": "LUNES A DOMINGO"
    }
  ],
}

```

Figura 30: JSON de la petición datosId_get de la clase AreasEspecialidad_SW

- **Clase Contribuidor_SW**
 - **Funcion index_post**

TABLA XLII: PRUEBA DE ACEPTACIÓN A LA CLASE CONTRIBUIDOR_SW DE LA FUNCIÓN INDEX_POST.

Nombre De La Clase De Prueba: Contribuidor_SW

Nombre Del Caso De Prueba:	index_post()
Tipo:	Prueba Funcional –Aceptación
Descripción:	Permite crear un nuevo contribuidor.
Condiciones De Ejecución	Que se haya accedido al servicio web
Entrada	<ul style="list-style-type: none"> • Nombre • Apellido • Correo • Teléfono
Resultado Esperado	ID del nuevo contribuidor creado
Estado Del Resultado	Exitoso

Resultado obtenido de realizar le petición index_post al servicio web.

Status: 200: OK ? Loading time: 1260 ms

Response headers 12 Request headers 3 Redirects 0 Timings

```

Date: Wed, 18 Jan 2017 19:43:42 GMT
Server: Apache/2.4.2 (Win64) PHP/5.4.3
X-Powered-By: PHP/5.4.3
Access-Control-Allow-Origin: *
Access-Control-Allow-Headers: X-API-KEY, Origin, X-Requested-With, Content-Type: application/json, Accept, Access-Control-Request-Method
Access-Control-Allow-Methods: GET, POST, OPTIONS, PUT, DELETE
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache
Set-Cookie: ci_session=e3b03bd2b95f0c94e900831a821d4b2c78cff1d0; expires=Wed, 18-Jan-2017 21:43:43 GMT; path=/; HttpOnly
Content-Length: 14
Content-Type: application/json; charset=utf-8

```

```

{
  "response": 9
}

```

Figura 31: JSON de la petición `index_post` de la clase `Contribuidor_SW`

- **Funcion `index_delete`**

TABLA XLIII: PRUEBA DE ACEPTACIÓN A LA CLASE `CONTRIBUIDOR_SW` DE LA FUNCIÓN `INDEX_DELETE`.

Nombre De La Clase De Prueba: `Contribuidor_SW`

Nombre Del Caso De Prueba:	<code>index_delete(\$id)</code>
Tipo:	Prueba Funcional –Aceptación
Descripción:	Permite eliminar un contribuidor por medio de su ID.
Condiciones De Ejecución	Que se haya accedido al servicio web
Entrada	ID del Contribuidor
Resultado Esperado	La información de las áreas de especialidad, unidad de salud y especialidad.
Estado Del Resultado	Exitoso

Resultado obtenido de realizar le petición `index_delete` al servicio web.

```

{
  "response": "Contribuidor Eliminado"
}

```

Figura 32: JSON de la petición `index_delete` de la clase `Contribuidor_SW`

- **Clase `Especialidad_SW`**
 - **Funcion `index_get`**

TABLA XLIV: PRUEBA DE ACEPTACIÓN A LA CLASE ESPECIALIDAD_SW DE LA FUNCIÓN INDEX_GET.

Nombre De La Clase De Prueba: Especialidad_SW

Nombre Del Caso De Prueba:	index_get()
Tipo:	Prueba Funcional –Aceptación
Descripción:	Permite obtener la información de todas las especialidades.
Condiciones De Ejecución	Que se haya accedido al servicio web
Entrada	
Resultado Esperado	La información de todas las especialidades.
Estado Del Resultado	Exitoso

Resultado obtenido de realizar le petición index_get al servicio web.

```
{
  - "response": [Array[4]
    -0: {
      "area_especialidadIDAREA_ESPECIALIDAD": "1",
      "IDESPECIALIDAD": "20",
      "NOMBRE": "Sala de Emergencia ",
      "DISPONIBILIDAD": "1"
    },
    -1: {
      "area_especialidadIDAREA_ESPECIALIDAD": "1",
      "IDESPECIALIDAD": "21",
      "NOMBRE": "Triaje",
      "DISPONIBILIDAD": "1"
    },
    -2: {
      "area_especialidadIDAREA_ESPECIALIDAD": "1",
      "IDESPECIALIDAD": "22",
      "NOMBRE": "Sala de Shock Trauma",
      "DISPONIBILIDAD": "1"
    },
    -3: {
      "area_especialidadIDAREA_ESPECIALIDAD": "2",
      "IDESPECIALIDAD": "23",
      "NOMBRE": "Medicina Interna",
      "DISPONIBILIDAD": "1"
    }
  ],
}
```

Figura 33: JSON de la petición index_get de la clase Especialidad_SW

- **Funcion find_get**

TABLA XLV: PRUEBA DE ACEPTACIÓN A LA CLASE ESPECIALIDAD_SW DE LA FUNCIÓN FIND_GET.

Nombre De La Clase De Prueba: Especialidad_SW

Nombre Del Caso De Prueba:	find_get(\$id)
Tipo:	Prueba Funcional –Aceptación
Descripción:	Obtiene la información de las especialidades por medio del ID del área de especialidad.
Condiciones De Ejecución	Que se haya accedido al servicio web
Entrada	ID del Área de Especialidad
Resultado Esperado	La información de todas las especialidades pertenecientes al ID enviado por parámetro.
Estado Del Resultado	Exitoso

Resultado obtenido de realizar le petición find_get al servicio web.

```

{
  -"response": [Array[3]
    -0: {
      "area_especialidadIDAREA_ESPECIALIDAD": "1",
      "IDESPECIALIDAD": "20",
      "NOMBRE": "Sala de Emergencia ",
      "DISPONIBILIDAD": "1",
      "TELEFONO": "",
      "NOMBRE_PROFESIONAL_RESPONSABLE": "JORGE GUAPULEMA",
      "HORARIO_ATENCION": "24H00",
      "DIAS_ATENCION": "LUNES A DOMINGO"
    },
    -1: {
      "area_especialidadIDAREA_ESPECIALIDAD": "1",
      "IDESPECIALIDAD": "21",
      "NOMBRE": "Triage",
      "DISPONIBILIDAD": "1",
      "TELEFONO": "",
      "NOMBRE_PROFESIONAL_RESPONSABLE": "ANGELICA GORDILLO",
      "HORARIO_ATENCION": "24H00",
      "DIAS_ATENCION": "LUNES A DOMINGO"
    },
    -2: {
      "area_especialidadIDAREA_ESPECIALIDAD": "1",
      "IDESPECIALIDAD": "22",
      "NOMBRE": "Sala de Shock Trauma",
      "DISPONIBILIDAD": "1",
      "TELEFONO": "",
      "NOMBRE_PROFESIONAL_RESPONSABLE": "JORGE GUAPULEMA",
      "HORARIO_ATENCION": "24H00",
      "DIAS_ATENCION": "LUNES A DOMINGO"
    }
  ],
}

```

Figura 34: JSON de la petición find_get de la clase Especialidad_SW

- **Clase Servicios_SW**
 - **Funcion id_get**

TABLA XLVI: PRUEBA DE ACEPTACIÓN A LA CLASE SERVICIOS_SW DE LA FUNCIÓN ID_GET.

Nombre De La Clase De Prueba: Servicios_SW

Nombre Del Caso De Prueba:	id_get(\$id)
Tipo:	Prueba Funcional –Aceptación
Descripción:	Obtiene la información de todos los servicio por medio de la ID de una Unidad de Salud.
Condiciones De Ejecución	Que se haya accedido al servicio web
Entrada	ID de la Unidad de Salud

Resultado Esperado	La información de todos los servicios que pertenecen a la ID Unidad de Salud
Estado Del Resultado	Exitoso

Resultado obtenido de realizar le petición widget al servicio web.

```

{
  - "response": [Array[1]
    - 0: {
      "unidad_SaludIDUNIDADSAUD": "165",
      "unidad_SaludUNICODIGO": "1145",
      "contribuidorIDCONTRIBUIDOR": "8",
      "IDSERVICIO": "31",
      "DESCRIPCION": "AMBULANCIAS",
      "DISPONIBILIDAD": "1",
      "TELEFONO": "72570540",
      "NOMBRE_RESPONSABLE": "FABIAN VALDIVIESO",
      "HORARIO": "24H00 LUNES A DOMINGO"
    }
  ],
}

```

Figura 35: JSON de la petición widget de la clase Servicios_SW

- **Funcion disponibilidad_get**

TABLA XLVII: PRUEBA DE ACEPTACIÓN A LA CLASE SERVICIOS_SW DE LA FUNCIÓN DISPONIBILIDAD_GET.

Nombre De La Clase De Prueba: Servicios_SW

Nombre Del Caso De Prueba:	disponibilidad_get(\$id)
Tipo:	Prueba Funcional –Aceptación
Descripción:	Permite obtener los servicios que estén pendientes de validación.
Condiciones De Ejecución	Que se haya accedido al servicio web
Entrada	
Resultado Esperado	La información de todos los servicios que estan con estado dos, es decir pendientes de validación.
Estado Del Resultado	Exitoso

Resultado obtenido de realizar le petición disponibilidad_get al servicio web.

```

{
  -"response": {
    -0: {
      "unidad_SaludIDUNIDADSalud": "168",
      "unidad_SaludUNICODIGO": "111",
      "contribuidorIDCONTRIBUIDOR": "6",
      "IDSERVICIO": "47",
      "DESCRIPCION": "Servicios",
      "DISPONIBILIDAD": "2",
      "TELEFONO": "987654321",
      "NOMBRE_RESPONSABLE": "Juan",
      "HORARIO": "No se"
    },
    "length": 1
  }
}

```

Figura 36: JSON de la petición disponibilidad_get de la clase Servicios

o **Funcion index_post**

TABLA XLVIII: PRUEBA DE ACEPTACIÓN A LA CLASE SERVICIOS_SW DE LA FUNCIÓN INDEX_POST.

Nombre De La Clase De Prueba: Servicios_SW

Nombre Del Caso De Prueba:	index_post()
Tipo:	Prueba Funcional –Aceptación
Descripción:	Función que permite crear un nuevo servicio.
Condiciones De Ejecución	Que se haya accedido al servicio web
Entrada	<ul style="list-style-type: none"> • Descripción • Teléfono • Nombre Responsable • Horario
Resultado Esperado	ID del nuevo servicio creado
Estado Del Resultado	Exitoso

Resultado obtenido de realizar le petición index_post al servicio web.

```

{
  "response": 40
}

```

Figura 37: JSON de la petición index_post de la clase Servicios_SW

- Clase UnidadSalud_SW
 - Funcion index_get

TABLA XLIX: PRUEBA DE ACEPTACIÓN A LA CLASE UNIDADSalUD_SW DE LA FUNCIÓN INDEX_GET.

Nombre De La Clase De Prueba: UnidadSalud_SW

Nombre Del Caso De Prueba:	index_get()
Tipo:	Prueba Funcional –Aceptación
Descripción:	Obtiene la información de todos los centros de salud, públicos y privados
Condiciones De Ejecución	Que se haya accedido al servicio web
Entrada	
Resultado Esperado	Los datos relevantes de las unidades de salud publicas y privadas
Estado Del Resultado	Exitoso

Resultado obtenido de realizar le petición index_get al servicio web.

```

{
  -"response": [Array[28]
    -0: {
      "IDUNIDADSalUD": "135",
      "UNICODIGO": "1086",
      "NOMBRE_OFICIAL": "TIERRAS COLORADAS",
      "DIRECCION": "AV. EUGENIO ESPEJO",
      "TELEFONO": "0991482192",
      "LUCRO": "NO DEFINIDO",
      "LONGITUD": "-79.15237426757812",
      "LATITUD": "-3.967011000868007",
      "FOTO": "uploads/US_defect.jpg"
    },
    -1: {
      "IDUNIDADSalUD": "136",
      "UNICODIGO": "1084",
      "NOMBRE_OFICIAL": "OBRAPIA",
      "DIRECCION": "AV. VILLONACO, JUNTO AL COLEGIO MANUEL IGNACIO MONTEROS",
      "TELEFONO": "072326700",
      "LUCRO": "NO DEFINIDO",
      "LONGITUD": "-79.22410860657692",
      "LATITUD": "-3.999411414978857",
      "FOTO": "uploads/US_defect.jpg"
    },
    -2: {
      "IDUNIDADSalUD": "137",
      "UNICODIGO": "1096",
      "NOMBRE_OFICIAL": "ZAMORA HUAYCO",
      "DIRECCION": "AV. RIO MARAÑON ENTRE RIO CURARAY Y RIO GUEPI",
      "TELEFONO": "072139351",
      "LUCRO": "NO DEFINIDO",
      "LONGITUD": "-79.18704316020012",
    }
  ]
}

```

Figura 38: JSON de la petición index_get de la clase UnidadSalud_SW

- **Funcion find_get**

TABLA L: PRUEBA DE ACEPTACIÓN A LA CLASE UNIDADSalUD_SW DE LA FUNCIÓN FIND_GET.

Nombre De La Clase De Prueba: UnidadSalud_SW

Nombre Del Caso De Prueba:	find_get(\$id)
Tipo:	Prueba Funcional –Aceptación
Descripción:	Obtiene la información de un centro de salud por medio del ID de la unidad de salud
Condiciones De Ejecución	Que se haya accedido al servicio web
Entrada	ID de la Unidad de Salud
Resultado Esperado	La información de la Unidad de Salud que envió como parámetro ID.
Estado Del Resultado	Exitoso

Resultado obtenido de realizar le petición find_get al servicio web.

