



UNL

Universidad
Nacional
de Loja



Carrera de Ingeniería en
Sistemas / Computación

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

CARRERA DE INGENIERÍA EN SISTEMAS

“DESARROLLO DE UNA APLICACIÓN INFORMÁTICA PARA LA BÚSQUEDA DE OFERTA DE ALQUILER DE INMUEBLES EN LA CIUDAD DE LOJA”

“Trabajo de Titulación previo a la obtención
del título de Ingeniero en Sistemas”

Autor:

- Steeven Michael Armijos Bravo

Director:

- Wilman P. Chamba Zaragocín Mg.Sc.

Loja-Ecuador
2022

Certificación de tesis:

Ing. Wilman Patricio Chamba Zaragocín, Mg. Sc

DIRECTOR DE TESIS

CERTIFICA:

Haber asesorado y revisado detenida y minuciosamente durante todo su desarrollo, al Trabajo de Titulación, titulado “**DESARROLLO DE UNA APLICACIÓN INFORMÁTICA PARA LA BÚSQUEDA DE OFERTA DE ALQUILER DE INMUEBLES EN LA CIUDAD DE LOJA**”, realizado por el Sr. Steeven Michael Armijos Bravo; cumple con los requisitos establecidos por las normativas para la graduación en la Universidad Nacional de Loja, tanto en aspecto como de contenido.

Loja, 20 de agosto de 2021

Ing. Wilman Patricio Chamba Zaragocín, Mg. Sc

DIRECTOR DE TESIS

Autoría.

Yo, **Steeven Michael Armijos Bravo**, declaro ser el autor del presente Trabajo de Titulación y eximo expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos de posibles reclamos o acciones legales, por el contenido de la misma. Adicionalmente acepto y autorizo a la Universidad Nacional de Loja, la publicación de mi trabajo de titulación en el Repositorio Institucional, Biblioteca Virtual.

Firma:

Cedula: 0706215738

Fecha: Loja, 24 de marzo de 2022

Correo Electrónico: smarmijosb@unl.edu.ec

Celular: 0989697404

CARTA DE AUTORIZACIÓN DE TESIS POR PARTE DEL AUTOR, PARA LA CONSULTA, REPRODUCCIÓN PARCIAL O TOTAL Y PUBLICACIÓN ELECTRÓNICA DEL TEXTO COMPLETO.

Yo, **STEEVEN MICHAEL ARMIJOS BRAVO** declaro ser autor del trabajo de titulación que versa: “**DESARROLLO DE UNA APLICACIÓN INFORMÁTICA PARA LA BÚSQUEDA DE OFERTA DE ALQUILER DE INMUEBLES EN LA CIUDAD DE LOJA**”, como requisito para optar al 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 (RDI):

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

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

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

Firma:

Autor: Steeven Michael Armijos Bravo

Cédula: 0706215738

Dirección: Barrio Daniel Álvarez Burneo

Correo Electrónico: smarmijosb@unl.edu.ec

Celular: 0989697404

DATOS COMPLEMENTARIOS:

Director de Tesis: Ing. Wilman Patricio Chamba Zaragocín, Mg.Sc

Tribunal de Grado: Ing. Edwin René Guamán Quinche.

Ing. Genoveva Jackelinne Suing Albito.

Ing. Mario Alexander Zambrano Mora.

Dedicatoria.

A mis padres Olga Bravo y Cosme Armijos por creer en mí y apoyarme en todo momento, por darme la oportunidad de estudiar y crecer humana y profesionalmente cada día, ya que con su cariño se ha forjado en mí, aquella ilusión de ser una excelente persona, así como lo son mis padres.

A mis hermanos/as Yuliana, Elías y Katy, por formar parte de mi vida y estar conmigo en los buenos y malos momentos, por brindarme su apoyo incondicional en todo momento que lo necesité.

A mis familiares y amigos que siempre estuvieron ahí para motivarme a que estudie y sea un gran profesional, pero más importante, que me forje como una persona humilde y agradecida.

Steeven Armijos

Agradecimiento.

Agradezco a Dios por permitirme llegar hasta este punto tan importante en mi vida, por cuidarme y guiarme hacia el camino del bien, brindándome salud y fortaleza para seguir adelante y ser mejor cada día.

A mi familia por apoyarme en todo momento, por su paciencia y cariño, por sus consejos y la confianza prestada en mí, ya que son el factor de motivación importante que me incita a superarme día a día.

A la Universidad Nacional de Loja, a la Facultad de la Energía, las Industrias y los Recursos Naturales no Renovables, y a la carrera de Ingeniería en Sistemas/Computación por abrirme las puertas para formarme profesionalmente.

Agradezco de manera muy cordial a mi director, Ing. Wilman Chamba, por haberme guiado y apoyado en todo momento, compartiendo sus conocimientos a lo largo del desarrollo de este proyecto para así culminarlo de manera exitosa. De igual manera agradezco a todos los docentes de la carrera de Ingeniería en Sistemas/Computación por su tiempo y dedicación en todo el transcurso de mi formación académica y humana, muchas gracias por todo.

Así mismo, expreso mis agradecimientos a las personas que colaboraron participando en las distintas etapas del desarrollo de este proyecto, familiares, amigos, compañeros y conocidos.

Finalmente agradecer a mis amigos y compañeros a quienes tuve la dicha de conocer y llevarme bien en todo momento, ya que pude contar con su apoyo y motivación, para lo cual estoy enteramente agradecido que formen parte de mi vida.

Steeven Armijos

Índice.

Carátula	i
Certificación de tesis:	ii
Autoría.	iii
Carta de autorización.....	iv
Dedicatoria.....	v
Agradecimiento.	vi
Índice.....	vii
Índice de figuras.....	ix
Índice de tablas.....	xii
1. Título.....	1
2. Resumen.....	2
Abstract	3
3. Introducción	4
4. Marco teórico	6
4.1. Elementos que intervienen en la búsqueda de alquiler de inmuebles.	7
4.1.1. Bien inmueble	7
4.1.2. Arrendador	7
4.1.3. Arrendatario	7
4.1.4. Arriendo o Arrendamiento	8
4.1.5. Canon de arrendamiento	8
4.2. Condiciones de los locales de arrendamiento.	8
4.3. Motivos por los que se busca un bien inmueble.....	8
4.4. Consecuencias de no encontrar un bien inmueble con buenas adecuaciones.	9
4.5. Comparativa de metodologías Ágiles de Desarrollo de Software.....	10
4.5.1. Metodologías de Desarrollo Ágil.	10
4.5.2. Metodología Extreme Programming (XP)	10
4.5.3. Metodología ICONIX	11
4.5.4. Metodología Mobile-D	12
4.5.5. Metodología Desarrollo Rápido de Aplicaciones (RAD)	13
4.5.6. SCRUM	13
4.6. Tabla comparativa de las Metodologías de Desarrollo Ágil.	14
4.7. Herramientas de desarrollo	17

4.8.	Trabajos relacionados.....	21
5.	Metodología.....	24
5.1.	Contexto.....	24
5.2.	Proceso.....	24
5.3.	Recursos.....	25
5.4.	Participantes.....	26
5.5.	Materiales.....	26
6.	Resultados.....	28
6.1.	Objetivo I: Determinar los requisitos necesarios para desarrollar la Aplicación Informática.....	28
6.1.1.	Revisión de trabajos similares para obtener requisitos generales..	28
6.1.2.	Formular y aplicar encuestas a la población seleccionada para obtener requisitos de funcionalidad y usabilidad.....	38
6.1.3.	Especificación de requisitos.....	41
6.2.	Objetivo II: Diseñar e implementar la Aplicación Informática para la búsqueda de alquiler de inmuebles en base a la metodología XP.....	44
6.2.1.	Planificación.....	44
6.2.2.	Diseño.....	65
6.2.3.	Codificación.....	76
6.3.	Objetivo 3: Evaluar la funcionalidad y usabilidad de la Aplicación Informática en un ambiente controlado o simulado.....	81
6.3.1.	Pruebas al Servicio Web API REST.....	81
6.3.2.	Pruebas de Caja Negra.....	83
6.3.3.	Pruebas de Usabilidad.....	85
7.	Discusión.....	97
	Objetivo 1: Determinar los requisitos necesarios para desarrollar la Aplicación Informática.....	97
	Objetivo 2: Diseñar e implementar la Aplicación Informática para la búsqueda de alquiler de inmuebles en base a la metodología XP.....	97
	Objetivo 3: Evaluar la funcionalidad y usabilidad de la Aplicación Informática en un ambiente controlado o simulado.....	98
8.	Conclusiones.....	101
9.	Recomendaciones.....	103
10.	Referencias.....	104
11.	Anexos.....	108

Índice de figuras

Figura 1. Pregunta relevante 1 de la encuesta (arrendatario)	39
Figura 2. Pregunta relevante 2 de la encuesta (arrendatario)	39
Figura 3. Pregunta relevante 3 de la encuesta (arrendatario)	40
Figura 4. Pregunta relevante 4 de la encuesta (arrendatario)	40
Figura 5. Pregunta relevante 5 de la encuesta (arrendatario)	41
Figura 6. Vista general del sistema.....	66
Figura 7. Diagrama de caso de uso - Usuario no registrado.	67
Figura 8. Diagrama de caso de uso - Usuario Arrendador	68
Figura 9. Diagrama de caso de uso - Usuario Arrendatario	68
Figura 10. Diagrama de caso de uso - Usuario Administrador	69
Figura 11. Modelo conceptual.....	70
Figura 12. Diagrama de clases	71
Figura 13. Diagrama de actividades - Publicar bien inmueble.....	72
Figura 14. Diagrama de componentes	73
Figura 15. Diagrama de despliegue de la aplicación LojaHouse	74
Figura 16. Arquitectura lógica de la aplicación LojaHouse	75
Figura 17. Página principal	76
Figura 18. Estructura del Servicio Web API-REST.....	77
Figura 19. Patrón de diseño Modelo Vista Controlador	77
Figura 20. Rutas para el módulo visita	78
Figura 21. Modelo para el módulo visita	78
Figura 22. Controlador para aceptar una visita	79
Figura 23. Respuesta en formato JSON para aceptar una visita.....	79
Figura 24. Página principal	80
Figura 25. Herramienta Postman, petición POST Crear Inmueble	82
Figura 26. Configuración de peticiones.....	84
Figura 27. Gráfico de resultados de la prueba de carga y estrés	85
Figura 28. Reporte de resumen de resultados de la ejecución de la prueba de carga y estrés	85
Figura 29. Prueba de Usabilidad - Pregunta 1	86
Figura 30. Prueba de Usabilidad - Pregunta 2	86
Figura 31. Prueba de Usabilidad - Pregunta 3	86
Figura 32. Prueba de Usabilidad - Pregunta 4	87
Figura 33. Prueba de Usabilidad - Pregunta 5	87
Figura 34. Prueba de Usabilidad - Pregunta 6	87

Figura 35. Prueba de Usabilidad - Pregunta 7	88
Figura 36. Prueba de Usabilidad - Pregunta 8	88
Figura 37. Prueba de Usabilidad - Pregunta 9	88
Figura 38. Prueba de Usabilidad - Pregunta 10	89
Figura 39. Prueba de Usabilidad - Pregunta 11	89
Figura 40. Prueba de Usabilidad - Pregunta 12	89
Figura 41. Prueba de Usabilidad - Pregunta 13	90
Figura 42. Prueba de Usabilidad - Pregunta 14	90
Figura 43. Prueba de Usabilidad - Pregunta 15	90
Figura 44. Prueba de Usabilidad - Pregunta 16	91
Figura 45. Grado de usabilidad total	94
Figura 46. Prueba de Validación - Pregunta 1	94
Figura 47. Prueba de Validación - Pregunta 2	95
Figura 48. Prueba de Validación - Pregunta 3	95
Figura 49. Prueba de Validación - Pregunta 4	95
Figura 50. Prueba de Validación - Pregunta 5	96
Figura 51. Prueba de Validación - Pregunta 6	96
Figura 52. Prueba de Validación - Pregunta 7	96
Figura 53. Creación de la prueba en apache Jmeter	188
Figura 54. Configuración de peticiones	189
Figura 55. Gráfico de resultados de la prueba de carga y estrés	189
Figura 56. Reporte de resumen de resultados de la ejecución de la prueba de carga y estrés	190
Figura 57. Diagrama de caso de uso -Usuario no registrado.	194
Figura 58. Diagrama de caso de uso -Usuario Arrendador	195
Figura 59. Diagrama de caso de uso -Usuario Arrendatario	195
Figura 60. Diagrama de caso de uso -Usuario Administrador	196
Figura 61. Modelo conceptual.....	197
Figura 62. Diagrama de clases	198
Figura 63. Diagrama de actividades - Registrar cuenta.....	199
Figura 64. Diagrama de actividades - Inicio de sesión	200
Figura 65. Diagrama de actividades - Filtrar inmuebles	201
Figura 66. Diagrama de actividades - Enviar mensaje	202
Figura 67. Diagrama de actividades - Resetear contraseña.....	203
Figura 68. Diagrama de actividades - Ver información del inmueble.....	204
Figura 69. Diagrama de actividades - Gestionar bien inmueble	205

Figura 70. Diagrama de actividades - Publicar bien inmueble.....	206
Figura 71. Diagrama de actividades - Filtrar solicitud de visita.....	207
Figura 72. Diagrama de actividades - Aceptar visita	208
Figura 73. Diagrama de actividades - Rechazar visita	209
Figura 74. Diagrama de actividades - Alquilar bien inmueble.....	210
Figura 75. Diagrama de actividades - Generar contrato de alquiler	211
Figura 76. Diagrama de actividades - Gestionar contrato de alquiler	212
Figura 77. Diagrama de actividades - Generar solicitud de visita de alquiler.....	213
Figura 78. Diagrama de actividades - Visualizar solicitud de visitas de alquiler	214
Figura 79. Diagrama de actividades - Eliminar solicitud de visitas de alquiler	215
Figura 80. Diagrama de actividades - Visualizar contrato de alquiler	216
Figura 81. Diagrama de actividades - Aceptar contrato de alquiler	217
Figura 82. Diagrama de actividades - Gestionar servicio básico	218
Figura 83. Diagrama de actividades - Gestionar mensaje.....	219
Figura 84. Diagrama de actividades - Activar usuario	220
Figura 85. Diagrama de actividades - Desactivar usuario	221
Figura 86. Diagrama de componentes	223
Figura 87. Vista física – Diagrama de despliegue	224
Figura 88. Arquitectura de la Aplicación.....	225
Figura 89. Prueba de Usabilidad - Pregunta 1	247
Figura 90. Prueba de Usabilidad - Pregunta 2	248
Figura 91. Prueba de Usabilidad - Pregunta 3	248
Figura 92. Prueba de Usabilidad - Pregunta 4	248
Figura 93. Prueba de Usabilidad - Pregunta 5	249
Figura 94. Prueba de Usabilidad - Pregunta 6	249
Figura 95. Prueba de Usabilidad - Pregunta 7	249
Figura 96. Prueba de Usabilidad - Pregunta 8	250
Figura 97. Prueba de Usabilidad - Pregunta 9	250
Figura 98. Prueba de Usabilidad - Pregunta 10	250
Figura 99. Prueba de Usabilidad - Pregunta 11	251
Figura 100. Prueba de Usabilidad - Pregunta 12	251
Figura 101. Prueba de Usabilidad - Pregunta 13	251
Figura 102. Prueba de Usabilidad - Pregunta 14	252
Figura 103. Prueba de Usabilidad - Pregunta 15	252
Figura 104. Prueba de Usabilidad - Pregunta 16	252
Figura 105. Grado de usabilidad total	256

Índice de tablas

Tabla I. Viviendas particulares arrendadas, de tipo casa, departamento, cuarto en casa de inquilinato, y otros, según ciudad investigada – ENALQUI 2013.....	6
Tabla II. Descripción de metodologías ágiles	14
Tabla III. Evaluación de las metodologías de desarrollo ágil	16
Tabla IV. Ponderación de criterios de selección de metodología.	16
Tabla V. Comparación entre Frameworks para el desarrollo FrontEnd	19
Tabla VI. Comparativa de frameworks de desarrollo móvil multiplataforma.....	20
Tabla VII. Materiales utilizados en el trabajo de titulación.....	26
Tabla VIII. Requerimientos de la aplicación para la publicación y búsqueda de bienes inmuebles para venta y arriendo.....	28
Tabla IX. Requerimientos de la aplicación para registrar el alquiler y ventas de una inmobiliaria	30
Tabla X. Descripción general de la aplicación web para compartir piso.....	31
Tabla XI. Descripción de la aplicación para la gestión de propiedades inmobiliarias...	32
Tabla XII. Requerimientos de la aplicación kleta app	32
Tabla XIII. Descripción de la aplicación estupiso segovia.....	35
Tabla XIV. Requerimientos comunes entre los trabajos relacionados	37
Tabla XV. Requerimientos funcionales.....	42
Tabla XVI. Requerimientos no funcionales	43
Tabla XVII. Conformación del equipo de trabajo	44
Tabla XVIII. Tipo de usuario	45
Tabla XIX. Modelo de historia de usuario	46
Tabla XX. Registrar usuarios.	47
Tabla XXI. Autenticar usuarios.....	48
Tabla XXII. Gestionar cuentas de usuarios.....	48
Tabla XXIII. Gestionar perfil de usuario.....	50
Tabla XXIV. Resetear contraseña	50
Tabla XXV. Cambiar contraseña	51
Tabla XXVI. Publicar anuncio.....	52
Tabla XXVII. Gestión de bienes inmuebles.....	53
Tabla XXVIII. Registrar bienes inmuebles.....	53
Tabla XXIX. Editar bienes inmuebles.....	54

Tabla XXX. Gestión de solicitudes de reserva de alquiler recibidas.....	55
Tabla XXXI. Gestionar contratos de alquiler.....	56
Tabla XXXII. Alquilar inmueble	57
Tabla XXXIII. Visualizar información de un bien inmueble.....	59
Tabla XXXIV. Gestión de solicitudes de reserva de alquiler solicitadas	60
Tabla XXXV. Seleccionar filtros de búsqueda.....	61
Tabla XXXVI. Enviar mensajes.....	62
Tabla XXXVII. Gestionar servicios básicos.....	62
Tabla XXXVIII. Módulos e historias de usuario	64
Tabla XXXIX. Arquitectura 4+1	66
Tabla XL. Pruebas unitarias servicio web api rest – postman.....	81
Tabla XLI. Casos de prueba de caja negra - crear inmueble	83
Tabla XLII. Puntuación de cada ítem.....	91
Tabla XLIII. Identificación de ítems	91
Tabla XLIV. Respuestas obtenidas de la encuesta.....	92
Tabla XLV. Frecuencia por ítem	93
Tabla XLVI. Recursos humanos, materiales, técnicos y tecnológicos	99
Tabla XLVII. Costo total de trabajo de titulación	100
Tabla XLVIII. Costo final total de trabajo de titulación.....	100
Tabla XLIX. Descripción general de la aplicación para la publicación y búsqueda de bienes inmuebles para venta y arriendo.....	110
Tabla L. Descripción general de la aplicación para registrar el alquiler y ventas de una inmobiliaria utilizando Scrum.....	111
Tabla LI. Descripción general de la aplicación web para compartir piso	111
Tabla LII. Descripción de la aplicación para la gestión de propiedades inmobiliarias.	112
Tabla LIII. Descripción de la aplicación Kleta app	113
Tabla LIV. Descripción de la aplicación Estupiso Segovia.....	114
Tabla LV. Requerimientos de la aplicación para la publicación y búsqueda de bienes inmuebles para venta y arriendo.	114
Tabla LVI. Requerimientos de la aplicación para registrar el alquiler y ventas de una inmobiliaria.	115
Tabla LVII. Requerimientos de la aplicación para compartir piso.	116

Tabla LVIII. Requerimientos de la aplicación para la gestión de propiedades inmobiliarias.....	117
Tabla LIX. Requerimientos de la aplicación Kleta app.....	117
Tabla LX. Requerimientos de la aplicación Estupiso Segovia.....	120
Tabla LXI. Requerimientos comunes entre los trabajos relacionados.....	122
Tabla LXII. Responsable del desarrollo del proyecto	125
Tabla LXIII. Director del trabajo de titulación.....	125
Tabla LXIV. Característica de usuario arrendador.....	128
Tabla LXV. Característica de usuario arrendatario.....	128
Tabla LXVI. Característica de usuario administrador.....	128
Tabla LXVII. Requerimiento funcional 1	129
Tabla LXVIII. Requerimiento funcional 2.....	130
Tabla LXIX. Requerimiento funcional 3	130
Tabla LXX. Requerimiento funcional 4	130
Tabla LXXI. Requerimiento funcional 5.....	131
Tabla LXXII. Requerimiento funcional 6.....	131
Tabla LXXIII. Requerimiento funcional 7.....	131
Tabla LXXIV. Requerimiento funcional 8.....	132
Tabla LXXV. Requerimiento funcional 9.....	132
Tabla LXXVI. Requerimiento funcional 10	133
Tabla LXXVII. Requerimiento funcional 11	133
Tabla LXXVIII. Requerimiento funcional 12.....	133
Tabla LXXIX. Requerimiento funcional 13	134
Tabla LXXX. Requerimiento funcional 14	134
Tabla LXXXI. Requerimiento funcional 15	134
Tabla LXXXII. Requerimiento funcional 16.....	134
Tabla LXXXIII. Requerimiento funcional 17	135
Tabla LXXXIV. Requerimiento funcional 18.....	135
Tabla LXXXV. Requerimiento funcional 19.....	135
Tabla LXXXVI. Requerimiento funcional 20.....	135
Tabla LXXXVII. Requerimiento funcional 21.....	136
Tabla LXXXVIII. Requerimiento funcional 22	136
Tabla LXXXIX. Requerimiento funcional 23.....	136

Tabla XC. Diseño de la pantalla principal	146
Tabla XCI. Diseño de la pantalla gestión de inmuebles	147
Tabla XCII. Diseño de la pantalla crear inmueble.....	148
Tabla XCIII. Diseño de la pantalla actualizar inmueble.....	149
Tabla XCIV. Diseño de la pantalla inmuebles publicados	150
Tabla XCV. Diseño de la pantalla generar contrato de alquiler.....	151
Tabla XCVI. Diseño de la pantalla gestión de visitas.....	152
Tabla XCVII. Diseño de la pantalla listado de contratos de alquiler	153
Tabla XCVIII. Diseño de la pantalla solicitar visita de alquiler	154
Tabla XCIX. Diseño de la pantalla actualizar perfil de usuario.....	155
Tabla C. Petición para registrar una cuenta de usuario.....	165
Tabla CI. Petición para el inicio de sesión	166
Tabla CII. Petición para el reseteo de password	167
Tabla CIII. Petición para el envío de mensajes.....	167
Tabla CIV. Petición para filtrar bienes inmuebles	168
Tabla CV. Petición para visualizar información del inmueble	168
Tabla CVI. Petición para crear un bien inmueble.....	169
Tabla CVII. Petición para publicar un bien inmueble.....	169
Tabla CVIII. Petición para generar una solicitud de alquiler	170
Tabla CIX. Petición para aceptar una visita.....	170
Tabla CX. Petición para generar un contrato de alquiler	171
Tabla CXI. Petición para aceptar un contrato de alquiler	171
Tabla CXII. Registrar usuarios.....	174
Tabla CXIII. Autenticar usuarios	174
Tabla CXIV. Gestionar cuentas de usuarios.....	175
Tabla CXV. Gestionar perfil de usuario	176
Tabla CXVI. Resetear contraseña.....	176
Tabla CXVII. Cambiar contraseña.....	177
Tabla CXVIII. Publicar anuncio	177
Tabla CXIX. Gestión de bienes inmuebles	178
Tabla CXX. Registrar bienes inmuebles.....	178
Tabla CXXI. Editar bienes inmuebles	179
Tabla CXXII. Gestión de solicitudes de reserva de alquiler recibidas	179

Tabla CXXIII. Gestionar contratos de alquiler	180
Tabla CXXIV. Alquilar inmueble.....	181
Tabla CXXV. Visualizar información de un bien inmueble	182
Tabla CXXVI. Gestión de solicitudes de reserva de alquiler solicitadas	182
Tabla CXXVII. Seleccionar filtros de búsqueda	183
Tabla CXXVIII. Enviar mensajes.....	183
Tabla CXXIX. Gestionar servicios básicos.....	184
Tabla CXXX. Arquitectura 4+1	193
Tabla CXXXI. Diseño de la pantalla principal	228
Tabla CXXXII. Diseño de la pantalla gestión de inmuebles	229
Tabla CXXXIII. Diseño de la pantalla crear inmueble	230
Tabla CXXXIV. Diseño de la pantalla actualizar inmueble.....	231
Tabla CXXXV. Diseño de la pantalla de inmuebles publicados	232
Tabla CXXXVI. Diseño de la pantalla generar contrato de alquiler	233
Tabla CXXXVII. Diseño de la pantalla gestión de visitas.....	234
Tabla CXXXVIII. Diseño de la pantalla listado de contratos de alquiler.....	235
Tabla CXXXIX. Diseño de la pantalla solicitar visita de alquiler	236
Tabla CXL. Diseño de la pantalla editar perfil de usuario.....	237
Tabla CXLI. Diseño de pantalla principal de la aplicación móvil	238
Tabla CXLII. Diseño de la pantalla solicitar visita	239
Tabla CXLIII. Diseño de la pantalla enviar mensaje.....	240
Tabla CXLIV. Diseño de pantalla listado de solicitudes de visita.....	241
Tabla CXLV. Diseño de pantalla listado de contratos	242
Tabla CXLVI. Instrucciones para el arrendador	244
Tabla CXLVII. Instrucciones para el arrendatario	245
Tabla CXLVIII. Puntuación de cada ítem	253
Tabla CXLIX. Identificación de ítems	253
Tabla CL. Respuestas obtenidas de la encuesta.....	254
Tabla CLI. Frecuencia por ítem.....	255

1. Título

**“DESARROLLO DE UNA APLICACIÓN INFORMÁTICA
PARA LA BÚSQUEDA DE OFERTA DE ALQUILER DE
INMUEBLES EN LA CIUDAD DE LOJA”**

2. Resumen

Encontrar un bien inmueble donde vivir para satisfacer la necesidad de vivienda cuando se reside, trabaja o estudia fuera de la ciudad natal, es algo importante que requiere de tiempo y dinero. En base a ello, el objetivo del presente Trabajo de Titulación (TT) es desarrollar una aplicación informática para la búsqueda de oferta de alquiler de inmuebles para los habitantes de la ciudad de Loja, con la finalidad de responder la siguiente pregunta de investigación: ¿El desarrollo de una aplicación informática optimizará el proceso de búsqueda de oferta de alquiler de inmuebles para los habitantes de la ciudad de Loja?

El presente TT constó de tres fases, en la primera fase se realizó una revisión de trabajos similares y la aplicación de encuestas para la obtención de requisitos de la aplicación, de igual manera se identificó la tecnología Stack MEAN (MongoDB, Express, Angular, Node) para el desarrollo del web service API REST y la aplicación Web; para la parte móvil se definió el Framework Flutter con el lenguaje de programación Dart obteniendo como resultado el documento de especificación de requerimientos según el estándar IEEE 830. En la segunda fase se llevó a cabo el desarrollo y diseño de la aplicación, por lo que se utilizó la metodología XP y el modelo 4+1 respectivamente para poder organizar la aplicación, de igual manera se utilizó la arquitectura cliente servidor multicapa para separar la lógica de presentación, lógica de negocio y la lógica de datos.

La aplicación web se implementó utilizando el patrón Modelo Vista Controlador y la aplicación móvil con el patrón Bloc. Finalmente, en la tercera fase se realizó pruebas de funcionalidad en base a los criterios de aceptación de las Historias de Usuario, por otro lado se llevó a cabo las pruebas de carga y estrés obteniendo tiempos de respuesta adecuados de la petición entre el cliente y el servidor, y las pruebas de usabilidad se llevaron a cabo en un ambiente controlado donde en base a una encuesta realizada a 63 personas se logró identificar que la aplicación es fácil de usar y ayuda a reducir el tiempo de búsqueda de un bien inmueble.

Palabras clave: Desarrollo de Software, Aplicación web, búsqueda de inmueble, Metodología XP, Stack MEAN.

Abstract

Finding a place to live in to satisfy the need for housing when residing, work or study outside the hometown is something important and that requires time and money, based on this the objective of this Degree Work (TT) is to develop a computer application for the search for rental offers of real estate for the inhabitants of the city of Loja, in order to answer the following research question: Will the development of a computer application optimize the process of searching for a real estate rental offer for the inhabitants of the city of Loja?

The present TT consisted of three phases, in the first phase a review of similar works was carried out and the application of a survey to obtain application requirements, in the same way the Stack MEAN technology (MongoDB, Express, Angular, Node) was identified the present TT consisted of three phases, in the first phase a review of similar works was carried out and the application of a survey to obtain application requirements, in the same way the Stack MEAN technology (MongoDB, Express, Angular, Node) was identified for the development of the web service API REST and the Web application, for the mobile part, the Flutter Framework was defined with the Dart programming language, resulting in the requirement specification document resulting in the requirements specification document according to the IEEE 830 standard. In the second phase, the development and design of the application was carried out, so the XP methodology and the 4+1 model were used respectively to organize the application, in the same way the multilayer client server architecture was used to separate the presentation logic, business logic and data logic.

The web application was implemented using the Model View Controller pattern and the mobile application with the Bloc pattern. Finally, in the third phase, functionality tests were carried out based on the acceptance criteria of the User Stories, on the other hand the load and stress tests were carried out obtaining adequate response times of the request between the client and the server, and the usability tests were carried out in a controlled environment where based on a survey of 63 users it was possible to identify that the app is easy to use and helps reduce search time for a property.

Keywords: Software Development, Web Application, Property Search, XP Methodology, MEAN Stack.

3. Introducción

Las personas que migran a otras ciudades por motivos de estudio o trabajo, es decir, en busca de mejores oportunidades para mejorar su calidad de vida [1] se enfrentan ante la realidad de buscar una vivienda con buenas adecuaciones y acorde a su situación económica, pero muchas de las veces suelen verse afectados debido a que salen a buscar por las distintas localidades (Ver Anexo 4, pregunta número 2) incurriendo a gastos en transporte, y más importante aún, el tiempo empleado para dicho proceso de búsqueda, esto provoca que, debido a la necesidad de conseguir alojamiento obliga a estas personas a optar por alquilar viviendas con malas adecuaciones y en ubicaciones peligrosas influyendo en el desarrollo integral y en las oportunidades laborales, sociales, económicas y de educación [2].

Es así que, el Art. 30 de la Constitución Ecuatoriana aprobada en 2008 menciona lo siguiente: “Las personas tienen derecho a un hábitat seguro y saludable, y a una vivienda adecuada y digna, con independencia de su situación social y económica”; con ello el Estado protege el derecho de las personas que no tienen la capacidad económica de adquirir una vivienda para que al menos puedan tener acceso a una vivienda alquilada a un precio justo y razonable.

Asimismo, en 2013 el Instituto Nacional de Estadística y Censos “INEC” ejecutó la Encuesta Nacional de Alquileres “ENALQUI – 2013” (la última encuesta realizada hasta la fecha) adquiriendo información acerca de las viviendas arrendadas en el área urbana del Ecuador, donde indica que, la ciudad de Loja cuenta con 5.551 viviendas de las cuales 2.111 son arrendadas [3]. Por otro lado, en el año 2018 a nivel nacional, el porcentaje de la población de cinco y más años que tienen teléfono celular inteligente alcanzó el 41,4% en relación al 2012 que fue de 6.2% [4], por otra parte, existen alrededor de 16 millones de habitantes, de los cuales 8,1 millones tienen cuentas de Internet móvil, es decir, el uso de Internet móvil ha incrementado desde 2010 de un 2.4% a un 48,7% [5].

Por lo mencionado anteriormente, el presente TT propone el desarrollo de una aplicación informática a la cual se denominó “LojaHouse” con el objetivo de dar respuesta a la pregunta de investigación: **¿El desarrollo de una Aplicación Informática optimizará el proceso de búsqueda de oferta de alquiler de inmuebles para los habitantes de la ciudad de Loja?**

Las secciones que componen el presente TT son las siguientes: en la sección de Revisión de literatura se encuentran los conceptos necesarios para el entendimiento del

presente TT; en los Materiales y Métodos se detalla el contexto y el proceso que se realizó para el desarrollo del presente TT, de igual manera se muestra los recursos utilizados y los participantes que intervinieron en su desarrollo. En la sección de Resultados se detalla los hallazgos encontrados en cada uno de los tres objetivos específicos propuestos para dar solución al objetivo general planteado; en la sección de Discusión se realiza la interpretación de los resultados obtenidos desde el punto de vista del autor, finalmente se presenta las Conclusiones encontradas durante el desarrollo de cada objetivo y las Recomendaciones para trabajos futuros relacionados al presente TT.

4. Marco teórico

En esta sección se presenta la revisión bibliográfica, la misma que consta de conceptos afines con el tema de estudio de la búsqueda de oferta de alquiler de inmuebles. Además, se identifican los diferentes estudios relacionados dentro de la línea de investigación.

Actualmente existen muchas personas que no poseen vivienda propia y se ven en la necesidad de alquilar una, ya sea para residir con sus familias o tener su propia independencia; cualquiera que sea el motivo, por lo general solicitan información a conocidos, a familiares si conocen de lugares en donde están alquilando una vivienda o simplemente van a recorrer los sectores de su interés en alquilar, siendo esta una vivencia que existe no solo en la ciudad de Loja, sino en cualquier ciudad [6].

Loja se encuentra dividida en parroquias urbanas y rurales, las mismas que se subdividen en barrios, ciudadelas y comunas. Considerando solo el área urbana, se puede mencionar las siguientes seis parroquias: El Sagrario, El Valle, Punzara, San Sebastián, Sucre y Carigán [7], los mismos que constan de diferentes barrios que constituyen a cada una de las parroquias, en donde el sector inmobiliario oferta sus bienes inmuebles en alquiler, cabe recalcar que para realizar una búsqueda de un bien inmueble se vuelve complicado salir a recorrer todos los sectores de interés.

En base a la investigación realizada por la ENALQUI – 2013. En Ecuador, en el área urbana de las nueve ciudades: Quito, Guayaquil, Cuenca, Ambato, Machala, Manta, Loja, Esmeraldas y Santo Domingo, en la Tabla I se puede observar que en la ciudad de Loja, en lo que se relaciona con el tipo de viviendas particulares arrendadas: las de tipo casa cuenta con un porcentaje del 18.51%, en lo que respecta a departamento, con un 49.25%, en el tipo cuarto sobresale del resto de las ciudades investigadas con el 29.59% y finalmente de otros tipos de vivienda con tan solo el 2.64% [3].

Tabla I. Viviendas particulares arrendadas, de tipo casa, departamento, cuarto en casa de inquilinato, y otros, según ciudad investigada – ENALQUI 2013.

Ciudad	Tipo de vivienda				Total
	Casa (%)	Departamento (%)	Cuarto (%)	Otros* (%)	
Cuenca	26,45%	53,87%	17,30%	2,38%	100,00%
Machala	31,39%	53,80%	11,89%	2,91%	100,00%
Esmeraldas	31,32%	41,13%	14,54%	13,00%	100,00%

Guayaquil	28,98%	63,46%	5,85%	1,73%	100,00%
Loja	18,51%	49,25%	29,59%	2,64%	100,00%
Manta	35,91%	56,82%	1,84%	5,42%	100,00%
Quito	11,28%	68,86%	16,92%	2,95%	100,00%

4.1. Elementos que intervienen en la búsqueda de alquiler de inmuebles.

Existen varios elementos que se identifican al momento de buscar un bien inmueble en alquiler, entre ellos están:

4.1.1. Bien inmueble

Se consideran inmuebles todos aquellos bienes considerados bienes raíces, por tener de común la circunstancia de estar íntimamente ligados al suelo, unidos de modo inseparable, física o jurídicamente, al terreno [7]. Por lo tanto, un bien inmueble no puede ser desplazado (haciendas, viviendas, terrenos), a más que por fuerzas naturales debido a que se encuentran en una posición geográfica fija, los mismos que pueden ser empleados por el hombre con fines lucrativos, en este caso para la actividad de arrendamiento del bien inmueble [8].

4.1.2. Arrendador

El arrendador es el propietario del bien inmueble, es la persona que cede el uso de ese bien a cambio de una remuneración económica, llamada renta, misma que puede ser pagada mes a mes o en un solo pago. Por lo tanto el arrendador es quién oferta su bien inmueble en alquiler, el mismo que debe cumplir con algunas condiciones estipuladas en la Ley de Inquilinato [9].

4.1.3. Arrendatario

El arrendatario o inquilino, es aquella persona que hace uso de cierto bien para su disfrute a cambio de pago de un canon. Es la persona que toma en arriendo alguna cosa. Cabe recalcar que el arrendatario está obligado con su arrendador a mantener en buen estado el bien inmueble, en caso de incumplir, se deberá acatar lo que determina la Ley de Inquilinato en su art. 7, que dice: "Si el inquilino fuere responsable de los daños ocasionados en el local arrendado, o en las instalaciones de agua potable, luz eléctrica y servicios higiénicos, estará obligado a la inmediata reparación, a su costa", en caso de incumplimiento en el tiempo fijado por el juez, el arrendador se hará cargo y deberá cobrar al arrendatario por los gastos incurridos con un aumento del 10% [7], [8], [10].

4.1.4. Arriendo o Arrendamiento

El arrendamiento es un contrato mediante el cual, una parte, llamado arrendador, se obliga a proporcionar de modo temporal el uso o el goce de una cosa a otra parte, llamada arrendatario, por un tiempo determinado quién a su vez debe pagar por ese uso o goce, un precio determinado. Entonces, el arriendo o arrendamiento, es la celebración de un contrato, mediante el cual las dos partes (arrendador y arrendatario) se ven obligados a entregar dicho bien inmueble y a pagar un precio (renta) respectivamente, bajo la normativa de la Ley de Inquilinato [9].

Un contrato de arrendamiento de forma verbal puede ser legalmente obligatorio, pero es mejor tener un contrato por escrito, ya que permite determinar los derechos y obligaciones entre las partes, del mismo modo, quedará establecido quién es el responsable de pagar por el bien inmueble arrendado, los servicios básicos (en caso de no estar incluidos en el contrato) y el cuidado del mismo [11].

4.1.5. Canon de arrendamiento

Es el precio que debe pagar el arrendatario por el uso del bien inmueble alquilado, el mismo que haya sido acordado entre las partes dentro del contrato de arrendamiento.

4.2. Condiciones de los locales de arrendamiento.

Dentro de la ley del inquilinato existen condiciones que deben cumplir los bienes inmuebles para poder ofrecer un lugar apto para una persona, es por ello que deben regirse a las condiciones que fijen las ordenanzas municipales [11]:

- a) Disponer de servicios higiénicos completos y permanentes, al menos uno para cada piso de la casa, dependiendo las características del lugar.
- b) Si en un mismo piso existieran 2 o más departamentos independientes, cada uno deberá tener siquiera un servicio higiénico.
- c) Tener buena ventilación y luz en las habitaciones
- d) Disponer de los servicios de agua potable y de luz eléctrica, de forma permanente.
- e) No ofrecer peligro de ruina.
- f) Que esté acreditado con un Certificado de Sanidad.

4.3. Motivos por los que se busca un bien inmueble.

Dentro del proceso de búsqueda de alquiler de inmuebles, existen varios motivos que generan esta necesidad, las mismas que se mencionan a continuación [12]:

- Realizar estudios en otra ciudad.

- Divorcios.
- Trabajar en otra ciudad.
- Personas sin vivienda.
- Parejas (casadas, unidas).
- Solteros/as.
- Personas con escasos recursos económicos.

En la lista anterior se mencionan varios motivos por los cuales se incurre a la búsqueda de alquiler de un bien inmueble, debido a que dichas personas no se encuentran en condiciones de optar por comprar una vivienda. En [2] menciona: “los costos de transacción en los que se incurre al obtener una vivienda: costos de búsqueda, cotización, movimiento, tarifas de transacciones legales, impuestos, seguros, entre otros”, son factores que influyen al momento de querer adquirir una vivienda propia.

4.4. Consecuencias de no encontrar un bien inmueble con buenas adecuaciones.

Al existir inmuebles en estado precario, éstos promueven a que surjan consecuencias que afectan directamente a las personas que deciden residir en un espacio en bajas condiciones a causa de sus escasas condiciones económicas, como lo menciona el Art. 30 de la Constitución Ecuatoriana [13]: Se considera digna a una vivienda que provea los elementos necesarios para satisfacer las necesidades básicas del ser humano, asimismo sus necesidades de convivencia. Entre las consecuencias que se pueden destacar, están [2]:

- Bajo rendimiento en estudiantes.
- Afectación en la salud de las personas.
- Riesgo en la seguridad de las personas.
- Ubicaciones alejadas de la zona céntrica de la ciudad.
- Servicios básicos deficientes.
- Gasto excesivo empleado en transporte.
- Desempeño laboral.
- Calidad de vida.

Existen criterios que consideran a una vivienda digna o adecuada, los cuales son: la seguridad, la disponibilidad de servicios y la habitabilidad [2].

4.5. Comparativa de metodologías Ágiles de Desarrollo de Software.

Para el desarrollo del presente Trabajo de Titulación es importante definir una metodología ágil de desarrollo de software, con el fin de poder llevar a cabo un proceso organizado que facilite el desarrollo del mismo. Dicho esto, se plantea a continuación una comparativa de las metodologías ágiles de desarrollo de software encontradas en los diferentes trabajos relacionados, mismos que se analizaron y que se detallan más adelante, para así poder determinar la que más se adecúe al presente Trabajo de Titulación.