```

{
  -"response": [Array[1]
    -0: {
      "IDUNIDADSalUD": "165",
      "UNICODIGO": "1145",
      "NOMBRE_OFICIAL": "HOSPITAL GENERAL ISIDRO AYORA",
      "DIRECCION": "J. J. SAMANIEGO E IBEROAMERICA",
      "TELEFONO": "072570540",
      "PROVINCIA": "LOJA",
      "LUCRO": "NO DEFINIDO",
      "NIVEL_ATENCION": "NIVEL 2",
      "TIPOLOGIA": "HOSPITAL GENERAL",
      "FOTO": "uploads/HIA.jpg"
    }
  ],
}

```

Figura 39: JSON de la petición find_get de la clase UnidadSalud_SW

- **Funcion lucro_get**

TABLA LI: PRUEBA DE ACEPTACIÓN A LA CLASE UNIDADSalUD_SW DE LA FUNCIÓN LUCRO_GET.

Nombre De La Clase De Prueba: UnidadSalud_SW

Nombre Del Caso De Prueba:	lucro_get(\$id)
Tipo:	Prueba Funcional –Aceptación
Descripción:	Obtiene información de los centros de salud de acuerdo al ID enviado por parámetro, si es 1 la informacion respondida será las unidades de salud sin fines de lucro, si el ID es 2 serán todas las unidades de salud, si el ID es 3 serán los centros de salud con fines de lucro.

Condiciones De Ejecución	Que se haya accedido al servicio web
Entrada	ID de la operación a realizar
Resultado Esperado	La información de todas las unidades de salud, dependiendo del ID enviando por parámetro.
Estado Del Resultado	Exitoso

Resultado obtenido de realizar le petición lucro_get al servicio web.

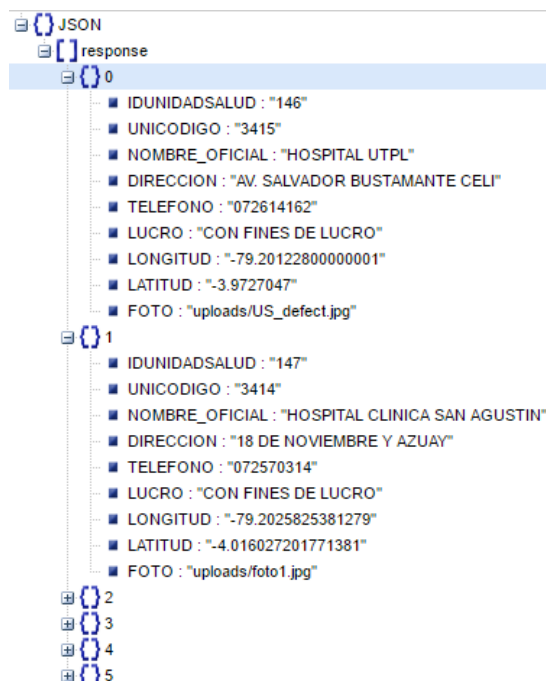


Figura 40: JSON de la petición lucro_get de la clase UnidadSalud_SW

○ **Funcion disponibilidad_get**

TABLA LII: PRUEBA DE ACEPTACIÓN A LA CLASE UNIDADSAUD_SW DE LA FUNCIÓN DISPONIBILIDAD_GET.

Nombre De La Clase De Prueba: UnidadSalud_SW

Nombre Del Caso De Prueba:	disponibilidad_get():
Tipo:	Prueba Funcional –Aceptación
Descripción:	Obtiene la información de los centros de salud cuyo estado sea 0.
Condiciones De Ejecución	Que se haya accedido al servicio web
Entrada	
Resultado Esperado	La información de todos los servicios que estan con estado dos, es decir pendientes de validación.
Estado Del Resultado	Exitoso

Resultado obtenido de realizar la petición disponibilidad_get al servicio web.

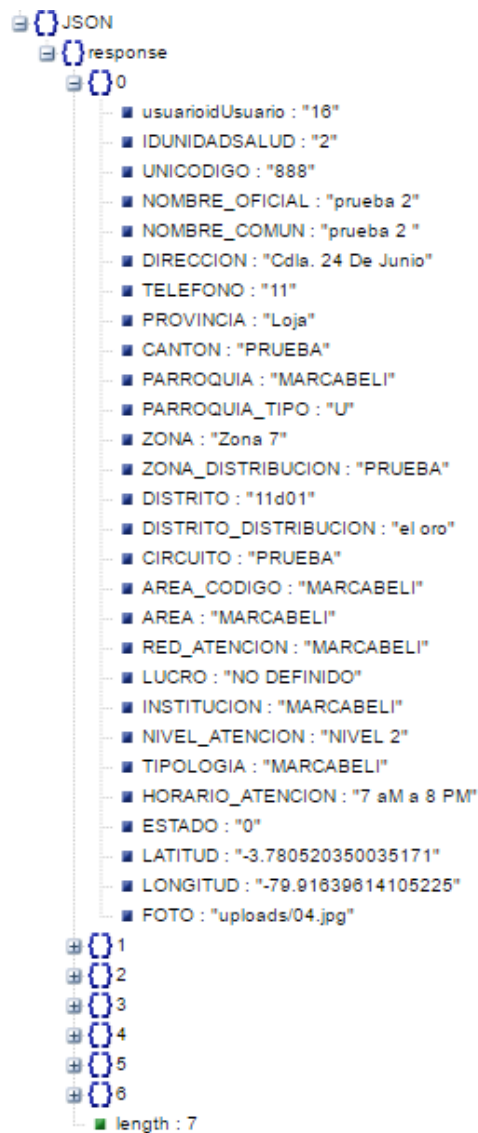


Figura 41: JSON de la petición disponibilidad_get de la clase UnidadSalud_SW

6.6.2.11. Pruebas de Aceptación

Las pruebas de aceptación permiten verificar la funcionalidad operativa del software, evaluando la calidad de las aplicaciones, si este funciona y lo hace de acuerdo con las especificaciones determinadas mediante la técnica de recolección de información como la encuesta realizada al Dr. Carlos Celi, (Véase en el anexo V).

6.6.2.12. Pruebas de Carga

Estas pruebas se las realiza para determinar lo velocidad, el rendimiento, funcionalidad con la que el sistema realiza una determinada tarea, bajo condiciones particulares de trabajo, permitiendo validar y verificar la calidad del software.

- **Prueba de Carga al Servicio Web**

Para la elaboración de las pruebas de carga se ha utilizado la herramienta JMeter, que nos permite evaluar el comportamiento y rendimiento del sistema, creando un grupo de hilos que simulan a un número de usuarios, que tienen la función de realizar peticiones al servicio web, para este caso de prueba se ha considerado 200 hilos (usuarios) con una frecuencia de 5 segundos.

Para mayor información de las pruebas realizadas, revisar el anexo 6, en donde se detallan los métodos get, post, delete, de las clases resatantes en esta sección solo se detalla de la clase unidad de salud.

- **Caso de Prueba de la Clase UnidadSalud_SW**
 - **Funcion index_get**

Petición HTTP

Nombre:

Comentarios:

Basic **Advanced**

Servidor Web

Nombre de Servidor o IP: Puerto:

Timeout (milisegundos)

Conexión: Respuesta:

Petición HTTP

Implementación HTTP: Protocolo: Método: Codificación del contenido:

Ruta:

Figura 42: Prueba de la clase UnidadSalud_SW, función index_get

Ver Resultados en Árbol

Nombre: Resultados de index_get

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo: win\Desktop\pruebas\Ver Resultados en Árbol.jmx Log/Mostrar sólo: Escribir en Log Sólo Errores Éxitos

Muestra #	Tiempo de co...	Nombre del hilo	Etiqueta	Tiempo de Mu...	Estado	Bytes	Sent Bytes	Latency	Connect Time(...)
1	10:07:42.726	Grupo 1 1-2	index_get	1284	✓	8354	207	1284	4
2	10:07:42.703	Grupo 1 1-1	index_get	1315	✓	8354	207	1315	3
3	10:07:42.844	Grupo 1 1-6	index_get	1202	✓	8354	207	1202	1
4	10:07:42.766	Grupo 1 1-4	index_get	1290	✓	8354	207	1290	1
5	10:07:42.844	Grupo 1 1-7	index_get	1273	✓	8354	207	1273	1
6	10:07:42.846	Grupo 1 1-8	index_get	1280	✓	8354	207	1280	1
7	10:07:42.886	Grupo 1 1-10	index_get	1332	✓	8354	207	1331	1
8	10:07:42.745	Grupo 1 1-3	index_get	1484	✓	8354	207	1484	1
9	10:07:42.844	Grupo 1 1-5	index_get	1393	✓	8354	207	1393	1
10	10:07:42.866	Grupo 1 1-9	index_get	1397	✓	8354	207	1397	1
11	10:07:42.906	Grupo 1 1-11	index_get	1397	✓	8354	207	1397	1
12	10:07:42.986	Grupo 1 1-15	index_get	2171	✓	8354	207	2171	1
13	10:07:42.966	Grupo 1 1-14	index_get	2217	✓	8354	207	2217	1
14	10:07:43.026	Grupo 1 1-17	index_get	2197	✓	8354	207	2197	1
15	10:07:43.066	Grupo 1 1-19	index_get	2161	✓	8354	207	2161	1
16	10:07:42.926	Grupo 1 1-12	index_get	2308	✓	8354	207	2308	1
17	10:07:43.047	Grupo 1 1-18	index_get	2190	✓	8354	207	2190	0
18	10:07:42.946	Grupo 1 1-13	index_get	2294	✓	8354	207	2294	1
19	10:07:43.007	Grupo 1 1-16	index_get	2249	✓	8354	207	2249	0
20	10:07:43.087	Grupo 1 1-20	index_get	2201	✓	8354	207	2201	0
21	10:07:43.107	Grupo 1 1-21	index_get	2197	✓	8354	207	2196	0
22	10:07:43.127	Grupo 1 1-22	index_get	2195	✓	8354	207	2195	0
23	10:07:43.206	Grupo 1 1-26	index_get	2159	✓	8354	207	2159	0
24	10:07:43.267	Grupo 1 1-29	index_get	2104	✓	8354	207	2104	1
25	10:07:43.166	Grupo 1 1-24	index_get	2217	✓	8354	207	2217	1
26	10:07:43.146	Grupo 1 1-23	index_get	2246	✓	8354	207	2246	1
27	10:07:43.247	Grupo 1 1-28	index_get	2150	✓	8354	207	2150	0

Figura 43: Resultado de la prueba realizada a la clase UnidadSalud_SW, función index_get

Reporte resumen

Nombre: Reporte resumen

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo: Log/Mostrar sólo: Escribir en Log Sólo Errores Éxitos

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Bytes
index_get	200	7352	1329	13887	4136.43	0,00%	10,9/sec	88,77	2,20	8354,0
Total	200	7352	1329	13887	4136.43	0,00%	10,9/sec	88,77	2,20	8354,0

Figura 44: Reporte resumen de la clase UnidadSalud_SW, función index_get

Como se puede evidenciar en la figura 44 el porcentaje de error es de 0% con una carga de 200 usuarios.

- **Funcion find_get**

Petición HTTP

Nombre: find_get

Comentarios: Permite obtener la informacion de un centro de salud en concreto por medio de la ID

Basic **Advanced**

Servidor Web

Nombre de Servidor o IP: Puerto: Timeout (milisegundos)

Conexión: Respuesta:

Petición HTTP

Implementación HTTP: Protocolo: Método: GET Codificación del contenido:

Ruta: http://localhost/sius/index.php/UnidadSalud_SW/165

Redirigir Automáticamente Seguir Redirecciones Utilizar KeepAlive Usar 'multipart/form-data' para HTTP POST Cabeceras compatibles con navegadores

Figura 45: Prueba de la clase UnidadSalud_SW, función find_get

Ver Resultados en Árbol

Nombre: Ver Resultados en Árbol

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo Navegar... Log/Mostrar sólo: Escribir en Log Sólo Errores Éxitos

Muestra #	Tiempo de co...	Nombre del hilo	Etiqueta	Tiempo de Mu...	Estado	Bytes	Sent Bytes	Latency	Connect Time...
1	14:14:49.257	Grupo Find 2-5	find_get	1421	✓	1018	211	1421	0
2	14:14:49.200	Grupo Find 2-3	find_get	1488	✓	1018	211	1488	0
3	14:14:49.220	Grupo Find 2-4	find_get	1480	✓	1018	211	1480	0
4	14:14:49.180	Grupo Find 2-2	find_get	1538	✓	1018	211	1538	0
5	14:14:49.155	Grupo Find 2-1	find_get	1646	✓	1018	211	1646	1
6	14:14:49.397	Grupo Find 2-11	find_get	1522	✓	1018	211	1522	0
7	14:14:49.260	Grupo Find 2-6	find_get	1667	✓	1018	211	1667	0
8	14:14:49.317	Grupo Find 2-7	find_get	1702	✓	1018	211	1702	0
9	14:14:49.337	Grupo Find 2-8	find_get	1711	✓	1018	211	1711	0
10	14:14:49.621	Grupo Find 2-24	find_get	1776	✓	1018	211	1776	0
11	14:14:49.380	Grupo Find 2-12	find_get	2032	✓	1018	211	2032	0
12	14:14:49.701	Grupo Find 2-28	find_get	1712	✓	1018	211	1712	1
13	14:14:49.501	Grupo Find 2-18	find_get	1916	✓	1018	211	1916	0
14	14:14:49.460	Grupo Find 2-16	find_get	1971	✓	1018	211	1970	0
15	14:14:49.320	Grupo Find 2-9	find_get	2112	✓	1018	211	2112	0
16	14:14:49.377	Grupo Find 2-10	find_get	2064	✓	1018	211	2064	0
17	14:14:49.642	Grupo Find 2-25	find_get	1812	✓	1018	211	1812	0
18	14:14:49.602	Grupo Find 2-23	find_get	1868	✓	1018	211	1868	0
19	14:14:49.458	Grupo Find 2-14	find_get	2040	✓	1018	211	2040	0
20	14:14:49.556	Grupo Find 2-21	find_get	2137	✓	1018	211	2137	0
21	14:14:49.400	Grupo Find 2-13	find_get	2566	✓	1018	211	2566	0
22	14:14:49.478	Grupo Find 2-15	find_get	2499	✓	1018	211	2499	0
23	14:14:49.480	Grupo Find 2-17	find_get	2507	✓	1018	211	2507	0
24	14:14:49.656	Grupo Find 2-26	find_get	2563	✓	1018	211	2563	0
25	14:14:49.541	Grupo Find 2-20	find_get	2684	✓	1018	211	2684	0
26	14:14:49.583	Grupo Find 2-22	find_get	2650	✓	1018	211	2650	0
27	14:14:49.521	Grupo Find 2-19	find_get	2721	✓	1018	211	2721	0

Figura 46: Resultado de la prueba realizada a la clase UnidadSalud_SW, función find_get

Reporte resumen

Nombre: Reporte resumen

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo Navegar... Log/Mostrar sólo: Escribir en Log Sólo Errores Éxitos

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Bytes
find_get	200	10018	1250	36992	11006,13	0,00%	4,8/sec	4,76	0,99	1018,0
Total	200	10018	1250	36992	11006,13	0,00%	4,8/sec	4,76	0,99	1018,0

Figura 47: Reporte resumen de la clase UnidadSalud_SW, función find_get

Como se puede evidenciar en la figura 47 el porcentaje de error es de 0% con una carga de 200 usuarios.

▪ Funcion lucro_get

Petición HTTP

Nombre: index_get

Comentarios: Obtener la informacion de las unidades de salud a traves de una Id, 1 si es publico, 2 todos y 3 privados

Basic Advanced

Servidor Web

Nombre de Servidor o IP: Puerto: Timeout (milisegundos)

Conexión: Respuesta:

Petición HTTP

Implementación HTTP: Protocolo: Método: GET Codificación del contenido:

Ruta: http://localhost/sius/index.php/UnidadSalud_SW/lucro/1

Figura 48: Prueba de la clase UnidadSalud_SW, función lucro_get

Ver Resultados en Árbol

Nombre: Resultados de lucro_get

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo: Users\Kevin\Desktop\pruebas\Ver Resultados en Árbol.jmx Log/Mostrar sólo: Escribir en Log Sólo Errores Éxitos

Muestra #	Tiempo de com...	Nombre del hilo	Etiqueta	Tiempo de Mue...	Estado	Bytes	Sent Bytes	Latency	Connect Time(...)
1	14:14:30.298	Grupo Get 1-1	lucro_get	1481	✓	8354	215	1481	3
2	14:14:30.381	Grupo Get 1-3	lucro_get	1550	✓	8354	215	1550	0
3	14:14:30.355	Grupo Get 1-2	lucro_get	1601	✓	8354	215	1601	1
4	14:14:30.431	Grupo Get 1-5	lucro_get	1571	✓	8354	215	1571	0
5	14:14:30.481	Grupo Get 1-7	lucro_get	1545	✓	8354	215	1545	0
6	14:14:30.514	Grupo Get 1-8	lucro_get	1512	✓	8354	215	1512	0
7	14:14:30.456	Grupo Get 1-6	lucro_get	1587	✓	8354	215	1587	0
8	14:14:30.406	Grupo Get 1-4	lucro_get	1638	✓	8354	215	1638	0
9	14:14:30.532	Grupo Get 1-9	lucro_get	1549	✓	8354	215	1549	0
10	14:14:30.556	Grupo Get 1-10	lucro_get	1555	✓	8354	215	1555	0
11	14:14:30.659	Grupo Get 1-12	lucro_get	1501	✓	8354	215	1501	0
12	14:14:30.632	Grupo Get 1-13	lucro_get	1592	✓	8354	215	1592	0
13	14:14:30.634	Grupo Get 1-11	lucro_get	1853	✓	8354	215	1853	0
14	14:14:30.707	Grupo Get 1-16	lucro_get	1837	✓	8354	215	1837	0
15	14:14:30.709	Grupo Get 1-14	lucro_get	1981	✓	8354	215	1981	0
16	14:14:30.834	Grupo Get 1-19	lucro_get	1917	✓	8354	215	1917	0
17	14:14:30.857	Grupo Get 1-22	lucro_get	1920	✓	8354	215	1920	0
18	14:14:30.807	Grupo Get 1-20	lucro_get	2028	✓	8354	215	2028	0
19	14:14:30.784	Grupo Get 1-17	lucro_get	2209	✓	8354	215	2209	0
20	14:14:30.882	Grupo Get 1-23	lucro_get	2118	✓	8354	215	2118	0
21	14:14:30.832	Grupo Get 1-21	lucro_get	2189	✓	8354	215	2189	0
22	14:14:30.734	Grupo Get 1-15	lucro_get	2358	✓	8354	215	2358	0
23	14:14:30.809	Grupo Get 1-18	lucro_get	2312	✓	8354	215	2312	0
24	14:14:30.907	Grupo Get 1-24	lucro_get	2293	✓	8354	215	2293	0
25	14:14:30.932	Grupo Get 1-25	lucro_get	2474	✓	8354	215	2474	0
26	14:14:30.957	Grupo Get 1-26	lucro_get	2532	✓	8354	215	2532	0
27	14:14:30.983	Grupo Get 1-27	lucro_get	2516	✓	8354	215	2516	0

Figura 49: Resultado de la prueba realizada a la clase UnidadSalud_SW, función lucro_get

Reporte resumen

Nombre: Reporte resumen

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo: Log/Mostrar sólo: Escribir en Log Sólo Errores Éxitos

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Bytes
lucro_get	200	8478	1481	22137	5596,95	0,00%	7,4/sec	60,78	1,56	8354,0
Total	200	8478	1481	22137	5596,95	0,00%	7,4/sec	60,78	1,56	8354,0

Figura 50: Reporte resumen de la clase UnidadSalud_SW, función lucro_get

▪ **Funcion lucro_get**

Petición HTTP

Nombre: disponibilidad_get

Comentarios: Obtener la informacion de todas las unidades de salud, que tienen el estado 0 (desactivado)

Servidor Web

Nombre de Servidor o IP: Puerto: Timeout (milisegundos)

Conexión: Respuesta:

Petición HTTP

Implementación HTTP: Protocolo: Método: GET Codificación del contenido:

Ruta: http://localhost/sius/index.php/UnidadSalud_SW/disponibilidad

Figura 51: Prueba de la clase UnidadSalud_SW, función disponibilidad_get

Ver Resultados en Árbol

Nombre: Resultados de disponibilidad_get

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo: Users\Kevin\Desktop\pruebas\Ver Resultados en Árbol.jmx Log/Mostrar sólo: Escribir en Log Sólo Errores Éxitos

Muestra #	Tiempo de com...	Nombre del hilo	Etiqueta	Tiempo de Mue...	Estado	Bytes	Sent Bytes	Latency	Connect Time(...)
1	15:40:20.129	Grupo Get 1-3	disponibilidad_...	1197	✓	5366	222	1197	0
2	15:40:20.154	Grupo Get 1-4	disponibilidad_...	1237	✓	5366	222	1237	0
3	15:40:20.107	Grupo Get 1-2	disponibilidad_...	1660	✓	5366	222	1660	0
4	15:40:20.077	Grupo Get 1-1	disponibilidad_...	1829	✓	5366	222	1829	1
5	15:40:20.206	Grupo Get 1-6	disponibilidad_...	1825	✓	5366	222	1825	1
6	15:40:20.188	Grupo Get 1-5	disponibilidad_...	1858	✓	5366	222	1858	0
7	15:40:20.235	Grupo Get 1-7	disponibilidad_...	1820	✓	5366	222	1820	1
8	15:40:20.285	Grupo Get 1-9	disponibilidad_...	1915	✓	5366	222	1915	1
9	15:40:20.360	Grupo Get 1-12	disponibilidad_...	1953	✓	5366	222	1953	1
10	15:40:20.266	Grupo Get 1-8	disponibilidad_...	2146	✓	5366	222	2146	0
11	15:40:20.310	Grupo Get 1-10	disponibilidad_...	2117	✓	5366	222	2117	1
12	15:40:20.385	Grupo Get 1-13	disponibilidad_...	2190	✓	5366	222	2190	1
13	15:40:20.411	Grupo Get 1-14	disponibilidad_...	2212	✓	5366	222	2212	1
14	15:40:20.335	Grupo Get 1-11	disponibilidad_...	2304	✓	5366	222	2304	1
15	15:40:20.436	Grupo Get 1-15	disponibilidad_...	2449	✓	5366	222	2449	1
16	15:40:20.461	Grupo Get 1-16	disponibilidad_...	2430	✓	5366	222	2430	1
17	15:40:20.512	Grupo Get 1-18	disponibilidad_...	2643	✓	5366	222	2643	0
18	15:40:20.486	Grupo Get 1-17	disponibilidad_...	2733	✓	5366	222	2733	1
19	15:40:20.536	Grupo Get 1-19	disponibilidad_...	2795	✓	5366	222	2795	1
20	15:40:20.561	Grupo Get 1-20	disponibilidad_...	2808	✓	5366	222	2808	1
21	15:40:20.587	Grupo Get 1-21	disponibilidad_...	2947	✓	5366	222	2947	1
22	15:40:20.611	Grupo Get 1-22	disponibilidad_...	2942	✓	5366	222	2942	1
23	15:40:20.662	Grupo Get 1-24	disponibilidad_...	2908	✓	5366	222	2908	1
24	15:40:20.712	Grupo Get 1-26	disponibilidad_...	2861	✓	5366	222	2861	0
25	15:40:20.637	Grupo Get 1-23	disponibilidad_...	2941	✓	5366	222	2941	1
26	15:40:20.687	Grupo Get 1-25	disponibilidad_...	2922	✓	5366	222	2922	1
27	15:40:20.736	Grupo Get 1-27	disponibilidad_...	2907	✓	5366	222	2907	1

Figura 52: Resultado de la prueba realizada a la clase UnidadSalud_SW, función disponibilidad_get

Reporte resumen

Nombre: Reporte resumen

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo: Log/Mostrar sólo: Escribir en Log Sólo Errores Éxitos

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Bytes
disponibilidad...	200	7536	1197	26643	5008,57	0,00%	6,5/sec	33,83	1,40	5366,0
Total	200	7536	1197	26643	5008,57	0,00%	6,5/sec	33,83	1,40	5366,0

Figura 53: Reporte resumen de la clase UnidadSalud_SW, función disponibilidad_get

6.6.2.13. Pruebas de Accesibilidad

Para realizar las pruebas de accesibilidad, se utilizó el siguiente URL: <http://examinator.ws> en el cual, el análisis de un módulo de la aplicación web obtuvo el siguiente resultado:

Informe

6.4

URI: <http://sius.000webhostapp.com/>

Título: Login

Elementos: 69

Tamaño: 2.2 KB (2220 bytes)

Fecha/Hora: 15/02/2017 - 20:27 GMT

Los resultados de la validación (X)HTML no están incluidos.



Figura 54: Pruebas de accesibilidad de la aplicación web

G. DISCUSIÓN

7.1. Evaluación del objeto de investigación

El presente trabajo de titulación el cual versa “**Aplicación móvil multiplataforma que brinda información de las unidades de salud públicas y privadas de la ciudad de Loja**” dio como resultado final el desarrollo de la aplicación móvil denominada IUS (Información de las Unidades de Salud) y de la aplicación web SIUS (Sistema de Información de las Unidades de Salud).

El desarrollo de la propuesta se basa en el cumplimiento de cada uno de los objetivos específicos que fueron abarcados en su totalidad, tal y como se describe a continuación:

- **Objetivo específico 1.** Realizar el análisis de requerimientos de usuario para definir los servicios y procesos que la aplicación ofrece.

Evaluación del Objetivo.

Para el cumplimiento de este objetivo se determinaron los requerimientos de usuario por medio de la elaboración del documento de Especificaciones de Requerimientos de Software (ERS), el mismo que aplica características del estándar IEEE 830-1998.

El ERS es fundamental para la obtención de las historias de usuario, que definen las características, procesos y servicios que serán desarrolladas, ya sea en la aplicación web como la móvil en fases posteriores.

- **Objetivo específico 2.** Elaborar el diseño preliminar de la aplicación móvil para obtener los diferentes diagramas.

Evaluación del Objetivo.

Para efectuar este objetivo se emplearon diagramas UML, para modelar las características, funciones, atributos, componentes y relaciones entre clases de la aplicación móvil y web.

El diagrama de casos de uso, representa la forma en como un Cliente (Actor) opera con el sistema. El diagrama de clases, permite visualizar las relaciones entre las clases que involucran el sistema. El diagrama de componentes, representa cómo un sistema es dividido en componentes y muestra las dependencias entre estos.

El diagrama de paquetes, muestra las dependencias entre los paquetes que componen el sistema y finalmente el diagrama de base de datos es un modelo de datos basado en la lógica de predicados y en la teoría de conjuntos.

- **Objetivo específico 3.** Desarrollar la aplicación móvil, utilizando las tecnologías previamente seleccionadas e implementar un servicio web para el consumo de datos.

Evaluación del Objetivo.

Para el cumplimiento a este objetivo, el desarrollo del software se lo dividió en tres componentes:

- Se programó la aplicación web SIUS usando el lenguaje PHP, además el framework Codeignite y la arquitectura Modelo-Vista-Controlador. Permitted llevar una estructura ordenada del código, separando los datos de la lógica del negocio, ahorrando tiempo en el desarrollo y reutilizando código.
- Para la comunicación entre la aplicación móvil y web, se desarrolló un servicio web, acoplado la librería CodeIgniter Rest Server, esto permitió construir el API de forma rápida, segura e intercambiando información en formato JSON, el cual será consumido por la aplicación móvil.

- Para la codificación de la aplicación móvil se seleccionó el framework Ionic, este provee la documentación, soporte, componentes, además integra tecnologías como Angular Js, HTML5, Gulp, Cordova, dando como resultado la creación de la aplicación para múltiples plataformas, además, reutilizando código y aprovechando el hardware que provee el dispositivo.
- **Objetivo específico 4.** Probar y evaluar las funcionalidades de la aplicación móvil y de consumo de información implementadas, para determinar su correcto funcionamiento.

Evaluación del Objetivo.

Para llevar a cabo la correcta aplicación de este objetivo, se procedió a realizar un plan de pruebas.

El mismo que permitió especificar lo que se desea probar y cómo ejecutar dichas pruebas, realizando una planeación de manera sistemática y sencilla, además se consideró el tipo de pruebas deben realizarse con la finalidad de evaluar el desempeño de las funcionalidades de la aplicación móvil y servicio web con el objeto de encontrar errores y solucionarlos.

7.2. Valoración Técnico – Económica – Ambiental

Para el desarrollo del presente trabajo de titulación (TT) “**Aplicación móvil multiplataforma que brinda información de las unidades de salud públicas y privadas de la ciudad de Loja**”, se contó con la colaboración de recursos humanos, materiales, técnicos y tecnológicos.

En el ámbito técnico, la aplicación móvil “IUS” (Información de las Unidades de Salud) al ser una herramienta tecnológica y desarrollada para la mayor cantidad de dispositivos móviles, tiene como finalidad brindar información completa, detallada y precisa de las unidades de salud, especialidades y los servicios que este ofrece, otorgando confianza, seguridad, ahorro tiempo y recursos a los usuarios que la utilicen.

En el ámbito económico la aplicación permite conocer de manera rápida y eficiente datos esenciales tales como, números de teléfono, dirección, nivel de atención, sector, áreas de especialidad, servicios y la ubicación exacta, permitiendo a los usuarios llegar al centro de salud más cercano, ahorrando tiempo en el traslado, haciendo uso eficaz de los servicios y recursos económicos.

Considerando el ámbito ambiental, el desarrollo de este TT no acarrea ningún impacto al medio ambiente, puesto que está orientado a hacer de carácter formativo e intelectual.

A continuación se presenta de manera más detallada los recursos utilizados con sus respectivos costos.

Recursos Humanos.

El recurso humano para el desarrollo del proyecto está conformado por el profesional encargado de la dirección de la tesis, la persona encargada de la elaboración de la tesis quien desempeña las tareas de análisis y desarrollo del sistema.

TABLA LIII: RECURSOS HUMANOS.

RECURSOS HUMANOS			
DESCRIPCIÓN	Numero Horas	Valor Unitario (\$)	Valor Total
AUTOR	700	7,00	4900
TUTOR	400	10,00	4000
SUBTOTAL			8900,00

Nota: Los costos concernientes a las horas de tutorías corren por cuenta de la Universidad Nacional de Loja.

Recursos Materiales

A continuación se detallan los materiales utilizados para el desarrollo del proyecto.

TABLA LIV: RECURSOS MATERIALES.

RECURSOS MATERIALES			
DESCRIPCIÓN	Cantidad	Valor Unitario (\$)	Valor Total
HOJAS	500	0,05	25
CARTUCHOS DE TINTA	2	25,00	50
ANILLADO	1	2,00	2
MEMORIA FLASH	1	16,00	16
TRANSPORTE	250	0,30	75
SUBTOTAL			168,00

Recursos técnicos

A continuación se detallan los recursos técnicos/tecnológicos utilizados para el desarrollo del proyecto.

TABLA LV: RECURSOS TÉCNICOS.

RECURSOS TÉCNICOS			
DESCRIPCIÓN	Cantidad	Valor Unitario (\$)	Valor Total
HARDWARE			
COMPUTADORA	1	1000,00	1000
IMPRESORA	1	320,00	320
SOFTWARE			
ATOM	1	0,00	0
NETBEANS	1	0,00	0
ANGULAR	1	0,00	0
IONIC	1	0,00	0
CODEIGNITER	1	0,00	0
INTERNET	1	25,00	25,00
SUBTOTAL			1345,00

Resumen del Presupuesto

TABLA LVI: COSTO TOTAL

COSTO TOTAL	
DESCRIPCIÓN	Valor Total