4.5.1. Metodologías de Desarrollo Ágil.

Son metodologías adaptativas, que permiten llevar a cabo, proyectos de desarrollo de software, adaptándolos a los distintos cambios que permitan mejorar el sistema y dar una mejor satisfacción al cliente, de esta manera, el proyecto puede ir adaptándose a lo largo del tiempo. Los problemas que se pretende solucionar con las metodologías ágiles son el de retrasar decisiones y la planificación adaptativa; permitiendo avanzar con el desarrollo de software a gran escala [14], [15].

4.5.2. Metodología Extreme Programming (XP)

Es una metodología ágil que permite tener revisiones continuas con el cliente, permitiendo de esta manera que el equipo de desarrollo pueda implementar la solución al problema planteado en base a la realimentación que fomenta esta metodología. Asimismo, tiene la cualidad de que su documentación se enfoca fuertemente en las Historias de Usuario y las tarjetas CRC y promueve los diseños simples para que no tengan mayor complejidad en su implementación [16].

Fases de la metodología XP

La metodología XP consta de 4 fases que son [17]:

- **Planificación.-** En esta fase es donde se definen las necesidades del cliente expresadas en Historias de Usuario, las mismas que tienen un propósito similar a los casos de uso, con la diferencia de que se redactan entre tres a cuatro líneas, escritas en un lenguaje no técnico. Esto permite que se pueda estimar el tiempo de desarrollo de la Aplicación Informática en base a lo que el usuario requiera. Como entregable de esta fase, se obtiene un documento llamado Historias de Usuario, que dependiendo de la complejidad de las mismas se

destina un tiempo para su desarrollo, por lo general van desde una a tres semanas.

- **Diseño.-** XP promueve que los diseños sean lo más simples y claros para que sea fácil de entender e implementar, permitiendo así, que tome menos tiempo y esfuerzo desarrollar.
- **Desarrollo.-** la participación del cliente es fundamental al momento de llevar a cabo la codificación de las historias de usuario ya que él es quién debe validar si cumple con la funcionalidad especificada.
- **Pruebas.-** Las pruebas unitarias se realizan antes de escribir el código y se ejecutan de manera constante frente a cada modificación realizada con la finalidad de que el software esté lo más libre de errores.

4.5.3. Metodología ICONIX

Se define como un proceso de desarrollo de software que unifica un conjunto de métodos de orientación a objetos con el objetivo de abarcar todo el ciclo de vida de un proyecto. Además, está adaptado a los patrones y ofrece el soporte de UML, dirigido por casos de uso y es un proceso iterativo e incremental, esta metodología considera el uso de UML para el modelado del sistema y la generación de la documentación, por ejemplo diagramas de caso de uso, modelo de dominio, diagrama de Robustez, diagrama de Actividades, diagramas de Secuencia, diagramas de Componentes, diagramas de Despliegue y diagramas de Implementación [18], [19].

Fases de la metodología ICONIX

La metodología ICONIX consta de 4 fases que son [20]:

- **Análisis de Requisitos.** – En esta fase se podrán identificar todos los requisitos necesarios para el sistema, obteniendo así un documento que lista todos los requisitos que el cliente necesita para el sistema a desarrollar. Asimismo, se realiza el modelado inicial del dominio del sistema conjuntamente con los casos de uso mediante el uso de la herramienta UML, obteniendo también los entregables respectivos (especificación de casos de uso, diagramas de secuencia, diagramas de dominio, diagramas de paquetes).
- **Análisis y Diseño Preliminar.** - En esta fase, a partir de los casos de uso se realiza la especificación de los mismos, donde no se hace énfasis en especificar a profundidad, sino que sea breve y que cada interacción sea descrita con sus precondiciones y pos condiciones; flujos normales y alternos. También se

realizan los diagramas de Robustez, los mismos que permiten visualizar mejor la interacción entre objetos y ayuda a entender mejor la lectura del sistema.

- **Diseño Detallado.** - En esta fase se hace énfasis en los diagramas de secuencia, donde se conoce todos los elementos que forman parte del sistema, éstos se modelan a partir de la especificación de casos de uso. Asimismo, se lleva a cabo el diseño del diagrama de clases y se define el modelo de datos que se implementará en el sistema. Se debe terminar el modelo estático, añadiendo los detalles del diseño en el diagrama de clases y verificar si el diseño satisface todos los requisitos identificados.
- **Implementación.** - En esta fase a partir del diseño alcanzado en la fase anterior se creará el software, donde se debe utilizar el diagrama de componentes como un apoyo para mostrar la distribución física de los elementos que componen la estructura interna del sistema. Así como escribir y generar el código.

4.5.4. Metodología Mobile-D

Esta metodología está diseñada para un ciclo de desarrollo rápido y para equipos pequeños (menos de diez programadores), el objetivo es que el ciclo de desarrollo propuesto se finalice en menos de 10 semanas con una aplicación completamente funcional [21].

Fases de la metodología Mobile-D

La metodología Mobile-D consta de 4 fases que son [22]:

- **Fase Exploración.** - En esta fase se realiza la planificación y los conceptos básicos del proyecto. Se define el alcance del proyecto y su establecimiento con las funcionalidades donde se va a llegar. Se debe prestar especial atención a la participación de los clientes.
- **Fase de Inicialización.** - En esta fase se configura el proyecto; se preparan y verifican todos los recursos físicos, tecnológicos y de comunicación necesarios para el proyecto. Esta es la fase en la que asegura el éxito o el fracaso de las demás fases del proyecto.
- **Fase de Producción.** - En esta fase se realiza la implementación de las funcionalidades necesarias de todos los módulos de manera iterativa, además, se realiza las pruebas del sistema para verificar el correcto funcionamiento del desarrollo.

- **Fase de Estabilización.** - En esta fase se integran todos los módulos donde se verifica el funcionamiento del sistema completo, en esta fase se asegura el éxito y la calidad de la implementación del proyecto. Adicionalmente, en esta fase se puede incluir la producción de documentación.
- **Fase de pruebas.** – La fase de pruebas, se realizan diferentes pruebas del sistema; se hace correcciones y reparación de posibles errores.

4.5.5. Metodología Desarrollo Rápido de Aplicaciones (RAD)

Esta metodología permite llevar a cabo el desarrollo iterativo e incremental, se apoya en la construcción de prototipos para modelar la aproximación de la solución real utilizando herramientas CASE para el diseño y construcción del software [17].

Fases de la metodología RAD

La metodología RAD consta de 4 fases que son [17]:

- **Fase de Planificación.** - en esta fase se reúnen el equipo encargado de la definición de los requisitos y los usuarios finales, de igual manera se define el alcance del proyecto mediante la definición de una lista de requisitos del sistema a implementar. Tiene una duración de entre 1 a 4 semanas. Al finalizar la fase se debería poder tener una base sólida para realizar una estimación del proyecto a desarrollar considerando los puntos de función.
- **Fase de Diseño.** - con la ayuda del usuario, los analistas detallan mejor los requisitos, luego diseñan el modelo de datos y se crean prototipos para poder ver los avances en cuanto a la funcionalidad que esperan los usuarios.
- **Fase de Construcción.** - en esta fase ya se procede a la implementación del software, donde se realizan de manera iterativa la codificación, pruebas y se hace un refinamiento de los requisitos.
- **Fase de Implantación.** - a esta fase la denominan también fase de despliegue, es decir, el sistema ya entra en producción en un entorno real. También se incluye la transición al nuevo sistema, y la capacitación de los usuarios.

4.5.6. SCRUM

Scrum es un marco de trabajo en equipo donde se lleva a cabo mediante iteraciones o Sprints (duración de 1 a 4 semanas). Esta metodología permite planificar y controlar proyectos en el que existe incertidumbre y se revisa cada entregable porque se presenten cambios en los requisitos del cliente. En función de los objetivos alcanzados en el Sprint, se priorizan las actividades para siguiente de manera iterativa [17].

Fases SCRUM

SCRUM consta de 5 fases que son [14]:

Reunión para la planificación del Sprint. - En esta etapa se reúne todo el equipo donde planifican qué se va a entregar en el próximo sprint y cómo se va a realizar el trabajo propuesto, es decir, queda definido lo que se va a realizar en ese sprint.

- **Scrum diario.** - se realiza una reunión de 15 minutos con el equipo, para así, informar al resto del equipo los avances de las actividades diarias del proyecto, para de esta manera poder obtener datos del progreso del sprint diario. Esta etapa es importante porque permite identificar posibles problemas que obstaculicen la continuidad del proyecto, permitiendo tomar decisiones de forma rápida.

- **Trabajo de desarrollo durante el Sprint.** - en este caso cuando el sprint está en curso se debe asegurar que no se realicen cambios y que los objetivos se cumplan, para ello, se debe mantener una retroalimentación constante con el cliente.

- **Revisión del Sprint.** - se da lugar al final de cada sprint y consiste en una reunión con el cliente y el equipo Scrum para poder revisar y estudiar el producto obtenido al finalizar el sprint. Aquí se definen los cambios a realizar, en caso de ser necesario, es importante para poder continuar con el siguiente sprint.

- **Retrospectiva del proyecto.** - en esta etapa el equipo Scrum tiene la oportunidad de poder evaluar y mejorar el proceso de cómo está realizando su trabajo, ya sea con las herramientas de trabajo como las personas que forman parte del equipo. Todo esto con el fin de mejorar el desempeño en el siguiente sprint.

4.6. Tabla comparativa de las Metodologías de Desarrollo Ágil.

En base a las metodologías antes descritas, se puede ver en la Tabla II la comparativa de las mismas.

Tabla II. Descripción de metodologías ágiles

Metodología	Tipo	Fases	Comunicación	Tipo de proyecto	Documentación
XP	Ágil	Planificación, Diseño, Desarrollo, Pruebas	Se basa en la retroalimentación continua entre el cliente y el equipo de desarrollo.	Aplicaciones web y móvil.	Lo principal son las historias de usuario y tarjetas CRC.
Mobile-D	Ágil	Exploración, Inicialización, Producción,	Trabaja con el cliente	Aplicaciones móviles.	Casos de Uso, Diagramas de Secuencia,

		Estabilización, Pruebas	a medida que avanza el proyecto		Diagramas Entidad Relación.
ICONIX	Pesada-Ágil	Análisis de requerimientos, Diseño preliminar, Diseño detallado, Implementación	Participación activa del cliente en el proyecto.	Aplicaciones web y móvil.	Diagramas de Casos de Uso, Descripción de Casos de Uso, Diagramas de Secuencia, Diagramas de Robustez
RAD	Ágil	Planificación, Diseño, Construcción, Implantación	Participación activa del cliente en el proyecto.	Aplicaciones web y móvil.	Uso de herramientas CASE para el modelado, prototipado, y generación de código automático y la reutilización de componentes de software desarrollados anteriormente
SCRUM	Ágil	Reunión para la planificación del Sprint, Scrum diario, Trabajo de desarrollo durante el Sprint, Revisión del Sprint, Retrospectiva del proyecto.	Participación del cliente en el proyecto.	Aplicaciones web y móvil.	No especificado.

Evaluación para la selección de la Metodología de Desarrollo

Se consideró el esquema de evaluación para la selección de la metodología en base a [23] en su tesis de grado, el cual propone los siguientes criterios en función de los conocimientos que el equipo tenga acerca de las metodologías. Estos criterios son:

- Grado de conocimiento
- Soporte orientado a objetos
- Adaptable a cambios
- Basado en casos de uso
- Posee documentación adecuada

- Facilita la integración entre las etapas de desarrollo
- Relación con UML
- Permite desarrollo software sobre cualquier tecnología

En función de los conocimientos que el equipo tenga, se establecen los porcentajes para cada criterio, los porcentajes propuestos son los siguientes:

- 20% para el Grado de conocimiento.
- 15% para Adaptable a cambios y Posee documentación adecuada.
- 10% para el resto de criterios.

Tabla III. Evaluación de las metodologías de desarrollo ágil

Criterios	%	XP	Mobile-D	ICONIX	RAD	SCRUM
Grado de conocimiento	20	15	10	10	10	10
Soporte orientado a objetos	10	10	10	10	10	-
Adaptable a cambios	15	15	15	10	10	10
Basado en casos de uso	10	5	10	10	10	-
Posee documentación adecuada	15	15	15	15	15	15
Facilita la integración entre las etapas de desarrollo	10	10	10	10	10	10
Relación con UML	10	8	8	10	8	-
Permite desarrollo software sobre cualquier tecnología	10	10	8	10	10	10
TOTAL:	100	88	86	85	83	55

En la Tabla IV se muestra la ponderación para los valores de los porcentajes asignados a cada criterio de la Tabla III.

Tabla IV. Ponderación de criterios de selección de metodología.

Criterios	Regular	Normal	Bueno	Alto
Grado de conocimiento	5	10	15	20
Soporte orientado a objetos	2	5	8	10
Adaptable a cambios	2	5	10	15
Basado en casos de uso	2	5	8	10
Posee documentación adecuada	2	5	10	15
Facilita la integración entre las etapas de desarrollo	2	5	8	10
Relación con UML	2	5	8	10
Permite desarrollo software sobre cualquier tecnología	2	5	8	10

En base a la evaluación realizada en la Tabla III se decidió utilizar la metodología XP debido a su alta puntuación, con 88% en relación al resto de metodologías, considerando la naturaleza del proyecto y el grado de conocimiento que se tiene respecto a esta metodología lo cual influyó bastante en su selección. Asimismo [24] manifiesta que las metodologías ágiles se enfocan en que el desarrollo de software funcione correctamente antes de conseguir una buena documentación, ya que estos tipos de documentos deben ser cortos y centrarse en lo fundamental.

4.7. Herramientas de desarrollo

Para el desarrollo de la Aplicación Informática en el entorno web del presente Trabajo de Titulación se utilizó las herramientas que se describen en el Stack MEAN, utilizando el lenguaje de programación JavaScript tanto del lado del cliente, como del lado del servidor [25]. Esto es una gran ventaja ya que solo se trabaja con un lenguaje de programación, otros enfoques, por ejemplo los basados en Python (Django), Ruby (Rails), Java (Spring, JEE), PHP (CodeIgniter, Laravel) obliga a los programadores a cambiar de contexto, no solo entre los lenguajes de programación, sino también entre los diferentes frameworks. JavaScript permite trabajar tanto en el backend como en el frontend lo cual da lugar a diferentes stacks: MEAN (MongoDB, Express, Angular, Node), MERN (MongoDB, Express, React, Node), MEVN (MongoDB, Express, Vue, Node), entre los más populares, donde la diferencia radica en la tecnología del frontend: Angular, React, Vue respectivamente [26].

MongoDB desempeña un papel importante para la gestión de la base de datos, mientras que Node.js y Express.js se utilizan para construir el API REST en el backend y Angular se utiliza en el frontend [27]. Para el desarrollo de la aplicación móvil se utilizó el framework Flutter con el lenguaje de programación Dart.

Los pilares de Stack MEAN:

- **MongoDB:** es una base de datos NoSQL basada en documentos, donde los datos se almacenan como objetos JSON. Cuando existe demasiado tráfico en la aplicación debido a la gran complejidad en el desarrollo y uso de la web, la base de datos relacional no puede hacer frente a la escala y la agilidad de las aplicaciones modernas, por ende la base de datos NoSQL tiene un mejor rendimiento y es más escalable [25], [28]. En esta tesis, se utilizó la base de datos NoSQL, que es MongoDB para el desarrollo rápido, la integración del código y para un menor tiempo de administración de la base de datos ya que

permite manejar un esquema de datos flexible a diferencia de SQL que se debe tener un esquema rígido y bien definido [29].

- **Express.js:** es un Framework para Node.js ya que le provee de componentes para el manejo de peticiones web para facilitar el desarrollo o implementación de un servidor web de manera sencilla, también permite que las aplicaciones sean extensibles, es decir, que se puede escalar la aplicación en caso que se desee agregar nuevas funcionalidades, además es posible utilizar código que ya ha sido creado por otros programadores, pudiendo así reutilizar código [25].
- **Angular.js:** es un Framework del lado del cliente y utiliza el patrón de desarrollo Modelo Vista Controlador (MVC), orientado a componentes, y provee toda la funcionalidad necesaria para manejar las entradas del usuario en el navegador. Permite el desarrollo de aplicaciones web de una sola página (SPA), es decir, la aplicación cuenta con una sola página obteniendo con esto un mejor rendimiento y la experiencia del usuario mejora ya que reduce la latencia y tiempos de espera en la navegación, por lo que las aplicaciones son más rápidas y más fluidas [28].
- **Node.js:** es un entorno de ejecución de JavaScript que permite la creación de servidores que soporten una alta concurrencia de usuarios, además, se puede crear aplicaciones en tiempo real [28]. La arquitectura de Node.js está basada en eventos, lo cual permite que el código se ejecute de forma asíncrona. Las tareas más comunes de los servidores incluyen responder consultas, almacenar datos en una base de datos, leer archivos del disco duro y establecer conexiones con otros componentes de la red. Todas estas acciones se agrupan bajo la categoría de E/S. En lenguajes como Java y C, la E/S se ejecuta de forma síncrona, es decir, las tareas se realizan de una en una, a diferencia de Node.js con la E/S asíncrona, lo cual permite delegar el trabajo al sistema operativo o a la base de datos. Esto hace posible que las tareas se realicen en paralelo, obteniendo una gran ventaja en cuanto a la velocidad de las aplicaciones que se desarrollan utilizando Node.js y JavaScript [29].

Stack MEVN:

Es otra combinación de pila de software que utiliza las mismas tecnologías que MEAN con la única diferencia que usa Vue.js en el lado del cliente. Vue.js es un Framework progresivo basado en componentes que se utiliza para crear interfaces de usuario dinámicas y de una sola página, la curva de aprendizaje de Vue.js es más sencilla que

React.js y Angular.js, además admite el enlace de datos bidireccional al mantener sincronizados el Modelo y la vista como en el patrón MVC [26].

Stack MERN:

En este caso se emplea React.js en el lado del cliente. React.js es una librería de JavaScript para crear interfaces de usuario. Está diseñado para mejorar el desarrollo de la interfaz de usuario interactiva al facilitar la actualización de la vista cuando cambian los datos. Se realiza dividiendo la vista en componentes más pequeños, que se pueden componer para crear interfaces de usuario complejas. Los componentes están integrados en JavaScript en lugar de plantillas, lo que permite un fácil flujo de datos [27]. La curva de aprendizaje depende de los conocimientos de JavaScript por parte de los desarrolladores ya que es el lenguaje de programación principal para este marco, al entender cómo funciona este Framework la curva de aprendizaje va disminuyendo [30].

Comparativa de Frameworks para el desarrollo en el FrontEnd.

En base a el análisis realizado en [26], [27], [30] se ha destacado los criterios de evaluación en la Tabla V, de los siguientes Frameworks: Angular, React y Vue, donde se evalúan los siguientes aspectos según el criterio y conocimiento del autor:

1. Basado en componentes.
2. Implementa la arquitectura Modelo Vista Controlador.
3. Curva de aprendizaje.
4. Lenguaje de programación.
5. Framework.
6. Estructura del proyecto.

Tabla V. Comparación entre Frameworks para el desarrollo FrontEnd

Criterio	Framework		
	Angular	React	Vue
Basado en componentes.	Si	Si	Si
Implementa la arquitectura Modelo Vista Controlador (MVC)	Si	No	No
Curva de aprendizaje	Alta	Media	Baja

Lenguaje de programación	TypeScript	JSX	JavaScript / TypeScript
Framework	Si	No	Si
Estructura del proyecto	Muy buena	Buena	Buena

En base a la información de la Tabla V se observa que tanto Angular como React y Vue se basan en componentes, pero únicamente Angular implementa la arquitectura MVC de forma nativa, React y Vue son más flexibles. Aprender React y Vue conlleva menor tiempo, a diferencia de Angular. Al ser un framework, Angular y Vue se puede utilizar el lenguaje de programación Typescript, en Vue no es obligatorio ya que puede usar JavaScript, en cambio React, que se cataloga como una librería, utiliza el lenguaje de programación JSX que es una combinación entre JavaScript y HTML. Angular permite tener una estructura del proyecto más legible, en React y Vue se puede llevar el control de cómo estructurar el proyecto.

Una vez realizado un análisis de los diferentes frameworks, el que se empleó dentro del Stack de desarrollo para el presente Trabajo de Titulación fue Angular, denominando así el Stack MEAN (MongoDB, Express, Angular, Node), debido a que Angular es considerada una plataforma de desarrollo, el cual está basado en componentes y utiliza el lenguaje de programación TypeScript, además brinda una estructura inicial del proyecto, empleando la arquitectura Modelo Vista Controlador. Aunque su curva de aprendizaje es alta, a diferencia de React y Vue, con Angular se puede tener un proyecto más robusto y tiene desde su core muchas librerías, entre ellas el routing, gestión de dependencias, la CLI de Angular que facilita la creación del proyecto de Angular.

Programación móvil con Flutter

En la Tabla VI se presenta una comparativa bajo criterio del autor entre los distintos frameworks para el desarrollo de aplicaciones móviles multiplataforma [31].

Tabla VI. Comparativa de frameworks de desarrollo móvil multiplataforma

	React Native	Ionic	Flutter
Tipo	Framework	Framework	Framework
Tipo de aplicaciones	Móviles	Móviles	Móviles

Lenguajes de programación	JavaScript/JSX	HTML, CSS, JavaScript	Dart
Curva de aprendizaje	Alta	Media	Baja
Aplicación compilada	No	No	Si
Rendimiento	Medio	Medio	Alto
Basado en componentes	Si	Si	No

En la Tabla VI se puede observar que el Framework Flutter permite el desarrollo de aplicaciones multiplataforma con un alto rendimiento debido a que se genera una aplicación compilada, es decir que el código escrito en Dart se compila a lenguaje nativo, lo que no sucede con Ionic que se ejecuta en un webview que es un navegador interno que se ejecuta dentro de la aplicación; por otro lado, el código de React Native se ejecuta en un motor de JavaScript (JavaScriptCore), el mismo que luego se comunica con un puente nativo que está escrito en C++ para IOS y Java para Android y éste a su vez se comunica con la aplicación, todo ese proceso es bidireccional. Flutter tiene una curva de aprendizaje más corta, a diferencia de React Native e Ionic, Flutter utiliza widgets, los cuales permiten que la interfaz de usuario se vea igual en cualquier dispositivo independiente del sistema operativo, por lo que se concluye que el Framework más idóneo para llevar a cabo el desarrollo de la aplicación móvil es Flutter.

4.8. Trabajos relacionados

En esta sección se resumen 7 trabajos que abordan el objeto de estudio desde otros autores en cuanto a la búsqueda de alquiler de inmuebles, para ello se presentan las síntesis de los trabajos relacionados desde el año 2013 al 2019.

Según [32] en su trabajo titulado “Diseño e implementación de una aplicación orientada a dispositivos móviles con sistema operativo Android para la publicación y búsqueda de bienes inmuebles para venta y arriendo”, desarrolló una aplicación enfocada a dispositivos móviles con el objetivo de publicar y buscar bienes inmuebles que se encuentren en venta o arriendo en la ciudad de Quito. Para el desarrollo del proyecto utilizó la metodología RAD con el fin de llevar a cabo un desarrollo rápido. Por lo que concluye que es factible el desarrollo de una aplicación para que los usuarios puedan visualizar, publicar, ubicar y encontrar información acerca de bienes raíces en venta o arriendo.

Frente a eso [33] en su tesis titulada “EstuPiso Segovia: buscador de viviendas en alquiler en Segovia para estudiantes.” presenta el desarrollo de una aplicación web con

el objetivo de permitir encontrar una vivienda en alquiler para los estudiantes que deciden realizar sus estudios en la ciudad de Segovia, permitiendo que dichos estudiantes puedan encontrar a compañeros que estén estudiando en la ciudad antes mencionada, para el desarrollo de esta aplicación web se optó por utilizar las bases de las metodologías ágiles sin especificar una en concreto debido al gran peso que tiene la documentación del proyecto.

Asimismo [34] en su tesis titulada “Desarrollo de una aplicación web y una aplicación móvil para registrar el alquiler y ventas de una inmobiliaria utilizando scrum”, trata del desarrollo de una aplicación web donde se utilizó el Framework Django y el lenguaje de programación Python, con el fin de automatizar el proceso de alquiler o venta de inmuebles, también el registro de inmuebles, clientes y empleados de una empresa de inmobiliaria, para lo cual se lo realizó basándose en la metodología SCRUM para su desarrollo. También hace uso de PostgreSQL para el diseño de la base de datos. No obstante, el estudio concluyó que la aplicación móvil permitió satisfacer las necesidades tanto para el usuario como para el empleado, teniendo un mejor control y organización en las actividades realizadas diariamente.

De igual manera [35] en su tesis titulada “Desarrollo de una aplicación web y una aplicación móvil para la gestión de alquiler y venta en una inmobiliaria”, donde desarrolla una aplicación web y móvil para una inmobiliaria que no dispone de un proceso automatizado, se lo realizó bajo la guía de la metodología XP (Xtreme Programming) para llevar a cabo el proceso de desarrollo de dicho proyecto. Para la parte web utiliza el Framework Django y el lenguaje de programación Python y para la parte móvil se desarrolla utilizando en SDK de Android Studio. También hace uso de PostgreSQL para el diseño de la base de datos. Como conclusión se tiene que aumentaron las ventas y arrendamientos de los inmuebles, extendiendo los ingresos de la inmobiliaria.

Así pues, [36] con tesis titulada: “Diseño y desarrollo de una aplicación web para compartir piso”, donde se desarrolló una aplicación web para compartir piso en España, al mismo tiempo de mitigar el problema de acceso a vivienda debido a los precios elevados que implica y que por lo general se encuentran en alquiler casas. Para ello la aplicación web se desarrolló utilizando el lenguaje de programación PHP y MySQL para la gestión de la base de datos, además se utilizó una arquitectura cliente-servidor.

En cuanto a la metodología utilizada, no se basó sino en su experiencia, sin considerar una metodología estandarizada. No obstante, concluyó que al ser un proyecto extenso en el cuál influyen varios factores, se desarrolló las funcionalidades básicas y que se pueden ampliar en trabajos futuros.

En [37] con tesis titulada “Desarrollo e implementación de un aplicativo para dispositivos móviles con sistema operativo Android y geolocalización que permita ofrecer y solicitar servicios de alquiler de casas con ubicación exacta y a la vez sugiera las más cercanas a la ubicación actual en cualquier parte de Guayaquil”, con el objetivo de permitir encontrar viviendas de alquiler publicadas con geolocalización, asimismo, que establezca la rutas más corta hacia la vivienda se desarrolló este proyecto que se enmarcó en la metodología Scrum-Bam, una metodología de desarrollo rápido y flexible; de igual manera, utilizó el IDE Android Studio para el desarrollo de la aplicación móvil y para la parte del administrador web, se desarrolló utilizando el lenguaje de programación PHP con una base de datos SQL Server sobre un Web Server IIS. No obstante, concluyó mediante encuesta realizada en el proyecto de titulación que usuarios con cuenta Gmail y Smartphone con plan de datos buscan vivienda con los siguientes criterios: sector, número de cuartos, número de baños, rango de precios, patio y garaje.

Además, [38] con título denominado “Implementación de un sistema web y una aplicación móvil para administrar los servicios y actividades de una empresa inmobiliaria”, con el objetivo de dar solución a la administración de actividades diarias; a la difusión, alquiler y venta de los bienes inmuebles; y a la organización de los datos de las oficinas, empleados, inquilinos, ventas, alquileres, anuncios, visitas e inspecciones que se realizan y son parte de la empresa inmobiliaria, se realizó la construcción de un sistema web sobre la plataforma de Python y su framework Django, con el fin de dar soporte a las actividades y al manejo de información de la empresa por parte de sus colaboradores con sus respectivos permisos de acceso. El proyecto se elaboró a través de la metodología de desarrollo de sistemas de Jonás Montilva denominada MEDSI, la cual permite documentar el progreso y los resultados de la propuesta tecnológica. No obstante, se concluyó que la creación de soluciones tecnológicas es de vital importancia dentro de la estructura de cualquier empresa, organización o negocio contemporáneo que se encuentra en la necesidad de crear lazos de comunicación y productividad dentro del entorno web.

5. Metodología

El tipo de investigación es aplicada ya que permite diseñar soluciones informáticas para solventar parte de los problemas de la sociedad, buscando optimizar la búsqueda de alquiler de inmuebles en la ciudad de Loja a través de una aplicación informática.

El proceso para culminar el desarrollo del presente TT se detalla en esta sección, para ello se define el contexto donde se desarrolló, el proceso realizado para el cumplimiento de los objetivos planteados, los recursos utilizados, y finalmente, los participantes.

5.1. Contexto

El presente TT fue desarrollado en la carrera de Ingeniería en Sistemas dentro de la Facultad de Energía, las Industrias y los Recursos Naturales no Renovables de la Universidad Nacional de Loja; el mismo que se enfocó para beneficio de los habitantes de la ciudad de Loja, principalmente para quienes buscan alquilar un bien inmueble, con la finalidad de cumplir con el objetivo general planteado, el cual es: “Desarrollar una Aplicación Informática para la búsqueda de ofertas de alquiler de inmuebles para los habitantes de la ciudad de Loja”, donde se plantearon 3 objetivos específicos con sus actividades respectivas, mismos que se detallan a continuación.

5.2. Proceso

El proceso que permitió lograr el cumplimiento del objetivo general del presente TT, es el siguiente:

Fase 1. Determinar los requisitos necesarios para desarrollar la Aplicación Informática.

- Se elaboró la revisión de trabajos similares al caso de estudio para obtener requisitos generales, ver Anexo 1: **Obtención de requisitos generales mediante la revisión de trabajos similares.**
- Se realizó el levantamiento de requisitos (funcionales y no funcionales), por medio de la técnica de la encuesta, los cuales fueron necesarios para establecer la frontera del sistema y la arquitectura de la misma.
- Se elaboró el documento de especificación de requisitos utilizando el estándar IEEE830.

Fase 2. Diseñar e implementar la Aplicación Informática para la búsqueda de alquiler de inmuebles en base a la metodología XP.

- Se desarrolló la fase de planificación de la Metodología XP mediante la elaboración de un prototipo de la aplicación informática y la definición de las historias de usuario y sus criterios de aceptación.

- Se desarrolló la fase de diseño de la Metodología XP, para lo cual se empleó el modelo arquitectónico 4+1, mismo que describe la arquitectura del software en base a múltiples vistas, como son: vista lógica, vista de procesos, vista de despliegue, vista física y vista de escenarios.
- Se desarrolló la fase de codificación de la aplicación informática.

Fase 3. Evaluar la funcionalidad y usabilidad de la Aplicación Informática en un ambiente controlado o simulado.

Se desarrolló la fase de pruebas de la Metodología XP:

- Pruebas al API REST.
- Pruebas de carga y estrés.
- Pruebas de caja negra.
- Pruebas de usabilidad.
- Pruebas de aceptación.

La aplicación informática fue evaluada en un ambiente controlado.

5.3. Recursos

Para el cumplimiento de las fases anteriormente mencionadas se emplearon los siguientes recursos.

5.3.1. Recursos Científicos

5.3.1.1. Experimento

Para aplicar este método se siguió un proceso experimental, el cual especifica las actividades a ejecutarse, así como las entradas y salidas de cada actividad. Fue utilizado en el tercer objetivo **pruebas de usabilidad**.

5.3.2. Recursos Técnicos

Para la elaboración del presente Trabajo de Titulación se utilizó la metodología Extreme-Programming (XP), la misma que se analizó y justificó basándose en los trabajos relacionados y evaluando algunos criterios para su elección (ver **Revisión de Literatura, sección 4.7, Comparativa de metodologías Ágiles de Desarrollo de Software**).

5.3.2.1. Encuesta

Permitió obtener parte de los requerimientos para el sistema, de igual manera permitió evaluar en un entorno controlado, la funcionalidad y aceptación de la aplicación informática por parte de los usuarios.

5.3.2.2. Revisión Bibliográfica

Permitió definir conceptos importantes respecto al presente Trabajo de Titulación, identificar trabajos relacionados que se han desarrollado y, en base a ello definir la metodología y las herramientas de desarrollo.

5.4. Participantes

El presente TT fue desarrollado por los siguientes participantes:

- Steeven Michael Armijos Bravo como investigador del TT, empezando con la formulación del problema hasta la culminación de los tres objetivos planteados.
- Ing. Wilman Patricio Chamba Zaragocín como director del TT, orientando en todo el desarrollo y cumplimiento de cada objetivo planteado hasta su culminación.
- Habitantes de la ciudad de Loja, como participantes en la aplicación de encuestas para obtener los requerimientos funcionales y no funcionales.
- Estudiantes de la carrera de Ingeniería en Sistemas/ Computación de la Universidad Nacional de Loja, familiares, amigos y conocidos para la fase de pruebas de usabilidad y aceptación de la Aplicación Informática.

5.5. Materiales

Los materiales utilizados para el desarrollo del presente TT se detallan en la Tabla VII.

Tabla VII. Materiales utilizados en el trabajo de titulación

Recursos Hardware	
Nombre	DESCRIPCIÓN
Laptop Toshiba	Dispositivo utilizado en el desarrollo del TT.
Recursos Software	
Nombre	Descripción
Zoom	Software utilizado para la comunicación y revisión del Trabajo de Titulación.
Visual Studio Code	Editor de código, herramienta fundamental para la implementación de la aplicación.
SendGrid	Plataforma para la gestión de envío de correos electrónicos.
Firebase Cloud Messaging	Plataforma para el envío de notificaciones.

Cloudinary	Plataforma para la gestión de imágenes y videos.
Draw.io	Software en versión gratuita para la elaboración de diagramas.
Pencil	Software gratuito utilizado para la elaboración de prototipos de la aplicación LojaHouse.
GitHub	Plataforma web utilizada para llevar el control de versiones en el desarrollo del presente TT.
Tecnología	
Nombre	Descripción
Flutter	Framweork utilizado para la implementación de la aplicación móvil utilizando el lenguaje de programación Dart.
Stack Mean	Stack de tecnologías (MongoDB, ExprerssJs, Angular, NodeJs) que utilizan el lenguaje de programación JavaScript que permitieron la creación de la aplicación LojaHouse.
MongoDB	Base de datos (NoSql) orientada a documentos utilizada para el almacenamiento de la información de la aplicación.
Dart	Lenguaje de programación utilizado para el desarrollo de la aplicación móvil.
JavaScript	Lenguaje de programación utilizado para el desarrollo del backend y frontend.
Firebase Hosting	Plataforma utilizada para el alojamiento del frontend de la aplicación.
Heroku	Palataforma utilizada en su versión gratuita para el alojamiento del backend de la aplicación.

6. Resultados

En el presente apartado se detalla el cumplimiento de los objetivos planteados para el desarrollo del presente Trabajo de Titulación.

6.1. Objetivo I: Determinar los requisitos necesarios para desarrollar la Aplicación Informática.

En el desarrollo de este objetivo se realizó una revisión de trabajos similares para poder obtener requerimientos generales, seguidamente se diseñó las encuestas y se las aplicó mediante la difusión a través de contactos de WhatsApp y redes sociales como Facebook a personas que ofertan bienes inmuebles en alquiler (arrendadores) y a personas que hacen uso del bien inmueble ofertado en alquiler (arrendatarios) en un tiempo de 10 días. Es importante mencionar que, por motivos de la emergencia sanitaria, no fue posible realizar entrevistas, en su defecto se aplicó dos encuestas, ver Anexo 4: **Encuesta para Arrendatarios**, Anexo 3: **Encuesta para Arrendadores**.

A continuación se detalla las actividades realizadas para el cumplimiento del presente objetivo.

6.1.1. Revisión de trabajos similares para obtener requisitos generales.

Se realizó una búsqueda de trabajos similares, con el objetivo de obtener requisitos que les han permitido llevar a cabo el desarrollo de las distintas aplicaciones, a continuación se describe mediante la Tabla VIII, IX, X, XI, XII, XIII. Para más detalle ver Anexo 1:

Obtención de requisitos generales.

Tabla VIII. Requerimientos de la aplicación para la publicación y búsqueda de bienes inmuebles para venta y arriendo

Título	Diseño e implementación de una aplicación orientada a dispositivos móviles con sistema operativo Android para la publicación y búsqueda de bienes inmuebles para venta y arriendo.
Año de publicación	2013
País	Ecuador
Plataforma donde fue encontrado	http://repositorio.puce.edu.ec
Entorno de desarrollo	Móvil
Requerimientos funcionales	

CÓDIGO	REQUERIMIENTO	DESCRIPCIÓN	Nivel de entendimiento	Utilizable en mi aplicación
RF001	Registrar usuario	No tiene	Alto	Si
RF002	Inicio de sesión de usuario	No tiene	Alto	Si
RF003	Administrar cuenta de usuario	No tiene	Alto	Si
RF004	Publicar inmueble	No tiene	Alto	Si
RF005	Administrar inmuebles	No tiene	Alto	Si
RF006	Visualizar inmuebles	No tiene	Alto	Si
RF007	Visualizar inmuebles destacados	No tiene	Alto	No
RF008	Buscar inmuebles	No tiene	Alto	Si
RF009	Agregar a favoritos	No tiene	Alto	No
RF010	Ver favoritos	No tiene	Alto	No
RF011	Ver ubicación de inmuebles	No tiene	Alto	No

Tabla IX. Requerimientos de la aplicación para registrar el alquiler y ventas de una inmobiliaria

Título		Desarrollo de una aplicación web y una aplicación móvil para registrar el alquiler y ventas de una inmobiliaria utilizando Scrum.		
Año de publicación		2015		
País		Ecuador		
Plataforma donde fue encontrado		http://repositorio.utmachala.edu.ec		
Entorno de desarrollo		Web - Móvil		
Requerimientos funcionales				
CÓDIGO	REQUERIMIENTO	DESCRIPCIÓN	Nivel de entendimiento	Utilizable en mi aplicación
RF001	Registrar oficinas	No tiene	Alto	No
RF002	Registrar empleados	No tiene	Alto	No
RF003	Registrar inmuebles	No tiene	Alto	Si
RF004	Registrar propietarios	No tiene	Alto	No
RF005	Registrar clientes	No tiene	Alto	No
RF006	Generar reportes	No tiene	Alto	No
RF007	Visualizar inmuebles disponibles	No tiene	Alto	Si
RF008	Reservar citas	No tiene	Alto	Si
RF009	Visualizar visitas reservadas	No tiene	Alto	Si

Tabla X. Descripción general de la aplicación web para compartir piso.

Título		Diseño y desarrollo de una aplicación web para compartir piso.		
Año de publicación		2018		
País		España		
Plataforma donde fue encontrado		http://openaccess.uoc.edu/		
Entorno de desarrollo		Web		
Requerimientos funcionales				
CÓDIGO	REQUERIMIENTO	DESCRIPCIÓN	Nivel de entendimiento	Utilizable en mi aplicación
RF001	Mostrar alquileres	Mostrar los últimos 10 alquileres que han sido añadidos a la aplicación web.	Alto	Si
RF002	Buscar alquileres	Realizar búsquedas con filtros.	Alto	Si
RF003	Crear alquileres	Añadir un nuevo alquiler a la aplicación y publicación automática en ella.	Bajo	No
RF004	Gestionar publicaciones	Perfil del usuario para gestionar sus publicaciones.	Alto	Si
RF005	Contactarse con el arrendador	Contacto con el arrendador del alquiler.	Alto	Si
RF006	Zonas de información	La aplicación deberá mostrar las siguientes zonas de información: <ul style="list-style-type: none"> • Aviso legal. • Funcionamiento de la aplicación. 	Alto	No

Tabla XI. Descripción de la aplicación para la gestión de propiedades inmobiliarias.