RECURSOS HUMANOS	8900
RECURSOS MATERIALES	168
RECURSOS TÉCNICOS	1345
SUBTOTAL	9068,00

H. Conclusiones

Luego de haber realizado el presente trabajo de titulación se puede concluir lo siguiente:

- El desarrollo de la aplicación móvil que brinda información de las unidades de salud pública y privada denominada IUS, facilita a las habitantes de la ciudad de Loja, conocer los servicios y especialidades que estos ofrecen.
- La construcción de la aplicación web SIUS, facilita al administrador la gestión y administración de la información de las unidades de salud, áreas de especialidad y servicios.
- La metodología de desarrollo ágil XP (Programación Extrema), fue de vital importancia, porque ayudó a seguir un proceso ordenado en la creación del software, partiendo desde la especificación de los requisitos con la creación de las historias de usuario que contribuyeron a definir correctamente la funcionalidad y propósito de las aplicaciones.
- Los diagramas UML (Lenguaje Unificado de Modelado) y las tarjetas CRC en la fase de diseño, ayudó a describir, comprender, documentar y modelar el comportamiento de las funciones, procesos y componentes que intervinieron en el software que se desarrolló.
- El framework CodeIgniter facilita el desarrollo de la aplicación web, porque este cumple con las características de la arquitectura MVC, además ofrece una extensa documentación y librerías complementarias que se pueden añadir, permitiendo al desarrollador poder reducir tiempo, reutilizar código y crear un software que cumpla con los requerimientos expuestos en las historias de usuario.
- La creación del servicio web REST, permite comunicar la aplicación web y móvil, brindando seguridad, fiabilidad, eficiencia e interoperatividad a los procesos que ofrece el Api, la cual será consumida por la aplicación móvil para el intercambio de información.
- Ionic framework combina lo mejor de la programación web y móvil permitiendo al programador desarrollar aplicación interactivas, rápidas, disponibles para una gran variedad de plataformas y se consigue aprovechar las características propias de cada uno de los dispositivos, haciendo de este framework muy potente y completo para la creación de aplicaciones móviles.

- La herramienta JMeter para la realización de las pruebas de carga, facilito el análisis y evaluación del desempeño de las funciones que ofrece el servicio web, porque la herramienta proporciona mecanismos que permite simular un entorno de estrés, además se obtiene un informe estadístico de resultados de forma gráfica o de árbol.

I. RECOMENDACIONES

Luego de haber realizado el presente trabajo de titulación se puede recomendar lo siguiente:

- Antes de iniciar un proyecto de desarrollo de software, el programador debe definir correctamente la arquitectura, lenguajes, frameworks y establecer las herramientas, entornos de desarrollo y tecnologías que serán usadas para la creación del software.
- Para un futuro trabajo de titulación, se puede modificar la aplicación móvil IUS o a partir de esta, crear nuevas aplicaciones que abarquen la provincia de Loja o la Zonal N°7, además se podría incrementar los módulos como la reserva de servicios, horarios de atención, consulta en línea con el especialista, consultar la disponibilidad de medicamentos y camas entre otros más servicios, lo que haría de este software una herramienta de información de suma importancia para el bienestar de los usuarios de la provincia de Loja.
- Para próximos trabajos de titulación referente al tema, se recomienda realizar un convenio interinstitucional entre la carrera de Ingeniería en Sistemas y la Coordinación Zonal N°7 de Salud, el mismo que busque un beneficio mancomunado entre ambas instituciones, proporcionando a los estudiantes el acceso a la información completa y actualizada de las unidades de salud de la provincia de Loja.
- Se recomienda el uso del framework híbrido Ionic, porque permite crear aplicaciones móviles multiplataforma de manera rápida, segura y sencilla, posee una documentación completa de todos sus componentes y se puede integrar otras tecnologías por medio de su interfaz de línea de comandos (CLI).

J. BIBLIOGRAFÍA

- [1] C. Bouras, A. Papazois, y N. Stasinou, «A Framework for Cross-Platform Mobile Web Applications Using HTML5», presentado en 2014 International Conference on Future Internet of Things and Cloud, 2014, pp. 420-424.
- [2] J. Cordeiro y K.-H. Krempels, Eds, «Web Information Systems and Technologies», vol. 140. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013.
- [3] L. Delía, N. Galdamez, P. Thomas, y P. Pesado, «Un Análisis Experimental de Tipo de Aplicaciones para Dispositivos Móviles», presentado en XVIII Congreso Argentino de Ciencias de la Computación, 2013, pp. 766-776.
- [4] B. Barrio, «Marketing móvil basado en aplicaciones», Universidad de La Rioja, 2013.
- [5] B. Raluca, «Mobile Web Apps vs . Mobile Native Apps : How to Make the Right Choice», p. 13, 2013.
- [6] IBM Corporation, «El desarrollo de aplicaciones móviles nativas, Web o híbridas». IBM Corporation, Abril del-2012.
- [7] R. Pressman, V. Campos, y J. Enríquez, «Ingeniería del software: un enfoque práctico». México: McGraw-Hill, 2010.
- [8] V. Szalvay, «An introduction to agile software development», *Danube Technol.*, pp. 1–9, 2004.
- [9] K. Beck, M. Beedle, A. van Bennekum, A. Cockburn, W. Cunningham, y M. Fowler, «12 Principles Behind the Agile Manifesto», *Agile Alliance*, 04-nov-2015. [En línea]. Disponible en: <https://www.agilealliance.org/agile101/12-principles-behind-the-agile-manifesto/>. [Accedido: 25-oct-2016].
- [10] P. Letelier y C. Penadés, «Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP)», *Cyta*, vol. 5, n.º 26, 15-abr-2006.
- [11] A. Navarro, J. Fernández, y J. Morales, «Revisión de metodologías ágiles para el desarrollo de software», *Prospectiva*, vol. 11, n.º 2, pp. 30–39, sep. 2013.
- [12] I. Sommerville, «Ingeniería del Software», Séptima. Madrid: Pearson Educación, 2005.
- [13] O. Pérez, «Cuatro enfoques metodológicos para el desarrollo de Software RUP–MSF–XP–SCRUM», *Rev. Inven.*, n.º 10, 2011.
- [14] D. Wells, «Extreme Programming Rules», *Extreme Programming: A gentle introduction*, 08-oct-2013. [En línea]. Disponible en: <http://www.extremeprogramming.org/rules.html>. [Accedido: 01-ago-2016].
- [15] M. Pérez, «Guía Comparativa de Metodologías Ágiles», Univ. Valladolid, vol. 1, pp. 3–117, 2012.
- [16] J. Palacio, «Gestión de proyectos Scrum Manager, Scrum Manager», vol. 2.5. 2014.

- [17] K. Schwaber y J. Sutherland, «La Guía de Scrum». Scrum Guides, jul-2013.
- [18] L. Valdez, A. Ruíz, E. López, G. Guerrero, y J. Brindis, «Aplicación de la metodología semi-ágil ICONIX para el desarrollo de software: implementación y publicación de un sitio WEB para una empresa SPIN-OFF en el Sur de Sonora, México.», presentado en Twelfth LACCEI Latin American and Caribbean Conference for Engineering and Technology (LACCEI'2014), Guayaquil, Ecuador, 2014, p. 10.
- [19] D. Rosenberg, M. Stephens, y M. Collins-Cope, «Agile development with ICONIX process: people, process, and pragmatism». Berkeley, CA: Apress, 2005.
- [20] Y. Balaguera, «Metodologías ágiles en el desarrollo de aplicaciones para dispositivos móviles. Estado actual», *Rev. Tecnol.*, vol. 12, n.o 2, 2015.
- [21] L. Craig, «UML y Patrones, Una introducción al análisis y diseño orientado a objetos y al proceso unificado», 2da Edición. Pearson Educación, 2004.
- [22] G. Villalobos, G. Sánchez, y D. Gutiérrez, «Diseño de framework web para el desarrollo dinámico de aplicaciones», *Sci. Tech.*, vol. 1, n.o 44, pp. 178–183, 2010.
- [23] A. Verdejo, «Conceptualización y diseño de una App. híbrida para Amigos de Silva ONGD», Master, Universidad oberta de catalunya, Catalunya, 2015.
- [24] J. L. Cobacho, «Top Games», Máster, UNIVERSIDAD DE POLITÉCNICA DE MADRID, Madrid, 2015.
- [25] C. Rodríguez y H. Enríquez, «Características del desarrollo en Frameworks multiplataforma para móviles», *Ingenium*, vol. 15, n.º 30, pp. 101–117, 2015.
- [26] J. Herrera, «Diseño e implementación de una aplicación móvil basada en la tecnología NFC para acceso a información de las piezas de arte de un museo», PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ, Lima, 2014.
- [27] D. Weider y H. Yuan, «An approach to explore mobile software engineering advances in cloud computing environment», en *Computer Software and Applications Conference Workshops (COMPSACW)*, 2011 IEEE 35th Annual, 2011, pp. 292–297.
- [28] R. Iskandar, «Estudio comparativo de alternativas y frameworks de programación, para el desarrollo de aplicaciones móviles en entorno Android», Universidad Politécnica de Cataluña, Barcelona, 2013.
- [29] M. Guncay y J. Mosquera, «Evaluación de frameworks para el desarrollo de aplicaciones móviles multiplataforma.», B.S. thesis, 2016.
- [30] C. Chisag y L. Margoth, «La realidad aumentada y su aplicación en el desarrollo del aprendizaje para los estudiantes de tercero y sexto semestre de la carrera de docencia en informática de la facultad de ciencias humanas y de la educación de la universidad técnica de ambato», 2013.
- [31] A. López y J. Villanueva, «Manual de CodeIgniter en Español», p. 302.
- [32] DesarrolloWeb.com, «Modelo - Vista - Controlador en CodeIgniter». [En línea]. Disponible en: <http://www.desarrolloweb.com/articulos/modelo-vista-controlador-codeigniter.html>. [Accedido: 02-ago-2016].

- [33] O. Burillo, «Domótica y servicios web», Monografía, Universidad Veracruzana, Xalapa Enríquez, 2008.
- [34] E. Enriquez, «Sistema de Gestion de Historias Clinicas para el Departamento de Bienestar Universitario de la UTN», Thesis, Universidad Técnica del Norte, Ibarra, 2011.
- [35] S. Otón, «Propuesta de una arquitectura software basada en servicios para la implementación de repositorios de objetos de aprendizaje distribuidos», doctoralThesis, Universidad de Alcalá, Madrid, 2006.
- [36] Juan Carlos Sánchez Tapia, «RESTful Web Services», *Web Services*, Abril-2013. [En línea]. Disponible en: <http://jc.pe/2013/04/18/restful-web-services/>. [Accedido: 15-feb-2016].
- [37] L. Richardson y S. Ruby, «RESTful web services». Farnham: O'Reilly, 2007.
- [38] R. Gogonea, «Cordova Generator: Generación automática de plugins Apache Cordova para Aplicaciones Híbridas», Thesis, Universidad Politécnica de Cataluña, Barcelona, 2016.
- [39] «AngularJS - Marco superheroic JavaScript MVW». [En línea]. Disponible en: <https://angularjs.org/>. [Accedido: 17-ene-2017].
- [40] «Apache Cordova». [En línea]. Disponible en: <https://cordova.apache.org/docs/es/latest/guide/overview/index.html>. [Accedido: 17-ene-2017].
- [41] «Bower». [En línea]. Disponible en: <https://bower.io/>. [Accedido: 17-ene-2017].
- [42] O. Heurtel, «PHP 5.6: desarrollar un sitio web dinámico e interactivo». Ediciones ENI, 2015.
- [43] S. S. ether Bakken *et al.*, «Manual de PHP», *Este Man. Escopyright Copyr.*, 1997.
- [44] F. Wempen, «HTML5 step by step». Pearson Education, 2011.
- [45] D. De Luca, «HTML5: Entiende el cambio, aproveche su potencial», vol. 216. USERSHOP, 2011.
- [46] F. Wempen, «Learn HTML5». Sebastopol, Calif: O'Reilly Media, 2012.
- [47] R. Bourdon, «WampServer». [En línea]. Disponible en: <http://www.wampserver.com/es/>. [Accedido: 13-ene-2017].
- [48] «NetBeans IDE 8.0 Release Information», *NetBeans*. [En línea]. Disponible en: <https://netbeans.org/community/releases/82/>. [Accedido: 13-ene-2017].
- [49] J. Sagredo, A. Espinosa, M. Reyes, y M. García, «Automatización de la codificación del patrón modelo vista controlador (MVC) en proyectos orientados a la Web», 2016.
- [50] F. Díaz, C. Banchoff Tzancoff, A. Rodríguez, y V. Soria, «Usando Jmeter para pruebas de rendimiento», en XIV Congreso Argentino de Ciencias de la Computación, 2008.