Título		Implementación de un sistema web y una aplicación móvil para administrar los servicios y actividades de una empresa inmobiliaria.	
Año de publicación		2018	
País		Ecuador	
Plataforma donde fue encontrado		http://repositorio.utmachala.edu.ec	
Entorno de desarrollo		Web - Móvil	
Requerimientos funcionales			
CÓDIGO	DESCRIPCIÓN	Nivel de entendimiento	Utilizable en mi aplicación
RF001	Crear empleados	Alto	No
RF002	Crear inquilinos	Alto	No
RF003	Crear contratos	Alto	Si
RF004	Crear oficinas	Medio	No
RF005	Crear visitas e inspecciones	Alto	Si
RF006	Publicar una galería de inmuebles	Alto	Si
RF007	Leer reportes	Medio	No
RF008	Conceder permisos a los usuarios: director, supervisor, empleados, y súper usuario.	Alto	No
RF009	Realizar reservas a inmuebles	Alto	Si

Tabla XII. Requerimientos de la aplicación kleta app

Título		Desarrollo e Implementación de un Aplicativo para Dispositivos Móviles con Sistema Operativo Android y Geolocalización que Permita Ofrecer y Solicitar Servicios de Alquileres de Casas con Ubicación Exacta y a la vez Sugiera las más Cercanas a la Ubicación Actual en Cualquier Parte de Guayaquil.	
Año de publicación		2016	

País	Ecuador			
Plataforma donde fue encontrado	http://repositorio.ug.edu.ec/			
Entorno de desarrollo	Web - Móvil			
Requerimientos funcionales				
CÓDIGO	REQUERIMIENTO	DESCRIPCIÓN	Nivel de entendimiento	Utilizable en mi aplicación
RF001	Preparación Servidor Público	Preparación de Ambiente del Servidor y creación de base de datos para Servicio Web.	Bajo	No
RF002	Implementación de Servicio Web	Implementación y Publicación de Servicio Web.	Bajo	No
RF003	Arranque del Sistema	Inicialización del sistema, preparando el ingreso	Bajo	No
RF004	Registro de Usuario	Registro de datos personales para saber que es un usuario real	Alto	Si
RF005	Ingreso al Sistema	Ingresar credenciales (usuario y clave) para tener acceso al sistema	Bajo	No
RF006	Conexión Google Maps Api con Android	Investigación e Implementación Google Maps Api en Android	Bajo	No
RF007	Pantalla de Mapa	Visualización de Google Maps	Bajo	No

RF008	Geolocalización Visual en Mapa	Mostrar punto de Geolocalización en mapa, vía GPS	Medio	No
RF009	Descarga de Datos	Descarga de Datos desde el Servicio Web, Presentación de Datos de Alquiler en Mapa, con coordenadas exactas.	Bajo	No
RF010	Búsqueda de Datos	Consulta de datos mediante filtros, como número de habitaciones, baños, garaje, patio	Medio	Si
RF011	Resultado de Búsqueda	Presentar un listado con los datos obtenidos en la búsqueda	Bajo	No
RF012	Detalle de Vivienda en Alquiler	Consulta de datos de la vivienda en alquiler que se seleccionó	Alto	Si
RF013	Ruta en Mapa	Creación de Ruta en Mapa, desde mi ubicación hasta la localización del inmueble deseado.	Medio	No
RF014	Llamada Telefónica oferente al alquiler	Proceso de marcación al número telefónico del oferente de la casa	Alto	Si
RF015	Envío de mensaje oferente al alquiler	Proceso de envío de mensaje al oferente	Alto	No

RF016	Mensajes Recibidos	Consultas los mensajes que se han recibido sobre las publicaciones	Bajo	No
RF017	Publicación de Alquiler	Registrar los datos de la casa en alquiler	Medio	Si
RF018	Subir Datos	Subir los datos al Servicio Web	Bajo	No

Tabla XIII. Descripción de la aplicación estupiso segovia.

Titulo		Estupiso Segovia: buscador de viviendas en alquiler en Segovia para estudiantes	
Año de publicación		2016	
País		España	
Plataforma donde fue encontrado		http://uvadoc.uva.es/	
Entorno de desarrollo		Web	
Requerimientos funcionales			
CÓDIGO	REQUERIMIENTO	Nivel de entendimiento	Utilizable en mi aplicación
RF001	El sistema almacenará los usuarios registrados en el sistema que se identificarán por un nombre de usuario	Alto	Si
RF002	El sistema almacenará la contraseña elegida por cada usuario para su autenticación.	Medio	No
RF003	El nombre de usuario deberá tener entre cuatro y doce caracteres permitiendo únicamente minúsculas, números y guion bajo.	Alto	No
RF004	La contraseña deberá tener entre ocho y doce caracteres permitiendo cualquier carácter excepto espacios, tabulaciones y saltos de línea	Alto	No

RF005	De un usuario el sistema almacenará su nombre y apellidos, foto de perfil, tipo y sus datos de contacto como son teléfono fijo y móvil, correo electrónico y si dicho usuario permite contactar vía WhatsApp	Alto	Si
RF006	El teléfono fijo y la foto de perfil de un usuario no serán obligatorios pudiéndose sustituir la foto por una por defecto	Alto	No
RF007	Un usuario puede ser "Particular" o "Estudiante"	Alto	No
RF008	El sistema almacenará las viviendas registradas por los usuarios que se identificarán por un número incremental y diferenciará entre viviendas en alquiler y viviendas compartidas	Medio	No
RF009	De una vivienda el sistema almacenará el usuario que la publica, si está disponible o alquilada, su fecha de publicación y de última actualización, un título, una descripción, su dirección y coordenadas, coste mensual en Euros, número de habitaciones y de baños, superficie en metros cuadrados, tipo y planta	Alto	Si
RF010	Para las viviendas compartidas el sistema almacenará el número de habitaciones que están libres y el coste mensual corresponderá al coste por habitación	Medio	No
RF011	El tipo de una vivienda puede ser "Piso", "Estudio", "Dúplex" o "Chalet" y	Alto	No

	la planta “Bajo”, “Primero”, “Plantas intermedias”, “Última planta” o “Chalet”		
RF012	El título de una vivienda se limitará a cincuenta caracteres y la descripción a cien	Alto	No
RF013	De una vivienda el sistema almacenará de una a diez fotos estableciéndose la primera como foto principal.	Alto	No
RF014	El sistema almacenará los gastos que sí y los que no incluye el coste de una vivienda diferenciando entre “Comunidad”, “Impuestos y seguro”, “Agua, electricidad y gas” y “Teléfono e internet”	Medio	No
RF015	El sistema almacenará las características de las que sí y de las que no dispone una vivienda siendo estas “Disponibilidad inmediata”, “Recién reformada”, “Amueblada”, “Cocina equipada”, “Ascensor”, “Garaje”, “Trastero”, “Calefacción individual”, “Aire acondicionado”, “Certificación energética y “Mascotas permitidas”	Alto	No

En la Tabla XIV se puede observar los requerimientos comunes extraídos en base al análisis y evaluación de los trabajos relacionados.

Tabla XIV. Requerimientos comunes entre los trabajos relacionados

Requisitos comunes seleccionados	
Requerimiento	Descripción
Mostrar alquileres	Mostrar los últimos 10 alquileres que han sido añadidos a la aplicación web.
Buscar alquileres	Realizar búsquedas con filtros.

Gestionar publicaciones	Perfil del usuario para gestionar sus publicaciones.
Contactarse con el arrendador	Contacto con el arrendador del alquiler.
Mostrar propiedades	El sistema muestra las propiedades que se encuentran en la aplicación
Registro de Usuario	Registro de datos personales para saber que es un usuario real
Búsqueda de Datos	Consulta de datos mediante filtros, como número de habitaciones, baños, garaje, patio
Detalle de Vivienda en Alquiler	Consulta de datos de la vivienda en alquiler que se seleccionó
Llamada Telefónica al oferente del alquiler	Proceso de marcación al número telefónico del oferente de la casa
Publicación de Alquiler	Registrar los datos de la casa en alquiler
Registrar vivienda	De una vivienda, el sistema almacenará el usuario que la publica, si está disponible o alquilada, su fecha de publicación y de última actualización, un título, una descripción, su dirección y coordenadas, coste mensual en Euros, número de habitaciones y de baños, superficie en metros cuadrados, tipo y planta
Realizar reservas a inmuebles	El usuario registrado podrá realizar reservas a inmuebles

6.1.2. Formular y aplicar encuestas a la población seleccionada para obtener requisitos de funcionalidad y usabilidad.

Para cumplir con esta actividad se definió las preguntas para las encuestas, tanto para las personas que ofertan en alquiler un bien inmueble (arrendador) y para las personas que buscan alquilar un bien inmueble (arrendatario), las mismas que fueron validadas por el director de tesis, el Ing. Wilman Patricio Chamba Zaragocín.

Para la selección de la población se utilizó la técnica de muestreo no probabilístico por conveniencia, fundamentado en la conveniente accesibilidad y proximidad de las personas a encuestar, donde la muestra seleccionada fue a estudiantes, habitantes de la ciudad de Loja y personas que desearían alquilar un bien inmueble, ya que por motivo de las restricciones sanitarias impuestas a nivel mundial no se pudo tener contacto

directo con la población general y se optó por solicitar ayuda a conocidos, amigos y familiares con el fin de lograr obtener respuestas de los usuarios arrendadores y arrendatarios, para ello se utilizó la herramienta Google Forms para la elaboración de las encuestas, las mismas que fueron difundidas por medio de la aplicación de mensajería WhatsApp, y la red social Facebook; obteniendo un total de 143 encuestas respondidas por parte de los arrendatarios, y 43 encuestas contestadas por parte de los arrendadores, donde las preguntas más relevantes para definir los requerimientos, fueron las siguientes:.

¿Qué medios ha utilizado para buscar un bien inmueble en alquiler?

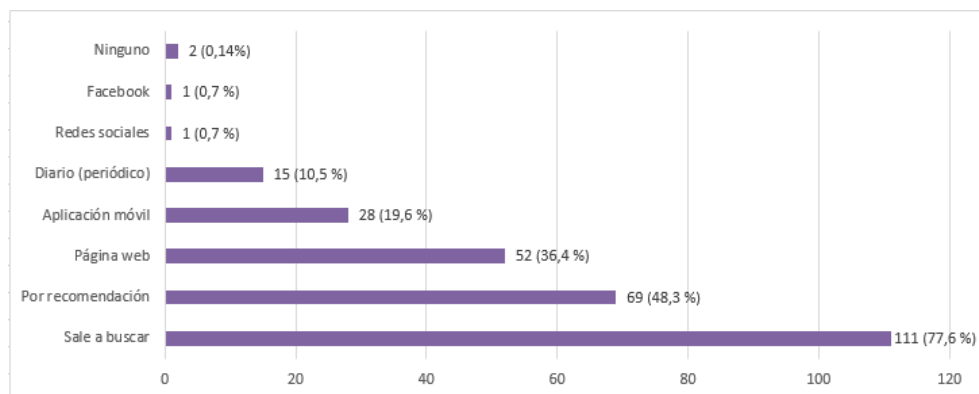


Figura 1. Pregunta relevante 1 de la encuesta (arrendatario)

En la Figura 1 se puede observar que el 77.6% de los encuestados sale a buscar un bien inmueble para alquilar, un 48% busca por medio de recomendaciones, por otro lado, el 36.4% realiza la búsqueda por medio de páginas web y el 19.6% mediante el uso de una aplicación móvil, en menor porcentaje con el 10.5% realizan este proceso de búsqueda por medio de periódicos, y con un 0.7% lo realizan por medio de redes sociales.

¿Qué medio de comunicación prefiere para contactarse con el arrendador de un bien inmueble?

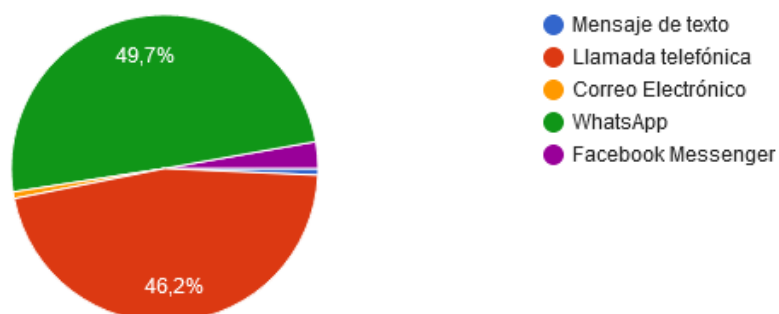


Figura 2. Pregunta relevante 2 de la encuesta (arrendatario)

En la Figura 2 se observa que los encuestados prefieren con un 49.7% comunicarse con el arrendador por medio de la aplicación WhatsApp y con un 46.2% mediante llamadas telefónicas, y en menor proporción mediante correo electrónico, Facebook Messenger y mensaje de texto. Se concluye que la aplicación debe permitir llamadas telefónicas y permitir el envío de mensajes a través de la aplicación WhatsApp.

Cuando arrienda un inmueble, desearía que le ofrezcan a más de los servicios básicos los siguientes:

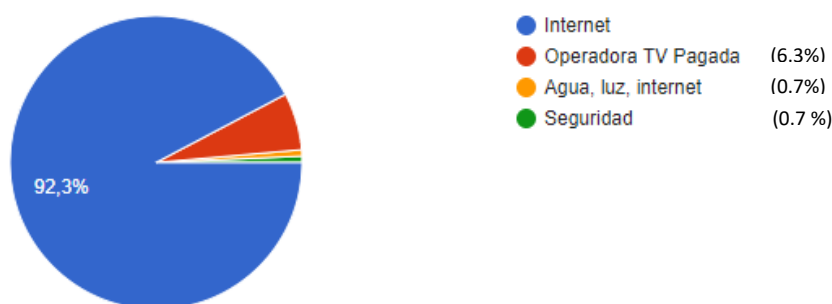


Figura 3. Pregunta relevante 3 de la encuesta (arrendatario)

En la figura 3 se puede observar que el 92.3% de los encuestados desearía que los inmuebles publicados en alquiler ofrezcan el servicio de internet, un 6.3% respondió que desearía el servicio de Operadora TV Pagada, un 0.7% respondió que desearía que le ofrezcan el servicio de agua, luz, y seguridad. Este resultado permitió determinar que la aplicación debe permitir gestionar los servicios básicos del bien inmueble ofertado.

Para la formalización del alquiler lo realiza a través de:

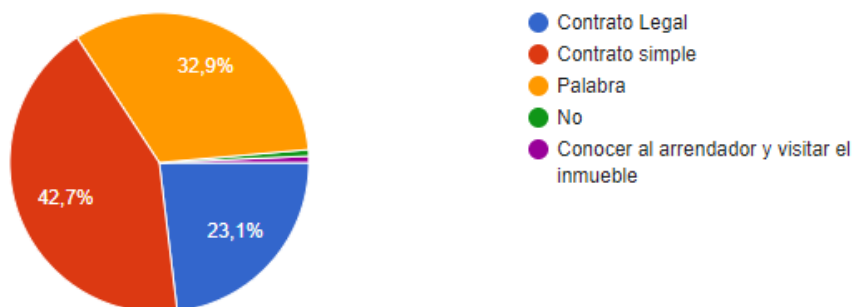


Figura 4. Pregunta relevante 4 de la encuesta (arrendatario)

En la Figura 4 se puede observar que el 42.7% de los encuestados formaliza un alquiler de inmueble mediante un contrato simple, el 32.9% mediante la palabra entre las partes, un 23.1% indicó que lo realiza a través de un contrato legal de arrendamiento, mientras

que el 0.7% respondió que no formaliza mediante un contrato de arrendamiento, que prefiere conocer al arrendador y visitar el inmueble. Este resultado permitió considerar la generación de un contrato simple mediante el uso de una plantilla preexistente de contrato de arrendamiento.

¿Cree usted que optimizaría su tiempo de búsqueda al utilizar una aplicación informática para buscar un bien inmueble en alquiler?

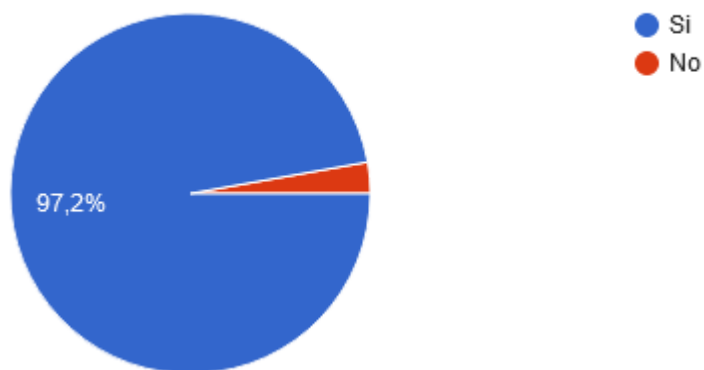


Figura 5. Pregunta relevante 5 de la encuesta (arrendatario)

Como se puede observar en la Figura 5, el 97.2% de los encuestados respondieron que una aplicación informática optimizaría su tiempo de búsqueda, por otro lado se tiene que el 2.8 % menciona que no optimizaría el tiempo de búsqueda. En base a ello se concluye que una aplicación informática ayudará en el proceso de búsqueda de un inmueble en alquiler.

6.1.3. Especificación de requisitos.

En base a la obtención de requisitos generales y a los resultados de las encuestas, se realizó la especificación de requerimientos, para ello se empleó el estándar IEEE 830, el cual consiste en la integración de los requerimientos de un sistema desde la perspectiva de los actores involucrados en su diseño, desarrollo y utilización: usuario, cliente y desarrollador [39], para mayor detalle ver el Anexo 2: **Especificación de Requerimientos**, donde se definen los requerimientos funcionales y no funcionales de la Aplicación Informática.

- **Requerimientos Funcionales**

En la Tabla XV se muestran los requerimientos funcionales de la Aplicación Informática.

Tabla XV. Requerimientos funcionales

REQUERIMIENTOS FUNCIONALES	
RF001	Registrar Usuarios.
RF002	Autenticar Usuarios.
RF003	Gestionar cuentas de usuarios.
RF004	Gestionar perfil de usuario.
RF005	Gestionar Inmueble.
RF006	Gestionar solicitudes de reserva de alquiler de un bien inmueble.
RF007	Gestionar contratos de alquiler.
RF008	Alquilar bien inmueble.
RF009	Visualizar contratos de alquiler interno.
RF010	Visualizar información de un bien inmueble.
RF011	Gestionar solicitudes de reserva de alquiler de un bien inmueble.
RF012	Buscar un bien inmueble
RF013	Cambiar contraseña.
RF014	Resetear contraseña.
RF015	Publicar un anuncio.
RF016	Enviar mensajes por WhatsApp.
RF017	Realizar una llamada telefónica.
RF018	Recibir notificación de un bien inmueble ofertado.
RF019	Seleccionar filtros de búsqueda.
RF020	Enviar mensajes al administrador
RF021	Visualizar mensajes
RF022	Eliminar mensajes
RF023	Gestionar servicios básicos

- **Requerimientos no Funcionales**

En la Tabla XVI se muestran los requerimientos no funcionales de la Aplicación Informática.

Tabla XVI. Requerimientos no funcionales

REQUERIMIENTOS NO FUNCIONALES		
CÓDIGO	REQUEREIMIENTO	DESCRIPCIÓN
RNF001	Requisitos de rendimiento	El tiempo de respuesta en las operaciones debe ser inferior o igual a 5 segundos.
RNF002	Seguridad	<ul style="list-style-type: none"> • La Aplicación Informática deberá encriptar la contraseña de los usuarios registrados mediante el método de encriptación bcrypt. • Para controlar el ingreso a las diferentes funcionalidades de la Aplicación Informática se lo realizará mediante el uso de JWT. • La Aplicación Informática funcionará bajo el protocolo HTTP. • La Aplicación Informática deberá restringir el acceso a los usuarios dependiendo de su rol dentro de la Aplicación Informática.
RNF003	Fiabilidad	La Aplicación Informática deberá registrar los datos ingresados por parte de los usuarios de manera correcta, y al producirse un error de cualquier índole, se deberá emitir un mensaje dentro de la Aplicación.
RNF004	Disponibilidad	La disponibilidad de la Aplicación Informática será de los 365 días del año, cuya afección producida dependerá del servicio de hosting que se contrate.
RNF005	Portabilidad	La Aplicación Informática podrá ser utilizada tanto en navegadores web y Smartphone con sistema operativo Android.
RNF006	Usabilidad	La Aplicación Informática debe ser desarrollada con una interfaz amigable para el usuario y de fácil comprensión, con la finalidad que pueda acceder a las diferentes funcionalidades de forma rápida e intuitiva.

6.2. Objetivo II: Diseñar e implementar la Aplicación Informática para la búsqueda de alquiler de inmuebles en base a la metodología XP.

En esta sección se describe las actividades realizadas en el diseño e implementación de la Aplicación Informática en base a la metodología XP, detallando cada fase en el desarrollo del presente TT.

La metodología XP consta de 4 fases, mismas que permitieron culminar con el presente TT, las fases se describen a continuación:

6.2.1. Planificación

En [14] con respecto a la metodología XP, señala que, en esta primera fase, la comunicación con el cliente es esencial para poder definir los requisitos, mismos que permiten crear las historias de usuario que describen la funcionalidad del software que se va a construir.

Equipos integrantes y roles

El equipo de trabajo encargado de la implementación de la aplicación LojaHouse, se detalla en la Tabla XVII.

Tabla XVII. Conformación del equipo de trabajo

ROL	PERSONA
Programador	Steeven Michael Armijos Bravo
Cliente	Arrendador
	Arrendatario
Encargado de Pruebas	Steeven Michael Armijos Bravo
Encargado de Seguimiento	Ing. Wilman Patricio Chamba Zaragocín
Entrenador	Ing. Wilman Patricio Chamba Zaragocín
Gestor	Steeven Michael Armijos Bravo

Se identificó tres tipos de usuarios que intervienen en la Aplicación Informática, los cuales se detallan en la Tabla XVIII.

Tabla XVIII. Tipo de usuario

TIPO DE USUARIO	DESCRIPCIÓN
Arrendador	<p>El usuario arrendador tiene acceso a las siguientes funcionalidades:</p> <ul style="list-style-type: none"> • Gestión de bienes inmuebles • Gestión de solicitudes de reservas de alquiler de un bien inmueble. • Gestión de contrato de alquiler interno. • Gestión de su perfil de usuario. • Enviar mensaje al administrador.
Arrendatario	<p>El usuario arrendatario tiene acceso a las siguientes funcionalidades:</p> <ul style="list-style-type: none"> • Visualización de bienes inmuebles. • Buscar bien inmueble. • Gestión de solicitudes de reservas de alquiler de un bien inmueble. • Aceptar contrato de alquiler. • Visualizar contratos de alquiler. • Enviar mensaje al administrador. • Gestión de su perfil de usuario.
Administrador	<p>El usuario administrador tiene acceso a las siguientes funcionalidades:</p> <ul style="list-style-type: none"> • Activar cuenta de usuario. • Desactivar cuenta de usuario. • Visualizar mensajes. • Eliminar mensajes. • Gestión de servicios básicos. • Gestión de su perfil de usuario.

Historias de Usuario

La metodología XP utiliza la técnica de Historias de Usuario (HU) para especificar los requisitos del software, describiendo brevemente las características que el sistema debe tener, ya sean requisitos funcionales como no funcionales. Las HU deben ser

comprensibles y delimitadas para que el programador pueda implementarla en el menor tiempo posible [14]. La metodología XP propone un esquema de tabla para redactar las HU, para ello se tomó como referencia el esquema propuesto en [40] la cual se muestra en la Tabla XIX.

Tabla XIX. Modelo de historia de usuario

Historia de Usuario	
Número:	Nombre de Historia de Usuario:
Usuario:	
Modificación de Historia Número:	Interacción Asignada:
Prioridad en el negocio (Alta/Media/Baja):	Puntos estimados:
Riesgo en el desarrollo (Alta/Media/Baja):	Puntos reales:
Descripción:	
Observaciones:	
Criterios de aceptación:	

Descripción de los campos de la Historia de Usuario:

- **Número:** Es la asignación de un número para cada Historia de Usuario.
- **Nombre de la historia de usuario:** Es un identificador descriptivo corto de la HU.
- **Usuario:** Nombre del responsable en la actividad descrita de la HU.
- **Modificación de Historia Número:** Número de modificación que se obtuvo en la HU. **Prioridad en el negocio:** Se asigna acorde a la importancia que tiene la actividad.
- **Riesgo en el desarrollo:** Complejidad de desarrollo de la actividad.
- **Interacción Asignada:** Número de interacción asignada a la actividad.
- **Puntos estimados:** Cada punto estimado hace referencia a cada día efectivo de desarrollo.
- **Puntos reales:** Puntos reales utilizados en la actividad.
- **Descripción:** Detalle de las actividades en las Historias de Usuario.
- **Observaciones:** Aspectos importantes en cada Historias de Usuarios.

- **Criterios de aceptación:** Criterios que el desarrollo debe cumplir para que la historia de usuario pueda ser aceptada.

A continuación se presentan las 18 Historias de Usuario.

Tabla XX. Registrar usuarios.

Historia de Usuario	
Número: 001	Nombre de Historia de Usuario: Registrar Usuarios
Usuario: Arrendador/Arrendatario	
Modificación de Historia Número: 2	Interacción Asignada: 1
Prioridad en el negocio (Alta/Media/Baja): Alta	Puntos estimados: 1
Riesgo en el desarrollo (Alta/Media/Baja): Alta	Puntos reales: 1
Descripción: Los usuarios podrán registrarse en la Aplicación Informática a través de un formulario donde se solicitará el ingreso de información básica como: correo electrónico, nombres, apellidos, número de celular y el rol, ya sea como arrendador o arrendatario.	
Observaciones:	
Criterios de aceptación: <ol style="list-style-type: none"> 1. Mostrar el formulario de registro a través de la opción "Regístrate". 2. Validar campos obligatorios vacíos del formulario de registro de usuario. 3. Mostrar un mensaje de aviso indicando que existen campos obligatorios vacíos. 4. Validar que el correo tenga un formato válido. 5. Validar que el correo electrónico no exista en la base de datos. 6. Mostrar un mensaje de aviso indicando que el correo ya existe. 7. Validar que se haya elegido un rol. 8. Generar una contraseña temporal. 9. Crear una cuenta de usuario con los datos de la persona con la información básica ingresada y su respectivo rol. 10. Mostrar un mensaje de aviso indicando que la cuenta ha sido creada y su clave se ha enviado al correo electrónico. 11. Notificar la contraseña al correo electrónico del usuario. 	

12. Mostrar una opción preguntando si el usuario tiene una cuenta, permitiéndole Iniciar Sesión.

Tabla XXI. Autenticar usuarios

Historia de Usuario	
Número: 002	Nombre de Historia de Usuario: Autenticar Usuarios
Usuario: Todos	
Modificación de Historia Número: 3	Interacción Asignada: 1
Prioridad en el negocio (Alta/Media/Baja): Alta	Puntos estimados: 2
Riesgo en el desarrollo (Alta/Media/Baja): Baja	Puntos reales: 2
Descripción: Los usuarios podrán autenticarse en la Aplicación Informática ingresando su correo electrónico y contraseña.	
Observaciones:	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. Mostrar el formulario de inicio de sesión a través de la opción "Iniciar Sesión". 2. Validar que el correo y la contraseña hayan sido ingresados. 3. Mostrar un mensaje de aviso indicando que los datos ingresados están incorrectos cuando el correo electrónico o contraseña no exista. 4. Mostrar una opción preguntando si el usuario ha olvidado su contraseña. 5. Mostrar una opción preguntando si el usuario no tiene una cuenta y que le permita acceder al formulario de registro. 6. Acceder al menú de la Aplicación dependiendo del rol del usuario autenticado. 	

Tabla XXII. Gestionar cuentas de usuarios

Historia de Usuario	
Número: 003	Nombre de Historia de Usuario: Gestionar Cuentas de Usuarios
Usuario: Administrador	
Modificación de Historia Número: 1	Interacción Asignada: 6

Prioridad en el negocio (Alta/Media/Baja): Alta	Puntos estimados: 2
Riesgo en el desarrollo (Alta/Media/Baja): Baja	Puntos reales: 2
Descripción: El usuario Administrador podrá leer, activar o desactivar las cuentas de los usuarios registrados.	
Observaciones:	
Criterios de aceptación: <ol style="list-style-type: none"> 1. En el panel del administrador, en la sección Arrendador y Arrendatario, podrá el administrador visualizar los usuarios registrados. 2. Debe permitir filtrar los usuarios por medio del estado de los mismos, para ello debe contener 3 opciones (Activados, Desactivados, Todos). 3. Al seleccionar el filtro, se deberán mostrar los usuarios que cumplan con esa condición. 4. En cada ítem se mostrará un botón para activar o desactivar el usuario, también la información de todos los usuarios arrendadores (estado, imagen, nombre, número de celular y teléfono, correo electrónico) y tres botones (Inmuebles, Visitas, Contratos). 5. Al dar clic en el botón Activar/Desactivar el usuario, se presentará una advertencia indicando lo siguiente: “¿Está seguro de realizar la siguiente acción?” 6. Al dar clic en el botón Aceptar, se notificará mediante una advertencia que el usuario ha sido Activado o Desactivado 7. Al desactivar un usuario las opciones de los botones (Inmuebles, Visitas, Contratos) quedarán inhabilitadas. 8. Al dar clic en el botón Inmuebles del usuario “ACTIVADO”, se mostrarán los inmuebles de ese usuario en una nueva sección. 9. Al dar clic en el botón Visitas del usuario “ACTIVADO”, se mostrarán las visitas de ese usuario en una nueva sección. 10. Al dar clic en el botón Contratos del usuario “ACTIVADO”, se mostrarán los contratos de ese usuario en una nueva sección. 	

Tabla XXIII. Gestionar perfil de usuario

Historia de Usuario	
Número: 004	Nombre de Historia de Usuario: Gestionar perfil de usuario
Usuario: Todos	
Modificación de Historia Número: 2	Interacción Asignada: 1
Prioridad en el negocio (Alta/Media/Baja): Alta	Puntos estimados: 2
Riesgo en el desarrollo (Alta/Media/Baja): Baja	Puntos reales: 2
Descripción: Los usuarios autenticados podrán leer y actualizar la información personal de su perfil de usuario.	
Observaciones:	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. Mostrar el formulario de perfil de usuario a través de la opción “Mi perfil” desde el panel administrativo. 2. Mostrar información (Nombres, Apellidos, Cédula, Celular, Convencional, foto) del usuario autenticado. 3. El campo cédula será obligatorio y debe validar el número de cédula ingresado, en caso de ser incorrecto deberá indicar que el número de cédula es inválido. 4. El campo convencional será opcional. 5. Al completar todos los campos y dar clic en el botón “Actualizar Información”, se desplegará un mensaje de aviso, indicando lo siguiente: “Actualización exitosa. Se ha actualizado su información correctamente”. 6. Existirá una sección para actualizar la foto de perfil, donde se podrá buscar una foto en el directorio del equipo del usuario y se habilitará el botón para actualizar la misma. 	

Tabla XXIV. Resetear contraseña

Historia de Usuario	
Número: 005	Nombre de Historia de Usuario: Resetear Contraseña
Usuario: Arrendador/Arrendatario	

Modificación de Historia Número: 2	Interacción Asignada: 1
Prioridad en el negocio (Alta/Media/Baja): Alta	Puntos estimados: 2
Riesgo en el desarrollo (Alta/Media/Baja): Baja	Puntos reales: 2
Descripción: Los usuarios registrados (arrendador, arrendatario) podrán resetear su contraseña escribiendo su correo electrónico.	
Observaciones:	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. Mostrar el formulario de recuperar contraseña a través de la opción ¿Olvidaste tu contraseña? 2. Validar que se haya ingresado el correo electrónico. 3. Validar que la cuenta de usuario exista a través del correo electrónico; en caso de que no exista se mostrará un mensaje de aviso. 4. Generar una contraseña temporal. 5. Notificar la contraseña al correo electrónico del usuario. 6. Mostrar un mensaje de aviso indicando que su contraseña ha sido actualizada y se ha enviado al correo electrónico. 	

Tabla XXV. Cambiar contraseña

Historia de Usuario	
Número: 006	Nombre de Historia de Usuario: Cambiar Contraseña
Usuario: Todos	
Modificación de Historia Número: 3	Interacción Asignada: 1
Prioridad en el negocio (Alta/Media/Baja): Alta	Puntos estimados: 1
Riesgo en el desarrollo (Alta/Media/Baja): Media	Puntos reales: 3
Descripción: Los usuarios autenticados (arrendador, arrendatario, administrador) podrán actualizar su contraseña.	
Observaciones:	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. Mostrar una acción al usuario autenticado que le permita actualizar su contraseña. 	

2. Mostrar un formulario de cambio de contraseña.
3. Validar que los campos nueva contraseña y confirmación de contraseña no estén vacíos; en caso de estarlos presentar un mensaje indicando que están vacíos.
4. Debe validar que la nueva contraseña cumpla con las siguientes seguridades mínimas:
 - Tamaño mínimo 6 caracteres.
 - Debe contener al menos una letra mayúscula.
 - Debe contener al menos un dígito.
5. Validar que la contraseña nueva y la confirmación de la contraseña coincidan.
6. Mostrar un mensaje de aviso indicando que alguna de las validaciones no se cumple.
7. Mostrar un mensaje de aviso indicando que su contraseña ha sido actualizada.

Tabla XXVI. Publicar anuncio

Historia de Usuario	
Número: 007	Nombre de Historia de Usuario: Publicar anuncio
Usuario: Arrendador	
Modificación de Historia Número: 2	Interacción Asignada: 2
Prioridad en el negocio (Alta/Media/Baja): Alta	Puntos estimados: 2
Riesgo en el desarrollo (Alta/Media/Baja): Alta	Puntos reales: 4
Descripción: El usuario arrendador tendrá la opción de publicar un anuncio del bien inmueble que tenga registrado.	
Observaciones:	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. Mostrar el listado de bienes inmuebles registrados por el usuario arrendador. 2. Publicar el bien inmueble que desee a través del botón publicar. 3. Mostrar una advertencia indicando lo siguiente: “¿Está seguro de realizar la siguiente acción? El inmueble estará: PÚBLICO” 	
<ol style="list-style-type: none"> 1. Cambiar el estado del bien inmueble a: publicado. 2. Inhabilitar el botón “publicar” y habilitar el botón “dejar de publicar” el inmueble. 3. Mostrar un mensaje de aviso indicando lo siguiente: “inmueble PÚBLICO”. 	

4. Mostrar el bien inmueble publicado en la pantalla principal de la Aplicación Informática.

Tabla XXVII. Gestión de bienes inmuebles

Historia de Usuario	
Número: 008	Nombre de Historia de Usuario: Gestión de bienes inmuebles
Usuario: Arrendador	
Modificación de Historia Número: 2	Interacción Asignada: 2
Prioridad en el negocio (Alta/Media/Baja): Alta	Puntos estimados: 2
Riesgo en el desarrollo (Alta/Media/Baja): Alta	Puntos reales: 4
Descripción: El usuario arrendador podrá gestionar sus bienes inmuebles (crear, actualizar, eliminar, visualizar, buscar y eliminar).	
Observaciones:	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. Acceder a la sección de gestión de bienes inmuebles a través de la opción Gestionar Inmuebles del menú del usuario arrendador. 2. Visualizar el listado de bienes inmuebles registrados. 3. Buscar bienes inmuebles según el nombre. 4. Filtrar inmuebles según el estado: "Disponibles", "Ocupados" o "Todos". 5. Mostrar los bienes inmuebles según el criterio de búsqueda, cuya información a mostrar es: estado, imágenes, nombre, precio, dirección, y acciones (editar, eliminar, publicar, dejar de publicar). 	

Tabla XXVIII. Registrar bienes inmuebles

Historia de Usuario	
Número: 009	Nombre de Historia de Usuario: Registrar bienes inmuebles
Usuario: Arrendador	
Modificación de Historia Número: 3	Interacción Asignada: 4
Prioridad en el negocio (Alta/Media/Baja): Alta	Puntos estimados: 5

Riesgo en el desarrollo (Alta/Media/Baja): Alta	Puntos reales: 15
Descripción: El usuario arrendador podrá registrar nuevos bienes inmuebles.	
Observaciones:	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. Mostrar el formulario de registro de un nuevo inmueble a través del botón Nuevo Inmueble de la pantalla gestión de inmuebles. 2. Cargar las opciones de tipo de inmueble como: casa, departamento, cuarto, mini-departamento en la pantalla de registro de nuevos inmuebles. 3. Seleccionar los servicios que incluye un bien inmueble. 4. Validar campos obligatorios. 5. Mostrar un mensaje indicando que hay campos obligatorios vacíos. 6. Controlar si el bien inmueble posee garantía, si no se ingresa un monto, por defecto será 0. 7. Guardar el inmueble al dar clic en el botón “Guardar inmueble”. 8. Registrar el bien inmueble y mostrar un mensaje notificando al usuario que se ha registrado su inmueble. 9. Redirigir a otra pantalla donde se podrá subir las imágenes del inmueble. 10. Cargar imágenes al bien inmueble. 11. Quitar imágenes (de ser necesario) del bien inmueble. 12. Mostrar un mensaje indicando lo siguiente: “Imágenes cargadas exitosamente”. 	

Tabla XXIX. Editar bienes inmuebles

Historia de Usuario	
Número: 010	Nombre de Historia de Usuario: Editar bienes inmuebles
Usuario: Arrendador	
Modificación de Historia Número: 3	Interacción Asignada: 4
Prioridad en el negocio (Alta/Media/Baja): Alta	Puntos estimados: 5
Riesgo en el desarrollo (Alta/Media/Baja): Alta	Puntos reales: 15
Descripción: El usuario arrendador podrá editar los bienes inmuebles registrados.	
Observaciones:	

Criterios de aceptación:

1. Mostrar el formulario de actualizar un inmueble a través de la opción Editar del listado de bienes Inmuebles.
2. Mostrar los datos del inmueble seleccionado.
3. Seleccionar o quitar imágenes seleccionadas (de ser necesario) antes de actualizar los datos del bien inmueble.
4. Cargar las opciones de tipo de inmueble como: casa, departamento, cuarto, mini-departamento.
5. Seleccionar los servicios que incluye un bien inmueble.
6. Validar campos obligatorios.
7. Mostrar un mensaje indicando que hay campos obligatorios vacíos.
8. Controlar si el bien inmueble posee garantía, si no se ingresa un monto, por defecto será 0.
9. Actualizar el inmueble al dar clic en el botón “Guardar inmueble”.
10. Registrar el bien inmueble y mostrar un mensaje notificando al usuario que se ha actualizado su inmueble.
11. Al dar clic en el botón cargar imágenes se deberá redirigir a otra pantalla donde se podrá subir las imágenes del inmueble.
12. Cargar imágenes al bien inmueble.
13. Quitar imágenes (de ser necesario) del bien inmueble.
14. Mostrar un mensaje indicando lo siguiente: “Imágenes cargadas exitosamente”.

Tabla XXX. Gestión de solicitudes de reserva de alquiler recibidas

Historia de Usuario	
Número: 011	Nombre de Historia de Usuario: Gestión de solicitudes de reserva de alquiler recibidas.
Usuario: Arrendador	
Modificación de Historia Número: 3	Interacción Asignada: 4
Prioridad en el negocio (Alta/Media/Baja): Alta	Puntos estimados: 5
Riesgo en el desarrollo (Alta/Media/Baja): Alta	Puntos reales: 15

Descripción: El usuario arrendador podrá gestionar las solicitudes de visita a sus bienes inmuebles, podrá: VER, APROBAR, RECHAZAR y generar CONTRATO.

Observaciones:

Criterios de aceptación:

1. Acceder a la sección de Gestión de Visitas a través de la opción en el menú del usuario arrendador.
2. Visualizar el listado de solicitud de visitas.
3. Filtrar las solicitudes de visitas de acuerdo al estado (PENDIENTE, ACEPTADA, ATENDIDA, TODAS).
4. Mostrar información de la solicitud: imagen del usuario arrendatario, nombre del usuario arrendatario, la descripción de la visita, el inmueble a visitar, fecha de la visita, el estado del mismo y 4 acciones (ACEPTAR, RECHAZAR, CONTRATO, VER).
5. Al seleccionar la opción ACEPTAR se notificará al usuario arrendatario que su solicitud de visita será atendida en la fecha solicitada.
6. Al seleccionar la opción RECHAZAR se notificará al usuario arrendatario que su solicitud de visita no será atendida por retraso en la fecha solicitada.
7. Mostrar la notificación en la aplicación del usuario arrendatario.
8. Al seleccionar la opción VER, se mostrará la solicitud con la siguiente información (la imagen del usuario arrendatario, el nombre, número de celular, el inmueble a visitar, fecha de visita y el estado de la visita).
9. Se deberá mostrar la información de la solicitud de la visita elegida.
10. Al seleccionar la opción CONTRATO se redirigirá a la sección de generar un contrato de alquiler.

Tabla XXXI. Gestionar contratos de alquiler

Historia de Usuario	
Número: 012	Nombre de Historia de Usuario: Gestionar contratos de alquiler
Usuario: Arrendador	
Modificación de Historia Número: 3	Interacción Asignada: 4
Prioridad en el negocio (Alta/Media/Baja): Alta	Puntos estimados: 5

Riesgo en el desarrollo (Alta/Media/Baja): Alta	Puntos reales: 15
Descripción: El usuario arrendador tendrá una sección de gestionar contratos de alquiler de un bien inmueble.	
Observaciones:	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. Acceder a la sección de Gestión de Contratos de Alquiler a través de la opción en el menú del usuario arrendador. 2. Visualizar el listado de contratos de alquiler generados. 3. Buscar contratos según el nombre del contrato. 4. Mostrar los contratos de alquiler, según el criterio de búsqueda en una tabla cuyas columnas debe contener: Nombre, Inicio, Finalización, Meses de alquiler, Estado y Acciones (ver, terminar contrato). 5. El botón "Terminar el contrato" estará habilitado y el arrendador deberá dar clic y se desplegará una advertencia indicando: "¿Está seguro de realizar la siguiente acción? El contrato será: TERMINADO", al dar clic en aceptar, el contrato se dará por terminado y se deshabilitará el botón. 6. Una vez terminado el contrato, se habilitará el botón "Publicar el inmueble", al dar clic se desplegará una advertencia indicando: "¿Está seguro de realizar la siguiente acción? El inmueble estará: DISPONIBLE", al dar clic en aceptar, el inmueble estará disponible para de nuevo ser publicado y se deshabilitará el botón. 7. Al dar clic en el botón de la columna "Acciones" (ver contrato) se desplegará el contrato generado, donde se podrá visualizarlo y descargarlo. 	