- [51] «Apache JMeter - Apache JMeter™». [En línea]. Disponible en:

<http://jmeter.apache.org/>. [Accedido: 07-dic-2016].

[52] «IEEE Standard for Software Test Documentation», *IEEE Std 829-1983*, pp. 1-48, 1983.

[53] G. Myers, T. Badgett, T. Thomas, y C. Sandler, «The art of software testing», 2nd ed. Hoboken, N.J: John Wiley & Sons, 2004.

K. Anexos

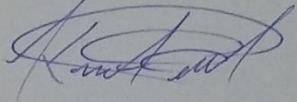
ANEXO I: ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE

Especificación de requisitos de software

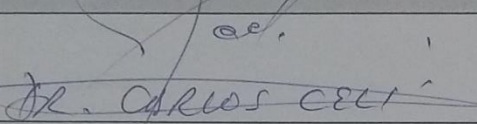
Proyecto: Aplicación móvil multiplataforma que brinda información de las unidades de salud públicas y privadas de la ciudad de Loja

Revisión 2.0

Ficha del documento

Revisión	Autor	Firma
2.0	Kevin Atencia	

Documento validado por las partes:

Por el cliente	Por la empresa suministradora
	
Nombre: Dr. Carlos E. Celi A. MEDICO CIRUJANO Mercadillo 15-09 y Sucre Mat. CML.Nº 192 T.576025	Nombre:

1. Introducción

Este documento es la especificación de requisitos de software (ERS) de la “Aplicación móvil multiplataforma que brinda información de las unidades de salud públicas y privadas de la ciudad de Loja”, todo su contenido ha sido elaborado en colaboración con los usuarios, mediante la aplicación de la técnica como la encuesta, para la obtención de información acerca de las necesidades de los usuarios y responsable del proyecto.

Esta especificación ha sido elaborada tomando en cuenta las directrices establecidas por el estándar “IEEE (Recommended Practice Software Requirements Specification) ANSI/IEEE 98”

2. Propósito

El presente documento tiene como propósito definir de manera clara, precisa y ordenada, todas las especificaciones funcionales, no funcionales, restricciones del sistema que se desea construir. El cual está integrada por la aplicación móvil, la aplicación web y el servicio web junto a otros componentes y herramientas que interactúan entre sí.

El documento va dirigido a los usuarios del sistema, en este caso es la población en general de la ciudad de Loja, esta especificación está sujeta a revisiones, que serán recogidas por medio de sucesivas versiones hasta su aprobación total, por ende, el contenido de la misma será adaptado a las necesidades de los stakeholders.

3. Alcance

El desarrollo total del sistema por el grado de complejidad, plantea que sea dividido en varios subsistemas dependiendo de las restricciones, funcionalidades, requisitos, etc. La aplicación móvil es la parte más trascendental e imprescindible de todo el sistema, funcionará en varias plataformas o sistemas operativos móviles, siempre y cuando dichos dispositivos cumplan con las condiciones mínimas para su correcto funcionamiento.

Por otra parte la aplicación web, es la responsable de administrar, gestionar y consultar la más completa información de las Unidades de Salud de la ciudad de Loja, esta información solo será accesible para un usuario administrador.

El objetivo primordial es presentar información confiable, útil y fehaciente, de las áreas de especialidad y los servicios que ofrecen las unidades de salud.

4. Personal involucrado

Nombre	Kevin Atiencia
Rol	Analista, Diseñador y Desarrollador
Categoría profesional	Ingeniero de Sistema.
Responsabilidades	Definir los requerimientos y desarrollarlos
Información de contacto	0969199863

5. Definiciones, acrónimos y abreviaturas

- APP: Aplicación.
- Administrador: Acción de agregar, modificar, eliminar la información que consume la app móvil.
- Usuario: Persona que puede ingresar a la zona privada del SISZOO, a través
- de un proceso de autenticación utilizando nombre de usuario y contraseña.
- Rol: Es un conjunto de permisos que puede asignarse a un usuario.
- RF: Requisitos Funcionales.
- RNF: Requisitos No Funcionales
- UNL: Universidad Nacional de Loja.
- US: Unidades de salud: establecimiento o institución en el cual se imparten los servicios y la atención de salud más básica y principal.
- ERS: Especificación de requerimientos de software.
- CRUD: Crear, Leer, Actualizar y Eliminar.

6. Descripción general

6.1 Perspectiva del producto

La aplicación será un producto diseñado para ser utilizada en varias plataformas móviles, el mismo que obtiene la información de las unidades de salud mediante el consumo de datos por medio de un servicio web, que proporciona el intercambio de información entre la app móvil y la base de datos central, la misma que será administrada vía web, con un sistema que se encarga de gestionar, administrar y actualizar la información que utilizara la app móvil.

Por ende se maneja el siguiente esquema, donde se diagraman las funcionalidades. Generales de la aplicación:

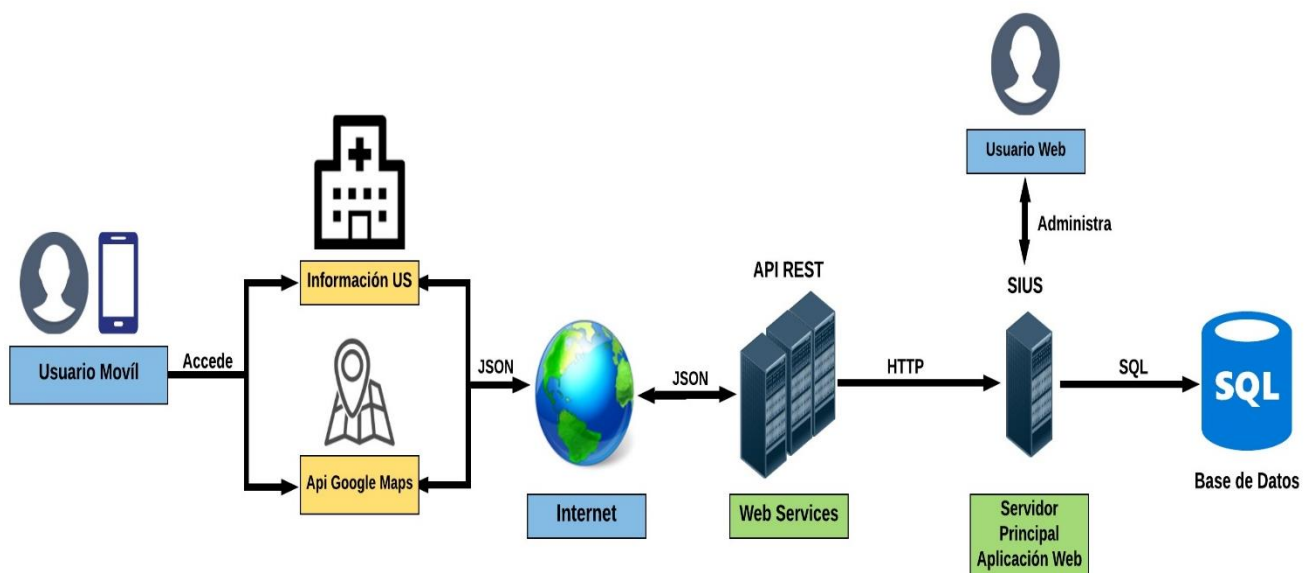


Figura 1: Diagrama de Funcionalidad General

La aplicación móvil envía una petición (Http) a través de internet al servicio web (Api Rest), este recibe la solicitud, la procesa y envía los resultados en un formato estándar (JSON), manteniendo al margen la manipulación de la información contenida en la base de datos central, la cual es la responsable de administrar y gestionar los datos a través de la aplicación web (SIUS).

6.2 Funcionalidad del producto

Las funcionalidades principales del sistema en conjunto, se las identifica de acuerdo a la complejidad del mismo y a las necesidades de los usuarios, por lo cual se ha dividido este proceso en tres componentes software, las cuales son:

Aplicación móvil:

- Buscar mediante Georreferencia las unidades operativas de salud.
- Visualizar información (como número telefónico, tipo de unidad, dirección, etc) de las unidades de salud de la ciudad de Loja.
- Mostrar la información de las especialidades o áreas de atención que ofrecen dichas unidades de salud.
- Mostrar la información de los servicios que disponen las unidades de salud.
- Buscar unidades de salud mediante su nombre.
- Buscar las especialidades por medio de su nombre.

Aplicación Web:

- El sistema de gestión web, permitirá realizar los procesos de CRUD (crear, modificar, actualizar y eliminar) información concerniente a:
 - Unidades de salud
 - Especialidades
 - Contribuidores
 - Servicios
- El sistema administrara la información que mostrara la aplicación móvil.
- Se debe crear un servicio web para la integración y comunicación de la aplicación móvil con la web.

6.3 Características de los usuarios

Tipo de usuario	Usuario Móvil
Formación	Ninguna
Habilidades	Ninguna
Actividades	Consultar e indagar información suministrada por la aplicación móvil

Tipo de usuario	Usuario Web (Administrador)
Formación	Segundo Nivel
Habilidades	Control y acceso total a la información almacenados en la base de datos.
Actividades	Gestionar la información la cual va a ser suministrada a la aplicación móvil

6.4 Restricciones

- Aplicación móvil desarrollada bajo un framework Open Source.
- Lenguajes y tecnologías en uso: HTML, JAVASCRIPT, PHP, JSON.
- La aplicación móvil debe ser multiplataforma.
- El sistema deberá tener un diseño, independiente de la plataforma o del lenguaje de programación.

6.5 Suposiciones y dependencias

La aplicación móvil en un principio se plantea desarrollarla para obtener información de las unidades de salud de la ciudad de Loja, lo cual en un futuro se podría integrar todas las ciudades de la Zona 7 o inclusive se podría escalar a nivel Nacional.

A sí mismo la app móvil debe adaptarse a las necesidades que puedan surgir en un futuro como la integración de varios idiomas, ya sea autóctonos o extranjeros, para mejorar su funcionalidad y abarcar mayor número de usuarios.

La aplicación móvil es multiplataforma, esta se ejecutará en varios sistemas operativos tales como Windows Phone, Symbian, iOS, etc. Recalcando que los dispositivos en los que se instale el software debe cumplir con las condiciones mínimas de funcionamiento.

6.6 Evolución previsible del sistema

El sistema estará en constante evolución, por lo tanto, estará sujeto a cambios significativos en su entorno de trabajo y el desempeño de la aplicación móvil y web debe ser mejor a la actual aplicando nuevas tecnologías que en un futuro podría integrarse como sistemas externos, nuevos scripts, etc, de acuerdo a lo que los usuarios requieran.

7. Requisitos funcionales

7.1 Requisitos comunes de los interfaces

- La interfaz con el usuario consistirá en un conjunto de ventanas con botones, listas y campos de textos.
- La interfaz del sistema deberá visualizarse correctamente utilizando un modelo minimalista con colores apropiados para los usuarios.
- El sistema deberá estar bien construido y en el idioma del usuario.

7.2 Requisitos de la Aplicación Web.

Nº Requisito	Nombre de requisito	Prioridad del requisito
RF-01	Acceder al sistema	Medio
RF-02	Modificar datos de Usuario.	Medio
RF-03	Crear Unidades de Salud	Alta
RF-04	Modificar Unidades de Salud	Medio
RF-05	Listar Unidades de Salud	Baja
RF-06	Subir Archivo de Unidades de Salud	Baja
RF-07	Eliminar Unidades de Salud	Alta
RF-08	Crear Áreas de Especialidad	Alta
RF-09	Modificar Áreas de Especialidad	Medio
RF-10	Listar Áreas de Especialidad	Baja
RF-11	Eliminar Áreas de Especialidad	Alta
RF-12	Subir Archivo de Áreas de Especialidad	Baja
RF-13	Crear un Servicio Web	Alta

RF-14	Crear Servicio	Medio
RF-15	Modificar Servicio	Baja
RF-16	Listar Servicios	Alto
RF-17	Eliminar Servicio	Baja
RF-18	Subir Archivo de Servicios	Medio
RF-19	Dar de alta Servicio	Baja
RF-20	Actualizar estado del Servicio	Alto
RF-21	Eliminar contribuidor	Alto
RF-22	Actualizar Latitud y Longitud de un centro de salud	Baja

7.3 Requisitos de la Aplicación Móvil.

Nº Requisito	Nombre de requisito	Prioridad del requisito
RF-23	Buscar Unidades de Salud	Alto
RF-24	Presentar información de las Unidades de Salud	Alto
RF-25	Presentar información de las Áreas Especialidad	Alto
RF-26	Crear Servicio	Alto
RF-27	Presentar información de los Servicios	Alto
RF-28	Crear Contribuidor	Medio
RF-29	Buscar Áreas Especialidad	Medio

7.4 Requisitos no funcionales

7.4.1 Requisitos de rendimiento

La aplicación móvil estará disponible sin costo en las diferentes plataformas de descarga que cuentan los distintos sistemas operativos móviles, esperando que el número de usuarios que se descarguen la aplicación debe ser moderadamente alta, sumando que el consumo de datos por parte de la aplicación debe tener un tiempo de respuesta alto, de 5 a 10 segundos dependiendo de las características del dispositivo móvil y de la velocidad del internet.

En la aplicación web se debe garantizar que las consultas u otro proceso no afecte a la integridad de la información almacenada en la base de datos, el tiempo de respuesta del sistema web en cada función que el usuario solicite sea de 8 a 10 segundos como máximo de demora.

7.4.2 Seguridad

La aplicación móvil no requiere de ninguna información del usuario o del dispositivo Smartphone.

Para acceder a la aplicación web, este solicitara antes que se identifique, mediante el correo y contraseña, además solicitara ingresar un código Re captcha, o llenando una pregunta de seguridad, con la cual, se planea garantizar la confiabilidad, integridad y seguridad de la información al usuario mostrando un desempeño satisfactorio del sistema. En este sentido la información almacenada o registros realizados podrán ser consultados y actualizados permanente y simultáneamente, sin que se afecte el tiempo de respuesta.

La contraseña del usuario será encriptadas, para incrementar el nivel de seguridad del sistema, además si el usuario web olvida su contraseña podrá acceder al sistema por medio de un correo secundario.

7.4.3 Fiabilidad

El sistema debe tener una interfaz de uso intuitivo y sencillo, debe ajustarse a las características del dispositivo móvil y a las características del equipo de cómputo en cuanto a la aplicación web, dentro de la cual estará incorporado el servicio web para el consumo de datos de la aplicación

8. Apéndices

Apéndice 1: Selección de Muestra

La fórmula para calcular el tamaño de la muestra suele utilizarse la siguiente fórmula: (Técnicas de muestreo y determinación del tamaño de la muestra en investigación cuantitativa)

$$n = \frac{N\sigma^2 Z^2}{(N-1)e^2 + \sigma^2 Z^2}$$

Dónde:

n = Tamaño de la muestra.

Z = Desviación estándar, para un intervalo de confianza de 95.55, es 1.96.

e = Límite aceptable de error muestral que, generalmente cuando no se tiene su valor, suele utilizarse un valor que varía entre el 1% (0,01) y 9% (0,09), valor que queda a criterio del encuestador.

N = Tamaño de la población.

σ = Desviación estándar de la población que, generalmente cuando no se tiene su valor, suele utilizarse un valor constante de 0,5.

La fórmula del tamaño de la muestra se obtiene de la fórmula para calcular la estimación del intervalo de confianza para la media, la cual es:

$$\bar{X} - Z \frac{\sigma}{\sqrt{n}} \sqrt{\frac{N-n}{N-1}} \leq \mu \leq \bar{X} + Z \frac{\sigma}{\sqrt{n}} \sqrt{\frac{N-n}{N-1}}$$

Donde el error es:

$$e = Z \frac{\sigma}{\sqrt{n}} \sqrt{\frac{N-n}{N-1}}$$

De esta fórmula del error de la estimación del intervalo de confianza para la media se despeja la n, para lo cual se sigue el siguiente proceso:

Elevando al cuadrado a ambos miembros de la fórmula se obtiene:

$$(e)^2 = \left(Z \frac{\sigma}{\sqrt{n}} \sqrt{\frac{N-n}{N-1}} \right)^2$$

$$e^2 = Z^2 \frac{\sigma^2 N - n}{n N - 1}$$

Multiplicando Fracciones:

$$e^2 = \frac{Z^2 \sigma^2 (N - n)}{n(N - 1)}$$

Eliminando denominadores:

$$e^2 n (N - 1) = Z^2 \sigma^2 (N - n)$$

Eliminando paréntesis

$$e^2 n N - e^2 n = Z^2 \sigma^2 N - Z^2 \sigma^2 n$$

Transponiendo n a la izquierda:

$$e^2 n N - e^2 n + Z^2 \sigma^2 n = Z^2 \sigma^2 N$$

Factor común y despejando n:

$$n = \frac{Z^2 \sigma^2 N}{e^2 N - e^2 + Z^2 \sigma^2}$$

Ordenando se obtiene la fórmula para calcular el tamaño de la muestra:

$$n = \frac{Z^2 \sigma^2 N}{e^2 (N - 1) + Z^2 \sigma^2}$$

Aplicación de la Formula para obtener el tamaño de la muestra.

El porcentaje de personas que tienen Smartphone activado es del 3,2% en el año 2011, mientras que en el año 2012, corresponde al 7.4%, es decir 4.2 puntos más. Si se mantiene la tendencia de crecimiento, se estima que el 24.2% de personas en la ciudad de Loja dispondrán de un teléfono Smartphone para el año 2016.

Tomando en cuenta el número de habitantes de la ciudad de Loja es de 214 855 personas según el censo realizado en el 2010 por el INEC.

Población Total: 214 855 = 100%

Población que tienen teléfono Smartphone para el año 2016 es de 51994 = 24.2%

Datos

Z = Nivel de confianza de 95%, es 1.96.

e = 0,05

N = 51994

$\sigma = 0,5$

$$n = \frac{Z^2 \sigma^2 N}{e^2 (N - 1) + Z^2 \sigma^2}$$

$$n = \frac{(1.96)^2 (0.5)^2 51994}{(0.5)^2 (51994 - 1) + (1.96)^2 (0.5)^2}$$

$$n = \frac{49935}{129.98 + 0.96} = 99$$

Por lo tanto el tamaño de la muestra es de 99 habitantes de la ciudad de Loja.

Apéndice 2: Encuesta Uno



UNIVERSIDAD NACIONAL DE LOJA
ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES NO
RENOVABLES
INGENIERÍA EN SISTEMAS

Encuesta dirigida a los usuarios

Como estudiante de la "Universidad Nacional de Loja" de la carrera de Ingeniería en Sistemas, solicito comedidamente responder la presente encuesta con el fin de obtener datos necesarios para el desarrollo del Trabajo de Titulación.

La encuesta tiene como objetivo recolectar información

¿Conoce usted la ubicación exacta de las unidades de salud públicas y privadas más cercanas a su localidad?

Si ()

No ()

Si la respuesta es afirmativa ¿De cuantas unidades de salud tiene información exacta de su ubicación?

¿Está informado de los horarios de atención de las unidades de salud?

Si ()

No ()

¿Tiene conocimiento de las áreas especializadas con las que cuenta dichas unidades?

Si ()

No ()

¿Conoce usted los médicos especialistas que labora en las áreas de las unidades salud?

Si ()

No ()

¿Tiene conocimiento de los turnos en la que se encuentran laborando dichos médicos?

Si ()

No ()

En el caso de ser una unidad de salud privada. ¿Sabe usted el valor de la consulta?

Si ()

No ()

¿Qué problemas considera usted que se generan por el desconocimiento de información?

Pérdida de tiempo al intentar localizar una unidad de salud ()

Transitar por rutas erróneas en caso de una emergencia ()

Generar aglomeración, a su vez retraso en la atención. ()

Llegar a la unidad de salud en horas no laborales ()

Estar expuesto a cobros indebidos ()

No encontrar al médico que necesitaba ()

Otros _____

¿Tiene conocimiento de alguna aplicación móvil que brinde información sobre las unidades de salud (Públicas y Privadas)?

Si ()

No ()

Si respondió de forma afirmativa la pregunta anterior ¿Qué aplicaciones conoce?

¿Quedó satisfecho con el desempeño mostrado por las aplicaciones que ha utilizado?

Si ()

No ()

¿Cuál de los siguientes medios, le parece más eficiente para obtener información de las unidades de salud?

A Través de periódicos ()

Por medio de una línea telefónica ()

Mediante una Página web ()

Mediante una aplicación en su celular ()

GRACIAS POR SU COLABORACIÓN

Apéndice 3: Resultados de la Encuesta Uno

El número de personas que fueron encuestas corresponde a 100.

¿Conoce usted la ubicación exacta de las unidades de salud públicas y privadas más cercanas a su localidad?

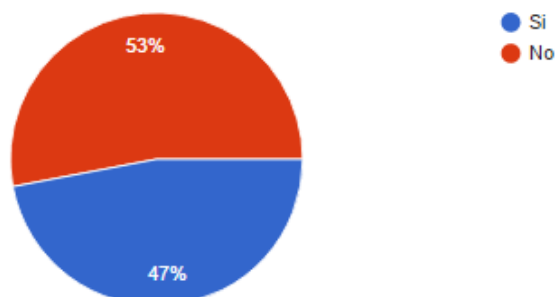


Figura 1: La población desconoce las ubicaciones exactas de ñas unidades de salud.

De una muestra de 100 personas encuestadas, 53 de ellas que corresponde el 53%, manifestaron que no conocen las ubicaciones exactas de las unidades de salud más cercanas a su localidad. Por otro lado, el 47% que pertenece a 47 personas de la muestra total, concluyeron que si conocen.

Si la respuesta es afirmativa ¿De cuantas unidades de salud tiene información exacta de su ubicación?

El número de personas que contestaron afirmativamente a esta pregunta corresponde a 47, en la cual la media obtenida es de 3 establecimientos de salud conoce la información exacta de su ubicación.

¿Está informado de los horarios de atención de las unidades de salud?

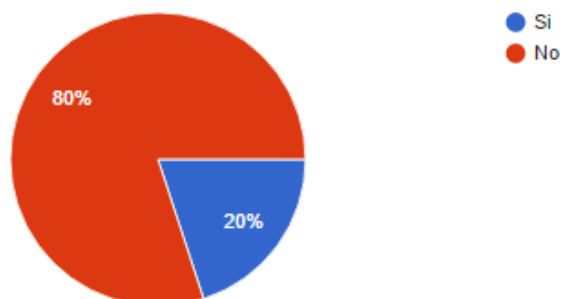


Figura 2: La mayor parte de los encuestados esta desinformado, con respecto a los horarios de atención.

De las respuestas obtenidas, se puede concluir que la opinión mayoritaria que corresponde al 80% de la muestra total, esta desinformado de los horario de atención de las unidades de salud, y el 20% de los encuestados considera que esta informado.

¿Tiene conocimiento de las áreas especializadas con las que cuenta dichas unidades?

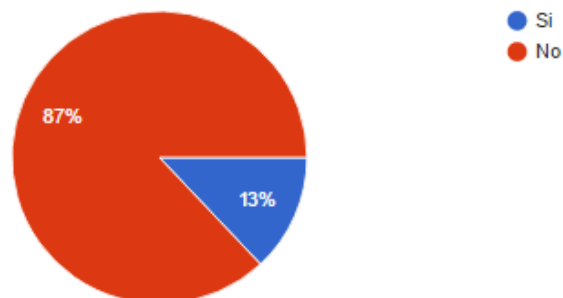


Figura 3: La mayor parte de los encuestados desconoce las áreas especializadas con las que cuentan las unidades de salud

La mayor parte de los encuestados desconoce de las áreas especializadas con las que cuenta las unidades de salud, es decir un 87% concerniente al mismo número de personas, por lo tanto el 13% considera que si tiene información.

¿Conoce usted los médicos especialistas que labora en las áreas de las unidades salud?

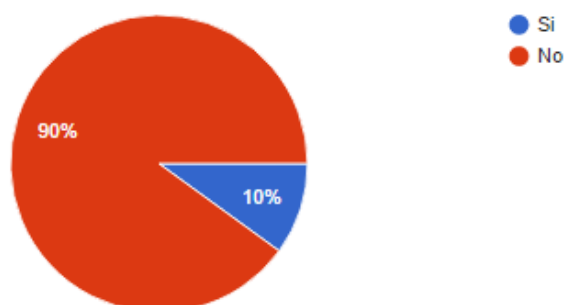


Figura 4: El total de la muestra, desconoce de los médicos especialistas que laboran en las unidades de salud.

El 90% de la muestra total, desconocen los médicos especialistas que laboran en las áreas de salud, mientras que un 10% que corresponde al mismo número de personas considera que si conoce.

¿Tiene conocimiento de los turnos en la que se encuentran laborando dichos médicos?

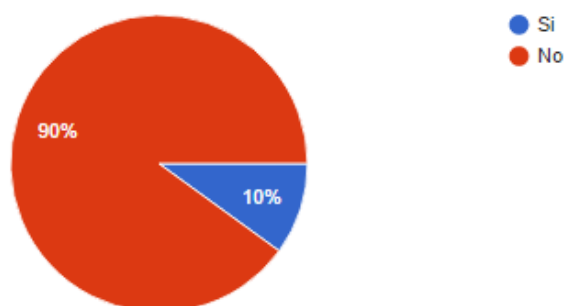


Figura 5: La mayor parte de los encuestados manifiesta que no conoce los turnos en las que se encuentran laborando lo médicos.

Mayor parte de los encuestados que corresponde al 90% manifiesta que no conoce los turnos en las que se encuentran laborando lo médicos de una unidad de salud en particular, por ende el 10% correspondiente a 10 personas manifiesta que si conoce.

En el caso de ser una unidad de salud privada. ¿Sabe usted el valor de la consulta?

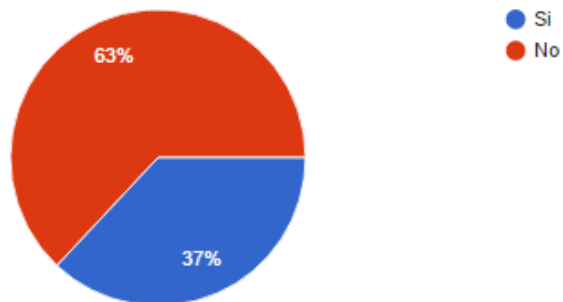


Figura 6: Las personas desconocen el valor de las consultas médicas.

De aquellas personas que desconocen el valor de las consultas médicas, el 63% de la muestra total afirma dicha aclaración, mientras que un 37% opina que si conocen el precio de una revisión médica.

¿Qué problemas considera usted que se generan por el desconocimiento de información?

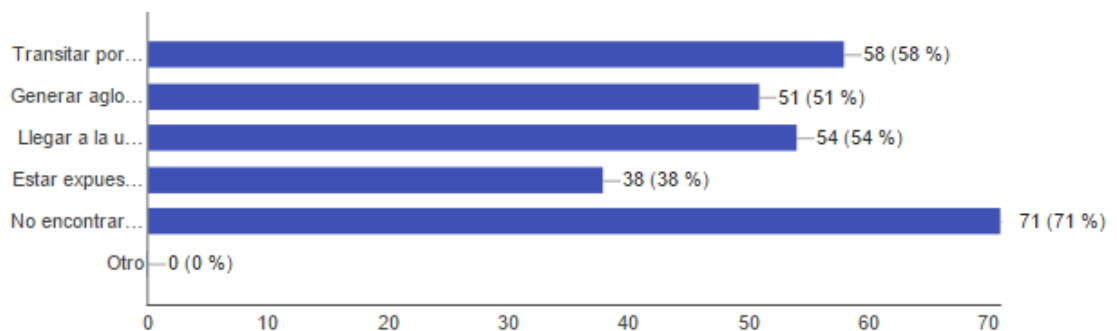


Figura 7: Población afirma que el mayor problema que genera el desconocimiento de información es el no encontrar el medico que necesitaba para su atención.

Los problemas que acarrea el desconocimiento de información de las unidades de salud, tenemos que el 71% manifestó que están expuestos a no encontrar al médico que necesitan, mientras que un 58% seleccionaron que el problema radica en transitar por rutas erróneas en el caso de una emergencia, así mismo el 54% considera que los problemas que podrían suscitarse es de llegar a la unidad de salud en horas no laborales, mientras que el 51% cree que el inconveniente es generar aglomeración, a su vez retraso en la atención y, finalmente el 38% cree se está expuesto a cobros indebidos.

¿Tiene conocimiento de alguna aplicación móvil que brinde información sobre las unidades de salud (Públicas y Privadas)?

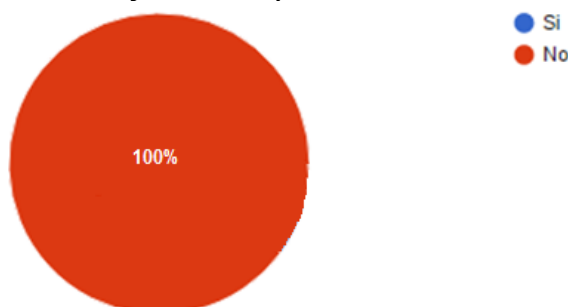


Figura 8: Toda la población afirma que desconoce de aplicaciones móviles que brinden información de las unidades de salud.

El 100% respondió negativamente que no conocen ninguna aplicación móvil que le permita brindar información de las unidades de salud públicas y privadas.

¿Quedó satisfecho con el desempeño mostrado por las aplicaciones que ha utilizado?

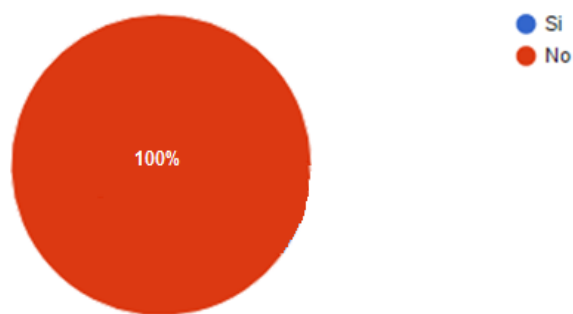


Figura 9: La población confirma su insatisfacción del desempeño de las aplicaciones que ha utilizado de información

El 100% respondió negativamente que están descontentos con el desempeño mostrado por la aplicación.

¿Cuál de los siguientes medios, le parece más eficiente para obtener información de las unidades de salud?

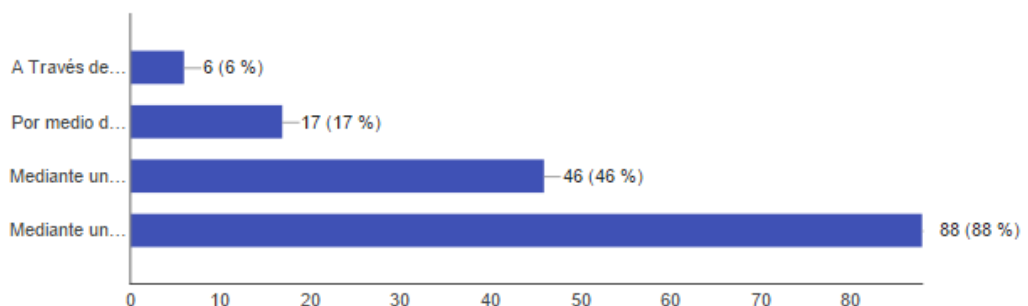


Figura 10: La población manifiesta que los medios más eficaces para obtener información de las unidades de salud son mediante una aplicación móvil.

Según los resultados expresados por la población, considera que los medios más eficientes para obtener información de las unidades de salud son: con el 88% manifiesta la creación de una aplicación para su celular, el 46% indicó que el medio más óptimo sería una página web, mientras que el 17% reveló que por medio de una línea telefónica es su mejor opción, finalmente el periódico ocupa el 6% de la muestra total.

Apéndice 4: Encuesta Dos



UNIVERSIDAD NACIONAL DE LOJA
ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES NO
RENOVABLES
INGENIERÍA EN SISTEMAS

Encuesta dirigida a los usuarios

Como estudiante de la "Universidad Nacional de Loja" de la carrera de Ingeniería en Sistemas, solicito comedidamente responder la presente encuesta con el fin de obtener datos necesarios para el desarrollo del Trabajo de Titulación.

La encuesta tiene como objetivo recolectar información

¿Desearía que la aplicación móvil, encuentre la Unidad de Salud más cercana dependiente de su ubicación?

Si ()

No ()

Qué información de los centros de Salud de la ciudad de Loja, desea que la aplicación móvil maneje.

Dirección ()

Nivel de Atención ()

Lucro ()

Número Telefónico ()

Parroquia ()