Tabla XXXII. Alquilar inmueble

Historia de Usuario	
Número: 013	Nombre de Historia de Usuario: Alquilar inmueble
Usuario: Arrendador	
Modificación de Historia Número: 2	Interacción Asignada: 4
Prioridad en el negocio (Alta/Media/Baja): Alta	Puntos estimados: 2

Riesgo en el desarrollo (Alta/Media/Baja): Alta	Puntos reales: 4
Descripción: El usuario arrendador tendrá la opción de alquilar un bien inmueble a un usuario desde su panel administrativo.	
Observaciones:	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. Mostrar el listado de bienes inmuebles públicos registrados por el usuario arrendador. 2. Alquilar el bien inmueble que desee a través del botón "Alquilar". 3. Redirigir a una nueva sección de un formulario de registro. 4. Buscar el usuario arrendatario mediante su correo electrónico. 5. Verificar que el usuario exista, en caso de no existir se deberá mostrar un mensaje al usuario arrendador que "El usuario con el correo no existe". 6. Verificar que no se pueda alquilar el bien inmueble a un usuario arrendador ni administrador, para ello se deberá desplegar una advertencia indicando lo siguiente: "No se puede asignar el contrato al usuario". 7. El usuario arrendador podrá registrar un usuario arrendatario en caso de no estar registrado en la Aplicación Informática para lo cual deberá ingresar los siguientes datos: Nombres, Apellidos, Celular, Correo electrónico. 8. Se validarán los campos vacíos, en caso de estarlos, indicar que se deben completar. 9. Al dar clic en el botón "Registrar arrendatario", quedará registrado el nuevo usuario con el rol predefinido como ARRENDATARIO. 10. Se deberá mostrar el nombre y apellido del usuario arrendatario en un campo de texto. 11. Se podrá seleccionar la fecha de inicio y finalización del contrato mediante un calendario. 12. Validar que se haya seleccionado una fecha de inicio y finalización del contrato, caso contrario indicar mediante una advertencia, lo siguiente: "Debe ingresar la fecha de finalización del contrato". 13. Validar que se seleccione por lo menos un mes en el tiempo del contrato, caso contrario indicar mediante una advertencia, lo siguiente: "El contrato debe por lo menos ser de un mes". 14. Se podrá modificar el monto mensual de alquiler y la garantía. 	

15. Al dar clic en el botón “Generar contrato”, se mostrará una advertencia, indicando lo siguiente: “¿Está seguro de generar el siguiente contrato de alquiler? Recuerde que no podrá modificar este contrato, por favor revise bien la información del contrato antes de continuar”.
16. Al dar clic en el botón “Generar contrato” con los datos correctamente ingresados, se mostrará en otra página el contrato generado.

Tabla XXXIII. Visualizar información de un bien inmueble

Historia de Usuario	
Número: 014	Nombre de Historia de Usuario: Visualizar información de un bien inmueble.
Usuario: Arrendatario	
Modificación de Historia Número: 2	Interacción Asignada: 4
Prioridad en el negocio (Alta/Media/Baja): Alta	Puntos estimados: 2
Riesgo en el desarrollo (Alta/Media/Baja): Alta	Puntos reales: 10
Descripción: Los usuarios podrán visualizar en la página principal, la información de los bienes inmuebles públicos, como: imagen, nombre, ubicación, descripción del inmueble, el precio de alquiler.	
Observaciones:	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. Los usuarios podrán visualizar una lista de bienes inmuebles públicos en la pantalla principal con la siguiente información: imagen, nombre, ubicación, descripción del inmueble, el precio de alquiler. 2. Al dar clic en el botón “Ver inmueble” del anuncio publicado, podrán acceder a la información más detallada de dicho bien inmueble seleccionado. 3. Mostrar la información de dicho inmueble por separado en una nueva sección. 	

Tabla XXXIV. Gestión de solicitudes de reserva de alquiler solicitadas

Historia de Usuario	
Número: 015	Nombre de Historia de Usuario: Gestionar solicitudes de reserva de alquiler solicitadas.
Usuario: Arrendatario	
Modificación de Historia Número: 2	Interacción Asignada: 4
Prioridad en el negocio (Alta/Media/Baja): Alta	Puntos estimados: 2
Riesgo en el desarrollo (Alta/Media/Baja): Alta	Puntos reales: 10
Descripción: El usuario arrendatario podrá realizar una solicitud de reserva de alquiler, visualizar las visitas solicitadas y dar de baja a las solicitudes realizadas.	
Observaciones:	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. Al seleccionar el inmueble a visitar, se mostrará un botón que permita solicitar una visita a dicho bien inmueble. 2. Al dar clic en el botón solicitar una visita se deberá verificar si el usuario está autenticado, caso contrario mostrar una advertencia indicando lo siguiente: "No estás logueado". 3. Al dar clic en el botón solicitar una visita y el usuario está logueado, se mostrará la pantalla donde podrá solicitar una visita a dicho inmueble previamente seleccionado. 4. Se mostrará una sección para solicitar la visita donde se deberá ingresar la fecha de visita y la descripción de la misma. 5. Deberá existir una descripción predefinida indicando lo siguiente: "Hola, estoy interesado en su inmueble "nombre del inmueble", y me gustaría estar en contacto con usted para poder llegar a un acuerdo. Muchas gracias por su tiempo, hasta luego". 6. Validar que los campos de texto estén completados, en caso de no estar, se deberá advertir al usuario que debe completarlos. 7. Al dar clic en el botón "Solicitar visita", se registrará la visita y redirigirá al usuario a la sección de visitas solicitadas. 	

8. Mostrar una tabla con las visitas realizadas por el usuario con los siguientes campos: Estado, Fecha solicitada, Ver inmueble.
9. En el campo ver inmueble existirá un botón que permita ver la visita a más detalle en una nueva página.
10. En el campo ver inmueble existirá un botón que permita eliminar o dar de baja la visita.

Tabla XXXV. Seleccionar filtros de búsqueda

Historia de Usuario	
Número: 016	Nombre de Historia de Usuario: Seleccionar filtros de búsqueda
Usuario: Todos	
Modificación de Historia Número: 2	Interacción Asignada: 4
Prioridad en el negocio (Alta/Media/Baja): Alta	Puntos estimados: 2
Riesgo en el desarrollo (Alta/Media/Baja): Alta	Puntos reales: 10
Descripción: La aplicación debe permitir a través de una lista de selección desplegable, elegir los ítems de búsqueda que el usuario requiera para buscar el inmueble, podrá filtrar entre: tipo de inmueble, ubicación, precio.	
Observaciones:	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. En la página principal se mostrará 3 opciones para seleccionar el tipo de inmueble, precio y ubicación para poder filtrar los inmuebles en base a dichos criterios de búsqueda. 2. El primer criterio de búsqueda "Tipo de inmueble" constará de todos los tipos de inmuebles que se hayan definido. 3. En el segundo criterio tendrá rangos de precios desde <50, 50-100, 100-150, 150-200 y >200. 4. En el tercer criterio de búsqueda contendrá todas las ubicaciones previamente cargadas. 5. Al dar clic en el botón "Buscar" se filtrarán los inmuebles que cumplan con todos los filtros seleccionados. 6. Mostrar inmuebles que hayan cumplido el filtro de búsqueda. 	

Tabla XXXVI. Enviar mensajes

Historia de Usuario	
Número: 017	Nombre de Historia de Usuario: Enviar mensajes al administrador
Usuario: Arrendador/Arrendatario	
Modificación de Historia Número: 2	Interacción Asignada: 4
Prioridad en el negocio (Alta/Media/Baja): Alta	Puntos estimados: 2
Riesgo en el desarrollo (Alta/Media/Baja): Alta	Puntos reales: 10
Descripción: La aplicación debe permitir al usuario arrendador y arrendatario enviar mensajes al administrador, para ello deberá ingresar el título, asunto, y correo.	
Observaciones:	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. En la página principal se mostrará una opción “Contáctanos” la cual permitirá al usuario enviar un mensaje al administrador. 2. Mostrar un formulario solicitando la siguiente información: Título, Asunto y correo. 3. Se deberá validar que se ingrese la información en el formulario. 4. Indicar al usuario que se deben completar los campos de texto. 5. Verificar que el correo del usuario exista, en caso de no existir, indicarle que el correo no existe. 6. Si todos los datos ingresados son correctos, al dar clic en el botón “Enviar mensaje” se enviará el mensaje al administrador. 7. Notificar al usuario que su mensaje ha sido enviado. 	

Tabla XXXVII. Gestionar servicios básicos

Historia de Usuario	
Número: 018	Nombre de Historia de Usuario: Gestionar Servicios Básicos
Usuario: Administrador	
Modificación de Historia Número: 2	Interacción Asignada: 4
Prioridad en el negocio (Alta/Media/Baja): Alta	Puntos estimados: 2

Riesgo en el desarrollo	Puntos reales: 10
(Alta/Media/Baja): Alta	
Descripción: La aplicación debe permitir al usuario administrador gestionar los servicios básicos de los bienes inmuebles, para ello deberá ingresar el nombre del servicio.	
Observaciones:	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. En el panel administrativo en la sección de Configuraciones, en la opción Servicios, al seleccionar, se mostrará todos los Servicios Básicos registrados. 2. Debe permitir registrar nuevos servicios básicos, a través de un botón "Registrar Servicio". 3. Se debe validar que el usuario ingrese el nombre del servicio básico, en caso de no ingresar el nombre del servicio básico, deberá indicar mediante una advertencia, lo siguiente: "El nombre del servicio es obligatorio". 4. Se debe validar que el nombre del servicio básico no se repita, en caso de repetirse, deberá indicar mediante una advertencia, lo siguiente: "El servicio ya existe, ingresa uno diferente". 5. Guardar el servicio básico al dar clic en el botón "Guardar servicio". 6. Registrar el servicio básico y mostrar un mensaje notificando al usuario que se ha registrado su servicio básico. 7. Al dar clic en el botón de editar, del campo Acciones, se deberá mostrar en una nueva sección el servicio básico seleccionado. 8. Cambiar el nombre del servicio básico, en caso de no existir el nombre, se deberá indicar mediante una advertencia, lo siguiente: "El nombre del servicio es obligatorio". 9. Actualizar el servicio básico al dar clic en el botón "Guardar servicio". 10. Registrar el servicio básico y mostrar un mensaje notificando al usuario que se ha actualizado su servicio básico. 11. Al dar clic en el botón eliminar, se deberá indicar mediante una advertencia, lo siguiente: "¿Está seguro de borrar el servicio? Está a punto de borrar el servicio". 12. Al dar clic en el botón eliminar de la advertencia, se eliminará el servicio básico y se notificará al usuario mediante una advertencia. 13. Mostrar los servicios básicos actualizados en la tabla. 	

14. Se podrá buscar los servicios básicos mediante el nombre del mismo.

Estimación de Historias de Usuario

En la Tabla XXXVII se presenta el tiempo estimado para cada una de las historias de usuario considerando una semana ideal (cinco días hábiles, cada una de 8 horas), de igual forma se diferencian de acuerdo al módulo al que pertenecen.

Tabla XXXVIII. Módulos e historias de usuario

Prioridad	Módulo	Nro.	Nombre de la historia de usuario	Tiempo estimado		
				Semanas	Días	Horas
1	Gestionar usuarios	1	Registrar Usuario	0,8	4	32
		2	Editar Usuario	0,8	4	32
		3	Activar Usuario	0,4	2	16
		4	Desactivar Usuario	0,4	2	16
		5	Buscar Usuario	0,8	4	32
3	Gestionar bien inmueble	6	Registrar Inmueble	1,6	8	64
		7	Mostrar Inmueble	0,8	4	32
		8	Modificar Inmueble	1,8	9	72
		9	Buscar Inmueble	1	5	40
		10	Eliminar Inmueble	0,8	4	32
		11	Publicar Inmueble	1,6	8	64
2	Gestionar Servicios Básicos	12	Registrar Servicios Básicos	0,8	4	32
		13	Mostrar Servicios Básicos	0,6	3	24
		14	Editar Servicios Básicos	0,6	3	24
		15	Buscar Servicios Básicos	0,4	2	16
		16	Eliminar Servicios Básicos	0,8	4	32
5	Gestionar solicitudes de reserva de alquiler	17	Solicitar Reserva de Alquiler	2	10	80
		18	Mostrar Solicitudes de Reserva	1,8	9	72

		19	Filtrar Solicitudes de Reserva	1,6	8	64
		20	Aprobar/Rechazar Solicitud de Reserva	0,4	2	16
		21	Eliminar Solicitud de Reserva	0,6	3	24
6	Gestionar contratos de alquiler	22	Generar Contrato de Alquiler	2	10	80
		23	Buscar Contrato	1	5	40
		24	Mostrar Contratos	0,4	2	16
		25	Terminar Contrato	0,2	1	8
		26	Visualizar Contrato	0,2	1	8
4	Alquilar bien inmueble	27	Mostrar Inmuebles Publicados	0,4	2	16
		28	Buscar Inmueble Publicado	0,2	1	8
		29	Alquilar Inmueble	0,8	4	32
7	Gestionar Mensajes	30	Enviar mensajes al administrador	1,2	6	48
		31	Mostrar mensajes	0,4	2	16
		32	Filtrar mensajes	0,4	2	16
		33	Eliminar mensajes	0,2	1	8
Tiempo estimado total				27,8	139	1112

6.2.2. Diseño

En esta sección se elabora el documento de la arquitectura de software de la Aplicación Informática LojaHouse en base a la metodología XP y el modelo arquitectónico 4+1 [41].

Vista General del Sistema

En la Figura 6, se puede observar la vista general del sistema LojaHouse, donde se especifica dos tipos de cliente (Frontend): cliente web, el cual se desarrolló con el framework Angular empleando el patrón de diseño Modelo Vista Controlador (MVC), y la aplicación móvil que se utilizó el framework Flutter aplicando el patrón Lógica de Negocio de Componente (BLoC). Para la creación del servicio web API-REST (Backend), se aplicó el patrón de diseño MVC y se utilizó NodeJs con el framework ExpressJs, el mismo que se comunica con la base de datos MongoDB a través del

Object Document Mapping (ODM) Mongoose, de igual forma se utilizó el servicio Cloudinary el cual permite almacenar imágenes, y finalmente se configuró el servidor web API REST para el envío de notificaciones mediante el servicio de Firebase Cloud Messaging.

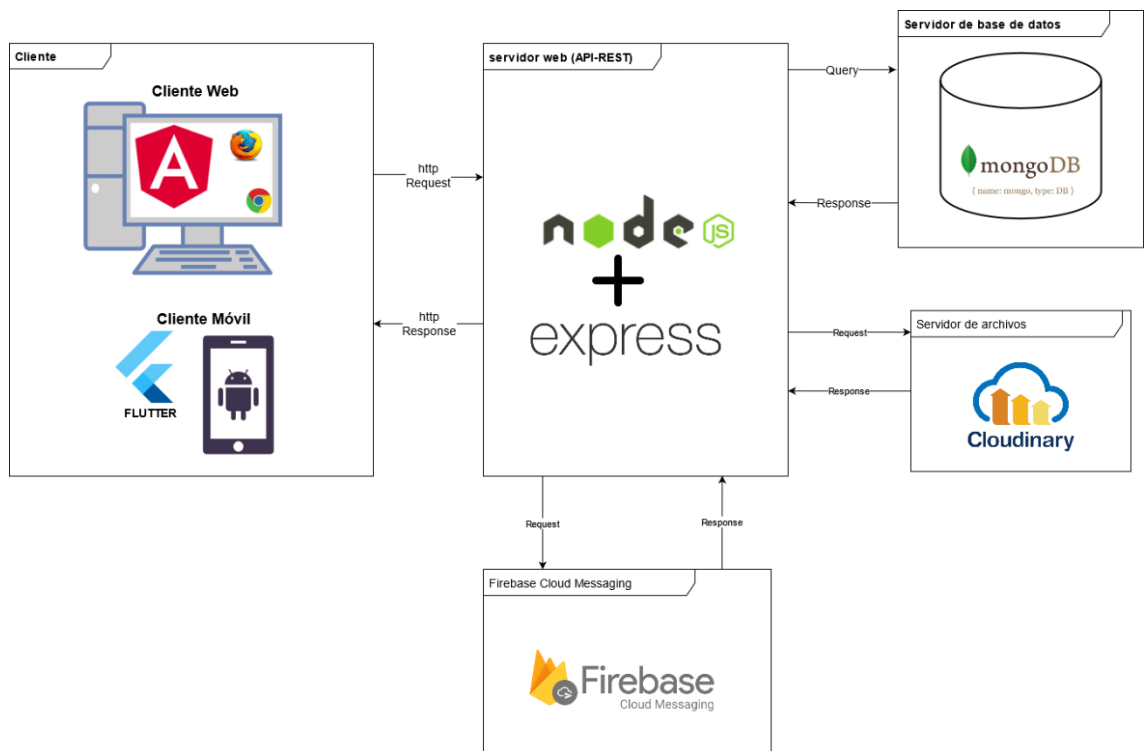


Figura 6. Vista general del sistema

Arquitectura de Software

En la Tabla XXXIX se puede visualizar como se encuentra definida la arquitectura de software, y en el Anexo 11: **Arquitectura de Software** se encuentra de forma detallada.

Tabla XXXIX. Arquitectura 4+1

Vista	Elemento Modelado	Descripción
Vista de Escenarios	Caso de Uso	Muestra la interacción que tienen los actores y los escenarios donde se va a desenvolver el sistema.
Vista Lógica	Modelo Conceptual Diagrama de Clases	Representa las funcionalidades y el servicio que proporciona a los usuarios.
Vista Física	Diagrama de Despliegue	Muestra los componentes físicos del sistema.

Vista de Despliegue	Diagrama de Componentes	Describe los componentes del sistema con el fin que el programador entienda las interacciones que existen.
Vista de Procesos	Diagrama de Actividad	Describe los procesos de la funcionalidad del sistema.

Vista de Escenarios

En la Figura 7 se puede observar el diagrama de caso del usuario no registrado, el mismo que tiene la funcionalidad de registrarse en la aplicación para poder enviar mensajes al administrador o resetear su contraseña, de igual manera podrá iniciar sesión y filtrar inmuebles desde la pantalla principal para poder visualizar la información del inmueble seleccionado.

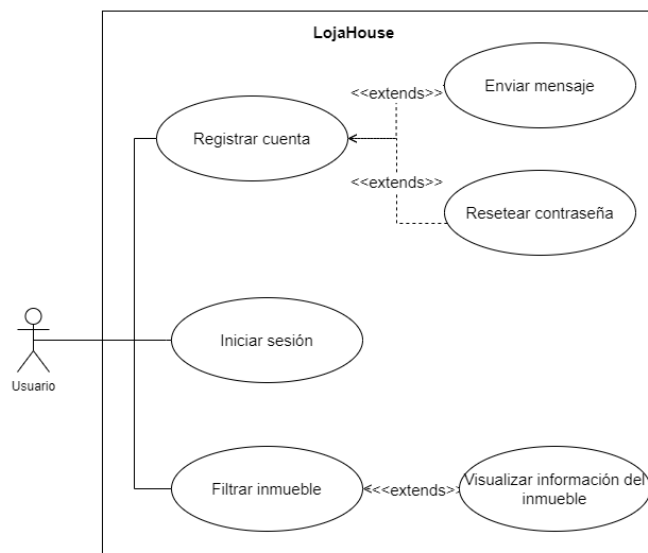


Figura 7. Diagrama de caso de uso - Usuario no registrado.

En la Figura 8 se puede observar el diagrama de caso de uso del usuario arrendador, el mismo que tiene la funcionalidad de gestionar bienes inmuebles, publicarlos, filtrar solicitudes de visita, aceptar o rechazar las mismas, alquilar un bien inmueble, generar un contrato de alquiler y gestionar los contratos.

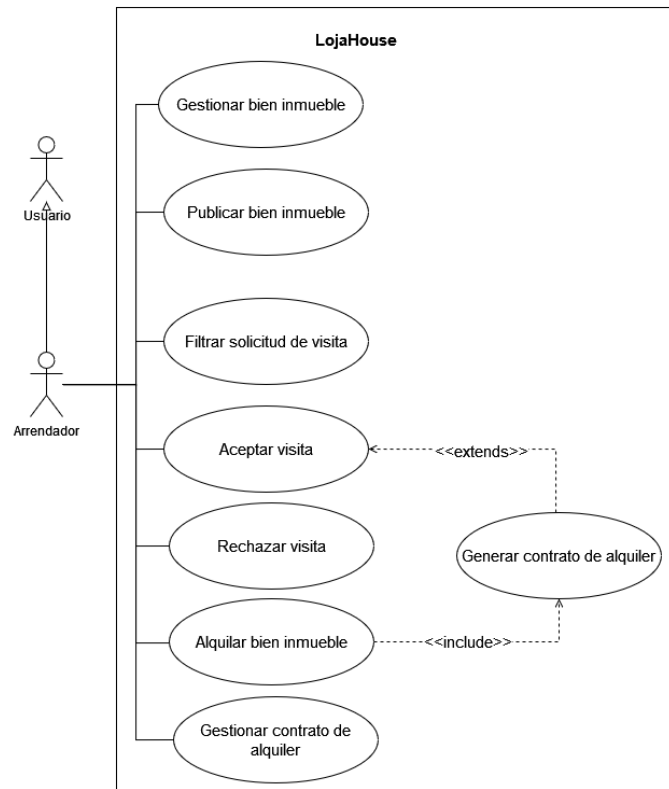


Figura 8. Diagrama de caso de uso - Usuario Arrendador

En la Figura 9 se puede observar el diagrama de caso de uso del usuario arrendatario, el mismo que tiene la funcionalidad de generar solicitud de alquiler, visualizarlas o eliminarlas; también puede visualizar y aceptar un contrato de alquiler.

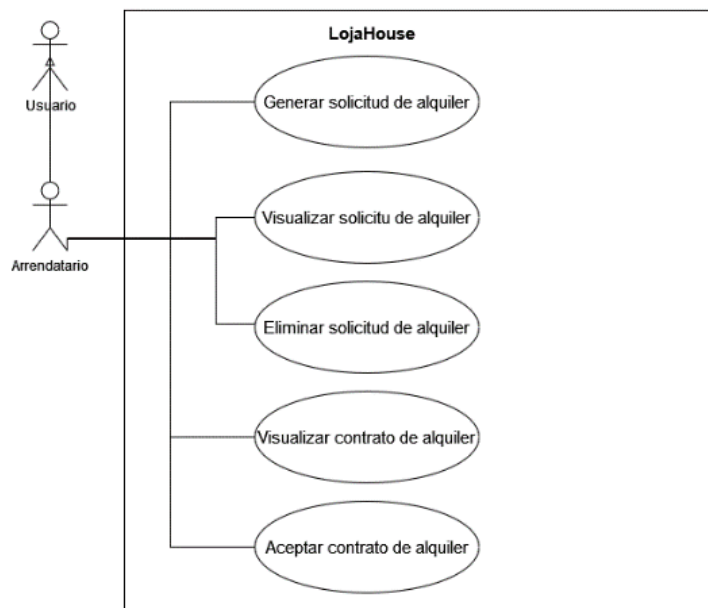


Figura 9. Diagrama de caso de uso - Usuario Arrendatario

En la Figura 10 se puede observar el diagrama de caso de uso del usuario administrador, el mismo que tiene la funcionalidad de iniciar sesión, gestionar los servicios básicos, gestionar los mensajes, activar o desactivar usuarios.

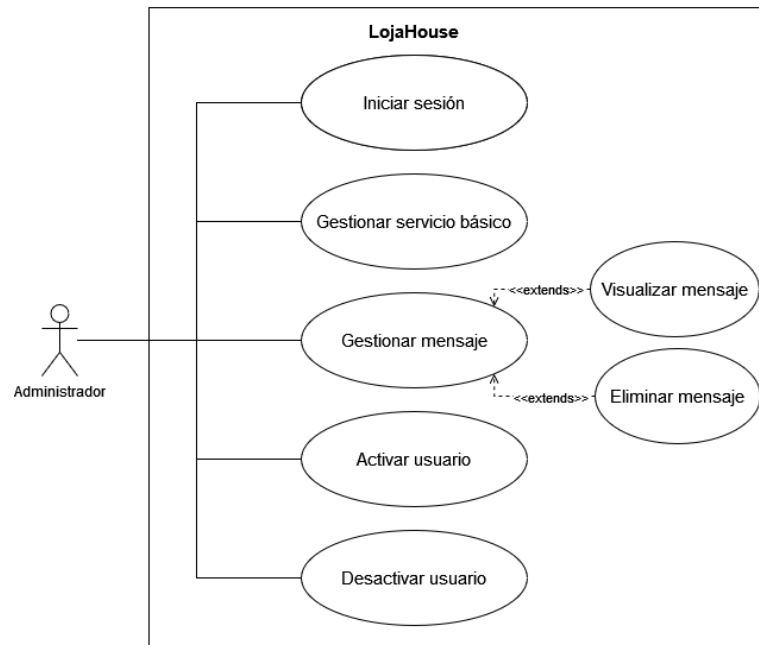


Figura 10. Diagrama de caso de uso - Usuario Administrador

Vista Lógica

En la Figura 11 se puede observar el modelo conceptual, la cual proporciona el entendimiento y solución relacionado a la aplicación informática.

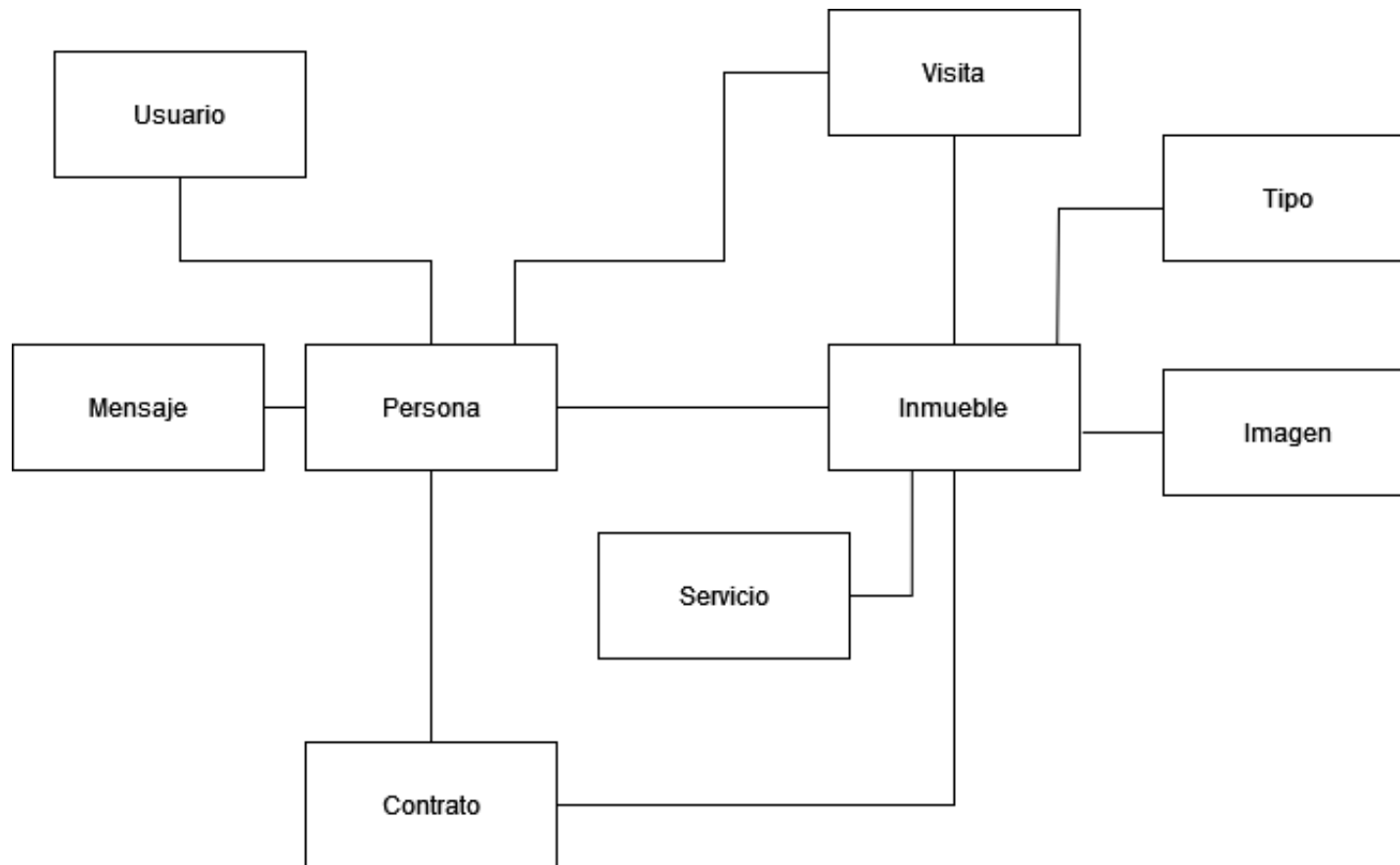


Figura 11. Modelo conceptual.

En la Figura 12 se puede observar el diagrama de clases la cual indica la estructura y el comportamiento de las diferentes clases.

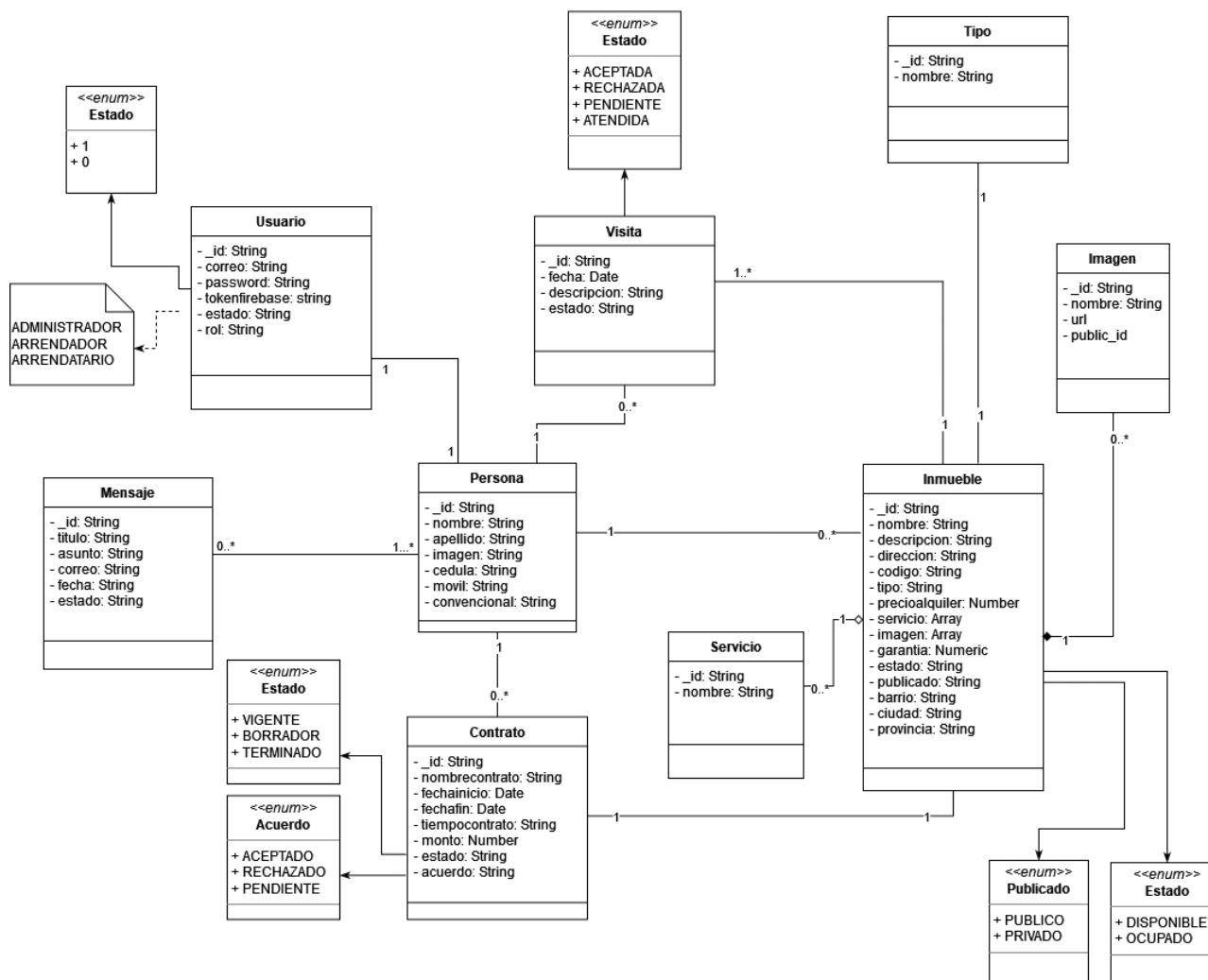


Figura 12. Diagrama de clases

Vista de Procesos

En esta vista se realizó los diagramas de actividad que muestran la interacción entre el usuario y el sistema, en la Figura 13 se observa el diagrama de actividad para publicar un inmueble, especificando los pasos para su correcta ejecución. En el Anexo 11 **Arquitectura de Software** se encuentran detallados todos los diagramas de procesos de la aplicación LojaHouse.

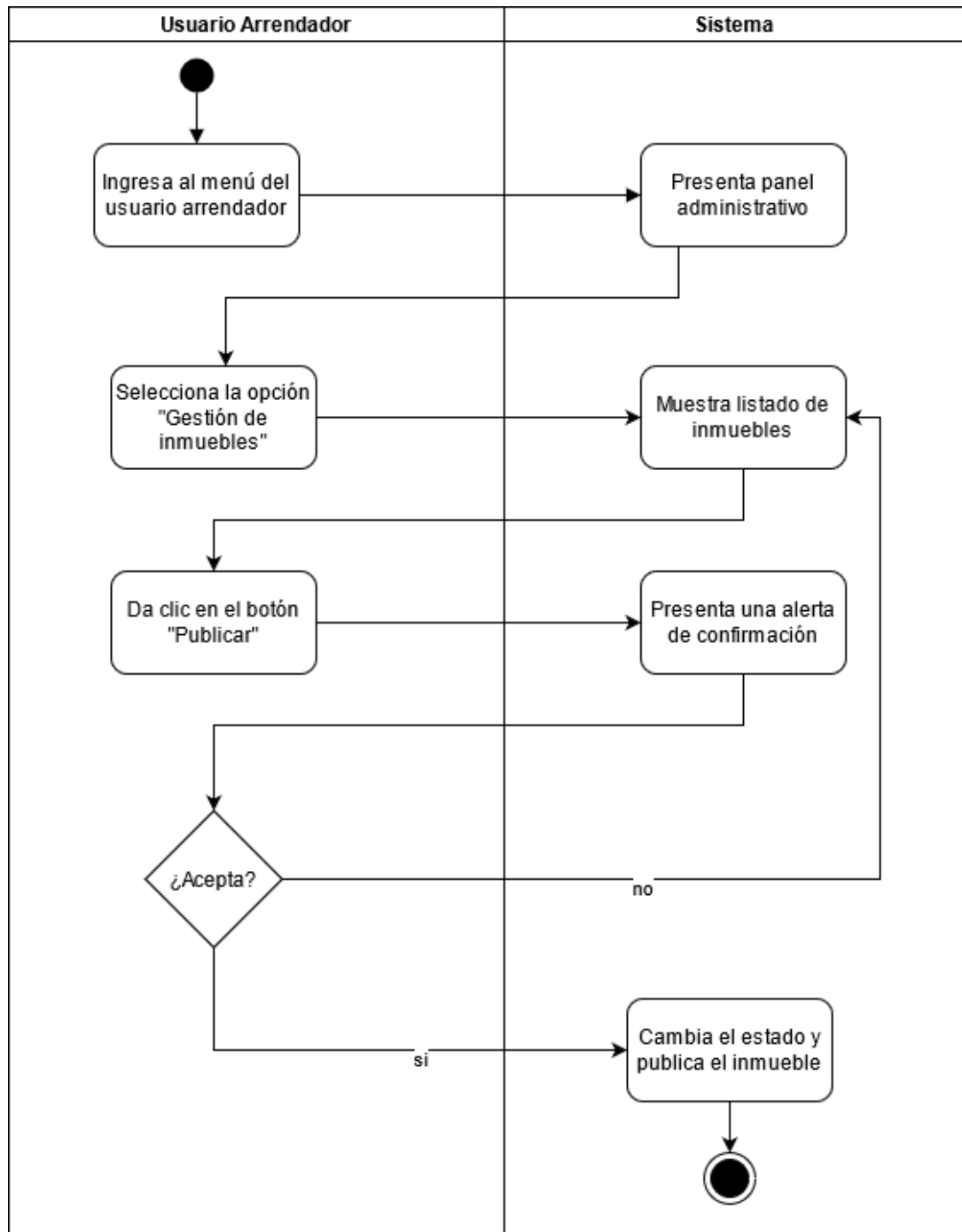


Figura 13. Diagrama de actividades - Publicar bien inmueble

Vista de Despliegue

En la figura 14 se puede observar la organización de los diferentes componentes de la Aplicación Informática LojaHouse.

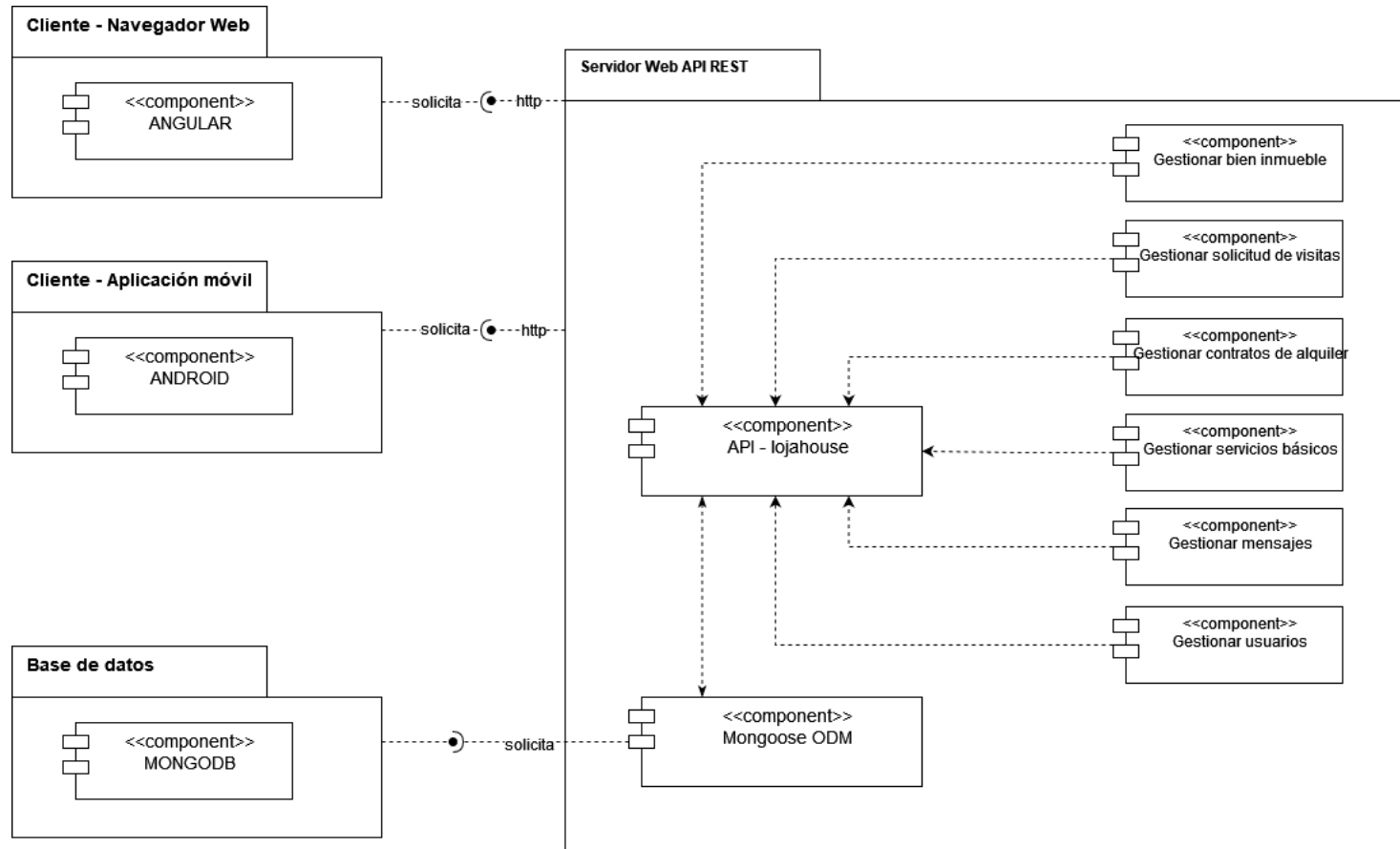


Figura 14. Diagrama de componentes

Vista Física

En la figura 15 se puede observar el diagrama de despliegue de la aplicación LojaHouse, el cual consta de dos clientes, el navegador web y el dispositivo móvil, mismos que se comunican con el servidor web API REST.