Otro _____

En cuanto a las áreas especializadas que cuentan los centros de salud, que datos prefiere usted que la aplicación móvil presente.

Nombre del médico responsable ()

Horario de Atención ()

Días de Atención ()

Otro _____

Que información cree que es necesaria que la aplicación móvil maneje en cuanto a los servicios que ofrece las Unidades de Salud.

Nombre del profesional responsable ()

Número Telefónico ()

Horario ()

Otro _____

Acerca de los horarios de trabajo, que información cree que le resulte útil conocer.

Nombre del Medico ()

Especialidad ()

Área Especialidad Asignada ()

Días de Atención ()

Otro _____

GRACIAS POR SU COLABORACIÓN

Apéndice 5: Resultados de la Encuesta Dos

El número de personas que fueron encuestas corresponde a 100.

¿Desearía que la aplicación móvil, encuentre la Unidad de Salud más cercana dependiente de su ubicación?

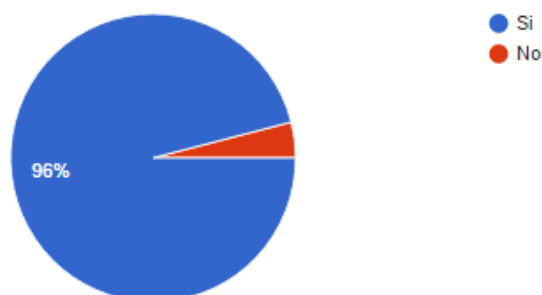


Figura 11: La población manifiesta que está de acuerdo que la aplicación móvil encuentre a la unidad de salud más cercana a su ubicación.

La mayor parte de la muestra que corresponde al 96% de personas encuestadas manifiesta que están de acuerdo que la aplicación móvil encuentre a la unidad de salud más cercana a su ubicación. Por otro lado, el 4% que pertenece a 4 personas de la muestra total, concluyeron que no están de acuerdo.

Qué información de los centros de Salud de la ciudad de Loja, desea que la aplicación móvil maneje.

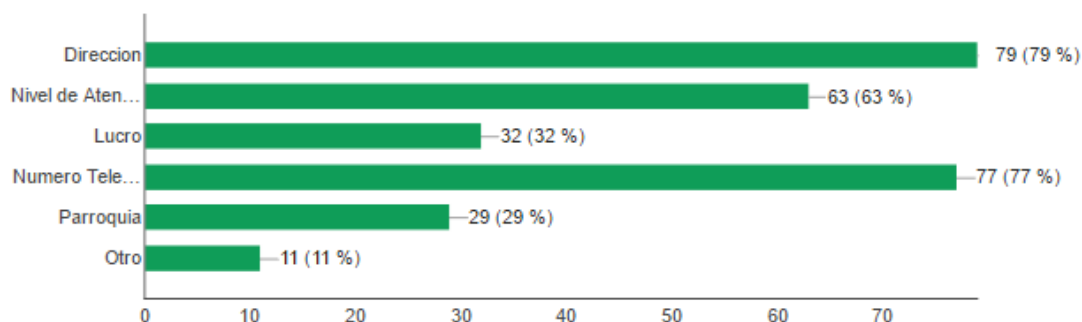


Figura 12: La población manifiesta que la dirección del establecimiento de salud es la información más importante que se debería manejar.

La información que la población desea que la aplicación móvil maneje son: Con el 79% la dirección del establecimiento de salud, mientras que el 77% manifiesta que desearía obtener información del número telefónico, por otro lado el 63% cree que se debería presentar información de Nivel de Atención, así mismo el 32% considera que se debería tener información sobre el establecimiento es o no con fines de lucro, el 29% manifiesta que se debería presentar información de la parroquia, y finalmente el 11% considera que debería manejarse otro tipo de datos.

En cuanto a las áreas especializadas que cuentan los centros de salud, que datos prefiere usted que la aplicación móvil presente.

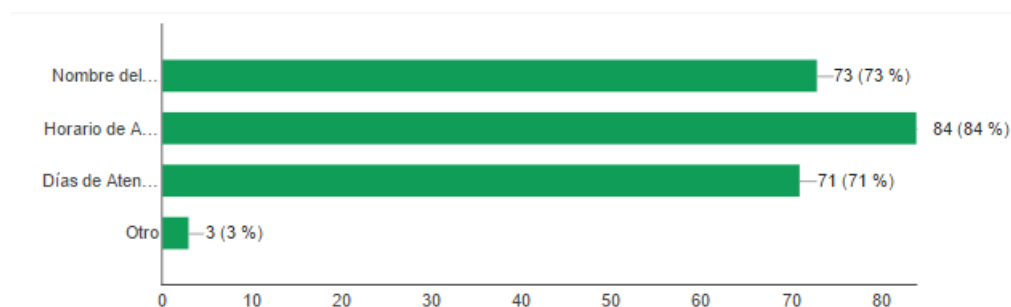


Figura 13: La población manifiesta que el horario de atención es la información más importante que se debería manejar la aplicación móvil.

En cuanto a las áreas de especialidad que disponen las unidades de salud, la información que la aplicación móvil debería presentar es: Con el 84% el horario de atención, mientras que un 73% desearía obtener información del nombre del médico responsable, por otra parte el 71% considera que los días de atención se debería mostrar en la aplicación, finalmente el 3% considera que debería manejarse otro tipo de datos.

Que información cree que es necesaria que la aplicación móvil maneje en cuanto a los servicios que ofrece las Unidades de Salud.

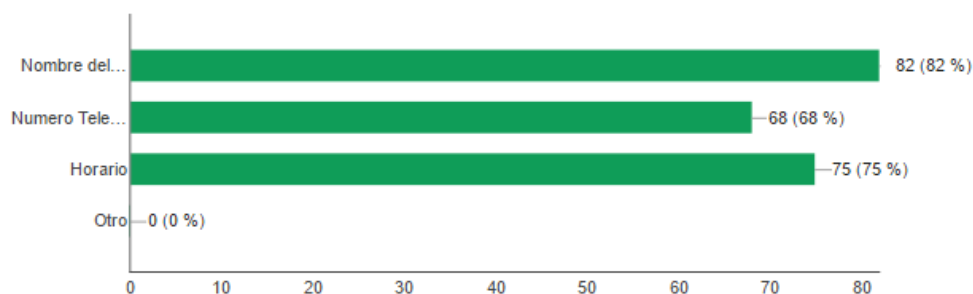


Figura 14: La población manifiesta que el nombre del profesional responsable es la información más importante que se debería manejar la aplicación móvil.

En cuanto a los servicios que ofrecen las unidades de salud, la información que la aplicación móvil debería presentar es: Con el 82% el nombre del profesional responsable, mientras que un 75% desearía obtener información del Horario de Atención, por otra parte el 68% considera que el número de teléfono se debería mostrar en la aplicación.

Acerca de los horarios de trabajo, que información cree que le resulte útil conocer.

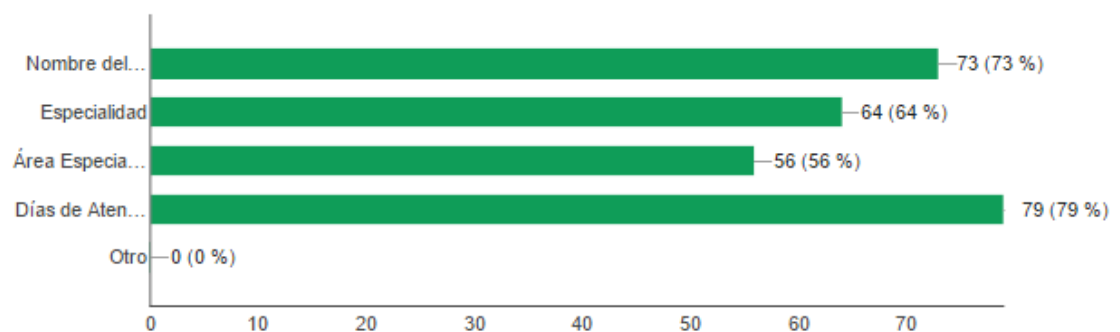


Figura 15: La población manifiesta que los días de atención es la información más importante que se debería manejar la aplicación móvil.

En cuanto a los horarios de trabajo, la información que la aplicación móvil debería presentar es: Con el 79% los días de atención, mientras que un 73% desearía obtener información del nombre del médico, por otra parte el 64% considera que la especialidad se debería mostrar en la aplicación, finalmente el 56% considera que debería manejar el área de especialidad asignada

ANEXO II: PROTOTIPO DE LA APLICACIÓN MÓVIL

- Interfaz de inicio

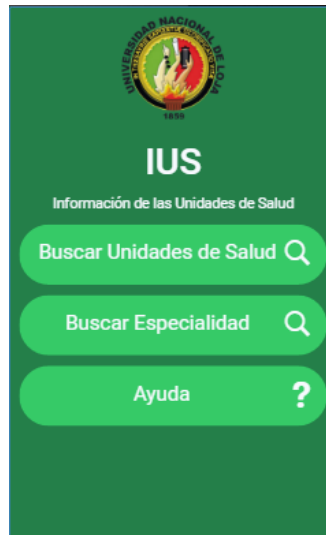


Figura 55: Interfaz Inicial

El usuario puede acceder por medio de los botones a las funcionalidades que ofrece la aplicación:

- Buscar Unidades de Salud
- Buscar Especialidades
- Ayuda

- Interfaz de Ayuda

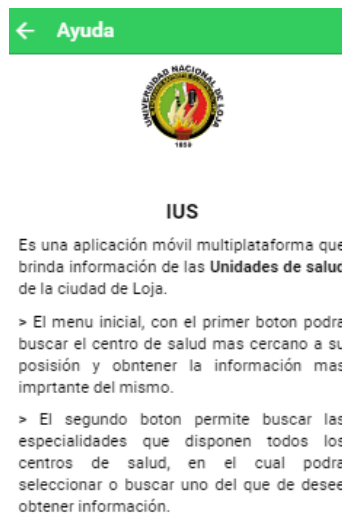


Figura 56: Interfaz de Ayuda

En la presente interfaz muestra la información relacionada al uso de la aplicación.

- **Interfaz de Google Maps**

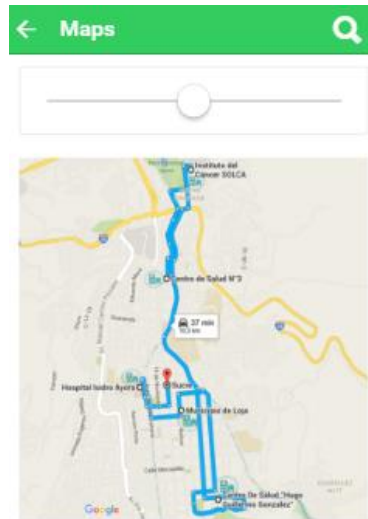


Figura 57: Interfaz de Google Maps

Esta interfaz permite localizar la unidad de salud más cercana a la posición del usuario, además dispone de un botón que permite filtrar la búsqueda, si este está a la izquierda, mostrara centros de salud públicos y si está a la derecha presentará centros de salud privados. En la parte superior izquierda se encuentra el botón de buscar, que permite cargar la interfaz para buscar unidades de salud por nombre.

- **Interfaz Buscar Unidad de Salud**



Figura 58: Interfaz de Información de Unidades de Salud

En esta interfaz el usuario podrá buscar el centro de salud de forma más directa, conociendo únicamente el nombre de la unidad de salud a la que desea obtener información.

- **Interfaz de Información Unidad de Salud**



Figura 59: Interfaz de Información de Unidades de Salud

En la presenta interfaz se muestra la información de un centro de salud que se haya elegido, en ella se presenta la imagen e información adicional como tipología, sector, número de teléfono, dirección referente al establecimiento de salud, además posee dos botones, la misma que cargan los datos dependiendo de su función:

- Áreas de Especialidad.
- Servicios.

- **Interfaz de Áreas de Especialidad**

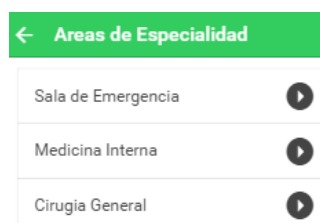


Figura 60: Interfaz de las Áreas de Especialidad que dispone una Unidad de Salud

El usuario puede seleccionar en esta interfaz, el área el cual desea obtener información.

- **Interfaz de información de las Especialidades**

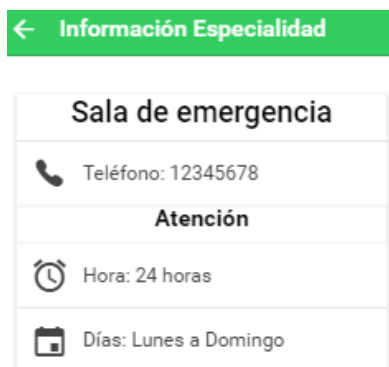


Figura 61: Interfaz de Información de las especialidades

Interfaz que presenta la información de las especialidades que ofrece un área en común de un centro de salud.

- **Interfaz de Servicios**

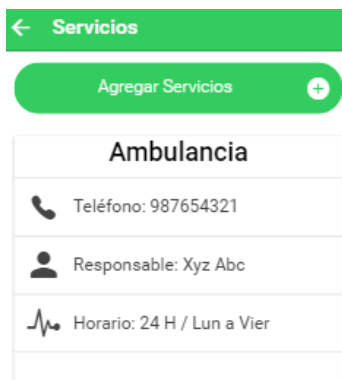


Figura 62: Interfaz de Información de los Servicios

Por medio de esta interfaz el usuario puede conocer los servicios que ofrecen un establecimiento de salud, además tiene la posibilidad de agregar un servicio, por medio del botón.

- **Interfaz del Formulario del Contribuidor**



Figura 63: Interfaz del Formulario del Contribuidor

El usuario debe ingresar la información del contribuidor solicitada en este formulario cuando se dispone a crear un nuevo servicio.

- **Interfaz de Formulario del Servicio**



Figura 64: Interfaz del Formulario de Servicio

El usuario debe ingresar la información del servicio solicitada en este formulario cuando se dispone a crear un nuevo servicio, el administrador de la aplicación web tendrá la prioridad de dar de alta o eliminar la información ingresada.

- **Interfaz de Buscar Especialidad**

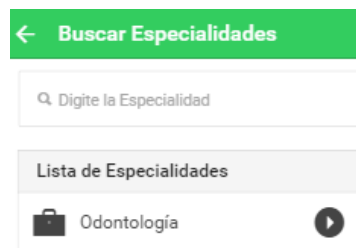


Figura 65: Interfaz de Buscar Especialidad

El usuario debe ingresar el nombre de la especialidad que desea obtener información, la aplicación automáticamente realizara un filtro de la búsqueda respecto a lo ingresado por el usuario.

- **Interfaz de Resultado de la Búsqueda**



Figura 66: Interfaz de Resultado de la Búsqueda

El usuario dispondrá en esta interfaz de una lista de las unidades de salud que disponen de la especialidad solicitada en la pantalla anterior.

- **Interfaz de Información de Unidad Salud.**



Figura 67: Interfaz de información del Resultado de Búsqueda

El usuario dispondrá en esta interfaz de la información completa de la Unidad de Salud y la especialidad que ha solicitado en la búsqueda realizada con anterioridad.

ANEXO III: PROTOTIPO DE LA APLICACIÓN WEB

- Interfaz de Login

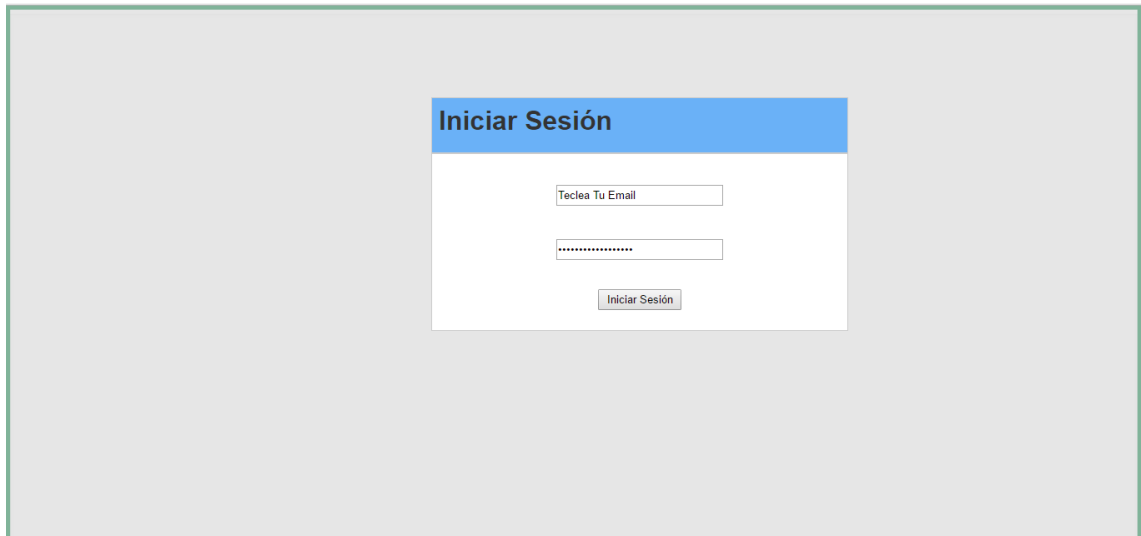


Figura 68: Interfaz de Login

El usuario debe ingresar su correo y contraseña, para tener acceso al sistema, en el caso que esta información sea correcta, caso contrario no podrá tener acceso.

- Interfaz de Inicio



Figura 69: Interfaz de Inicio

La interfaz de inicio permite tener acceso a todas las funcionalidades que ofrece la aplicación web, como son crear y modificar unidades de salud, especialidades, servicios.

- **Interfaz de Crear Unidad de Salud**

Figura 70: Interfaz de Crear Unidad de Salud

La interfaz permite al usuario por medio de un formulario crear una unidad de salud.

- **Interfaz de Crear Especialidad**

Figura 71: Interfaz de Crear Especialidad

La interfaz permite al usuario crear una especialidad por medio de un formulario.

- **Interfaz de Crear Servicio**

Figura 72: Interfaz de Crear Servicio

La interfaz permite al usuario crear un servicio por medio de un formulario.

- **Interfaz de Lista de Unidades de Salud**

Opciones	Nombre	Lucro	Telefono	Provincia	Dirección
	HOSPITAL GENERAL ISIDRO AYORA	SIN FINES	7294	Loja	J. J. SAMANIEGO E IBEROAME
	DANIEL ALVAREZ	SIN FINES	7294	Loja	ANTONIO NEUMANE Y ANTONI
	HOSPITAL UTPL	CON FINES	2570540	Loja	AV. SALVADOR BUSTAMANTE C
	HOSPITAL CLINICA SAN AGUSTIN	CON FINES	7288	Loja	18 DE NOVIEMBRE Y AZUAY

Figura 73: Interfaz de Listar Unidades de Salud

La interfaz presenta todas y cada una de las centros de salud, con la opción de filtrar unidades de salud públicas y privadas, además la opción de editar y eliminar según lo requiere el usuario.

- Interfaz de Información de Unidad de Salud

Sistema de Información Aplicación Móvil Bienvenido: admin | Tipo Usuario: ADMINISTRADOR [[Mis Datos] [Salir]

HOSPITAL PROVINCIAL ISIDRO AYORA Administrar información

NOMBRE DE LA UNIDAD OPERATIVA: HOSPITAL PROVINCIAL ISIDRO AYORA LOJA TELEFONO: 072570540

NIVEL DE ATENCION Y COMPLEJIDAD: SEGUNDO NIVEL CORREO ELECTRONICO: mecueva@hotmail.com

OPCIONES	UNICODIGO	DIRECCION	PROVINCIA	CANTON	PARROQUIA	PARROQUIA TIPO	ZONA	ZONA DISTRIBUCION	DISTRITO	DISTRITO DISTRIBUCION
	001145	J. J. SAMANIEGO E IBEROAMERICA	LOJA	LOJA	LOJA	U	ZONA 7	EL ORO, LOJA, ZAMORA CHINCHIPE	11D01	LOJA

OPCIONES	CIRCUITO	AREA CODIGO	AREA	RED ATENCION	LUCRO	INSTITUCION	JEFATURA	NIVEL ATENCION	TIPOLOGIA	HORARIO ATENCION
	11D01C08	11A02	CENTRO DE SALUD Nº 2	RED PUBLICA	NO DEFINIDO	MSP	NO	NIVEL 2	HOSPITAL GENERAL	24 Horas

Figura 74: Interfaz de Información Unidades de Salud

La interfaz presenta la información completa de un centro de salud específico, con la opción de editar.

- Interfaz de Lista de Especialidades

Sistema de Información Aplicación Móvil Bienvenido: admin | Tipo Usuario: ADMINISTRADOR [[Mis Datos] [Salir]

HOSPITAL REGIONAL ISIDRO AYORA Nuevo

Listado De Especialidades

Opciones	Nombre	Disponibilidad	Telefono	Profesional Responsable	Horario de Atención	Días de Atención
	Sala de Emergencia	SI		JORGE GUAPULEMA	24H00	LUNES A I
	Triaje	SI		ANGELICA GORDILLO	24H00	LUNES A I
	Sala de Shock Trauma	SI		JORGE GUAPULEMA	24H00	LUNES A I
	Sala de Reanimación	SI		JORGE GUAPULEMA	24H00	LUNES A I
	Sala de Observación para Adultos	NO		JORGE GUAPULEMA	24H00	LUNES A I
	Sala de Observación Pediátrica	NO			08H00	OTRO
	Quirófano de Emergencias	NO			08H00	OTRO
	Sala de Recuperación	NO			08H00	OTRO

Figura 75: Interfaz de Listar Especialidades

La interfaz muestra una lista de todas las especialidades con las que cuenta un centro de salud en concreto, además cuenta con la opción de crear, editar y eliminar según lo requiere el usuario.

- Interfaz de Lista de Servicios

Sistema de Información Aplicación Móvil Bienvenido: admin | Tipo Usuario: ADMINISTRADOR [[Mis Datos] [Salir]

HOSPITAL REGIONAL ISIDRO AYORA Nuevo

Menú Principal

- Unidades de Salud
 - Publicas
 - ISIDRO AYORA
 - Especialidad
 - Servicios
 - DANIEL ALVAREZ
 - Privadas

Listado De Servicios

Opciones	Descripción	Disponibilidad	Telefono	Nombre del Responsable	Horario de Atención
	Farmacia	SI	7294	JENNY LEON	24 H de Lunes a Viernes
	Ambulancias	SI	7294	FABIAN VALDIVIESO	12H de Lunes a Viernes
	Farmacia	SI	2570540	ANGELICA GORDILLO	24 H de Lunes a Viernes
	Ambulancias	SI	7288	DANIEL LEON	12 H de Lunes a Viernes

Figura 76: Interfaz de Listar Servicios

La interfaz muestra una lista de todos los servicios con las que cuenta un centro de salud en concreto, además cuenta con la opción de crear, editar y eliminar según lo requiere el usuario.

ANEXO IV: PRUEBAS DE CARGA

Haciendo uso de la herramienta JMeter, se efectuaron las pruebas al servicio web, para cada caso de prueba se ha considerado 200 hilos (usuarios) con una frecuencia de 5 segundos.

- Caso de Prueba de la Clase AreaEspecialidad_SW
 - Función Find_get

Petición HTTP

Nombre: find_get

Comentarios: Permite obtener la informacion de las especialidades por medio del ID de la unidad de salud

Basic Advanced

Servidor Web

Nombre de Servidor o IP: Puerto:

Petición HTTP

Implementación HTTP: Protocolo: Método: GET

Ruta: http://localhost/sius/index.php/AreaEspecialidad_SW/165

Figura 77: Prueba de la clase AreaEspecialidad_SW, función find_get

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error
find_get	200	179546	2015	552832	251839,47	0,00%
Total	200	179546	2015	552832	251839,47	0,00%

Figura 78: Reporte resumen de la clase AreaEspecialidad_SW, función find_get

Como se puede evidenciar en la figura 78 el porcentaje de error es de 0% con una carga de 200 usuarios.

- **Función datos_get**

Petición HTTP

Nombre:

Comentarios: Obtiene informacion de las unidades, areas de espeialidad por medio del la especialidad que es enviada por parametro

Basic | Advanced

Servidor Web

Nombre de Servidor o IP: Puerto: Timeout: Conexión:

Petición HTTP

Implementación HTTP: Protocolo: Método: Codificación del contenido:

Ruta:

Figura 79: Reporte resumen de la clase AreaEspecialidad_SW, función datos_get

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error
datos_get	200	6435	1405	11715	3126,83	0,00%
Total	200	6435	1405	11715	3126,83	0,00%

Figura 80: Reporte resumen de la clase AreaEspecialidad_SW, función datos_get

Como se puede evidenciar en la figura 80 el porcentaje de error es de 0% con una carga de 200 usuarios.

- **Función datosId_get**

Petición HTTP

Nombre:

Comentarios: Obtiene informacion de las unidades, areas de espeialidad por medio del la ID de la especialidad

Basic | Advanced

Servidor Web

Nombre de Servidor o IP: Puerto:

Petición HTTP

Implementación HTTP: Protocolo: Método: Codifica:

Ruta:

Figura 81: Reporte resumen de la clase AreaEspecialidad_SW, función datosId_get

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error
datos_get	200	6550	1363	12261	3328,18	0,00%
Total	200	6550	1363	12261	3328,18	0,00%

Figura 82: Reporte resumen de la clase AreaEspecialidad_SW, función datosId_get

Como se puede evidenciar en la figura 82 el porcentaje de error es de 0% con una carga de 200 usuarios.

- **Caso de Prueba de la Clase Contribuidor_SW**
 - **Función Index_get**

Figura 83: Reporte resumen de la clase Contribuidor_SW, función Index_get

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error
index_get	200	8504	1452	15448	4136,92	0,00%
Total	200	8504	1452	15448	4136,92	0,00%

Figura 84: Reporte resumen de la clase Contribuidor_SW, función index_get

Como se puede evidenciar en la figura 84 el porcentaje de error es de 0% con una carga de 200 usuarios.

- **Función index_delete**

Petición HTTP

Nombre:

Comentarios: Esta función permite eliminar un contribuidor po medio del ID

Basic **Advanced**

Servidor Web

Nombre de Servidor o IP: Puerto:

Petición HTTP

Implementación HTTP: Protocolo: Método:

Ruta:

Figura 85: Reporte resumen de la clase Contribuidor_SW, función Index_delete

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error
index_delete	200	9092	1193	17860	5110,62	0,00%
Total	200	9092	1193	17860	5110,62	0,00%

Figura 86: Reporte resumen de la clase Contribuidor_SW, función index_delete

Como se puede evidenciar en la figura 86 el porcentaje de error es de 0% con una carga de 200 usuarios.

- **Caso de Prueba de la Clase Especialidad_SW**
 - **Función Index_get**

Petición HTTP

Nombre:

Comentarios: Permite obtener la informacion de las especialidades

Basic **Advanced**

Servidor Web

Nombre de Servidor o IP: Puerto:

Petición HTTP

Implementación HTTP: Protocolo: Método:

Ruta:

Figura 87: Reporte resumen de la clase Especialidad_SW, función index_get

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error
index_get	200	9977	1245	19390	5853,81	0,00%
Total	200	9977	1245	19390	5853,81	0,00%

Figura 88: Reporte resumen de la clase Especialidad_SW, función index_get

Como se puede evidenciar en la figura 88 el porcentaje de error es de 0% con una carga de 200 usuarios.

- **Función Find_get**

Figura 89: Reporte resumen de la clase Especialidad_SW, función find_get

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error
find_get	200	11309	1209	22128	6466,80	0,00%
Total	200	11309	1209	22128	6466,80	0,00%

Figura 90: Reporte resumen de la clase Especialidad_SW, función find_get

Como se puede evidenciar en la figura 90 el porcentaje de error es de 0% con una carga de 200 usuarios.

- **Caso de Prueba de la Clase Servicios_SW**

- **Función Id_get**

Petición HTTP

Nombre: find_get

Comentarios: Permite obtener la información de las especialidades por medio del ID del area

Basic Advanced

Servidor Web

Nombre de Servidor o IP: Puerto:

Petición HTTP

Implementación HTTP: Protocolo: Método: GET

Ruta: http://localhost/sius/index.php/Especialidad_SW/1

Figura 91: Reporte resumen de la clase Servicios_SW, función find_get

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error
find_get	200	11309	1209	22128	6466,80	0,00%
Total	200	11309	1209	22128	6466,80	0,00%

Figura 92: Reporte resumen de la clase Servicios_SW, función find_get

Como se puede evidenciar en la figura 92 el porcentaje de error es de 0% con una carga de 200 usuarios.

- **Función Disponibilidad_get**

Petición HTTP

Nombre: disponibilidad_get

Comentarios: Obtiene información de las especialidades que tienen como estado igual a 2

Basic Advanced

Servidor Web

Nombre de Servidor o IP: Puerto:

Petición HTTP

Implementación HTTP: Protocolo: Método: GET

Ruta: http://localhost/sius/index.php/Servicios_SW/disponibilidad

Figura 93: Reporte resumen de la clase Servicios_SW, función disponibilidad_get

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error
disponibilida...	200	9379	1207	19532	5438,87	0,00%
Total	200	9379	1207	19532	5438,87	0,00%

Figura 94: Reporte resumen de la clase Servicios_SW, función disponibilidad_get

Como se puede evidenciar en la figura 94 el porcentaje de error es de 0% con una carga de 200 usuarios.

- **Función Index_post**

Petición HTTP

Nombre: index_post

Comentarios: Esta función permite crear servicios por medio de la función post

Basic | **Advanced**

Servidor Web

Nombre de Servidor o IP: Puerto: Timeout (milisegundos) Conexión: Respuesta:

Petición HTTP

Implementación HTTP: Protocolo: Método: **POST** Codificación del contenido:

Ruta: http://localhost/sius/index.php/Servicios_SW

Redirigir Automáticamente Seguir Redirecciones Utilizar KeepAlive Usar 'multipart/form-data' para HTTP POST Cabeceras compatibles con navegadores

Parameters | **Body Data** | **Files Upload**

Enviar Parámetros Con la Petición:

Nombre:	Valor	¿Codificar?	¿Incluir Equals?
unidad_SaludIDUNIDADSAUD	165	<input type="checkbox"/>	<input checked="" type="checkbox"/>
unidad_SaludUNICODIGO	1145	<input type="checkbox"/>	<input checked="" type="checkbox"/>
contribuidorIDCONTRIBUIDOR	2	<input type="checkbox"/>	<input checked="" type="checkbox"/>
DESCRIPCION	Prueba	<input type="checkbox"/>	<input checked="" type="checkbox"/>
TELEFONO	123456789	<input type="checkbox"/>	<input checked="" type="checkbox"/>
NOMBRE_RESPONSABLE	Dr Prueba	<input type="checkbox"/>	<input checked="" type="checkbox"/>
HORARIO	8 a 17 H/ Lun a Vier	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Figura 95: Reporte resumen de la clase Servicios_SW, función index_post

Etiqueta	# Muestras	Media	Mín	Máx	Dev. Estándar	% Error	Rendimiento
index_post	200	9991	1703	19359	5473,00	1,00%	8,4/sec
Total	200	9991	1703	19359	5473,00	1,00%	8,4/sec

Figura 96: Reporte resumen de la clase Servicios_SW, función index_post

Como se puede evidenciar en la figura 96 el porcentaje de error es de 1% con una carga de 200 usuarios.

ANEXO V: PRUEBAS DE ACEPTACIÓN REALIZADAS AL USUARIO



UNIVERSIDAD NACIONAL DE LOJA
 ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES NO
 RENOVABLES
 INGENIERÍA EN SISTEMAS

Como estudiante de la "Universidad Nacional de Loja" de la carrera de Ingeniería en Sistemas, solicito comedidamente responder la presente encuesta con el fin de obtener datos necesarios para el desarrollo del Trabajo de Titulación.
 La encuesta tiene como objetivo recolectar información

La aplicación web y móvil satisface las necesidades expuestas en los requerimientos solicitados.

Si ()
 No ()

El diseño de la aplicación móvil resultó fácil de comprender.

Si ()
 No ()

La información presentada en la aplicación móvil y web resultó ser útil.

Si ()
 No ()

Marque su respuesta en la casilla según crea conveniente, considerando los siguientes lineamientos:

- Bueno
- Regular
- Malo

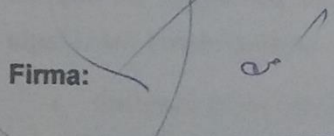
Criterios a Evaluar	Valor		
	Bueno	Regular	Malo
Diseño de la interfaz	<input checked="" type="checkbox"/>		
Velocidad de respuesta	<input checked="" type="checkbox"/>		
Visibilidad del texto	<input checked="" type="checkbox"/>		

Marque su respuesta en la casilla según crea conveniente, considerando los siguientes lineamientos:

- **Exitoso:** El resultado obtenido ha sido el correcto o esperado.
- **Detenido:** La ejecución del proceso ha tenido inconvenientes se ha detenido.
- **Fallido:** El resultado de la prueba no ha sido el esperado.

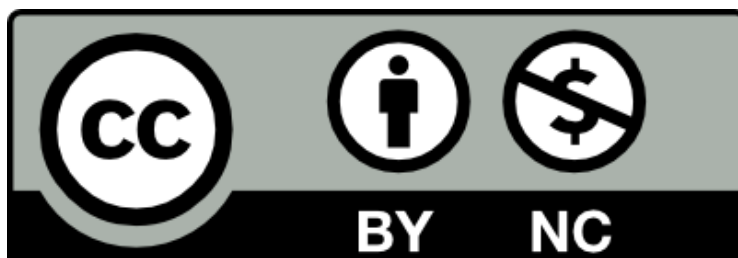
Criterios a Evaluar de la aplicación.	Valor		
	Exitoso	Detenido	Fallido
Buscar Unidad de Salud	X		
Buscar Unidad de Salud publica	X		
Buscar Unidad de Salud privada	X		
Buscar Especialidades	X		
Crear Contribuidor	X		
Crear Servicio	X		
Obtener información de la Unidad de Salud.	X		
Obtener información de la especialidad	X		
Obtener información del servicio	X		
Criterios a Evaluar de la aplicación web	Valor		
	Exitoso	Detenido	Fallido
Crear Unidad de Salud	X		
Modificar Unidad de Salud	✓		
Eliminar Unidad de Salud	X		
Buscar Unidad de Salud	X		
Listar Unidad de Salud	X		
Crear Especialidad	X		
Modificar Especialidad	X		
Eliminar Especialidad	X		
Crear Servicio	X		
Modificar Servicio	✓		
Gestionar contribuidor	X		

Nombre: Dr. Carlos Celi

Firma: 

Dr. Carlos E. Celi A.
 MEDICO CIRUJANO
 Mercadillo 15-09 y Sucre
 Mat. CML-N9 192 T.576025

ANEXO VI: LICENCIA CREATIVE COMMONS



Aplicación móvil multiplataforma que brinda información de las unidades de salud públicas y privadas de la ciudad de Loja por Atencia Pogo Kevin Luis se distribuye bajo una Licencia Creative Commons Atribución-NoComercial 4.0 Internacional. Para ver una copia de esta licencia, visite <http://creativecommons.org/licenses/by/40/> .