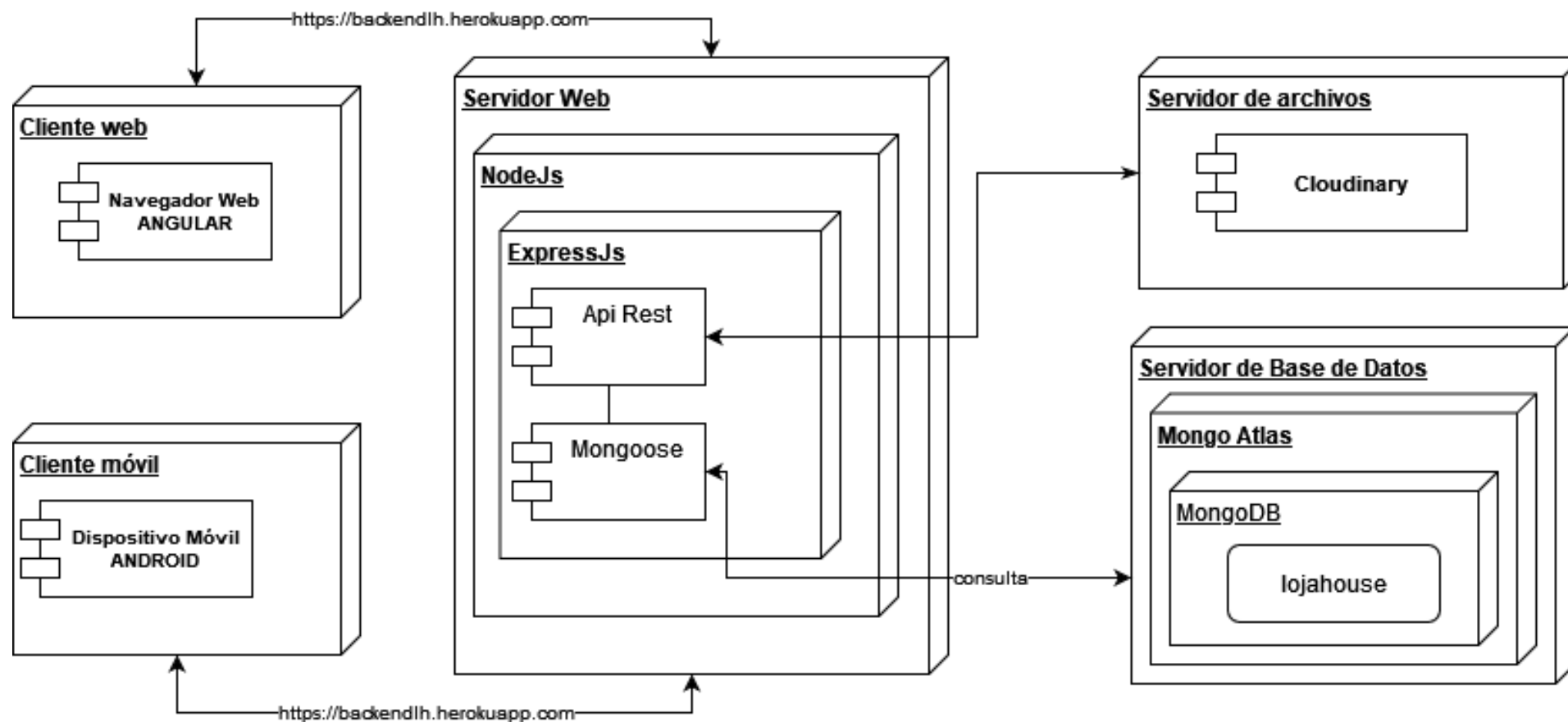


Figura 15. Diagrama de despliegue de la aplicación LojaHouse

Arquitectura de la Aplicación

La arquitectura utilizada es Cliente/Servidor para lo cual se compone de dos clientes (web y móvil) y el servicio web API REST, ver Figura 16.

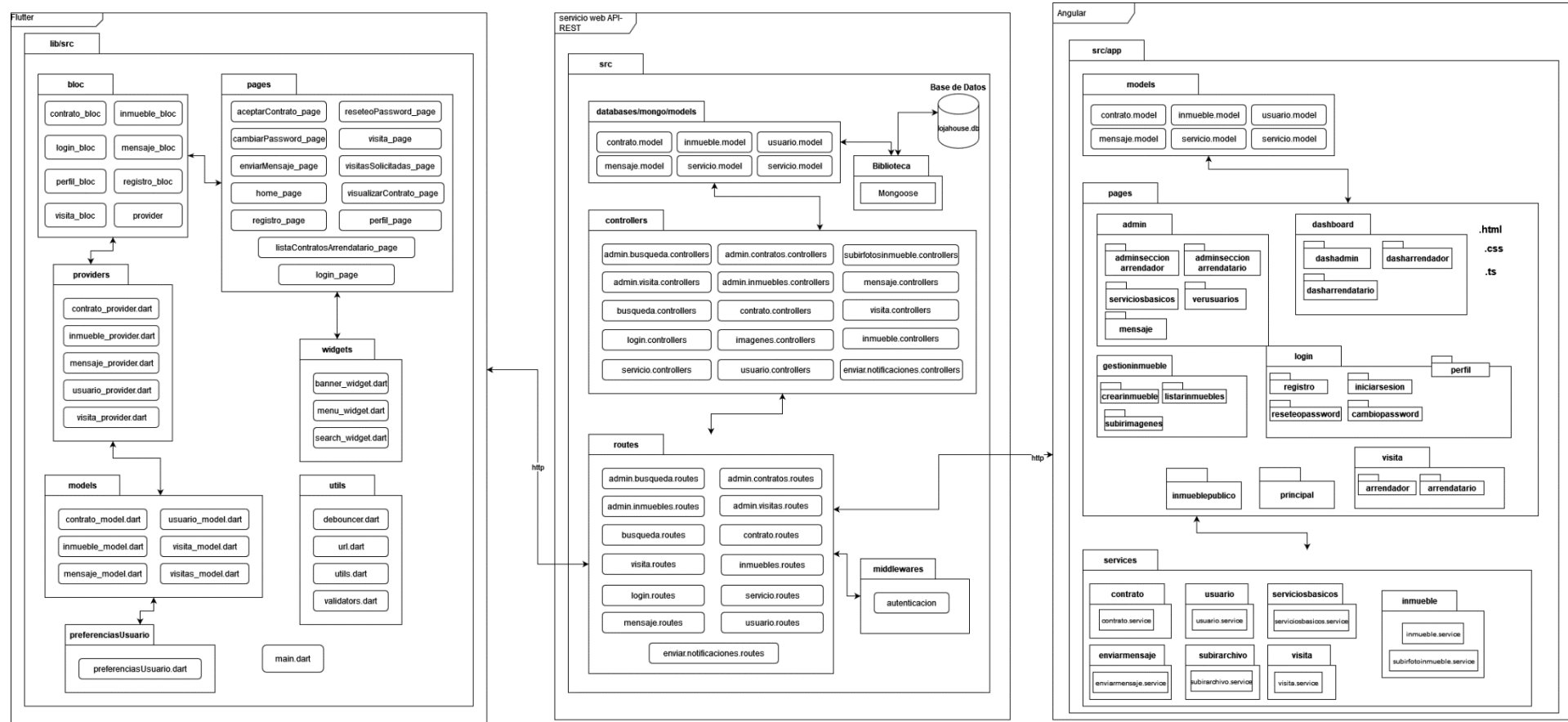


Figura 16. Arquitectura lógica de la aplicación LojaHouse

Prototipos iniciales de las Interfaces de Usuario de la Aplicación Informática

Se realizó una versión inicial mediante el uso del software Pencil para la creación de prototipos de la interfaz, misma que permitió tener una idea global de cómo quedaría la aplicación para posteriormente iniciar la fase de implementación. A continuación se muestra en la Figura 17 el diseño inicial de la pantalla principal. En el Anexo 5 **Prototipos iniciales de interfaces** se encuentra más detallado.

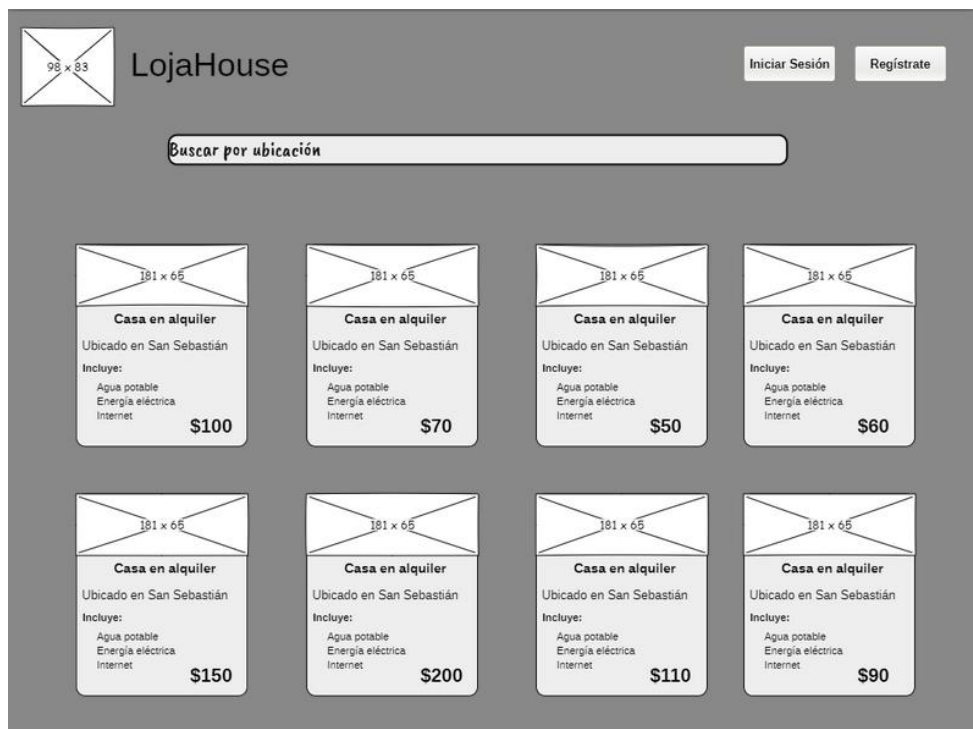


Figura 17. Página principal

6.2.3. Codificación

En esta sección se codificó la aplicación LojaHouse, utilizando un stack de tecnologías basadas en el lenguaje de programación JavaScript denominado Stack MEAN (MongoDB, Express, Angular, Node) y para la parte móvil se utilizó el framework Flutter y el lenguaje de programación Dart.

Para el desarrollo del backend de la Aplicación LojaHouse, se utilizó el patrón de diseño Modelo Vista Controlador (MVC), en la Figura 18 se muestra la estructura del servicio web API REST, de igual manera en la Figura 19 se muestra el patrón MVC utilizado.

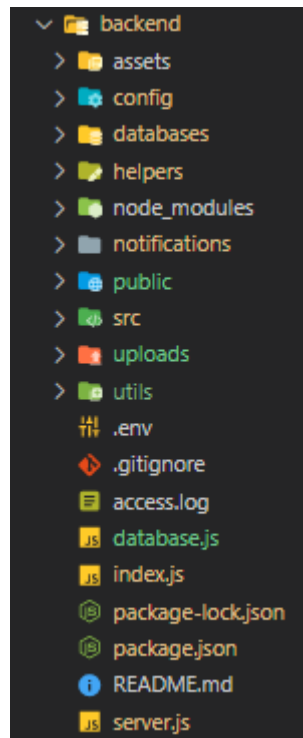


Figura 18. Estructura del Servicio Web API-REST

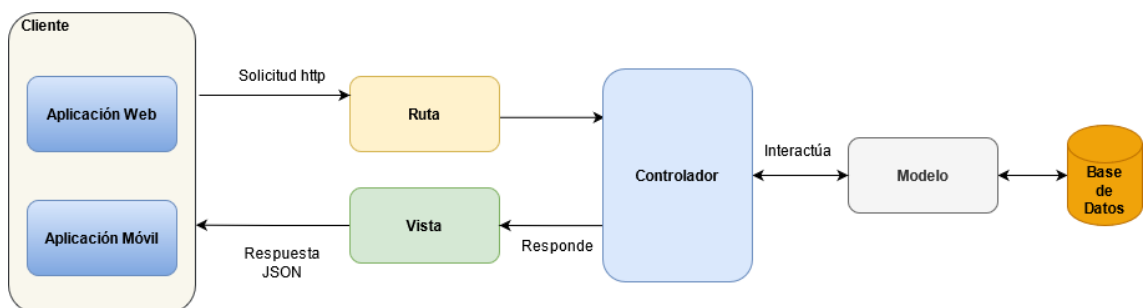


Figura 19. Patrón de diseño Modelo Vista Controlador

- La ruta se encarga de reenviar las solicitudes admitidas hacia el controlador apropiado.

La Figura 20 muestra la definición de las rutas para el módulo de visitas, las mismas que redirigen la petición hacia el controlador específico.

```

const express = require('express');
var mdwVerificarToken = require('../middlewares/autenticacion');
const router = express();

> const { ...
} = require('../controllers/visita.controllers')

//SE OBTIENE SOLO LAS VISITAS A LOS INMUEBLES DEL ARRENDADOR
router.get('/obtenervisitas/:desde', mdwVerificarToken.verificaToken, obtenerVisitas);
//SE OBTIENE SOLO LAS VISITAS A LOS INMUEBLES DEL ARRENDADOR, este es para el dashboard
router.get('/obtenervisitas-contador', mdwVerificarToken.verificaToken, obtenerSolicitudVisitasPendientes);

//SE OBTIENE SOLO LAS VISITAS A LOS INMUEBLES DEL ARRENDADOR-ADMINISTRADOR
router.get('/administrador/arrendador/visitas/:desde', mdwVerificarToken.verificaToken, obtenerVisitasArrendatarioAdministrador);

//CRUD ARRENDADOR (VISITA)
router.post('/crearvisita', mdwVerificarToken.verificaToken, crearVisita);
router.put('/actualizarvisita/:id', mdwVerificarToken.verificaToken, actualizarVisita);
router.put('/eliminarvisita/:id', mdwVerificarToken.verificaToken, eliminarVisita);

//RUTAS DEL USUARIO ARRENDADOR PARA ACEPTAR LAS VISITAS
router.put('/arrendador/aceptarvisita/:id', aceptarVisita);

//RUTA PARA OBTENER TODAS LAS VISITAS QUE EL USUARIO ARRENDATARIO HA SOLICITADO
router.get('/arrendatario/visitasolicitada/:desde', mdwVerificarToken.verificaToken, obtenerVisitasSolicitadas);
router.get('/arrendatario/visitasolicitadamovil', mdwVerificarToken.verificaToken, obtenerVisitasSolicitadasMovil);
//RUTAS DEL USUARIO ARRENDATARIO PARA LAS VISITAS
router.get('/arrendatario/obtenervisita/:id', mdwVerificarToken.verificaToken, obtenerVisitaEspecificaArrendatario);
router.get('/arrendador/obtenervisita/:id', mdwVerificarToken.verificaToken, obtenerVisitaEspecificaArrendador);

module.exports = router;

```

Figura 20. Rutas para el módulo visita

- El modelo contiene la representación de los datos del dominio de la aplicación, es decir, las clases.

La Figura 21 muestra la definición del modelo para el módulo de visitas utilizando el ODM Mongoose, el cual permite crear el esquema de la base de datos, además permite realizar validaciones y crear consultas.

```

1  let mongoose = require('mongoose');
2
3  let Schema = mongoose.Schema;
4
5
6  let estadosValidos = {
7    values: ['ACEPTADA', 'RECHAZADA', 'PENDIENTE', 'ATENDIDA', 'ELIMINADA'],
8    message: '{VALUE} no es un estado permitido'
9  }
10
11 let visitaSchema = new Schema({
12   fecha: {type: Date, required: [true, 'La fecha es necesaria']},
13   descripcion: {type: String, required: [true, 'La descripción es necesaria']},
14   estado: {type: String, required: false, default: 'PENDIENTE', enum: estadosValidos},
15   inmueble: {type: Schema.Types.ObjectId, ref: 'Inmueble'},
16   usuarioarrendatario: {type: Schema.Types.ObjectId, ref: 'Usuario'},
17 }, {timestamps: true, versionKey: false}, {collection: 'visitas'});
18
19 module.exports = mongoose.model('Visita', visitaSchema);

```

Figura 21. Modelo para el módulo visita

- El controlador es el que interactúa con el modelo para obtener los datos solicitados y representarlos a través de la vista.

En la Figura 22 se muestra el código del controlador de aceptar visita, aquí es donde se procesan los datos de la solicitud y recuperará la información necesaria del modelo para luego entregárselos a la vista para que los procese el cliente.

```
aceptarVisita: async (req, res) => {
  let id = req.params.id;
  const { estado } = req.body;
  await visitaModel.findById(id, (err, visita) => {
    if (err) {
      return res.status(500).json({
        ok: false,
        mensaje: "Error al buscar visita",
        errors: err,
      });
    }
    if (!visita) {
      return res.status(400).json({
        ok: false,
        mensaje: "El visita con el id: " + id + " no existe",
        errors: { message: "No existe un visita con ese ID" },
      });
    }
    visita.estado = estado;
    visita.save((err, visitaGuardado) => {
      if (err) {
        return res.status(400).json({
          ok: false,
          mensaje: "Error al aceptar la visita",
          errors: err,
        });
      }
      res.status(200).json({
        ok: true,
        visita: visitaGuardado,
        mensaje: `La visita está: ${estado}`,
      });
    });
  });
};
```

Figura 22. Controlador para aceptar una visita

- La vista es utilizada por el controlador para representar los datos, en este caso la respuesta en formato JSON.

En la Figura 23 se muestra la respuesta que se envía desde el controlador en formato JSON para que el cliente pueda procesarla.

```
if (err) {
  return res.status(400).json({
    ok: false,
    mensaje: "Error al aceptar la visita",
    errors: err,
  });
}
res.status(200).json({
  ok: true,
  visita: visitaGuardado,
  mensaje: `La visita está: ${estado}`,
});
```

Figura 23. Respuesta en formato JSON para aceptar una visita

Diseño final de Interfaces de Usuario de la Aplicación Informática

En las Figuras 24 se puede observar el diseño de interfaz de la pantalla principal de la aplicación web LojaHouse, en el Anexo 12 **Prototipo final de interfaz** se encuentra más detallado.

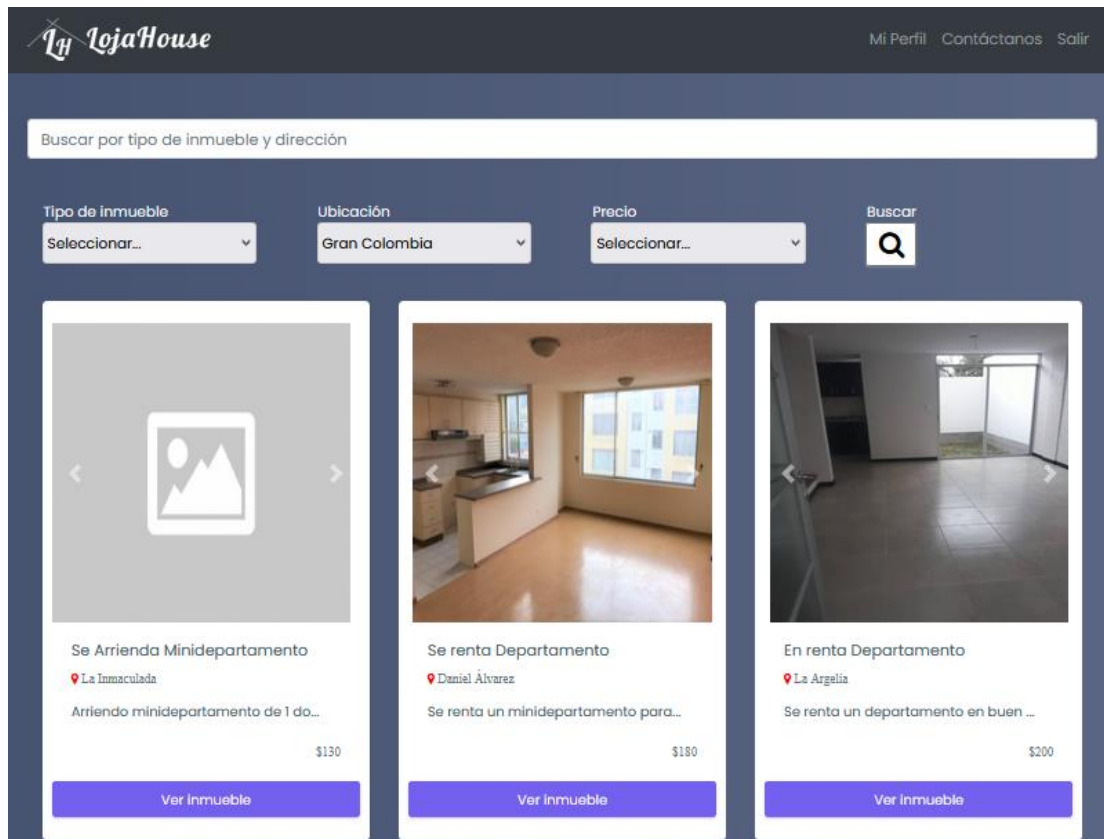


Figura 24. Página principal

6.3. Objetivo 3: Evaluar la funcionalidad y usabilidad de la Aplicación Informática en un ambiente controlado o simulado.

En esta sección se describe la cuarta fase de la metodología XP, la fase de pruebas del sistema, para dar cumplimiento al tercer objetivo del presente TT.

6.3.1. Pruebas al Servicio Web API REST

Para las pruebas del API REST se utilizó la herramienta Postman, la cual permitió el envío de peticiones http (get, put, delete, post) hacia el servicio web API REST, obteniendo una respuesta en formato json, el mismo que permitió verificar el correcto funcionamiento del Backend e identificar el tiempo que demora la respuesta en cada petición realizada.

En la Tabla XL se muestra los test realizados donde se utilizó la herramienta Postman considerando los siguientes parámetros: url, token, método, resultado y tiempo de respuesta de la petición, para ello se realizó el test a las peticiones principales del API REST.

Tabla XL. Pruebas unitarias servicio web api rest – postman

URL	Token	Método	Resultado	Tiempo de respuesta
localhost:3000/usuario/crearusuario	No	POST	Permite registrar la cuenta del usuario.	2160 ms
http://localhost:3000/login	No	POST	Permite iniciar sesión.	238 ms
http://localhost:3000/usuario/resetpassword	No	PUT	Permite resetear el password.	1860 ms
http://localhost:3000/mensaje/crearmensaje	No	POST	Permite enviar un mensaje	273 ms
http://localhost:3000/búsqueda/coleccion/inmuebles/CASA/Gran Colombia/150-200	No	GET	Permite filtrar bienes inmuebles	297 ms
localhost:3000/inmueble/obtenerinmueble/public	No	GET	Permite visualizar información del inmueble	241 ms

o/60a88386fdb902cfc00fc22				
localhost:3000/inmueble/crearinmueble	Si	POST	Permite crear un bien inmueble	244 ms
localhost:3000/inmueble/desactivarinmueble/60ed2b0f064bcc3ba8f21378	Si	PUT	Permite publicar un bien inmueble	578 ms
localhost:3000/visita/crearvisita	Si	POST	Permite generar una solicitud de alquiler	250 ms
localhost:3000/inmueble/desactivarinmueble/60ed2b0f064bcc3ba8f21378	Si	PUT	Permite aceptar una visita de alquiler	239 ms
localhost:3000/contrato/crearcontrato	Si	POST	Permite generar un contrato de alquiler	357 ms
localhost:3000/contrato/acuerdo/60ed32a0db551b3bbcd02525/aceptar	Si	POST	Permite aceptar un contrato de alquiler	411 ms

En la Figura 25 se muestra la respuesta en formato Json de una petición POST a la URL para crear inmuebles. En el Anexo 8: **Pruebas al Servicio Web API REST** se encuentra más detallado.

The screenshot shows a Postman interface for a POST request. The URL is localhost:3000/inmueble/crearinmueble?token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c3Vhc... The response body is displayed in JSON format:

```

1 {
2   "ok": true,
3   "inmueble": {
4     "direccion": "Av. universitaria",
5     "codigo": "453",
6     "tipo": "departamento",
7     "servicio": "agua, luz, internet",
8     "precioalquiler": "100",
9     "garantia": "100",
10    "usuario": "60df23774f45bb4f84a03943",
11    "barrio": "Sauces Norte",
12    "ciudad": "Loja",
13    "provincia": "Loja"
14  },
15   "mensaje": "El inmueble ha sido creado"
16 }

```

Figura 25. Herramienta Postman, petición POST Crear Inmueble

6.3.2. Pruebas de Caja Negra

Las pruebas de caja negra se llevaron a cabo en base a los criterios de aceptación de las Historias de Usuario, utilizando una lista de chequeo para determinar si el software cumple o no con los requisitos de software. Las pruebas de caja negra se encuentran detalladas en el Anexo 9 **Pruebas de Caja Negra**, con su respectiva aprobación.

La Tabla XLI muestra el ejemplo para el caso de prueba de caja negra para el registro de un inmueble.

Tabla XLI. Casos de prueba de caja negra - crear inmueble

Caso de prueba de Caja Negra 1			
Historia de Usuario 1: Registrar bienes inmuebles			
Criterios de Aceptación	Cumple		Observación
	Si	No	
Mostrar el formulario de registro de un nuevo inmueble a través del botón Nuevo Inmueble de la pantalla gestión de inmuebles.	X		
Cargar las opciones de tipo de inmueble como: casa, departamento, cuarto, mini-departamento en la pantalla de registro de nuevos inmuebles.	X		
Seleccionarlos servicios que incluye un bien inmueble.	X		
Validar campos obligatorios.	X		
Mostrar un mensaje indicando que hay campos obligatorios vacíos.	X		
Controlar si el bien inmueble posee garantía, si no se ingresa un monto, por defecto será 0.	X		
Guardar el inmueble al dar clic en el botón "Guardar inmueble".	X		
Registrar el bien inmueble y mostrar un mensaje notificando al usuario que se ha registrado su inmueble.	X		
Habilitar el botón para cargar imágenes del inmueble guardado y al dar clic se deberá redirigir a otra pantalla donde se podrá subir las imágenes del inmueble.	X		
Cargar imágenes al bien inmueble.	X		
Quitar imágenes (de ser necesario) del bien inmueble.	X		
Mostrar un mensaje indicando lo siguiente: "Imágenes cargadas exitosamente".	X		

Con la ejecución de las pruebas realizadas por parte del ing. Wilman Patricio Chamba Zaragocín, encargado de la revisión y aprobación del documento de pruebas se identificaron funcionalidades que se habían omitido y mediante las mismas se lograron completar, concluyendo de esta manera con el cumplimiento de todos los criterios de aceptación descritos en los diferentes casos de prueba de caja negra.

Pruebas de carga y estrés

El backend de la aplicación LojaHouse se encuentra alojado en los servidores de Heroku, y el frontend en los servidores de Firebase Hosting, de google. Para conocer el nivel de carga que puede soportar la aplicación y su rendimiento, se realizó pruebas de carga y estrés.

En la Figura 26 se muestra la configuración del grupo de hilos, donde se define las peticiones de tipo GET y POST que se estima podrían producir cuellos de botella, para ello se configuró 1000 peticiones para un periodo de un segundo que se ejecutarán de manera concurrente.

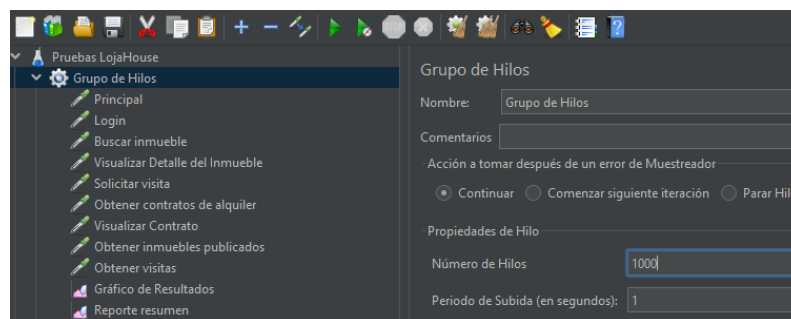


Figura 26. Configuración de peticiones

En el resultado que se muestra en la Figura 27, se obtuvo un valor para la media de 4603, lo cual representa el tiempo promedio en milisegundos para el conjunto de resultados, cumpliendo de esta forma el requerimiento no funcional acerca del tiempo de respuesta de la aplicación que debe ser inferior o igual a 5 segundos. Por otro lado, en la Figura 28 se puede observar el resultado en forma de resumen, lo cual permite conocer el error obtenido en la prueba, en este caso se obtuvo un porcentaje de 6.36%, es decir, la aplicación LojaHouse tiende a fallar más al realizar peticiones concurrentes en la página principal, por otro lado, la página de login tiende a fallar en menor medida. Por lo que se puede evidenciar que al ejecutar 1000 peticiones concurrentes presenta un rendimiento de 70,4/sec, por lo que se concluye que la aplicación a medida que incrementan los usuarios tiende a fallar, aun así se obtuvo tiempos de respuesta mínimos de 79 milisegundos y como máximo de 42418 milisegundos.

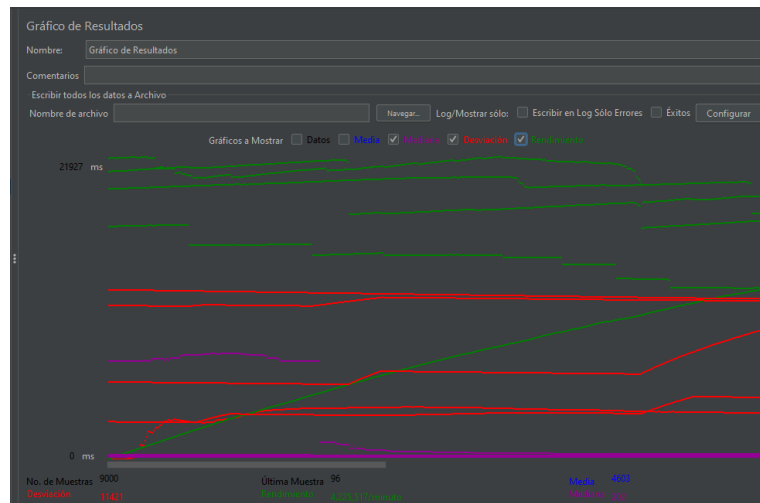


Figura 27. Gráfico de resultados de la prueba de carga y estrés

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 Configurar

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Byt...
Principal	1000	29284	7079	42418	14185,40	47,90%	23,6/sec	70,72	1,48	3072,2
Login	1000	9151	278	42124	13774,98	9,20%	12,9/sec	40,66	2,37	3222,2
Buscar inmu...	1000	1540	80	42081	5417,05	0,10%	8,4/sec	26,45	1,01	3222,2
Visualizar De...	1000	221	80	4717	312,69	0,00%	8,4/sec	26,40	1,01	3222,5
Solicitar visita	1000	472	278	4274	331,49	0,00%	8,4/sec	26,59	4,38	3252,0
Obtener con...	1000	197	80	3047	255,77	0,00%	8,4/sec	26,44	1,01	3222,5
Visualizar Co...	1000	212	79	2825	267,28	0,00%	8,4/sec	26,45	1,01	3222,6
Obtener inm...	1000	183	81	3141	195,45	0,00%	8,4/sec	26,46	1,01	3222,6
Obtener visitas	1000	168	80	2357	182,83	0,00%	8,4/sec	26,47	1,01	3222,6
Total	9000	4603	79	42418	11421,85	6,36%	70,4/sec	220,69	11,65	3208,8

Figura 28. Reporte de resumen de resultados de la ejecución de la prueba de carga y estrés

El documento de pruebas de carga y estrés se encuentra detallado en el **Anexo 10: pruebas de carga y estrés**.

6.3.3. Pruebas de Usabilidad

Para las pruebas de usabilidad de la aplicación LojaHouse se utilizó el método del experimento (Ver Anexo13: **Desarrollo del experimento**) y el método empírico de la encuesta (Se utilizó una encuesta estandarizada Cuestionario de Usabilidad del Sistema Posterior al Estudio (PSSUQ)); realizando así la validación académica de acuerdo a [42], en donde se define que la validación académica se lo realiza en un entorno experimental, usualmente utilizando estudiantes como sujetos.

El resultado de cada pregunta de la encuesta realizada acerca de las pruebas de usabilidad de la aplicación LojaHouse es la siguiente:

1. En general, estoy satisfecho con lo fácil que es utilizar este sistema

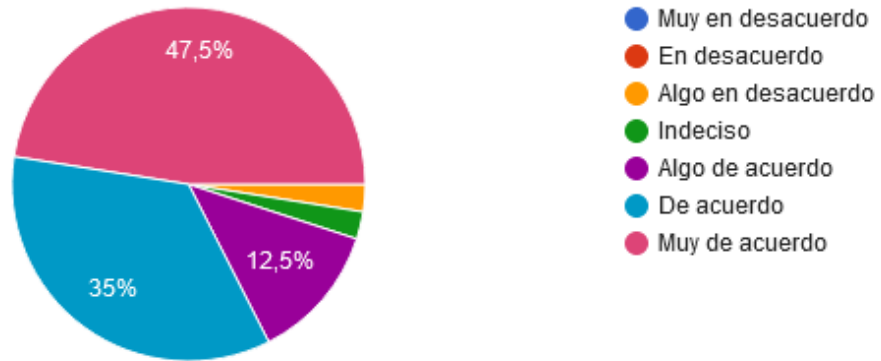


Figura 29. Prueba de Usabilidad - Pregunta 1

2. Fue sencillo utilizar este sistema

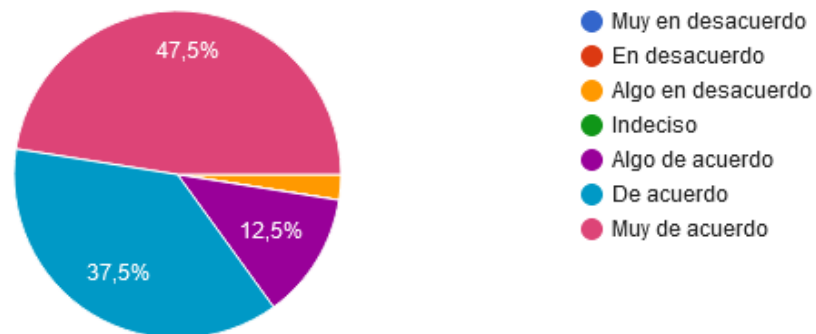


Figura 30. Prueba de Usabilidad - Pregunta 2

3. Pude completar las tareas y escenarios rápidamente usando este sistema

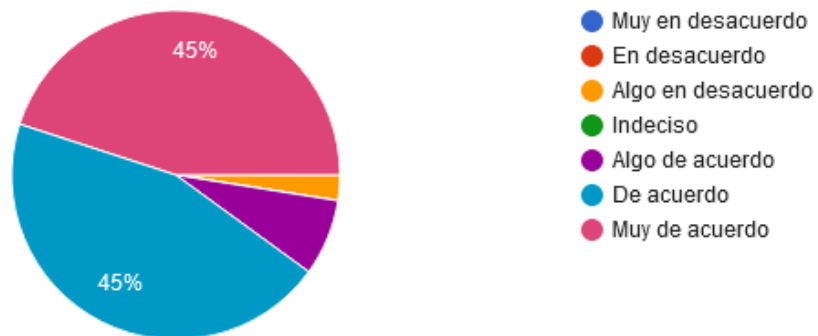


Figura 31. Prueba de Usabilidad - Pregunta 3

4. Me sentí cómodo usando este sistema

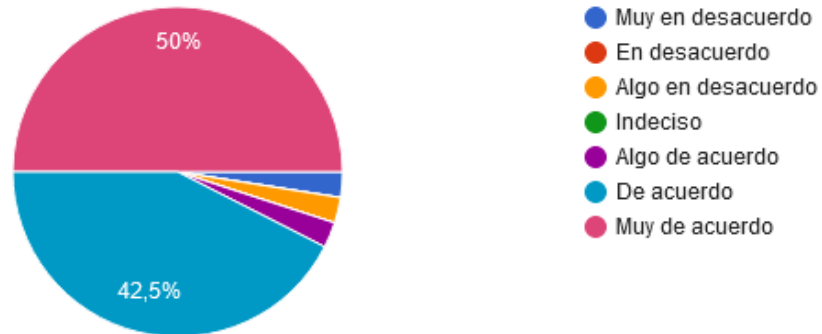


Figura 32. Prueba de Usabilidad - Pregunta 4

5. Fue fácil aprender a usar este sistema

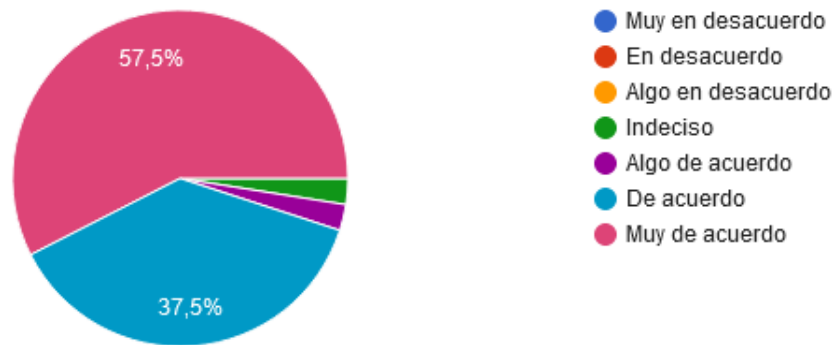


Figura 33. Prueba de Usabilidad - Pregunta 5

6. Creo que podría volverme productivo rápidamente usando este sistema

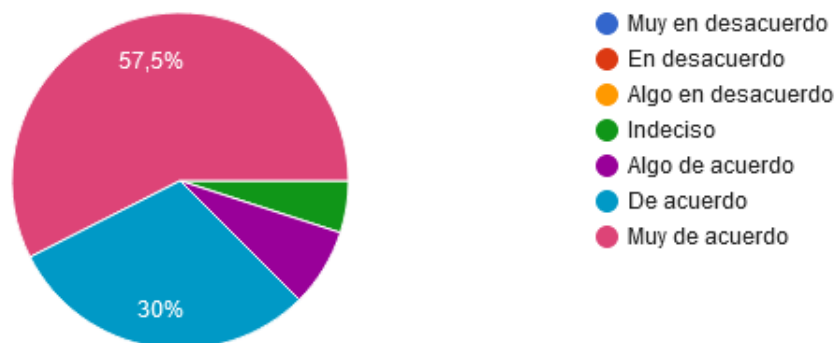


Figura 34. Prueba de Usabilidad - Pregunta 6

7. El sistema dio mensajes de error que me indicaban claramente cómo solucionar problemas

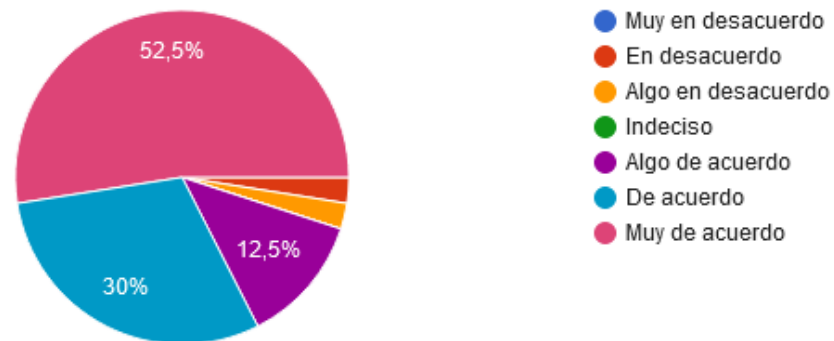


Figura 35. Prueba de Usabilidad - Pregunta 7

8. Siempre que cometía un error al utilizar el sistema, podía recuperarme fácil y rápidamente.

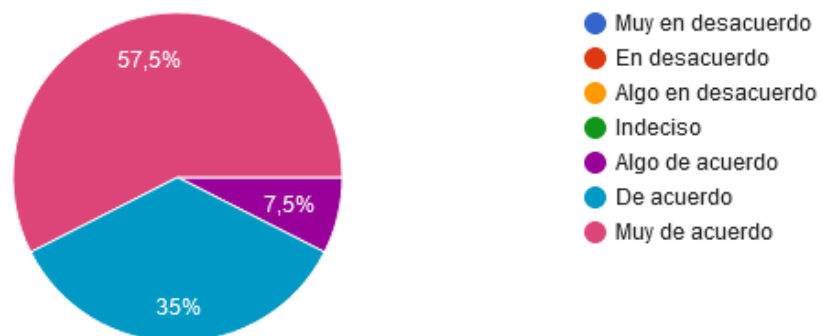


Figura 36. Prueba de Usabilidad - Pregunta 8

9. La información (como ayuda en línea, mensajes en pantalla y otra documentación) proporcionada con este sistema era clara.

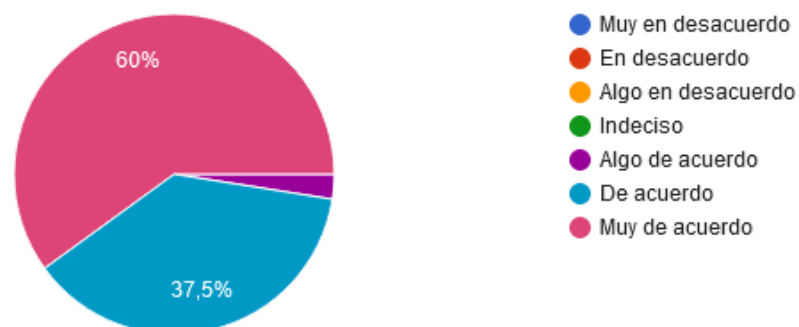


Figura 37. Prueba de Usabilidad - Pregunta 9

10. Fue fácil encontrar la información que necesitaba.

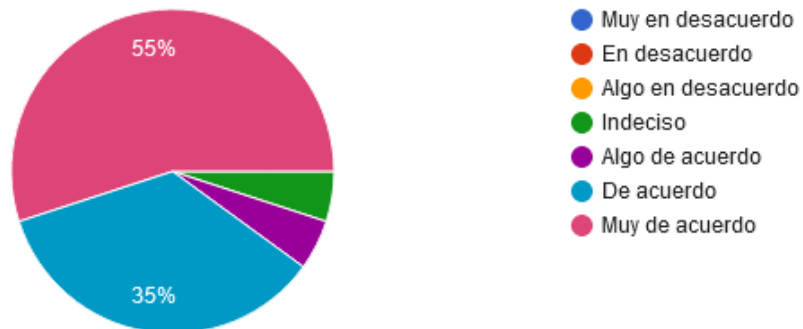


Figura 38. Prueba de Usabilidad - Pregunta 10

11. La información fue eficaz para ayudarme a completar las tareas y los escenarios.

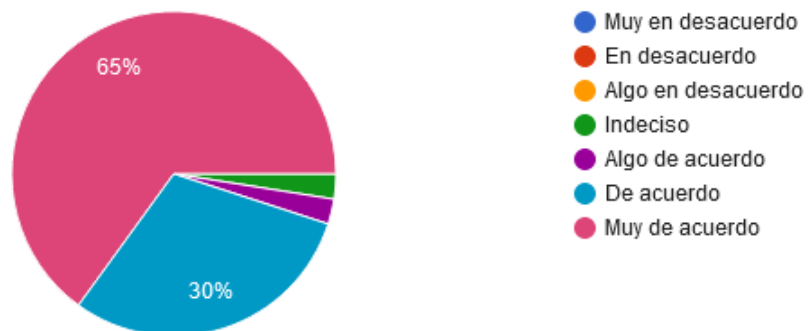


Figura 39. Prueba de Usabilidad - Pregunta 11

12. La organización de la información en las pantallas del sistema fue clara.

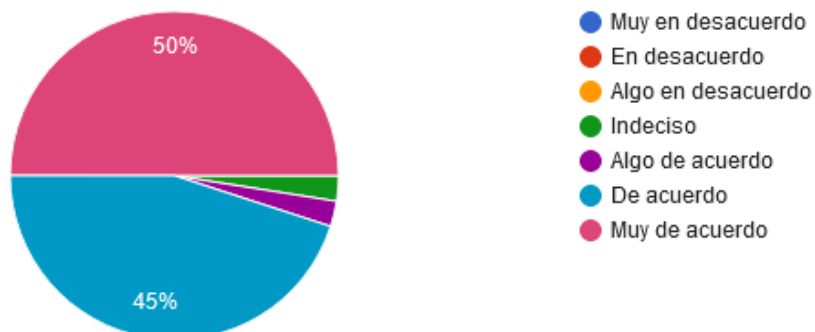


Figura 40. Prueba de Usabilidad - Pregunta 12

13. La interfaz de este sistema fue agradable.

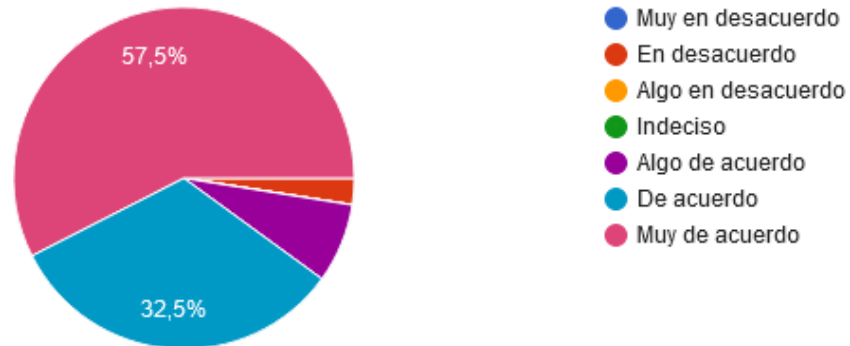


Figura 41. Prueba de Usabilidad - Pregunta 13

14. Me gustó usar la interfaz de este sistema.

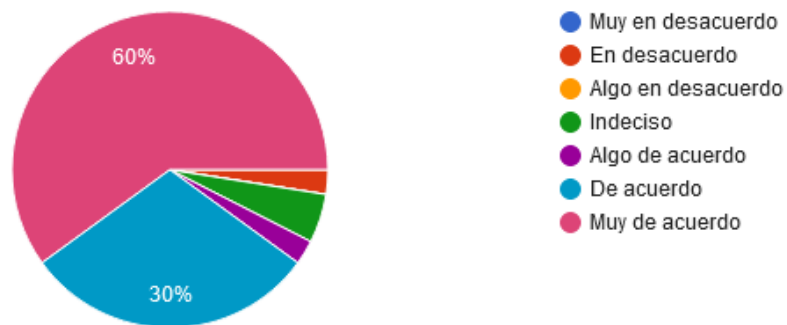


Figura 42. Prueba de Usabilidad - Pregunta 14

15. Este sistema tiene todas las funciones y capacidades que espero que tenga.

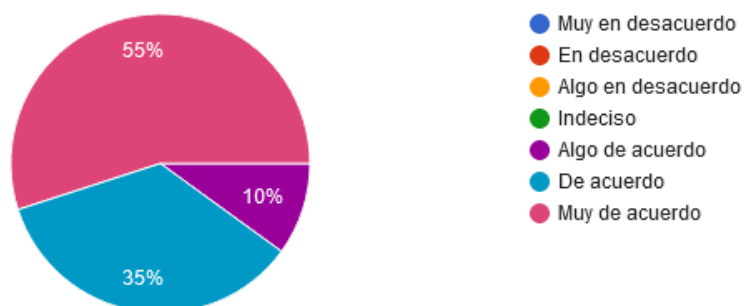


Figura 43. Prueba de Usabilidad - Pregunta 15

16. En general, estoy satisfecho con este sistema.

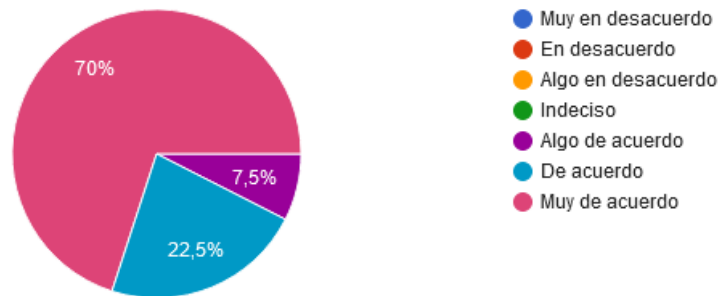


Figura 44. Prueba de Usabilidad - Pregunta 16

Para el análisis de las respuestas obtenidas, se asignó una puntuación o nivel a cada ítem como se muestra en la Tabla XLII para una mejor manipulación de los resultados.

Tabla XLII. Puntuación de cada ítem

Grado de usabilidad	Niveles
Muy en desacuerdo	1
En desacuerdo	2
Algo en desacuerdo	3
Indeciso	4
Algo de acuerdo	5
De acuerdo	6
Muy de acuerdo	7

A cada pregunta se identificó como ítem para mayor facilidad de manipulación de los datos (ver Tabla XLIII).

Tabla XLIII. Identificación de ítems

Pregunta	Ítem
En general, estoy satisfecho con lo fácil que es utilizar este sistema	1
Fue sencillo utilizar este sistema	2
Pude completar las tareas y escenarios rápidamente usando este sistema	3
Me sentí cómodo usando este sistema	4
Fue fácil aprender a usar este sistema	5
Creo que podría volverme productivo rápidamente usando este sistema	6
El sistema dio mensajes de error que me indicaban claramente cómo solucionar problemas	7
Siempre que cometía un error al utilizar el sistema, podía recuperarme fácil y rápidamente.	8
La información (como ayuda en línea, mensajes en pantalla y otra documentación) proporcionada con este sistema era clara.	9
Fue fácil encontrar la información que necesitaba.	10
La información fue eficaz para ayudarme a completar las tareas y los escenarios.	11

La organización de la información en las pantallas del sistema fue clara.	12
La interfaz de este sistema fue agradable.	13
Me gustó usar la interfaz de este sistema.	14
Este sistema tiene todas las funciones y capacidades que espero que tenga.	15
En general, estoy satisfecho con este sistema.	16

Para la asignación de cada puntuación se realizó en base a las respuestas de cada encuesta realizada a los 40 usuarios como se muestra en la Tabla XLIV.

Tabla XLIV. Respuestas obtenidas de la encuesta

Encuestados	Ítem															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
2	7	6	6	7	7	7	7	6	7	7	7	7	6	6	7	7
3	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
4	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
5	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
6	6	6	7	7	6	6	7	6	7	7	7	6	7	7	6	7
7	7	6	6	7	7	6	7	7	7	6	7	6	7	7	6	7
8	7	5	6	7	7	7	7	7	7	7	7	7	7	7	7	7
9	7	7	6	6	5	7	7	7	7	7	7	7	7	7	7	7
10	5	5	5	6	6	7	7	7	7	7	7	7	7	7	7	7
11	5	6	7	7	6	7	7	7	7	7	7	7	7	7	7	7
12	7	5	7	7	7	7	7	7	7	7	7	7	7	7	7	7
13	7	5	6	7	7	7	7	7	7	7	7	7	7	7	7	7
14	6	7	6	6	7	6	5	7	7	7	7	7	7	6	7	7
15	7	7	7	1	7	7	6	7	7	7	7	7	7	7	7	7
16	5	6	6	6	7	6	6	6	6	6	6	6	6	6	6	6
17	6	6	6	6	6	6	6	6	6	6	6	6	5	6	6	6
18	6	6	6	6	6	6	5	5	7	7	6	6	5	6	5	6
19	6	6	6	6	4	4	5	6	5	5	6	6	6	4	5	6
20	6	7	6	7	6	7	6	6	6	7	7	7	6	7	6	7
21	6	6	6	6	7	7	7	7	7	6	7	6	6	7	6	7
22	6	7	7	7	7	7	3	5	6	5	6	6	6	6	6	6
23	6	6	6	6	6	6	2	7	6	6	6	6	6	6	6	6
24	6	6	7	6	6	7	6	6	7	6	6	7	6	7	6	7
25	7	7	7	7	6	7	6	6	6	6	7	6	7	7	7	6
26	7	7	7	7	7	7	6	7	7	7	7	7	7	7	7	7
27	6	7	6	6	7	5	7	6	7	6	7	6	6	6	6	7
28	7	6	6	7	6	7	5	5	6	7	7	6	7	6	7	6
29	6	7	6	6	6	5	6	6	6	6	6	6	5	6	6	7
30	4	6	5	5	7	6	5	7	6	6	6	6	6	6	6	6

31	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
32	5	5	3	6	6	4	7	7	6	6	6	4	6	4	7	5
33	5	6	5	6	6	6	6	6	6	4	6	6	6	5	5	5
34	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
35	6	7	6	6	6	6	6	6	6	7	6	6	6	6	6	7
36	6	6	7	7	7	7	6	6	6	6	7	6	7	7	6	7
37	7	7	7	7	7	7	7	7	6	4	5	6	7	7	7	7
38	3	3	7	3	6	6	6	5	6	6	4	5	2	2	5	5
39	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
40	7	7	6	6	6	6	7	7	7	7	7	7	7	7	7	7

Para el análisis de los datos obtenidos se realizó la sumatoria de la frecuencia de las respuestas obtenidas en cada ítem como se muestra en la Tabla XLV.

Tabla XLV. Frecuencia por ítem

Grado de usabilidad	Ítem																Grado de usabilidad Total
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
Muy en desacuerdo	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1
En desacuerdo	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0	0	3
Algo en desacuerdo	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	5
Indeciso	1	0	0	0	1	2	0	0	0	2	1	1	0	2	0	0	10
Algo de acuerdo	5	5	3	1	1	2	5	4	1	2	1	1	3	1	4	3	42
De acuerdo	14	15	18	16	16	12	12	13	15	13	12	18	13	12	14	9	222
Muy de acuerdo	19	19	18	21	22	24	21	23	24	23	26	20	23	24	22	28	357
Total	40	40	40	40	40	40	40	40	40	40	40	40	40	40	40	40	640

Finalmente se graficó la columna del grado de usabilidad y el grado de usabilidad total contenidos en la Tabla XLV como se muestra en la Figura 45, donde el 56% de los encuestados están muy de acuerdo en que la aplicación es fácil de usar, el 35% está de acuerdo y el 9% está algo en desacuerdo.

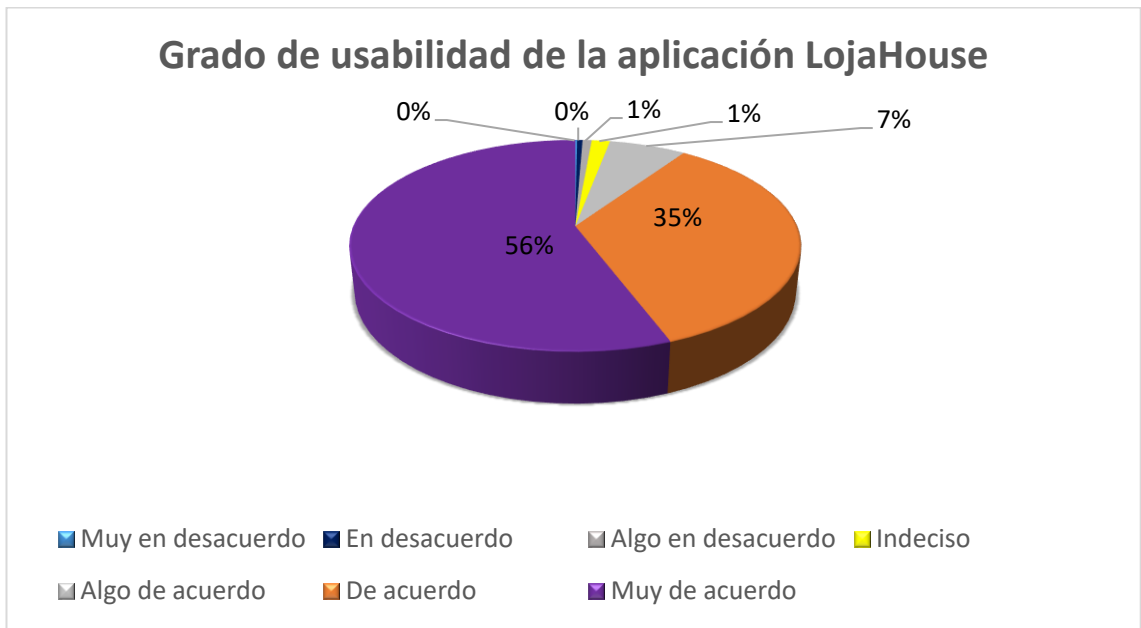


Figura 45. Grado de usabilidad total

Encuesta de aceptación de la aplicación LojaHouse.

Para determinar la influencia que tuvo la Aplicación LojaHouse en la búsqueda de oferta de alquiler de inmuebles se aplicó una encuesta a los usuarios arrendatarios donde los resultados obtenidos en cada pregunta son los que se muestran a continuación:

1. Antes del uso de la aplicación LojaHouse ¿Cuánto tiempo empleaba en la búsqueda de un bien inmueble en la ciudad de Loja?

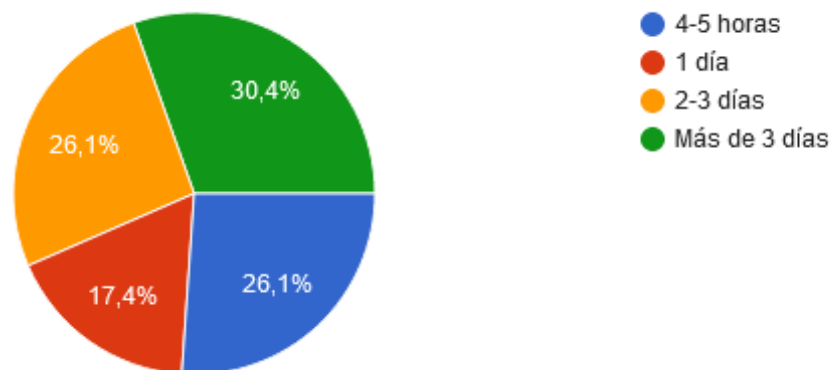


Figura 46. Prueba de Validación - Pregunta 1

2. Luego de haber interactuado con la aplicación LojaHouse ¿Considera Ud., que le permitió agilizar el tiempo de búsqueda de un bien inmueble?

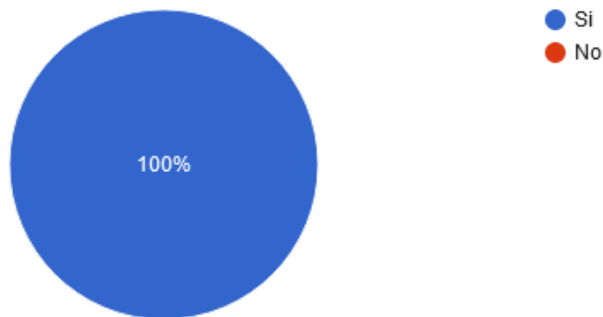


Figura 47. Prueba de Validación - Pregunta 2

3. ¿En qué medida cree que la aplicación “LojaHouse” ayuda en la búsqueda de alquiler de inmuebles en la Ciudad de Loja?

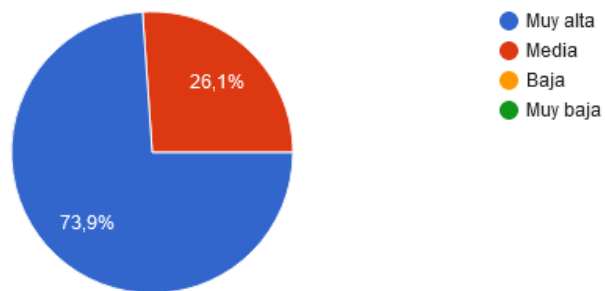


Figura 48. Prueba de Validación - Pregunta 3

4. Luego del uso de la aplicación LojaHouse ¿Cuánto tiempo empleó en la búsqueda de un bien inmueble en la ciudad de Loja?

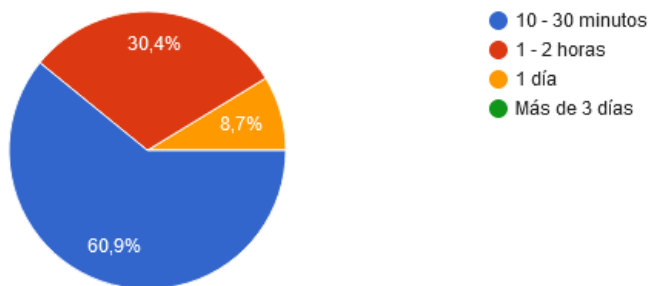


Figura 49. Prueba de Validación - Pregunta 4

5. ¿Sugeriría a otros usuarios que usen la aplicación?

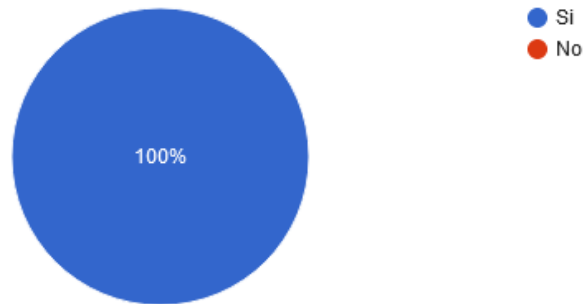


Figura 50. Prueba de Validación - Pregunta 5

6. Indique alguna sugerencia o comentario sobre la aplicación “LojaHouse”.

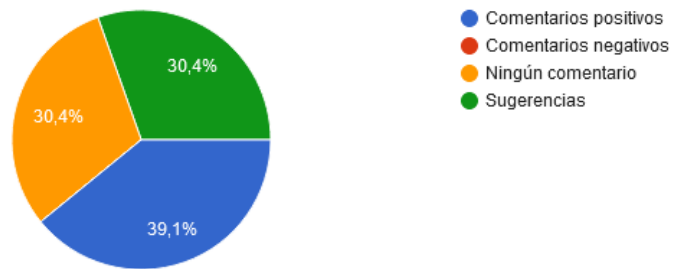


Figura 51. Prueba de Validación - Pregunta 6

7. En medios generales. ¿Qué te pareció la aplicación?

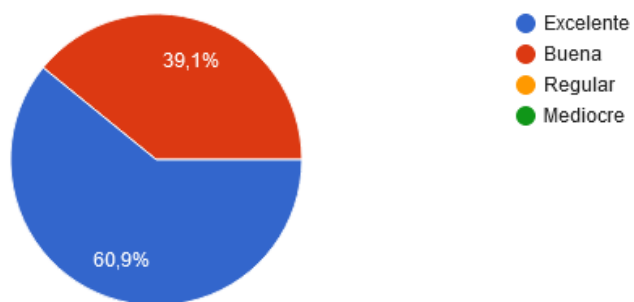


Figura 52. Prueba de Validación - Pregunta 7

7. Discusión

Objetivo 1: Determinar los requisitos necesarios para desarrollar la Aplicación Informática.

La revisión de los trabajos [32], [33], [34], [35], [36], [37], [38] ayudó a identificar 68 requisitos de los cuales 12 se tomaron en cuenta, para ello se realizó una evaluación bajo el criterio del autor con el objetivo de determinar el nivel de entendimiento y su utilidad en el presente TT. Este trabajo planteó el uso de la entrevista para la obtención de requisitos, pero a consecuencia de las medidas restrictivas que surgieron por motivos de la crisis sanitaria a nivel mundial, COVID-19, se utilizó la técnica de la encuesta en un formato digital haciendo uso de la herramienta Google Forms, las cuales fueron difundidas a través de las Aplicaciones de Facebook y WhatsApp logrando llegar a las personas que residen en la ciudad de Loja. Los resultados obtenidos permitieron corroborar con los requisitos de los trabajos similares, además, los requisitos propuestos se diferencian en cuanto al envío de mensajes por WhatsApp, la solicitud de visitas de alquiler, el envío de mensajes al administrador, generación de contrato de alquiler interno, filtrar por el tipo de inmueble, la dirección y el precio; de igual manera, como requisito no funcional se consideró desarrollar la aplicación web y móvil. Como resultado final se obtuvo el documento de Especificación de Requerimientos de Software en base al estándar IEEE-830, mismo que ha sido objeto de estudio en los trabajos revisados anteriormente, por lo que se identificó un total de 23 requisitos funcionales y 9 requisitos no funcionales, además la tecnología a utilizar (Ver **Sección 4.9. Herramientas de Desarrollo**), a diferencia de los trabajos revisados que utilizaban varios lenguajes de programación para el Backend (php, python) y para el Frontend (HTML, CSS, JavaScript), por lo que para este estudio se definió un stack de tecnologías basados en el lenguaje de programación JavaScript y para la aplicación móvil el lenguaje de programación Dart.

Objetivo 2: Diseñar e implementar la Aplicación Informática para la búsqueda de alquiler de inmuebles en base a la metodología XP.

Las fases de la metodología XP permitieron desarrollar el presente TT de manera exitosa como el caso del trabajo elaborado por [35]. En primera instancia se logró definir las HU en base al documento de requerimientos, obteniendo así un conjunto de 18 HU que a su vez se realizó una estimación de tiempo de implementación general, ya que una metodología no es restrictiva, es por ello que se optó por realizar una estimación completa inicial.

Para reforzar el diseño del sistema, se utilizó el modelo arquitectónico 4+1 el cual permitió modelar y documentar el diseño de la aplicación informática para un mejor entendimiento en base a 5 vistas bien definidas. A partir de ello, en la fase de Codificación de la metodología XP, se facilitó en gran medida, ya que se eligió los módulos de forma alternativa para llevar a cabo la implementación de los mismos. Cabe recalcar que existieron funcionalidades que generaron retrasos en la fase de codificación, por ejemplo el manejo de imágenes, envío de notificaciones push, envío de correos electrónicos, entre otras. Al elegir el Framework Flutter con el lenguaje de programación Dart, la curva de aprendizaje fue fácil con un grado de dificultad por falta de conocimientos en cuanto al emplear el patrón bloc para el manejo de estados, pero permitió llevar a cabo el desarrollo de la aplicación móvil de manera exitosa ya que se puede crear aplicaciones elegantes que se ven igual tanto en IOS como en Android, además es una tecnología que está apoderándose de la programación web y de escritorio, cabe recalcar que la aplicación móvil se ha generado específicamente para Android porque no se dispone del hardware necesario para realizar pruebas en dispositivos IOS. Por otro lado, la aplicación LojaHouse permite agregar futuras funcionalidades ya que se compone de una arquitectura cliente servidor multicapa, la cual permite que tanto la aplicación web y móvil se comuniquen a través del Servicio Web API REST.

Objetivo 3: Evaluar la funcionalidad y usabilidad de la Aplicación Informática en un ambiente controlado o simulado.

La evaluación de la aplicación LojaHouse se llevó a cabo mediante diversas pruebas que permitieron garantizar que el software funciona correctamente y cumple con las especificaciones definidas en el documento de Especificación de Requerimientos. Se realizaron casos de prueba de caja negra en base a las Historias de Usuario lo que permitió encontrar algunas fallas en la funcionalidad de la aplicación, las cuales se lograron solventar de manera exitosa y cumplir con los requerimientos especificados. Por otro lado, las pruebas de carga y estrés permitieron detectar en qué páginas tiende a fallar la aplicación, por lo que la página principal y la de iniciar sesión fueron las que presentan cuellos de botella al tener más de 1000 usuarios concurrentes, en este caso influye el servicio de hosting que se utilizó, al tener una versión gratuita en los servidores de Heroku y Firebase, el tiempo de respuesta tiende a ser más lento. Finalmente las pruebas de usabilidad realizadas en base a un cuestionario estandarizado PSSUQ (Cuestionario de Usabilidad del Sistema Posterior al Estudio) utilizado en [43], permitió establecer que la aplicación es aceptada por los usuarios, es fácil de usar y sobretodo,

permite agilizar la búsqueda de oferta de alquiler de un bien inmueble, donde normalmente los usuarios emplean de 1 día como mínimo, hasta más de 3 días, por lo cual, mediante las pruebas realizadas dicha actividad se demora de 10 a 30 minutos como mínimo, hasta máximo 1 día.

De forma general, la aplicación facilita la búsqueda y control de los bienes inmuebles, de los contratos de alquiler interno y las visitas, por lo que se obtuvo como resultado que la aplicación es fácil de usar y es aceptada por los usuarios.

Valoración Técnica económica ambiental

Para el desarrollo del presente Trabajo de Titulación, se utilizaron recursos técnicos, económicos y ambientales.

Los recursos técnicos como Frameworks, Lenguajes de Programación permitieron llevar a cabo el desarrollo y culminación de la aplicación web y móvil. Las herramientas utilizadas, al ser de código abierto y algunos servicios utilizados en su versión gratuita no incurrieron en gastos elevados lo que permitió el correcto desarrollo de la aplicación. En el aspecto ambiental, al permitir realizar una búsqueda de ofertas de alquiler de inmuebles mitiga en gran medida el uso del transporte a los arrendatarios, se reduce el uso de volantes informativos acerca de un bien inmueble en alquiler, y de igual manera se reduce el uso de contratos de alquiler internos en formato físico. En la Tabla XLVI se detalla los recursos utilizados.

Tabla XLVI. Recursos humanos, materiales, técnicos y tecnológicos

RECURSOS HUMANOS			
Cargo	Número de Horas	Precio/Hora	Valor Total
Estudiante	800	3	\$ 2400
Tutor	300	10	\$ 3000
Profesor de la asignatura	140	10	\$ 1400
Total			\$ 6800
RECURSOS MATERIALES			
Descripción	Meses	Precio unitario	Valor Total
Internet	12	\$ 20	\$ 240
Google Play Store	12	\$ 25	\$ 25
Firma Electrónica	12	\$22,4	\$22,4
RECURSOS TÉCNICOS Y TECNOLÓGICOS			
Descripción	Cantidad	Precio Unitario	Valor Total
Laptop	1	\$ 800	\$ 800
SmartPhone	1	\$ 240	\$ 240
Draw.io	1	\$ -	\$ -
Angular	1	\$ -	\$ -
Flutter	1	\$ -	\$ -
GitHub	1	\$ -	\$ -
Firebase Hosting	1	\$ -	\$ -

Firestore Cloud Messaging	1	\$ -	\$ -
Cloudinary	1	\$ -	\$ -
MongoAtlas	1	\$ -	\$ -
PostMan	1	\$ -	\$ -
Pencil	1	\$ -	\$ -
Visual Studio Code	1	\$ -	\$ -
Lenguajes de Programación	2	\$ -	\$ -
Total			\$ 1040

En la Tabla XLVII se muestra un resumen de los costos de los recursos detallados en la tabla anterior, determinando el costo final del presente Trabajo de Titulación.

Tabla XLVII. Costo total de trabajo de titulación

Descripción	Valor Total
Recursos Humanos	\$ 6800
Recursos Materiales	\$ 287,4
Recursos Técnicos y Tecnológicos	\$ 1040
Subtotal	\$ 8127,4
Imprevistos (10%)	\$ 812,74
TOTAL	\$ 8940,14

Al ser un Trabajo de Titulación la Universidad Nacional de Loja ha solventado los gastos de los recursos humanos, en la Tabla XLVIII se muestra el resultado final del costo total del presente Trabajo de Titulación.

Tabla XLVIII. Costo final total de trabajo de titulación

Descripción	Valor Total
Recursos Materiales	\$ 287,4
Recursos Técnicos y Tecnológicos	\$ 1040
Subtotal	\$ 1327,4
Imprevistos (10%)	\$ 132,74
TOTAL	\$ 1460,14

8. Conclusiones

Una vez finalizado el Trabajo de Titulación se concluye lo siguiente:

- El desarrollo de la aplicación LojaHouse permite optimizar la búsqueda de bienes inmuebles a los usuarios que desean alquilar una vivienda en la ciudad de Loja, con respecto al tiempo de búsqueda, donde normalmente empleaban de 1 día como mínimo, hasta más de 3 días, por lo cual, mediante las pruebas realizadas dicha actividad se demora de 10 a 30 minutos como mínimo, hasta máximo 1 día, y como ahorra el tiempo de búsqueda también ayuda económicamente ya que el arrendatario no se dirige al bien inmueble hasta que exista una previa comunicación entre ambas partes, por otro lado, se determinó que la aplicación es aceptada por los usuarios, y que la mayoría está muy de acuerdo en que la aplicación es fácil de usar.
- La revisión de trabajos similares ayudó en gran medida en la recopilación de requisitos generales, y con la encuesta se logró identificar y detallar 23 requisitos funcionales y 9 requisitos no funcionales que la aplicación debe contemplar, quedando establecidos en el documento de Especificación de Requisitos en base al estándar IEEE 830.
- El desarrollo de la aplicación siguiendo la metodología XP ayuda en gran medida a organizar las actividades de forma general para el cumplimiento de las 18 historias de usuario a través de cada una de sus fases, como son: la fase de planificación, diseño, codificación y pruebas para un desarrollo exitoso del proyecto, de igual manera se utilizó el modelo arquitectónico 4+1 para tener una mejor comprensión y comunicación en el desarrollo del sistema.
- La fase de pruebas ayudó a validar que la aplicación cumple con los requisitos especificados, que la aplicación es fácil de usar y sobretodo la aceptación del sistema por parte de los usuarios. Además permitieron corroborar que el rendimiento de la aplicación tenga tiempos de respuesta aceptables.
- Utilizar tecnologías basadas en JavaScript como el Stack MEAN (MongoDB, Express, Angular, Node) ayudado con el Framework Flutter, permitió llevar a cabo el desarrollo de la aplicación de manera exitosa, logrando obtener un producto de software funcional que cumple con los requisitos especificados, además, reduce el tiempo de desarrollo y la curva de aprendizaje.
- La pandemia incidió en la recolección de los requisitos debido a las restricciones de movilidad en todo el mundo pero con la ayuda de red social Facebook y la

aplicación WhatsApp se logró obtener mediante encuestas los requisitos que corroboraron con los obtenidos en la revisión de trabajos similares, logrando de esta manera obtener el documento de Especificación de Requisitos.

9. Recomendaciones

Una vez finalizado el Trabajo de Titulación se recomienda lo siguiente:

- Evaluar las tecnologías que se va a utilizar en el desarrollo de la aplicación para evitar problemas futuros o para estar acorde a las nuevas tecnologías y que van a permanecer por mucho tiempo.
- Realizar una revisión de trabajos similares para los proyectos de desarrollo de aplicaciones informáticas para la obtención de requisitos y poder contrarrestar con los obtenidos mediante la técnica de recolección de datos que se emplee en el Trabajo de Titulación ya que esto ayuda en gran medida para la especificación de requisitos.
- Utilizar el modelo arquitectónico 4+1 para la fase de diseño de la metodología XP ya que define 5 vistas (escenarios, lógica, física, procesos y despliegue) que permiten describir la arquitectura del sistema.
- Desarrollar la aplicación informática con el Stack MEAN ya que el lenguaje de programación es JavaScript y permite codificar tanto para el Backend, el Frontend y en la base de datos.
- Aprender el Framework Flutter y el lenguaje de programación Dart ya que estos van a permitir el desarrollo de aplicaciones no solo móviles sino también web y de escritorio.
- Generar la aplicación móvil para IOS ya que por cuestiones de tiempo y limitaciones en el hardware no se logró en el presente Trabajo de Titulación.

a. Trabajos Futuros:

- Integrar el módulo de gestión de pagos desde la aplicación para llevar un registro de los meses cancelados de alquiler y que la aplicación emita una notificación al usuario indicando que se acerca la fecha de pago.
- Ampliar el inicio de sesión utilizando los servicios de redes sociales para mayor facilidad en cuanto al registro de los usuarios en la aplicación.
- Implementar la geolocalización utilizando MapBox con la finalidad de realizar las búsquedas de los inmuebles de manera más eficiente y fácil para los usuarios.
- Mejorar la funcionalidad de generar el contrato de alquiler interno, permitiendo al usuario arrendador modificar las opciones en base a la normativa de la ley del inquilinato a más de tener la plantilla predefinida actualmente en la aplicación.
- Implementar el módulo del arrendador en la aplicación móvil para brindarle la facilidad de una comunicación más directa y síncrona con sus ofertas solicitadas.

10. Referencias

- [1] J. M. Jara Escobar, "migración estudiantil. un análisis exploratorio sobre el impacto en la adaptación de los estudiantes de la carrera de licenciatura en gestión y desarrollo turístico de la universidad estatal península de santa elena, ecuador," 2019.
- [2] C. Cevallos, "Programa de arrendamiento de vivienda social: Alternativa para reducir el déficit de vivienda social, generar mayor asequibilidad y disminuir la segregación espacial en el Distrito Metropolitano de Quito (DMQ)," 2019.
- [3] Inec, "encuesta nacional de alquileres enalqui 2013," 2013.
- [4] Inec, "Encuesta de Seguimiento al Plan Nacional de Desarrollo," 2019.
- [5] C. F. Iniguez Pineda, "análisis comparativo del servicio de streaming de video de youtube entre las operadoras de datos móviles 4g en el ecuador, para proponer parámetros mínimos de calidad," 2018.
- [6] F. D. Cajacuri Zacarias, C. A. Medina Rivera, W. C. J. Ordoñez Alcantara, and R. I. Rosales Perez, "propuesta de negocio para la intermediación de alquileres de habitaciones para estudiantes a través de una plataforma virtual," 2018.
- [7] C. Micolta Riascos, "estudio de factibilidad para la creación de una agencia inmobiliaria de compraventa y arrendamiento de inmuebles en el distrito de buenaventura," 2014.
- [8] N. G. Salazar Andrade, "las pensiones maximas de arrendamiento de la ley de inquilinato," 2015.
- [9] J. A. Estrada Moya, "análisis jurídico de la ley de inquilinato falta de aplicabilidad entre arrendador y arrendatario cantón babahoyo," 2017.
- [10] A. A. Acosta Escalante, "Causas de incumplimiento de los contratos de arrendamiento," 2020.
- [11] "Ley de inquilinato," 2016. Accessed: Feb. 04, 2021. [Online]. Available: www.lexis.com.ec.
- [12] A. S. Valarezo Díaz, "propuesta para la creación de una empresa de construcción y alquiler de condominios de viviendas modulares en la ciudad de guayaquil," 2018.

- [13] C. Ecuador Constitución de la república del Ecuador, *artículo 30*. 2008, p. 136.
- [14] A. López Gil, “Estudio comparativo de metodologías tradicionales y ágiles para proyectos de desarrollo de software,” 2018.
- [15] M. Navarro, M. Moreno, J. Aranda, L. Parra, J. Rueda, and J. Cruz Pantano, “Selección de Metodologías Ágiles e Integración de Arquitecturas de Software en el Desarrollo de Sistemas de Información,” 2017, Accessed: Feb. 04, 2021. [Online]. Available: http://sedici.unlp.edu.ar/bitstream/handle/10915/62179/Documento_completo.pdf-PDFA.pdf?sequence=1&isAllowed=y.
- [16] P. Letelier and C. Penadés, “metodologías ágiles para el desarrollo de software: extreme programming (xp),” *Métodologías ágiles para el Desarro. Softw. Extrem. Program.*, p. 17, 2017.
- [17] R. Graña Salgado, “Metodologías de desarrollo de proyectos informáticos en entornos web,” 2011.
- [18] L. O. Amavizca Valdez, A. C. García Ruíz, E. Jiménez López, G. L. Duarte Guerrero, and J. C. Vázquez 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.,” *Lat. Am. Caribb. Conf. Eng. Technol.*, vol. 12, pp. 1–10, 2014, [Online]. Available: <http://www.laccei.org/LACCEI2014-Guayaquil/RefereedPapers/RP246.pdf>.
- [19] L. M. Rizzo Villamar and L. G. Cordova Avellaneda, “análisis, desarrollo e implementación de una aplicación web para la gestión de hojas de vida y portafolio de trabajo de profesionales freelance, utilizando extreme programming y tecnologías open source,” 2015.
- [20] S. P. Stalin Joe, “propuesta de una metodología para el desarrollo de aplicaciones móviles en el campo educativo (mobile learning) para la carrera de ingeniería en sistemas de la universidad nacional de loja,” 2017.
- [21] Hernández Rojas Jaime Germán, “desarrollo del módulo de inspecciones de campo para actividad económica integrado al sistema de gestión tributaria (siget) aplicando dispositivos móviles android y metodología ágil mobile-d para la empresa bypros sistemas incorporados cia.ltda.,” 2018.

- [22] D. F. Hallo Yáñez, “desarrollo de una aplicación para la gestión del mercado inmobiliario en la ciudad de quito usando la plataforma iphone.,” 2014.
- [23] E. G. Ríos Pinzón, “Desarrollo de un sistema informático para los procesos de cosecha y post-cosecha de la camaronera ‘pampas de Cayanca,’” 2008.
- [24] F. Moreno del Río, “Desarrollo multiplataforma de tipo Full Stack: creación de un Front-End,” 2017.
- [25] M. Gil Durá, “Aplicación web para el seguimiento online de lecturas basadas en el stack MEAN,” 2015.
- [26] S. M. Johnson, “The building of Web Pages,” no. February, 2020, [Online]. Available: <https://www.diva-portal.org/smash/record.jsf?pid=diva2:1416891>.
- [27] E. Saks, “JavaScript frameworks: Angular vs React vs Vue,” 2019.
- [28] Ó. Fernández Aranda, “Aplicación de metodología Scrum para el desarrollo de una API web de gestión médica con MEAN Stack.”
- [29] A. Samikshya, “MERN STACK WITH MODERN WEB PRACTICES-Developers Connecting Application,” 2020.
- [30] E. Wohlgethan, “SupportingWeb Development Decisions by Comparing Three Major JavaScript Frameworks: Angular, React and Vue.js,” 2018.
- [31] V. Vázquez, “Desarrollo de aplicaciones móviles multiplataforma con Flutter,” 2019.
- [32] C. D. Cobo Pinzón, “diseño e implementación de una aplicación orientada a dispositivos móviles con sistema operativo android para la publicación y búsqueda de bienes inmuebles para venta y arriendo,” 2013.
- [33] M. R. Arranz, “estupiso segovia: buscador de viviendas en alquiler en segovia para estudiantes,” 2016.
- [34] J. A. Jara Castillo, “desarrollo de una aplicación web y una aplicación móvil para registrar el alquiler y ventas de una inmobiliaria utilizando scrum,” 2015.
- [35] M. B. Herrera Arizaga, “desarrollo de una aplicación web y una aplicación móvil para la gestión de alquiler y venta en una inmobiliaria,” 2015.
- [36] C. Lázaro Hernández, “diseño y desarrollo de una aplicación web para compartir

piso,” 2018.

- [37] A. D. Loor Villamar and M. A. Morocho Ramos, “desarrollo e implementación de un aplicativo para dispositivos móviles con sistema operativo android y geolocalización que permita ofrecer y solicitar servicios de alquiler de casas con ubicación exacta y a la vez sugiera las más cercanas a la ubicación a,” 2016.
- [38] J. F. Herrera Nole, “implementación de un sistema web y una aplicación móvil para administrar los servicios y actividades de una empresa inmobiliaria.,” 2016.
- [39] M. Diéguez, S. Sepúlveda, and D. Canullan, “Diseño de un Documento para la Elicitación y Especificación de Requerimientos: Caso Práctico,” *Cuarto Encuentro Informática y Gestión*, no. October, p. 10, 2010.
- [40] K. V. Suaza, “Definición de equivalencias entre historias de usuario y especificaciones en UN - LEN CEP para el desarrollo ágil de software,” p. 96, 2013.
- [41] P. Kruchten, “Planos Arquitectónicos: El Modelo de ‘ 4 + 1 ’ Vistas de la La Arquitectura del Software,” vol. 12, no. 6, pp. 1–16, 2006.
- [42] M. Genero Bocco, J. A. Cruz Lemus, and M. G. Piattini Velthuis, *Métodos de investigación en ingeniería de Software.pdf*. 2015.
- [43] O. D. Tobar Ibarra, “Evaluación de usabilidad de plataforma educativa acceso multidispositivos,” 2018.
- [44] C. Lázaro Hernández, “Diseño y desarrollo de una aplicación web para compartir piso,” 2018.
- [45] C. Cuesta Rodríguez, “aplicación web para la gestión de propiedades inmobiliarias,” 2019.
- [46] M. Ruano Arranz, “estupiso segovia: buscador de viviendas en alquiler en segovia para estudiantes,” 2016.

11. Anexos.

Anexo 1: Obtención de requisitos generales mediante la revisión de trabajos similares	109
Anexo 2: Especificación de Requerimientos	124
Anexo 3: Encuesta para Arrendadores.....	138
Anexo 4: Encuesta para Arrendatarios.....	141
Anexo 5: Prototipo inicial de interfaz.....	144
Anexo 6: Encuesta para evaluar la usabilidad de la aplicación LojaHouse....	156
Anexo 7: Prueba de aceptación Usuarios Finales	161
Anexo 8: Pruebas al Servicio Web API REST.....	163
Anexo 9: Pruebas de Caja Negra	172
Anexo 10: Pruebas de Carga y Estrés	186
Anexo 11: Arquitectura de Software.....	191
Anexo 12: Prototipo final de interfaz	226
Anexo 13. Desarrollo del experimento.....	243

Anexo 1: Obtención de requisitos generales mediante la revisión de trabajos similares

Obtención de requisitos generales mediante la revisión de trabajos similares

Proyecto: Desarrollo de una Aplicación Informática para la búsqueda de oferta de alquiler de inmuebles en la ciudad de Loja.

1. Introducción

1.1. Propósito

El presente documento tiene como propósito la obtención de requisitos generales en base a una revisión bibliográfica de trabajos similares al proyecto titulado “**Desarrollo de una Aplicación Informática para la búsqueda de oferta de alquiler de inmuebles en la ciudad de Loja**”; los mismos que son necesarios para el desarrollo de la Aplicación Informática.

2. Descripción general

2.1. Proyectos encontrados

A continuación se presenta la descripción general y el resumen de algunos trabajos que tiene relación con el presente Trabajo de Titulación, con la finalidad de conocer los aportes y conclusiones relevantes a las que llegaron los autores.

Tabla XLIX. Descripción general de la aplicación para la publicación y búsqueda de bienes inmuebles para venta y arriendo

Trabajo relacionado N°: 1	
Título	Diseño e implementación de una aplicación orientada a dispositivos móviles con sistema operativo Android para la publicación y búsqueda de bienes inmuebles para venta y arriendo.
Año de publicación	2013
País	Ecuador
Plataforma donde fue encontrado	http://repositorio.puce.edu.ec
Entorno de desarrollo	Móvil
Descripción	
Según [32] en su trabajo titulado “Diseño e implementación de una aplicación orientada a dispositivos móviles con sistema operativo Android para la publicación y búsqueda de bienes inmuebles para venta y arriendo”, desarrolló una aplicación enfocada a dispositivos móviles con el objetivo de publicar y buscar bienes inmuebles que se encuentren en venta o arriendo en la ciudad de Quito. Para el desarrollo del proyecto utilizó la metodología RAD con el fin de llevar a cabo un desarrollo rápido. Por lo que concluye que es factible el desarrollo de una aplicación para que los usuarios puedan visualizar, publicar, ubicar y encontrar información acerca de bienes raíces en venta o arriendo.	

Tabla L. Descripción general de la aplicación para registrar el alquiler y ventas de una inmobiliaria utilizando Scrum.

Trabajo relacionado N°: 2	
Título	Desarrollo de una aplicación web y una aplicación móvil para registrar el alquiler y ventas de una inmobiliaria utilizando Scrum
Año de publicación	2015
País	Ecuador
Plataforma donde fue encontrado	http://repositorio.utmachala.edu.ec
Entorno de desarrollo	Web - Móvil
Descripción	
<p>Asimismo [34] en su tesis titulada “Desarrollo de una aplicación web y una aplicación móvil para registrar el alquiler y ventas de una inmobiliaria utilizando scrum”, trata del desarrollo de una aplicación web donde se utilizó el Framework Django y el lenguaje de programación Python, y la aplicación móvil utilizó el lenguaje de programación Java, con el fin de automatizar el proceso de alquiler o venta de inmuebles, asimismo, el registro de inmuebles, clientes y empleados de una empresa de inmobiliaria, para lo cual se lo realizó basándose en la metodología SCRUM para su desarrollo. También hace uso de PostgreSQL para el diseño de la base de datos. No obstante, el estudio concluyó que la aplicación móvil permitió satisfacer las necesidades tanto para el usuario como para el empleado, teniendo un mejor control y organización en las actividades realizadas diariamente.</p>	

Tabla LI. Descripción general de la aplicación web para compartir piso

Trabajo relacionado N°: 3	
Título	Diseño y desarrollo de una aplicación web para compartir piso.
Año de publicación	2018
País	España
Plataforma donde fue encontrado	http://openaccess.uoc.edu/
Entorno de desarrollo	Web
Descripción	
<p>En la tesis [44] titulada: “Diseño y desarrollo de una aplicación web para compartir piso”, se desarrolló una aplicación web para compartir piso en España, haciendo énfasis en que las personas que más se decantan por alquilar un inmueble, son: estudiantes que se mudan de</p>	

ciudad por motivos de estudio o trabajo, hasta personas que ofertan sus casas con el objetivo de adquirir un pago mensual y poder subsistir, al mismo tiempo de mitigar el problema de acceso a vivienda debido a los precios elevados que implica comprar una casa nueva, y que por lo general optan por alquilar casas. Para ello la aplicación web se desarrolló utilizando el lenguaje de programación PHP, y MySQL para la gestión de la base de datos, además, se utilizó una arquitectura cliente-servidor. En cuanto a la metodología utilizada, se basó en su experiencia, sin considerar una metodología estandarizada. No obstante, concluyó que al ser un proyecto extenso en el cuál influyen varios factores, se desarrolló las funcionalidades básicas y que se pueden ampliar en trabajos futuros.

Tabla LII. Descripción de la aplicación para la gestión de propiedades inmobiliarias.

Trabajo relacionado N°: 4	
Título	Implementación de un sistema web y una aplicación móvil para administrar los servicios y actividades de una empresa inmobiliaria
Año de publicación	2016
País	Ecuador
Plataforma donde fue encontrado	http://repositorio.utmachala.edu.ec
Entorno de desarrollo	Web - Móvil
Descripción	
<p>En la tesis [45] Además, [34] con título denominado “Implementación de un sistema web y una aplicación móvil para administrar los servicios y actividades de una empresa inmobiliaria”, con el objetivo de dar solución a la administración de actividades diarias; a la difusión, alquiler y venta de los bienes inmuebles; y a la organización de los datos de las oficinas, empleados, inquilinos, ventas, alquileres, anuncios, visitas e inspecciones que se realizan y son parte de la empresa inmobiliaria, se realizó la construcción de un sistema web sobre la plataforma de Python y su framework Django, con el fin de dar soporte a las actividades y al manejo de información de la empresa por parte de sus colaboradores con sus respectivos permisos de acceso. El proyecto se elaboró a través de la metodología de desarrollo de sistemas de Jonás Montilva denominada MEDSI, la cual permite documentar el progreso y los resultados de la propuesta tecnológica. No obstante, se concluyó que la creación de soluciones tecnológicas es de vital importancia dentro de la estructura de cualquier empresa, organización o negocio contemporáneo que se encuentra en la necesidad de crear lazos de comunicación y productividad dentro del entorno web</p>	

Tabla LIII. Descripción de la aplicación Kleta app

Trabajo relacionado N°: 5	
Título	Desarrollo e Implementación de un Aplicativo para Dispositivos Móviles con Sistema Operativo Android y Geolocalización que Permita Ofrecer y Solicitar Servicios de Alquileres de Casas con Ubicación Exacta y a la vez Sugiera las más Cercanas a la Ubicación Actual en Cualquier Parte de Guayaquil.
Año de publicación	2016
País	Ecuador
Plataforma donde fue encontrado	http://repositorio.ug.edu.ec/
Entorno de desarrollo	Web y Móvil
Descripción	
<p>En [37] con tesis titulada “Desarrollo e implementación de un aplicativo para dispositivos móviles con sistema operativo Android y geolocalización que permita ofrecer y solicitar servicios de alquiler de casas con ubicación exacta y a la vez sugiera las más cercanas a la ubicación actual en cualquier parte de Guayaquil”, la misma que se la denominó Kleta App, con el objetivo de permitir encontrar viviendas de alquiler publicadas con geolocalización, que establezca la ruta más corta hacia la vivienda, el desarrollo de este proyecto se enmarcó en la metodología Scrum-Bam, una metodología de desarrollo rápido y flexible; de igual manera, utilizó el IDE Android Studio para el desarrollo de la aplicación móvil y para la parte del administrador web, se desarrolló utilizando el lenguaje de programación PHP con una base de datos SQL Server sobre un Web Server IIS. No obstante, concluyó mediante encuesta realizada en el proyecto de titulación que usuarios con cuenta Gmail y Smartphone con plan de datos buscan vivienda con los siguientes criterios; sector, número de cuartos, número de baños, rango de precios, patio y garaje.</p>	

Tabla LIV. Descripción de la aplicación Estupiso Segovia.

Trabajo relacionado N°: 6	
Título	Estupiso Segovia: buscador de viviendas en alquiler en Segovia para estudiantes
Año de publicación	2016
País	España
Plataforma donde fue encontrado	http://uvadoc.uva.es/
Entorno de desarrollo	Web
Descripción	
En [46] con tesis titulada “EstuPiso Segovia: buscador de viviendas en alquiler en Segovia para estudiantes.” presenta el desarrollo de una aplicación web con el objetivo de permitir encontrar una vivienda en alquiler para los estudiantes que deciden realizar sus estudios en la ciudad de Segovia, permitiendo que dichos estudiantes puedan encontrar a compañeros que estén estudiando en la ciudad antes mencionad. Para el desarrollo de esta aplicación web se optó por utilizar las bases de las metodologías ágiles sin especificar una en concreto debido al gran peso que tiene la documentación del proyecto.	

3. Requisitos

3.1. Requisitos Individuales

A continuación, mediante el uso de tablas se presenta los requisitos de cada proyecto seleccionado respectivamente, los mismos que se evaluaron según el nivel de entendimiento (Alto, Medio, Bajo) y si es utilizable en mi aplicación (Si, No):

Tabla LV. Requerimientos de la aplicación para la publicación y búsqueda de bienes inmuebles para venta y arriendo.

Requisitos del trabajo relacionado N°: 1			
Diseño e implementación de una aplicación orientada a dispositivos móviles con sistema operativo Android para la publicación y búsqueda de bienes inmuebles para venta y arriendo.			
REQUERIMIENTO	DESCRIPCIÓN	Nivel de entendimiento	Utilizable en mi aplicación

Registrar usuario	No tiene	Alto	Si
Inicio de sesión de usuario	No tiene	Alto	Si
Administrar cuenta de usuario	No tiene	Alto	Si
Publicar inmueble	No tiene	Alto	Si
Administrar inmuebles	No tiene	Alto	Si
Visualizar inmuebles	No tiene	Alto	Si
Visualizar inmuebles destacados	No tiene	Alto	No
Buscar inmuebles	No tiene	Alto	Si
Agregar a favoritos	No tiene	Alto	No
Ver favoritos	No tiene	Alto	No
Ver ubicación de inmuebles	No tiene	Alto	No

Tabla LVI. Requerimientos de la aplicación para registrar el alquiler y ventas de una inmobiliaria.

Requisitos del trabajo relacionado N°: 2			
Desarrollo de una aplicación web y una aplicación móvil para registrar el alquiler y ventas de una inmobiliaria utilizando scrum.			
REQUERIMIENTO	DESCRIPCIÓN	Nivel de entendimiento	Utilizable en mi aplicación
Registrar oficinas	No tiene	Alto	No
Registrar empleados	No tiene	Alto	No
Registrar inmuebles	No tiene	Alto	Si
Registrar propietarios	No tiene	Alto	No
Registrar clientes	No tiene	Alto	No
Generar reportes	No tiene	Alto	No
Visualizar inmuebles disponibles	No tiene	Alto	Si

Reservar citas	No tiene	Alto	Si
Visualizar visitas reservadas	No tiene	Alto	Si

Tabla LVII. Requerimientos de la aplicación para compartir piso.

Requisitos del trabajo relacionado N°: 3			
Diseño y desarrollo de una aplicación web para compartir piso.			
REQUERIMIENTO	DESCRIPCIÓN	Nivel de entendimiento	Utilizable en mi aplicación
Mostrar alquileres	Mostrar los últimos 10 alquileres que han sido añadidos a la aplicación web.	Alto	Si
Buscar alquileres	Realizar búsquedas con filtros.	Alto	Si
Crear alquileres	Añadir un nuevo alquiler a la aplicación y publicación automática en ella.	Alto	No
Gestionar publicaciones	Perfil del usuario para gestionar sus publicaciones.	Alto	Si
Contactarse con el arrendador	Contacto con el arrendador del alquiler.	Alto	Si
Zonas de información	La aplicación deberá mostrar las siguientes zonas de información: <ul style="list-style-type: none"> • Aviso legal. • Funcionamiento de la aplicación. 	Alto	No

Tabla LVIII. Requerimientos de la aplicación para la gestión de propiedades inmobiliarias.

Requisitos del trabajo relacionado N°: 4			
Implementación de un sistema web y una aplicación móvil para administrar los servicios y actividades de una empresa inmobiliaria.			
CÓDIGO	DESCRIPCIÓN	Nivel de entendimiento	Utilizable en mi aplicación
RF001	Crear empleados	Alto	No
RF002	Crear inquilinos	Alto	No
RF003	Crear contratos	Alto	Si
RF004	Crear oficinas	Medio	No
RF005	Crear visitas e inspecciones	Alto	Si
RF006	Publicar una galería de inmuebles	Alto	Si
RF007	Leer reportes	Medio	No
RF008	Conceder permisos a los usuarios: director, supervisor, empleados, y súper usuario.	Alto	No
RF009	Realizar reservas a inmuebles	Alto	Si

Tabla LIX. Requerimientos de la aplicación Kleta app.

Requisitos del trabajo relacionado N°: 5				
Desarrollo e Implementación de un Aplicativo para Dispositivos Móviles con Sistema Operativo Android y Geolocalización que Permita Ofrecer y Solicitar Servicios de Alquileres de Casas con Ubicación Exacta y a la vez Sugiera las más Cercanas a la Ubicación Actual en Cualquier Parte de Guayaquil.				
CÓDIGO	REQUERIMIENTO	DESCRIPCIÓN	Nivel de entendimiento	Utilizable en mi aplicación
RF001	Preparación Servidor Publico	Preparación de Ambiente del Servidor y creación de base de datos para Servicio Web.	Bajo	No

RF002	Implementación de Servicio Web	Implementación y Publicación de Servicio Web.	Bajo	No
RF003	Arranque del Sistema	Inicialización del sistema, preparando el ingreso	Bajo	No
RF004	Registro de Usuario	Registro de datos personales para saber que es un usuario real	Alto	Si
RF005	Ingreso al Sistema	Ingresar credenciales (usuario y clave) para tener acceso al sistema	Bajo	No
RF006	Conexión Google Maps Api con Android	Investigación e Implementación Google Maps Api en Android	Bajo	No
RF007	Pantalla de Mapa	Visualización de Google Maps	Bajo	No
RF008	Geolocalización Visual en Mapa	Mostrar punto de Geolocalización en mapa, vía GPS	Medio	No
RF009	Descarga de Datos	Descarga de Datos desde el Servicio Web, Presentación de Datos de Alquiler en Mapa, con coordenadas exactas.	Bajo	No
RF010	Búsqueda de Datos	Consulta de datos mediante filtros,	Medio	Si

		como número de habitaciones, baños, garaje, patio		
RF011	Resultado de Búsqueda	Presentar un listado con los datos obtenidos en la búsqueda	Bajo	No
RF012	Detalle de Vivienda en Alquiler	Consulta de datos de la vivienda en alquiler que se seleccionó	Alto	Si
RF013	Ruta en Mapa	Creación de Ruta en Mapa, desde mi ubicación hasta la localización del inmueble deseado.	Medio	No
RF014	Llamada Telefónica al oferente del alquiler	Proceso de marcación al número telefónico del oferente de la casa	Alto	Si
RF015	Envío de mensaje al oferente del alquiler	Proceso de envío de mensaje al oferente	Alto	Si
RF016	Mensajes Recibidos	Consultas los mensajes que se han recibido sobre las publicaciones	Bajo	No
RF017	Publicación de Alquiler	Registrar los datos de la casa en alquiler	Medio	Si
RF018	Subir Datos	Subir los datos al Servicio Web	Bajo	No

Tabla LX. Requerimientos de la aplicación Estupiso Segovia.

Requisitos del trabajo relacionado N°: 6			
Estupiso Segovia: buscador de viviendas en alquiler en Segovia para estudiantes			
CÓDIGO	REQUERIMIENTO	Nivel de entendimiento	Utilizable en mi aplicación
RF001	El sistema almacenará los usuarios registrados en el sistema que se identificarán por un nombre de usuario	Alto	Si
RF002	El sistema almacenará la contraseña elegida por cada usuario para su autenticación.	Medio	No
RF003	El nombre de usuario deberá tener entre cuatro y doce caracteres permitiendo únicamente minúsculas, números y guion bajo.	Alto	No
RF004	La contraseña deberá tener entre ocho y doce caracteres permitiendo cualquier carácter excepto espacios, tabulaciones y saltos de línea	Alto	No
RF005	De un usuario el sistema almacenará su nombre y apellidos, foto de perfil, tipo y sus datos de contacto como son teléfono fijo y móvil, correo electrónico y si dicho usuario permite contactar vía WhatsApp	Alto	Si
RF006	El teléfono fijo y la foto de perfil de un usuario no serán obligatorios pudiéndose sustituir la foto por una por defecto	Alto	No

RF007	Un usuario puede ser "Particular" o "Estudiante"	Alto	No
RF008	El sistema almacenará las viviendas registradas por los usuarios que se identificarán por un número incremental y diferenciará entre viviendas en alquiler y viviendas compartidas	Medio	No
RF009	De una vivienda el sistema almacenará el usuario que la publica, si está disponible o alquilada, su fecha de publicación y de última actualización, un título, una descripción, su dirección y coordenadas, coste mensual en Euros, número de habitaciones y de baños, superficie en metros cuadrados, tipo y planta	Alto	Si
RF010	Para las viviendas compartidas el sistema almacenará el número de habitaciones que están libres y el coste mensual corresponderá al coste por habitación	Medio	No
RF011	El tipo de una vivienda puede ser "Piso", "Estudio", "Dúplex" o "Chalet" y la planta "Bajo", "Primero", "Plantas intermedias", "Última planta" o "Chalet"	Alto	No
RF012	El título de una vivienda se limitará a cincuenta caracteres y la descripción a cien	Alto	No
RF013	De una vivienda el sistema almacenará de una a diez fotos	Alto	No

	estableciéndose la primera como foto principal.		
RF014	El sistema almacenará los gastos que sí y los que no incluye el coste de una vivienda diferenciando entre “Comunidad”, “Impuestos y seguro”, “Agua, electricidad y gas” y “Teléfono e internet”	Medio	No
RF015	El sistema almacenará las características de las que sí y de las que no dispone una vivienda siendo estas “Disponibilidad inmediata”, “Recién reformada”, “Amueblada”, “Cocina equipada”, “Ascensor”, “Garaje”, “Trastero”, “Calefacción individual”, “Aire acondicionado”, “Certificación energética y “Mascotas permitidas”	Alto	No

3.2. Requisitos Comunes

En la Tabla LIV se puede observar los requerimientos comunes extraídos en base al análisis y evaluación de los cuatro trabajos relacionados, donde existen requerimientos que se los ha considerado y se los reformulará en la siguiente sección de requisitos comunes seleccionados.

Tabla LXI. Requerimientos comunes entre los trabajos relacionados

Requisitos comunes seleccionados	
Requerimiento	Descripción
Mostrar alquileres	Mostrar los últimos 10 alquileres que han sido añadidos a la aplicación web.
Buscar alquileres	Realizar búsquedas con filtros.
Gestionar publicaciones	Perfil del usuario para gestionar sus publicaciones.
Contactarse con el arrendador	Contacto con el arrendador del alquiler.

Mostrar propiedades	El sistema muestra las propiedades que se encuentran en la aplicación.
Registro de Usuario	Registro de datos personales para saber que es un usuario real.
Búsqueda de Datos	Consulta de datos mediante filtros, como número de habitaciones, baños, garaje, patio.
Detalle de Vivienda en Alquiler	Consulta de datos de la vivienda en alquiler que se seleccionó.
Llamada Telefónica al oferente del alquiler	Proceso de marcación al número telefónico del oferente de la casa.
Publicación de Alquiler	Registrar los datos de la casa en alquiler
Registrar vivienda	De una vivienda, el sistema almacenará el usuario que la publica, si está disponible o alquilada, su fecha de publicación y de última actualización, un título, una descripción, su dirección y coordenadas, coste mensual en Euros, número de habitaciones y de baños, superficie en metros cuadrados, tipo y planta.
Realizar reservas a inmuebles	El usuario registrado podrá realizar reservas a inmuebles.

Anexo 2: Especificación de Requerimientos

Especificación de requisitos de software IEEE 830

Proyecto: Desarrollo de una Aplicación Informática para la búsqueda de oferta de alquiler de inmuebles en la ciudad de Loja.

11.1.1. Introducción

El presente documento presenta los requerimientos del proyecto “Desarrollo de una Aplicación Informática para la búsqueda de oferta de alquiler de inmuebles en la ciudad de Loja”, permitió desarrollar el presente Trabajo de Titulación.

11.1.2. Propósito

El presente documento tiene como propósito obtener los requerimientos funcionales y no funcionales de la Aplicación Informática, necesarios para la elaboración de la aplicación web y la aplicación móvil.

11.1.3. Alcance

La Aplicación Informática denominada “LojaHouse”, tiene como objetivo optimizar la búsqueda de bienes inmuebles en alquiler para que los usuarios que se encuentren dentro de la ciudad de Loja puedan de manera rápida y sencilla visualizar información de un bien inmueble; contactarse con el arrendador y alquilar un bien inmueble.

11.1.4. Personal Involucrado

Tabla LXII. Responsable del desarrollo del proyecto

Nombre	Steeven Michael Armijos Bravo
Rol	Desarrollador del proyecto
Categoría profesional	Estudiante de Ingeniería en Sistemas
Responsabilidades	Análisis, Diseño y Desarrollo de la Aplicación Informática
Información de contacto	smarmijosb@unl.edu.ec

Tabla LXIII. Director del trabajo de titulación

Nombre	Wilman Patricio Chamba Zaragocín
Rol	Director de Trabajo de Titulación
Categoría profesional	Ingeniero en Informática
Responsabilidades	Supervisar y asesorar en el desarrollo del Trabajo de Titulación
Información de contacto	wpchamba@unl.edu.ec

11.1.5. Definiciones, acrónimos y abreviaturas

Definiciones:

- **Arrendatario:** persona que hace uso del inmueble (inquilino).
- **Arrendador:** persona que oferta en alquiler un bien inmueble (dueño).
- **Bien inmueble:** aquel bien que no puede ser transportado del lugar en donde se encuentra.
- **Usuario:** Persona que utilizará la Aplicación Informática.

Acrónimos:

- **ERS:** Especificación de Requerimientos de Software.
- **RF:** Requerimientos Funcionales
- **RNF:** Requerimientos No Funcionales
- **XP:** Extreme Programming
- **REST:** Transferencia de Estado Representacional
- **JWT:** Json Web Token.

11.1.6. Referencias

El presente documento se basa en el estándar IEEE 830.

11.1.7. Resumen

El presente documento de SRS se constituye de 3 secciones:

La primera sección detalla una pequeña introducción para poder tener una visión global de la Aplicación Informática.

La segunda sección describe de forma general la solución de la Aplicación Informática, para conocer la perspectiva y las funcionalidades de la aplicación web y móvil en cuanto a la búsqueda y alquiler de inmuebles dentro de la ciudad de Loja.

La tercera sección se detalla cada uno de los RF y los RNF para satisfacer el desarrollo de la Aplicación Informática y que cumpla con los requerimientos que los usuarios involucrados esperan del producto final.

11.2. Descripción general

A continuación, se describe los factores que afectan directamente a la Aplicación Informática y requerimientos tanto funcionales como no funcionales.

11.2.1. Perspectiva del producto

La Aplicación Informática “LojaHouse”, tiene como propósito ser una aplicación web y móvil, con la finalidad de apoyar y optimizar la búsqueda de oferta de alquiler de inmuebles en la ciudad de Loja; donde podrán visualizar toda la información de los inmuebles publicados, alquilar y generar un contrato de alquiler.

11.2.2. Funcionalidad del producto

La Aplicación Informática tendrá las siguientes funcionalidades:

Entorno Web (Usuario Arrendador):

- Gestión de un bien inmueble.
- Gestión de solicitudes de reserva de alquiler de un bien inmueble.
- Gestión de contratos de alquiler.
- Alquilar un bien inmueble.

Entorno Web y Móvil (Usuario Arrendatario):

- Visualización de bienes inmuebles.
- Buscar bien inmueble.
- Gestión de solicitudes de reservas de alquiler de un bien inmueble.
- Aceptar un contrato de alquiler.
- Visualizar contratos de alquiler.

Entorno Web (Usuario Administrador):

- Gestionar servicios básicos.
- Gestionar mensaje al administrador.
- Activar usuario.
- Desactivar usuario.

La Aplicación Informática pretende ser un apoyo en la búsqueda de oferta de alquiler de inmuebles en la ciudad de Loja, optimizando el tiempo empleado en la búsqueda y alquiler de un bien inmueble.

11.2.3. Características de los usuarios

La Aplicación LojaHouse identifica como usuarios a: personas que buscan un bien inmueble en alquiler, personas que ofertan bienes inmuebles en alquiler y el administrador. A continuación se presenta su interpretación con la Aplicación Informática.

Tabla LXIV. Característica de usuario arrendador

Tipo de usuario	Arrendador
Descripción	Persona que oferta un bien inmueble en alquiler (arrendador).
Habilidades	Habilidades básicas de manejo de una PC
Actividades	Gestión de bienes inmuebles. Gestión de solicitudes de reservas de alquiler de un bien inmueble. Gestión de contrato de alquiler interno. Envío de mensajes al administrador. Gestión de su perfil de usuario.

Tabla LXV. Característica de usuario arrendatario

Tipo de usuario	Arrendatario
Descripción	Persona que busca un bien inmueble donde alquilar (arrendatario).
Habilidades	Habilidades básicas de manejo de una PC o Smartphone
Actividades	Visualización de bienes inmuebles. Buscar bien inmueble. Gestión de solicitudes de reservas de alquiler de un bien inmueble. Envío de mensajes al administrador. Visualizar contratos de alquiler. Gestión de su perfil de usuario.

Tabla LXVI. Característica de usuario administrador

Tipo de usuario	Administrador
Descripción	Persona con conocimientos básicos en la manipulación y administración de la Aplicación Informática.
Habilidades	Administrar la Aplicación Informática, el entorno web.
Actividades	Activar cuentas de usuarios. Desactivar cuentas de usuarios. Gestión de mensajes. Gestión de servicios básicos.

	Gestión de su perfil de usuario.
--	----------------------------------

11.2.4. Restricciones

- La aplicación se desarrollará en base a la metodología XP.
- El lenguaje de programación y tecnologías a utilizar: JavaScript, Angular, Node.js, Express, MongoDB respectivamente.
- La aplicación móvil se desarrollará utilizando el framework Flutter, con el lenguaje de programación Dart.
- La Aplicación móvil será usada en Smartphone con sistema operativo Android.

11.2.5. Suposiciones y dependencias

Los requisitos de la solución informática fueron obtenidos en base a la revisión de trabajos relacionados y encuestas realizadas a personas que ofertan y buscan un inmueble en alquiler dentro de la ciudad de Loja.

Los solución informática tendrá una arquitectura cliente-servidor multicapa con estilo REST, por lo cual, la aplicación web y móvil dependerán de un servicio web API REST (Backend).

11.2.6. Evolución previsible del sistema

La Aplicación Informática permitirá la integración de nuevas funcionalidades dependiendo de las necesidades de los usuarios involucrados, tendrá esa facilidad de evolución tanto en la aplicación web como la móvil.

11.3. Requisitos específicos

11.3.1. Interfaces de usuario

La interfaz de la aplicación constará de botones, cuadros de texto, tablas, menú de opciones, etiquetas de texto, mensajes de alerta, con el fin de facilitar la experiencia al usuario que utilice la aplicación tanto en entorno web como móvil.

11.3.2. Requisitos funcionales

En el presente apartado se detalla los requisitos funcionales que serán utilizados en el sistema:

Tabla LXVII. Requerimiento funcional 1

Número de requisito	RF001
Nombre de requisito	Registrar Usuarios

Descripción del requisito	La Aplicación debe permitir a los usuarios tanto arrendador como arrendatario registrarse, en el entorno web, ingresando (nombres, apellidos, celular, correo electrónico, rol); en la aplicación móvil se solicitarán los mismos datos omitiendo el rol, el cual será asignado automáticamente como arrendatario.
Prioridad del requisito	Alta

Tabla LXVIII. Requerimiento funcional 2

Número de requisito	RF002
Nombre de requisito	Autenticar Usuarios
Descripción del requisito	La Aplicación debe permitir al usuario identificarse para hacer uso de ciertas funcionalidades del sistema, para ello debe ingresar su correo electrónico y contraseña para realizar las diferentes funciones que tendrá cada uno dependiendo de su rol.
Prioridad del requisito	Alta

Tabla LXIX. Requerimiento funcional 3

Número de requisito	RF003
Nombre de requisito	Gestionar cuentas de usuarios
Descripción del requisito	La Aplicación debe permitir al usuario administrador: leer, activar y desactivar cuentas de los usuarios registrados.
Prioridad del requisito	Alta

Tabla LXX. Requerimiento funcional 4

Número de requisito	RF004
Nombre de requisito	Gestionar perfil de usuario

Descripción del requisito	La Aplicación debe permitir al usuario arrendador, arrendatario y administrador: leer y actualizar su perfil de usuario.
Prioridad del requisito	Alta

Tabla LXXI. Requerimiento funcional 5

Número de requisito	RF005
Nombre de requisito	Gestionar Inmueble
Descripción del requisito	La Aplicación debe permitir al usuario arrendador gestionar un bien inmueble (Crear, actualizar, visualizar, buscar y dar de baja).
Prioridad del requisito	Alta

Tabla LXXII. Requerimiento funcional 6

Número de requisito	RF006
Nombre de requisito	Gestionar solicitudes de reserva de alquiler de un bien inmueble.
Descripción del requisito	La Aplicación debe permitir al usuario arrendador : <ul style="list-style-type: none"> • Visualizar solicitudes de reserva de alquiler de un bien inmueble. • Aprobar o rechazar solicitudes de reserva del bien inmueble.
Prioridad del requisito	Alta

Tabla LXXIII. Requerimiento funcional 7

Número de requisito	RF007
Nombre de requisito	Gestionar contratos de alquiler
Descripción del requisito	La Aplicación debe permitir al usuario arrendador: <ul style="list-style-type: none"> • Generar un contrato de alquiler a partir de una plantilla preexistente.

	<ul style="list-style-type: none"> • Visualizar contratos de alquiler generados. • Descargar contrato de alquiler. • Terminar un contrato de alquiler.
Prioridad del requisito	Alta

Tabla LXXIV. Requerimiento funcional 8

Número de requisito	RF008
Nombre de requisito	Alquilar bien inmueble
Descripción del requisito	La Aplicación debe permitir al usuario arrendador alquilar un bien inmueble a un usuario registrado como arrendatario, permitiéndole buscarlo por el nombre de usuario (correo electrónico); en caso de no estar registrado, debe poder registrarlo ingresando los datos (nombres, apellidos, celular, correo electrónico), el rol se asignará automáticamente como arrendatario. Una vez seleccionado el usuario (arrendatario), y el bien inmueble se podrá elegir la fecha de inicio y finalización del contrato, el monto mensual de alquiler y la garantía.
Prioridad del requisito	Alta

Tabla LXXV. Requerimiento funcional 9

Número de requisito	RF009
Nombre de requisito	Visualizar información de un bien inmueble.
Descripción del requisito	La Aplicación debe permitir al usuario público, visualizar información de los bienes inmuebles ofertados (nombre, tipo, precio, estado, dirección, servicios básicos, descripción, imágenes).
Prioridad del requisito	Alta

Tabla LXXVI. Requerimiento funcional 10

Número de requisito	RF010
Nombre de requisito	Buscar un bien inmueble
Descripción del requisito	La Aplicación debe permitir al usuario arrendatario buscar un bien inmueble, considerando los criterios de búsqueda: tipo, precio y ubicación.
Prioridad del requisito	Alta

Tabla LXXVII. Requerimiento funcional 11

Número de requisito	RF011
Nombre de requisito	Gestionar solicitudes de reserva de alquiler de un bien inmueble
Descripción del requisito	<p>La Aplicación debe permitir al usuario arrendatario:</p> <ul style="list-style-type: none"> • Enviar una solicitud de reserva de alquiler de un bien inmueble. • Visualizar solicitudes de reserva de un bien inmueble realizadas. • Dar de baja solicitudes de reserva de alquiler de un bien inmueble.
Prioridad del requisito	Alta

Tabla LXXVIII. Requerimiento funcional 12

Número de requisito	RF012
Nombre de requisito	Visualizar contratos de alquiler interno
Descripción del requisito	La Aplicación debe permitir al usuario arrendatario visualizar el contrato de alquiler generado al momento de alquilar o arrendar un bien inmueble.
Prioridad del requisito	Alta

Tabla LXXIX. Requerimiento funcional 13

Número de requisito	RF013
Nombre de requisito	Resetear contraseña
Descripción del requisito	La Aplicación debe permitir al usuario registrado resetear su contraseña.
Prioridad del requisito	Alta

Tabla LXXX. Requerimiento funcional 14

Número de requisito	RF014
Nombre de requisito	Cambiar contraseña
Descripción del requisito	La Aplicación debe permitir al usuario autenticado actualizar su contraseña.
Prioridad del requisito	Alta

Tabla LXXXI. Requerimiento funcional 15

Número de requisito	RF015
Nombre de requisito	Publicar un anuncio
Descripción del requisito	La Aplicación debe permitir al usuario arrendador publicar o anunciar sus bienes inmuebles en alquiler.
Prioridad del requisito	Alta

Tabla LXXXII. Requerimiento funcional 16

Número de requisito	RF016
Nombre de requisito	Enviar mensajes por WhatsApp
Descripción del requisito	La aplicación móvil debe permitir al usuario arrendatario enviar mensajes por medio de la aplicación WhatsApp.
Prioridad del requisito	Media

Tabla LXXXIII. Requerimiento funcional 17

Número de requisito	RF017
Nombre de requisito	Realizar una llamada telefónica.
Descripción del requisito	La Aplicación móvil debe permitir al usuario arrendatario realizar una llamada por medio de la operadora o aplicación WhatsApp.
Prioridad del requisito	Alta

Tabla LXXXIV. Requerimiento funcional 18

Número de requisito	RF018
Nombre de requisito	Recibir notificación de un bien inmueble ofertado.
Descripción del requisito	La aplicación móvil debe permitir al usuario arrendatario recibir notificaciones cuando se publique un nuevo bien inmueble en alquiler.
Prioridad del requisito	Alta

Tabla LXXXV. Requerimiento funcional 19

Número de requisito	RF019
Nombre de requisito	Seleccionar filtros de búsqueda
Descripción del requisito	La aplicación debe permitir a través de una lista de selección desplegable, elegir los ítems de búsqueda que el usuario requiera para buscar el inmueble, podrá filtrar entre: tipo de inmueble, ubicación (ciudad), precio (rango).
Prioridad del requisito	Alta

Tabla LXXXVI. Requerimiento funcional 20

Número de requisito	RF020
Nombre de requisito	Enviar mensajes al administrador
Descripción del requisito	La aplicación debe permitir al usuario arrendador y arrendatario enviar mensajes al administrador,

	para ello deberá ingresar el título, asunto, y correo.
Prioridad del requisito	Alta

Tabla LXXXVII. Requerimiento funcional 21

Número de requisito	RF021
Nombre de requisito	Visualizar mensajes
Descripción del requisito	La aplicación debe permitir al usuario administrador visualizar mensajes
Prioridad del requisito	Alta

Tabla LXXXVIII. Requerimiento funcional 22

Número de requisito	RF022
Nombre de requisito	Eliminar mensajes
Descripción del requisito	La aplicación debe permitir al usuario administrador eliminar mensajes.
Prioridad del requisito	Alta

Tabla LXXXIX. Requerimiento funcional 23

Número de requisito	RF023
Nombre de requisito	Gestionar servicios básicos
Descripción del requisito	La aplicación debe permitir al usuario administrador gestionar los servicios básicos de los bienes inmuebles, para ello deberá ingresar el nombre del servicio.
Prioridad del requisito	Alta

11.3.3. Requisitos no funcionales

11.3.3.1. Requisitos de rendimiento

- El tiempo de respuesta en las operaciones debe ser inferior o igual a 5 segundos.

Seguridad

La Aplicación Informática deberá encriptar la contraseña de los usuarios registrados mediante el método de encriptación bcrypt.

Para controlar el ingreso a las diferentes funcionalidades de la Aplicación Informática se lo realizará mediante el uso de JWT.

La Aplicación Informática funcionará bajo el protocolo HTTP.

La Aplicación Informática deberá restringir el acceso a los usuarios dependiendo de su rol dentro de la Aplicación Informática.

Fiabilidad

La Aplicación Informática deberá registrar los datos ingresados por parte de los usuarios de manera correcta y al producirse un error de cualquier índole, se deberá emitir un mensaje dentro de la Aplicación.

Disponibilidad

La disponibilidad de la Aplicación Informática será de los 365 días del año, cuya afección producida dependerá del servicio de hosting que se contrate.

Portabilidad

La Aplicación Informática podrá ser utilizada tanto en navegadores web y Smartphone con sistema operativo Android.

Usabilidad

La Aplicación Informática debe ser desarrollada con una interfaz amigable para el usuario y de fácil comprensión, con la finalidad que pueda acceder a las diferentes funcionalidades de forma rápida e intuitiva.

Anexo 3: Encuesta para Arrendadores



UNIVERSIDAD NACIONAL DE LOJA

Encuesta para dueños de bienes inmuebles (arrendador)

Estimado/a:

La presente encuesta tiene como objetivo obtener información importante que ayude en el desarrollo de una Aplicación Informática para la búsqueda de oferta de alquiler de inmuebles. Dicha encuesta servirá en la realización de un trabajo de titulación para la carrera de Ingeniería en Sistemas. Por favor responda de forma veraz. Gracias por su tiempo y colaboración.

1. ¿Qué medios ha utilizado para ofertar un bien inmueble en alquiler?

- Letreros o volantes informativos
- Página web
- Aplicación móvil
- Redes Sociales

2. ¿Qué tipo de bien inmueble oferta?

- Casa
- Departamento
- Cuarto
- Mini departamento
- Otro: _____

3. ¿Qué servicios incluye dentro del valor del arriendo del inmueble?

- Agua Potable
- Energía Eléctrica
- Internet
- Operadora TV Pagada
- Otro: _____

4. ¿Cuál es el medio de comunicación más frecuente que usted utiliza para realizar la entrevista sobre el bien inmueble para su posible arrendatario?

- Telefonía fija
- Teléfono móvil

- Correo electrónico
 - WhatsApp
 - Facebook Messenger
 - Entrevista personal
 - Otro: _____
- 5. ¿Hacia qué tipo de personas va dirigido su bien inmueble en alquiler?**
- Estudiantes
 - Parejas
 - Mujeres
 - Hombres
 - No hay preferencia.
- 6. Para la formalización del alquiler lo realiza a través de:**
- Contrato Legal
 - Contrato simple
 - Palabra
 - Otro: _____
- 7. ¿Ha utilizado alguna de las siguientes plataformas web para ofertar un bien inmueble en alquiler?**
- Airbnb
 - Trivago
 - Facebook
 - OLX
 - Ninguna
- 8. ¿Considera Ud., necesaria una Aplicación Informática para ofertar bienes inmuebles en alquiler?**
- Si
 - No
- 9. ¿Le gustaría gestionar el bien inmueble que oferta en alquiler a través de una aplicación informática?**
- Si
 - No
- 10. ¿Qué información desearía presentar en los avisos de arrendamiento de su bien inmueble?**
- Información de contacto
 - Ubicación del inmueble

- Imágenes del inmueble
- Otro: _____

11. Al alquilar su bien inmueble, ¿Solicita algún monto inicial como garantía?

- Si
- No

Anexo 4: Encuesta para Arrendatarios



UNIVERSIDAD NACIONAL DE LOJA

Encuesta para las personas que buscan un inmueble para arrendar

Estimado/a:

La presente encuesta tiene como objetivo obtener información importante que ayude en el desarrollo de una Aplicación Informática para la búsqueda de oferta de alquiler de inmuebles. Dicha encuesta servirá en la realización de un trabajo de titulación para la carrera de Ingeniería en Sistemas. Por favor responda de forma veraz.

Gracias por su tiempo y colaboración.

1. ¿Ha realizado alguna búsqueda de un bien inmueble para alquilar?
 - Si
 - No
2. ¿Qué medios ha utilizado para buscar un bien inmueble en alquiler?
 - Sale a buscar
 - Por recomendación
 - Página web
 - Aplicación móvil
 - Diario (periódico)
 - Otro: _____
3. Cuando busca un bien inmueble para alquilar ¿Cuáles son sus criterios para la elección?
 - Precio
 - Tipo de inmueble
 - Comodidad de las instalaciones
 - Ubicación
 - Servicios básicos
 - Cercanía a lugares comerciales
 - Cercanía a Universidad
 - Otro: _____

4. ¿Qué medio de comunicación prefiere para contactarse con el arrendador de un bien inmueble?
 - Mensaje de texto
 - Llamada telefónica
 - Correo Electrónico
 - WhatsApp
 - Facebook Messenger
5. ¿Qué tipo de dispositivo de uso frecuente dispone para revisar anuncios electrónicos?
 - Computador
 - Tablet
 - Smartphone (Teléfono inteligente)
 - No utilizo medios electrónicos
6. ¿Utiliza o utilizó alguna aplicación móvil para buscar bienes inmuebles en alquiler?
 - Si
 - No
7. ¿Ha utilizado alguna de las siguientes plataformas web para buscar un bien inmueble en alquiler?
 - Airbnb
 - Trivago
 - Facebook
 - OLX
 - Ninguna
8. Cuando está en el proceso de búsqueda de un bien inmueble, ¿desearía recibir notificaciones de los inmuebles en oferta a través de una aplicación móvil?
 - Si
 - No
9. Cuando arrienda un inmueble, desearía que le ofrezcan a más de los servicios básicos los siguientes:
 - Internet
 - Operadora TV Pagada
 - Otro: _____
10. Para la formalización del alquiler lo realiza a través de:

- Contrato Legal
 - Contrato simple
 - Palabra
 - Otro: _____
11. De acuerdo a su experiencia buscando un bien inmueble en alquiler ¿Le ha sido fácil encontrar un bien inmueble acorde a sus necesidades y posibilidades?
- Si
 - No
12. ¿Cree usted que optimizaría su tiempo de búsqueda al utilizar una aplicación informática para buscar un bien inmueble en alquiler?
- Si
 - No
13. ¿Dispone de un Smartphone actualmente?
- Si
 - No
14. ¿Qué sistema operativo tiene su Smartphone?
- Android
 - IOS
 - Windows Phone
 - Symbian
 - Otro: _____
15. ¿Tiene usted contratado algún plan de Internet móvil en su Smartphone?
- Si
 - No

Anexo 5: Prototipo inicial de interfaz

Pantallas de Aceptación de la Aplicación Informática LojaHouse.

Proyecto: Desarrollo de una aplicación informática para la búsqueda de oferta de alquiler de inmuebles en la ciudad de Loja.

LEGALIZACIÓN DE APROBACIÓN DE LAS INTERFACES O PANTALLAS DE LA APLICACIÓN LOJAHOUSE.

E = Encargado que realizó la elaboración de la aplicación LojaHouse.

R = Encargado que revisó del Prototipo de la aplicación LojaHouse.

A = Encargado de la Aprobación de la aplicación LojaHouse.

Nombre	Cargo	E	R	A	Firma
Steeven Michael Armijos Bravo	Tesista	X			
Wilman Patricio Chamba Zaragocín	Revisor Técnico		X	X	

PROTOTIPO INICIAL DE INTERFACES DE USUARIO DE LA APLICACIÓN LOJAHOUSE.

El presente documento contiene el diseño inicial de la aplicación LojaHouse, la cual fue validada por el director del presente TT; se utilizó la herramienta Pencil para la creación del diseño preliminar de la aplicación, adicionalmente se obtuvo observaciones para su rediseño final.

Tabla XC. Diseño de la pantalla principal

Pantalla principal de la aplicación LojaHose		
Nro., de pantalla	001	
Estado de prueba	Aceptado sin cambios	
	Aceptado con cambios	X
	Rediseñar	
Observaciones	Añadir opciones de búsqueda: tipo de inmueble, precio y ubicación.	

Tabla XCI. Diseño de la pantalla gestión de inmuebles

Pantalla gestión de inmuebles		
Nro., de pantalla	002	
Estado de prueba	Aceptado sin cambios	
	Aceptado con cambios	
	Rediseñar	X
Observaciones	Rediseñar la pantalla para mejorar la experiencia del usuario.	

Gestión de Bienes inmuebles

Nombre	Precio	Estado	Acciones
Casa	\$250	publicado	<input type="button" value="EDITAR"/> <input type="button" value="ELIMINAR"/> <input type="button" value="PUBLICAR"/>
Departamento	\$120	no publicado	<input type="button" value="EDITAR"/> <input type="button" value="ELIMINAR"/> <input type="button" value="PUBLICAR"/>
Cuarto	\$90	publicado	<input type="button" value="EDITAR"/> <input type="button" value="ELIMINAR"/> <input type="button" value="PUBLICAR"/>
Mini-Departamento	\$100	no publicado	<input type="button" value="EDITAR"/> <input type="button" value="ELIMINAR"/> <input type="button" value="PUBLICAR"/>

Tabla XCII. Diseño de la pantalla crear inmueble

Crear inmueble		
Nro., de pantalla	003	
Estado de prueba	Aceptado sin cambios	
	Aceptado con cambios	X
	Rediseñar	
Observaciones	Organizar el formulario, permitir al usuario escribir otros servicios básicos, la carga de imágenes realizarla en otra pantalla y agregar un botón para regresar a la sección anterior.	

Registra un bien inmueble

Información del bien inmueble

Nombre para mostrar en la publicación del inmueble Dirección

Descripción Número de la casa Precio

Tipo de inmueble casa, departamento, cuarto, mini departamento ▾

Sube imágenes de tu bien inmueble **Servicios básicos que incluye**

Subir imágenes del inmueble Subir imagen

Agua internet luz
 amoblado operadora tv pagada

¿Solicitas garantía?

Si solicita un monto inicial como garantía, por favor ingresa su valor

\$ 0

Guardar

Cancelar

Tabla XCIII. Diseño de la pantalla actualizar inmueble

Actualizar inmueble		
Nro., de pantalla	004	
Estado de prueba	Aceptado sin cambios	
	Aceptado con cambios	X
	Rediseñar	
Observaciones	Organizar el formulario, permitir al usuario escribir otros servicios básicos, la carga de imágenes realizarla en otra pantalla y agregar un botón para regresar a la sección anterior.	

Editar un bien inmueble

Información del bien inmueble

Nombre Dirección

Descripción Número de la casa Precio

Tipo de inmueble

Sube imágenes de tu bien inmueble **Servicios básicos que incluye**

Agua internet luz
 amoblado operadora tv pagada

¿Solicitas garantía?

Si solicita un monto inicial como garantía, por favor ingresa su valor

\$ 0

Tabla XCIV. Diseño de la pantalla inmuebles publicados

Listar inmuebles disponibles para alquilar																													
Nro., de pantalla	005																												
Estado de prueba	Aceptado sin cambios																												
	Aceptado con cambios																												
	Rediseñar		X																										
Observaciones	Agregar una barra de búsqueda, rediseñar pantalla para mejorar la experiencia del usuario.																												
<h3>Alquila un bien inmueble</h3> <table border="1"> <thead> <tr> <th>item</th> <th>Nombre</th> <th>Descripción</th> <th>Precio</th> <th>Acciones</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Casa en buen estado</td> <td>Hermosa casa con jardín, garaje, buena ventilación</td> <td>\$250</td> <td>Generar un contrato de alquiler</td> </tr> <tr> <td>2</td> <td>Cuarto para estudiante</td> <td>Cuarto ideal para estudiantes con cocina</td> <td>\$90</td> <td>Generar un contrato de alquiler</td> </tr> <tr> <td>3</td> <td>Departamento</td> <td>Departamento de estreno</td> <td>\$150</td> <td>Generar un contrato de alquiler</td> </tr> <tr> <td>4</td> <td>Casa para familia</td> <td>Casa solo para familia con todas las comodidades</td> <td>\$200</td> <td>Generar un contrato de alquiler</td> </tr> </tbody> </table>					item	Nombre	Descripción	Precio	Acciones	1	Casa en buen estado	Hermosa casa con jardín, garaje, buena ventilación	\$250	Generar un contrato de alquiler	2	Cuarto para estudiante	Cuarto ideal para estudiantes con cocina	\$90	Generar un contrato de alquiler	3	Departamento	Departamento de estreno	\$150	Generar un contrato de alquiler	4	Casa para familia	Casa solo para familia con todas las comodidades	\$200	Generar un contrato de alquiler
item	Nombre	Descripción	Precio	Acciones																									
1	Casa en buen estado	Hermosa casa con jardín, garaje, buena ventilación	\$250	Generar un contrato de alquiler																									
2	Cuarto para estudiante	Cuarto ideal para estudiantes con cocina	\$90	Generar un contrato de alquiler																									
3	Departamento	Departamento de estreno	\$150	Generar un contrato de alquiler																									
4	Casa para familia	Casa solo para familia con todas las comodidades	\$200	Generar un contrato de alquiler																									

Tabla XCV. Diseño de la pantalla generar contrato de alquiler

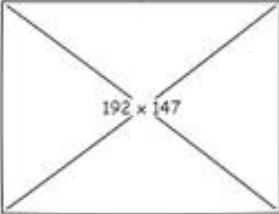
Generar contrato de alquiler	
Nro., de pantalla	006
Estado de prueba	Aceptado sin cambios
	Aceptado con cambios
	Rediseñar
Observaciones	Rediseñar la pantalla para mejorar la experiencia del usuario.
	
<p>Nombre del usuario: Usuario C</p> <p>Celular: 0912345678</p> <p>Inmueble de interés: Departamento amplio</p> <p>Precio de alquiler: \$120</p> <p>Fecha de visita: 14 de junio</p> <p>Hora de visita: 11:00</p> <p>Estado: Pendiente</p> <p style="text-align: center;">Generar un contrato de alquiler</p> <p style="text-align: center;">Eliminar solicitud</p>	

Tabla XCVI. Diseño de la pantalla gestión de visitas

Gestión de visitas																																				
Nro., de pantalla	007																																			
Estado de prueba	<table border="1"> <tr> <td>Aceptado sin cambios</td> <td></td> </tr> <tr> <td>Aceptado con cambios</td> <td></td> </tr> <tr> <td>Rediseñar</td> <td>X</td> </tr> </table>	Aceptado sin cambios		Aceptado con cambios		Rediseñar	X																													
	Aceptado sin cambios																																			
	Aceptado con cambios																																			
Rediseñar	X																																			
Observaciones	Agregar un filtro de solicitudes: pendientes aceptadas, atendidas, todas, y rediseñar el diseño de la pantalla para mejorar la experiencia del usuario.																																			
<p>Solicitudes de visitas de arrendatarios interesados</p> <table border="1"> <thead> <tr> <th>item</th> <th>Usuario</th> <th>Fecha de visita</th> <th>Hora de visita</th> <th>Inmueble</th> <th>Estado</th> <th>Acciones</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario A</td> <td>14-06-20</td> <td>11:00</td> <td>Casa en buen estado</td> <td>pendiente</td> <td> <input type="button" value="VER"/> <input type="button" value="APROBAR"/> <input type="button" value="RECHAZAR"/> </td> </tr> <tr> <td>2</td> <td>Usuario B</td> <td>14-06-20</td> <td>11:00</td> <td>Cuarto para estudiante</td> <td>cancelado</td> <td> <input type="button" value="VER"/> <input type="button" value="APROBAR"/> <input type="button" value="RECHAZAR"/> </td> </tr> <tr> <td>3</td> <td>Usuario C</td> <td>14-06-20</td> <td>11:00</td> <td>Departamento amplio</td> <td>visitado</td> <td> <input type="button" value="VER"/> <input type="button" value="APROBAR"/> <input type="button" value="RECHAZAR"/> </td> </tr> <tr> <td>4</td> <td>Usuario D</td> <td>14-06-20</td> <td>11:00</td> <td>Casa para familia</td> <td>pendiente</td> <td> <input type="button" value="VER"/> <input type="button" value="APROBAR"/> <input type="button" value="RECHAZAR"/> </td> </tr> </tbody> </table>		item	Usuario	Fecha de visita	Hora de visita	Inmueble	Estado	Acciones	1	Usuario A	14-06-20	11:00	Casa en buen estado	pendiente	<input type="button" value="VER"/> <input type="button" value="APROBAR"/> <input type="button" value="RECHAZAR"/>	2	Usuario B	14-06-20	11:00	Cuarto para estudiante	cancelado	<input type="button" value="VER"/> <input type="button" value="APROBAR"/> <input type="button" value="RECHAZAR"/>	3	Usuario C	14-06-20	11:00	Departamento amplio	visitado	<input type="button" value="VER"/> <input type="button" value="APROBAR"/> <input type="button" value="RECHAZAR"/>	4	Usuario D	14-06-20	11:00	Casa para familia	pendiente	<input type="button" value="VER"/> <input type="button" value="APROBAR"/> <input type="button" value="RECHAZAR"/>
item	Usuario	Fecha de visita	Hora de visita	Inmueble	Estado	Acciones																														
1	Usuario A	14-06-20	11:00	Casa en buen estado	pendiente	<input type="button" value="VER"/> <input type="button" value="APROBAR"/> <input type="button" value="RECHAZAR"/>																														
2	Usuario B	14-06-20	11:00	Cuarto para estudiante	cancelado	<input type="button" value="VER"/> <input type="button" value="APROBAR"/> <input type="button" value="RECHAZAR"/>																														
3	Usuario C	14-06-20	11:00	Departamento amplio	visitado	<input type="button" value="VER"/> <input type="button" value="APROBAR"/> <input type="button" value="RECHAZAR"/>																														
4	Usuario D	14-06-20	11:00	Casa para familia	pendiente	<input type="button" value="VER"/> <input type="button" value="APROBAR"/> <input type="button" value="RECHAZAR"/>																														

Tabla XCVII. Diseño de la pantalla listado de contratos de alquiler

Listar contratos de alquiler							
Nro., de pantalla	008						
Estado de prueba	<table border="1"> <tr> <td>Aceptado sin cambios</td> <td></td> </tr> <tr> <td>Aceptado con cambios</td> <td>X</td> </tr> <tr> <td>Rediseñar</td> <td></td> </tr> </table>	Aceptado sin cambios		Aceptado con cambios	X	Rediseñar	
	Aceptado sin cambios						
	Aceptado con cambios	X					
Rediseñar							
Observaciones	Eliminar el botón para generar contrato de alquiler, mostrar los datos de: nombre del contrato, inicio y finalización del contrato, meses de alquiler, estado del contrato y dentro de las acciones disponibles agregar una opción para terminar el contrato de alquiler y visualizar el contrato.						


Sección de gestión de contratos de alquiler

[Generar un contrato de alquiler](#)

item	Inquilino	Inmueble	Fecha	Precio	Acciones
1	Usuario A	Casa en buen estado	19-04-2020	\$250	<div style="display: flex; gap: 10px;"> <div style="border: 1px solid #ccc; padding: 2px 5px;">VER</div> <div style="border: 1px solid #ccc; padding: 2px 5px;">DESCARGAR</div> </div>
2	Usuario B	Cuarto para estudiante	31-09-2020	\$90	<div style="display: flex; gap: 10px;"> <div style="border: 1px solid #ccc; padding: 2px 5px;">VER</div> <div style="border: 1px solid #ccc; padding: 2px 5px;">DESCARGAR</div> </div>
3	Usuario C	Departamento	20-04-2020	\$150	<div style="display: flex; gap: 10px;"> <div style="border: 1px solid #ccc; padding: 2px 5px;">VER</div> <div style="border: 1px solid #ccc; padding: 2px 5px;">DESCARGAR</div> </div>
4	Usuario D	Casa para familia	05-05-2020	\$200	<div style="display: flex; gap: 10px;"> <div style="border: 1px solid #ccc; padding: 2px 5px;">VER</div> <div style="border: 1px solid #ccc; padding: 2px 5px;">DESCARGAR</div> </div>

Tabla XCVIII. Diseño de la pantalla solicitar visita de alquiler

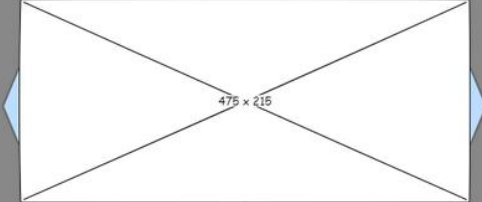
Solicitar visita de alquiler	
Nro., de pantalla	009
Estado de prueba	Aceptado sin cambios
	Aceptado con cambios
	Rediseñar
Observaciones	Rediseñar la pantalla para mejorar la experiencia del usuario.



LojaHouse

Casa en alquiler

Ubicado en San Sebastián




ESTE INMUEBLE SOLICITA UN MONTO INICIAL COMO GARANTÍA

Servicios que incluye:

- Agua potable
- Energía eléctrica
- Internet

\$100

Tabla XCIX. Diseño de la pantalla actualizar perfil de usuario

Actualizar perfil de usuario		
Nro., de pantalla	0010	
Estado de prueba	Aceptado sin cambios	
	Aceptado con cambios	X
	Rediseñar	
Observaciones	Agregar un botón para regresar a la sección anterior y el nombre de usuario (correo electrónico), rol encima del formulario.	
		

Anexo 6: Encuesta para evaluar la usabilidad de la aplicación LojaHouse.



UNIVERSIDAD NACIONAL DE LOJA

Encuesta para evaluar la usabilidad de la aplicación LojaHouse

Este cuestionario tiene por objetivo recolectar información para el trabajo de titulación denominado: “Desarrollo de una aplicación informática para la búsqueda de oferta de alquiler de inmuebles en la ciudad de Loja”. Es importante que tenga en cuenta que su participación es de carácter voluntaria, por lo que tiene plena facultad de decidir si desea o no participar de este proceso. Además, podrá desvincularse en cualquier momento sin que ello lo/la perjudique de forma alguna. Sus respuestas serán resguardadas bajo la más estricta confidencialidad siendo usadas para fines académicos.

1. En general, estoy satisfecho con lo fácil que es utilizar este sistema.

- Muy en desacuerdo
- En desacuerdo
- Algo en desacuerdo
- Indeciso
- Algo de acuerdo
- De acuerdo
- Muy de acuerdo

2. Fue sencillo utilizar este sistema

- Muy en desacuerdo
- En desacuerdo
- Algo en desacuerdo
- Indeciso
- Algo de acuerdo
- De acuerdo
- Muy de acuerdo

3. Pude completar las tareas y escenarios rápidamente usando este sistema

- Muy en desacuerdo

- En desacuerdo
- Algo en desacuerdo
- Indeciso
- Algo de acuerdo
- De acuerdo
- Muy de acuerdo

4. Me sentí cómodo usando este sistema

- Muy en desacuerdo
- En desacuerdo
- Algo en desacuerdo
- Indeciso
- Algo de acuerdo
- De acuerdo
- Muy de acuerdo

5. Fue fácil aprender a usar este sistema

- Muy en desacuerdo
- En desacuerdo
- Algo en desacuerdo
- Indeciso
- Algo de acuerdo
- De acuerdo
- Muy de acuerdo

6. Creo que podría volverme productivo rápidamente usando este sistema

- Muy en desacuerdo
- En desacuerdo
- Algo en desacuerdo
- Indeciso
- Algo de acuerdo
- De acuerdo
- Muy de acuerdo

7. El sistema dio mensajes de error que me indicaban claramente cómo solucionar problemas

- Muy en desacuerdo
- En desacuerdo
- Algo en desacuerdo

- Indeciso
 - Algo de acuerdo
 - De acuerdo
 - Muy de acuerdo
- 8. Siempre que cometía un error al utilizar el sistema, podía recuperarme fácil y rápidamente**
- Muy en desacuerdo
 - En desacuerdo
 - Algo en desacuerdo
 - Indeciso
 - Algo de acuerdo
 - De acuerdo
 - Muy de acuerdo
- 9. La información (como ayuda en línea, mensajes en pantalla y otra documentación) proporcionada con este sistema era clara**
- Muy en desacuerdo
 - En desacuerdo
 - Algo en desacuerdo
 - Indeciso
 - Algo de acuerdo
 - De acuerdo
 - Muy de acuerdo
- 10. Fue fácil encontrar la información que necesitaba**
- Muy en desacuerdo
 - En desacuerdo
 - Algo en desacuerdo
 - Indeciso
 - Algo de acuerdo
 - De acuerdo
 - Muy de acuerdo
- 11. La información fue eficaz para ayudarme a completar las tareas y los escenarios**
- Muy en desacuerdo
 - En desacuerdo
 - Algo en desacuerdo

- Indeciso
- Algo de acuerdo
- De acuerdo
- Muy de acuerdo

12. La organización de la información en las pantallas del sistema fue clara

- Muy en desacuerdo
- En desacuerdo
- Algo en desacuerdo
- Indeciso
- Algo de acuerdo
- De acuerdo
- Muy de acuerdo

13. La interfaz de este sistema fue agradable

- Muy en desacuerdo
- En desacuerdo
- Algo en desacuerdo
- Indeciso
- Algo de acuerdo
- De acuerdo
- Muy de acuerdo

14. Me gustó usar la interfaz de este sistema

- Muy en desacuerdo
- En desacuerdo
- Algo en desacuerdo
- Indeciso
- Algo de acuerdo
- De acuerdo
- Muy de acuerdo

15. Este sistema tiene todas las funciones y capacidades que espero que tenga

- Muy en desacuerdo
- En desacuerdo
- Algo en desacuerdo
- Indeciso
- Algo de acuerdo
- De acuerdo

- Muy de acuerdo

16. En general, estoy satisfecho con este sistema

- Muy en desacuerdo
- En desacuerdo
- Algo en desacuerdo
- Indeciso
- Algo de acuerdo
- De acuerdo
- Muy de acuerdo

Anexo 7: Prueba de aceptación Usuarios Finales



UNIVERSIDAD NACIONAL DE LOJA

Pruebas de Aceptación

La presente encuesta tiene la finalidad de conocer cómo influyó la aplicación LojaHouse en cuanto a la búsqueda de ofertas de alquiler de inmuebles en la ciudad de Loja. Sus respuestas serán resguardadas bajo la más estricta confidencialidad siendo usadas estrictamente con fines académicos.

- 1. Antes del uso de la aplicación LojaHouse ¿Cuánto tiempo empleaba en la búsqueda de un bien inmueble en la ciudad de Loja?**
 - 4-5 horas
 - 1 día
 - 2-3 días
 - Más de 3 días
- 2. Luego de haber interactuado con la aplicación LojaHouse ¿Considera Ud., que le permitió agilizar el tiempo de búsqueda de un bien inmueble?**
 - Si
 - No
- 3. ¿En qué medida cree que la aplicación “LojaHouse” ayuda en la búsqueda de alquiler de inmuebles en la Ciudad de Loja?**
 - Muy alta
 - Media
 - Baja
 - Muy baja
- 4. Luego del uso de la aplicación LojaHouse ¿Cuánto tiempo empleó en la búsqueda de un bien inmueble en la ciudad de Loja?**
 - Menos de 5 minutos
 - 10 - 30 minutos
 - 1 día
 - Más de 3 días
- 5. ¿Sugeriría a otros usuarios que usen la aplicación?**

- Si
- No

6. Indique alguna sugerencia o comentario sobre la aplicación “LojaHouse”

7. En medios generales. ¿Qué te pareció la aplicación?

- Excelente
- Buena
- Regular
- Mediocre

Anexo 8: Pruebas al Servicio Web API REST

Pruebas al Servicio Web API REST

Proyecto: Desarrollo de una Aplicación Informática para la búsqueda de oferta de alquiler de inmuebles en la ciudad de Loja.

VALIDACIÓN DE LAS PRUEBAS AL SERVICIO WEB API REST PARA LA APLICACIÓN LOJAHOUSE.

E = Encargado de la elaboración del documento.

R = Encargado de la revisión del documento.

A = Encargado de la aprobación del documento de pruebas al servicio web API REST.

Nombre	Cargo	E	R	A	Firma
Steeven Michael Armijos Bravo	Tesista	X			
Wilman Patricio Chamba Zaragocín	Revisor Técnico		X	X	

1. Introducción

El presente documento muestra las pruebas al servicio web API REST de la aplicación LojaHouse, para comprobar el correcto funcionamiento de los end points, controlando que las respuestas devueltas sean correctas.

2. Propósito

Detectar posibles errores cometidos en la implementación del servicio web API REST, así mismo, probar que los end points respondan de manera correcta las peticiones; para lo cual se utilizó la herramienta de software Postman.

3. Referencias

Referencia	Sitio Web
Postman	https://www.postman.com/

4. Pruebas al Servicio Web API REST

Para las pruebas del API REST se utilizó la herramienta Postman, la cual permitió el envío de peticiones http (get, put, delete, post) hacia el servicio web API REST, obteniendo una respuesta en formato json.

Tabla C. Petición para registrar una cuenta de usuario

URL	Token	Método	Descripción
localhost:3000/usuario/crearusuario	No	POST	Permite registrar la cuenta del usuario
Resultado			

POST localhost:3000/usuario/crearusuario

form-data x-www-form-urlencoded raw binary

Key	Value	Description
nombre	Steeven Michael	
apellido	Armijos Bravo	
correo	smarmijosb@un.edu.ec	
movil	09896974004	
estado	1	
rol	ARRENDATARIO	

Body Cookies Headers (13) Test Results Status: 200 OK Time: 2160 ms

```

1 {
2   "ok": true,
3   "mensaje": "Usuario Steeven Michael Armijos Bravo creado exitosamente"
4 }
  
```

Tabla CI. Petición para el inicio de sesión

URL	Token	Método	Descripción
http://localhost:3000/login	No	POST	Permite iniciar sesión.

Resultado

POST http://localhost:3000/login

correo smarmijosb@un.edu.ec
password up5kdH

Body Cookies Headers (13) Test Results Status: 200 OK Time: 238 ms

```

1 {
2   "ok": true,
3   "mensaje": "Login correcto",
4   "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJic3VhcmVlIjpb7I19pZCI6IjYwZlZkZjksIjN1OTFmIzcxNDM3MzI3IjEiInJvcCI6Ikp5SUKVOREFUUVjYyYjY9LzJpYXQ1QjE2MjYxNTI5N2csImV4cCI6IjE3NTYxYjE5MjE3N30.9yubhrqmjq058koL4ngNGK6sDa9SsKGfMfthexq0",
5   "menu": [
6     {
7       "titulo": "Arrendatario",
8       "icono": "mdi mdi-account-outline",
9       "submenu": [
10        {
11          "titulo": " Dashboard ",
12          "url": "/dasharrendatario"
13        },
14        {
15          "titulo": " Visitas solicitadas ",
16          "url": "/visitas-arrendatario"
17        },
18        {
19          "titulo": " Contratos de Alquiler ",
20          "url": "/contratoarrendatario"
21        }
22      ]
23   }
24 }
  
```

Tabla CII. Petición para el reseteo de password

URL	Token	Método	Descripción
http://localhost:3000/usuario/resetepassw ord	No	PUT	Permite resetear el password

Resultado															
<div style="border: 1px solid #ccc; padding: 5px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> PUT http://localhost:3000/usuario/resetepassw Params Send Save </div> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td><input checked="" type="checkbox"/></td> <td>correo</td> <td>smarmijosb@unl.edu.ec</td> <td></td> </tr> <tr> <td><input type="checkbox"/></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td>New key</td> <td>Value</td> <td>Description</td> </tr> </table> <div style="display: flex; justify-content: space-between; align-items: center; margin-top: 5px;"> Body Cookies Headers (13) Test Results Status: 200 OK Time: 1860 ms </div> <div style="margin-top: 5px;"> Pretty Raw Preview JSON Save Response </div> <pre style="background-color: #f0f0f0; padding: 5px;"> 1 { 2 "ok": true, 3 "usuario": { 4 "imagen": [], 5 "estado": "1", 6 "_id": "60ed1f9923e91f3714373b7d", 7 "nombre": "Steeven Michael", 8 "apellido": "Armijos Bravo", 9 "correo": "smarmijosb@unl.edu.ec", 10 "password": ":"}, 11 "movil": "09896974004", 12 "rol": "ARRENDATARIO", 13 "createdAt": "2021-07-13T05:07:38.092Z", 14 "updatedAt": "2021-07-13T05:20:50.598Z" 15 } 16 }</pre> </div>				<input checked="" type="checkbox"/>	correo	smarmijosb@unl.edu.ec		<input type="checkbox"/>					New key	Value	Description
<input checked="" type="checkbox"/>	correo	smarmijosb@unl.edu.ec													
<input type="checkbox"/>															
	New key	Value	Description												

Tabla CIII. Petición para el envío de mensajes

URL	Token	Método	Descripción
http://localhost:3000/mensaje/crearmensaj e	No	POST	Permite enviar un mensaje

Resultado																							
<div style="border: 1px solid #ccc; padding: 5px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> POST http://localhost:3000/mensaje/crearmensaje Params Send Save </div> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td><input checked="" type="checkbox"/></td> <td>titulo</td> <td>Recuperar cuenta</td> <td></td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>asunto</td> <td>Quisiera recuperar mi cuenta</td> <td></td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>fecha</td> <td>2020-01-02</td> <td></td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>correo</td> <td>smarmijosb@unl.edu.ec</td> <td></td> </tr> <tr> <td></td> <td>New key</td> <td>Value</td> <td>Description</td> </tr> </table> <div style="display: flex; justify-content: space-between; align-items: center; margin-top: 5px;"> Body Cookies Headers (13) Test Results Status: 201 Created Time: 273 ms </div> <div style="margin-top: 5px;"> Pretty Raw Preview JSON Save Response </div> <pre style="background-color: #f0f0f0; padding: 5px;"> 1 { 2 "ok": true, 3 "mensaje": { 4 "estado": "ENVIADO", 5 "_id": "60ed26b80ecc5639e0638e71", 6 "titulo": "Recuperar cuenta", 7 "asunto": "Quisiera recuperar mi cuenta", 8 "fecha": "2020-01-02T00:00:00.000Z", 9 "correo": "smarmijosb@unl.edu.ec", 10 "createdAt": "2021-07-13T05:38:00.289Z", 11 "updatedAt": "2021-07-13T05:38:00.289Z" 12 } 13 }</pre> </div>				<input checked="" type="checkbox"/>	titulo	Recuperar cuenta		<input checked="" type="checkbox"/>	asunto	Quisiera recuperar mi cuenta		<input checked="" type="checkbox"/>	fecha	2020-01-02		<input checked="" type="checkbox"/>	correo	smarmijosb@unl.edu.ec			New key	Value	Description
<input checked="" type="checkbox"/>	titulo	Recuperar cuenta																					
<input checked="" type="checkbox"/>	asunto	Quisiera recuperar mi cuenta																					
<input checked="" type="checkbox"/>	fecha	2020-01-02																					
<input checked="" type="checkbox"/>	correo	smarmijosb@unl.edu.ec																					
	New key	Value	Description																				

Tabla CIV. Petición para filtrar bienes inmuebles

URL	Token	Método	Descripción
http://localhost:3000/busqueda/coleccion/inmuebles/CASA/Gran Colombia/150-200	No	GET	Permite filtrar bienes inmuebles
Resultado			
<pre> 1 - { 2 "inmuebles": [3 { 4 "servicio": [5 "garaje", 6 "agua", 7 "teléfono" 8], 9 "imagen": [10 { 11 "_id": "5a0335aa-d43f-43b1-a48b-9761b26a9d06", 12 "url": "http://res.cloudinary.com/lojahouse/image/upload/v1625889035/inmueble/60a205ce84bbd200155c38c8/qsbyiw68qsb8glx0qyw.jpg", 13 "inmueble": "60a205ce84bbd200155c38c8", 14 "public_id": "inmueble/60a205ce84bbd200155c38c8/qsbyiw68qsb8glx0qyw" 15 }, 16 { 17 "_id": "5a19f0e9-ae8d-45eb-8a09-ce839c290fd4", 18 "url": "http://res.cloudinary.com/lojahouse/image/upload/v1625889035/inmueble/60a205ce84bbd200155c38c8/wy2abqe6zjknyoiulxpq.jpg", 19 "inmueble": "60a205ce84bbd200155c38c8", 20 "public_id": "inmueble/60a205ce84bbd200155c38c8/wy2abqe6zjknyoiulxpq" 21 } 22] 23 } 24] 25 } </pre>			

Tabla CV. Petición para visualizar información del inmueble

URL	Token	Método	Descripción
localhost:3000/inmueble/obtenerinmueble/publico/60a88386fdbc902cfc00fc22	No	GET	Permite visualizar información del inmueble
Resultado			
<pre> 1 - { 2 "ok": true, 3 "inmueble": [4 { 5 "servicio": [6 "garaje", 7 "luz", 8 "teléfono" 9], 10 "imagen": [11 { 12 "_id": "a98044fd-4590-434f-8540-0c1aa1b345ca", 13 "url": "http://res.cloudinary.com/lojahouse/image/upload/v1622102801/inmueble/60a88386fdbc902cfc00fc22/zfmkkvvf7iow7sa19nne.jpg", 14 "inmueble": "60a88386fdbc902cfc00fc22", 15 "public_id": "inmueble/60a88386fdbc902cfc00fc22/zfmkkvvf7iow7sa19nne" 16 }, 17 { 18 "_id": "1d563761-3e04-4bb2-b176-af676b851931", 19 "url": "http://res.cloudinary.com/lojahouse/image/upload/v1622102801/inmueble/60a88386fdbc902cfc00fc22/pqcp7ukqapzsjku3qfrz.jpg", 20 "inmueble": "60a88386fdbc902cfc00fc22", 21 "public_id": "inmueble/60a88386fdbc902cfc00fc22/pqcp7ukqapzsjku3qfrz" 22 }, 23 { 24 "_id": "26a75379-d19f-42c3-91a7-73b606c2473d", 25 "url": "http://res.cloudinary.com/lojahouse/image/upload/v1622102801/inmueble/60a88386fdbc902cfc00fc22/amewg113dfs0nosvs0fe.jpg", 26 "inmueble": "60a88386fdbc902cfc00fc22", 27 "public_id": "inmueble/60a88386fdbc902cfc00fc22/amewg113dfs0nosvs0fe" 28 } 29] 30 } 31] 32 } </pre>			

Tabla CVI. Petición para crear un bien inmueble

URL	Token	Método	Descripción
localhost:3000/inmueble/crearinmueble	Si	POST	Permite crear un bien inmueble
Resultado			

Tabla CVII. Petición para publicar un bien inmueble

URL	Token	Método	Descripción
localhost:3000/inmueble/desactivarinmueble/60ed2b0f064bcc3ba8f21378	Si	PUT	Permite publicar un bien inmueble
Resultado			

Tabla CVIII. Petición para generar una solicitud de alquiler

URL	Token	Método	Descripción
localhost:3000/visita/crearvisita	Si	POST	Permite generar una solicitud de alquiler

Resultado																
<p>The screenshot shows a REST client interface with the following details:</p> <ul style="list-style-type: none"> Method: POST URL: localhost:3000/visita/crearvisita?token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VhcnVlIjoi... Body: x-www-form-urlencoded Form Data: <table border="1"> <thead> <tr> <th>Key</th> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>fecha</td> <td>4/9/2021</td> <td></td> </tr> <tr> <td>descripcion</td> <td>Lorem ipsum dolor, sit amet consectetur adipisicing...</td> <td></td> </tr> <tr> <td>usuarioarrendatario</td> <td>60ed1f9923e91f3714373b7d</td> <td></td> </tr> <tr> <td>inmueble</td> <td>60df23774f45bb4f84a03943</td> <td></td> </tr> </tbody> </table> Status: 201 Created Time: 250 ms Response (JSON): <pre> 1 { 2 "ok": true, 3 "visita": { 13 "mensaje": "Se ha registrado su visita" 14 } </pre> 		Key	Value	Description	fecha	4/9/2021		descripcion	Lorem ipsum dolor, sit amet consectetur adipisicing...		usuarioarrendatario	60ed1f9923e91f3714373b7d		inmueble	60df23774f45bb4f84a03943	
Key	Value	Description														
fecha	4/9/2021															
descripcion	Lorem ipsum dolor, sit amet consectetur adipisicing...															
usuarioarrendatario	60ed1f9923e91f3714373b7d															
inmueble	60df23774f45bb4f84a03943															

Tabla CIX. Petición para aceptar una visita

URL	Token	Método	Descripción
localhost:3000/inmueble/desactivarinmueble/60ed2b0f064bcc3ba8f21378	Si	PUT	Permite aceptar una visita solicitada

Resultado							
<p>The screenshot shows a REST client interface with the following details:</p> <ul style="list-style-type: none"> Method: PUT URL: localhost:3000/visita/arrendador/aceptarvisita/60ed2fafdb551b3bbcd02523?token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VhcnVlIjoi... Body: x-www-form-urlencoded Form Data: <table border="1"> <thead> <tr> <th>Key</th> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>estado</td> <td>ACEPTADA</td> <td></td> </tr> </tbody> </table> Status: 200 OK Time: 239 ms Response (JSON): <pre> 1 { 2 "ok": true, 3 "visita": { 4 "estado": "ACEPTADA", 5 "id": "60ed2fafdb551b3bbcd02523", 6 "fecha": "2021-04-09T05:00:00.000Z", 7 "descripcion": "Lorem ipsum dolor, sit amet consectetur adipisicing elit. Corrupti inventore vel recusandae facere adipisci 8 incidunt debitis itaque sunt dolorum libero? ", 9 "inmueble": "60df23774f45bb4f84a03943", 10 "usuarioarrendatario": "60ed1f9923e91f3714373b7d", 11 "createdAt": "2021-07-13T06:16:16.012Z", 12 "updatedAt": "2021-07-13T06:23:38.567Z" 13 }, 14 "mensaje": "La visita está: ACEPTADA" </pre> 		Key	Value	Description	estado	ACEPTADA	
Key	Value	Description					
estado	ACEPTADA						

Tabla CX. Petición para generar un contrato de alquiler

URL	Token	Método	Descripción
localhost:3000/contrato/crearcontrato	Si	POST	Permite generar un contrato de alquiler

Resultado		

Tabla CXI. Petición para aceptar un contrato de alquiler

URL	Token	Método	Descripción
localhost:3000/contrato/acuerdo/60ed32a0db551b3bbcd02525/aceptar	Si	POST	Permite aceptar un contrato de alquiler

Resultado		

Anexo 9: Pruebas de Caja Negra

Pruebas de Caja Negra

Proyecto: Desarrollo de una Aplicación Informática para la búsqueda de oferta de alquiler de inmuebles en la ciudad de Loja.

VALIDACIÓN DE LAS PRUEBAS DE CAJA NEGRA PARA LA APLICACIÓN LOJAHOUSE.

E = Encargado de la elaboración del documento.

R = Encargado de la revisión del documento.

A = Encargado de la aprobación del documento de pruebas de caja negra.

Nombre	Cargo	E	R	A	Firma
Steeven Michael Armijos Bravo	Tesista	X			
Wilman Patricio Chamba Zaragocín	Revisor Técnico		X	X	

Las pruebas de caja negra se llevaron a cabo en base a los criterios de aceptación de las Historias de Usuario, utilizando una lista de chequeo para determinar si el software cumple o no con los requisitos de software.

Tabla CXII. Registrar usuarios

Caso de prueba de Caja Negra 1			
Historia de Usuario 1: Registrar Usuarios			
Criterios de Aceptación	Cumple		Observación
	Si	No	
Mostrar el formulario de registro a través de la opción Regístrate.	X		
Validar campos obligatorios vacíos del formulario de registro de usuario.	X		
Mostrar un mensaje de aviso indicando que existen campos obligatorios vacíos.	X		
Validar que el correo tenga un formato válido	X		
Validar que el correo electrónico no exista en la base de datos.	X		
Mostrar un mensaje de aviso indicando que el correo ya existe.	X		
Validar que se haya elegido un rol.	X		
Generar una contraseña temporal.	X		
Crear una cuenta de usuario con los datos de la persona con la información básica ingresada y su respectivo rol.	X		
Mostrar un mensaje de aviso indicando que la cuenta ha sido creada y su clave se ha enviado al correo electrónico.	X		
Notificar la contraseña al correo electrónico del usuario.	X		
Mostrar una opción preguntando si el usuario tiene una cuenta, permitiéndole Iniciar Sesión.	X		

Tabla CXIII. Autenticar usuarios

Caso de prueba de Caja Negra 2			
Historia de Usuario 2: Autenticar Usuarios			
Criterios de Aceptación	Cumple		Observación
	Si	No	
Mostrar el formulario de inicio de sesión a través de la opción Iniciar Sesión.	X		
Validar que el correo y la contraseña hayan sido ingresados.	X		
Mostrar un mensaje de aviso indicando que los datos ingresados están incorrectos cuando el correo electrónico o contraseña no exista.	X		
Mostrar una opción preguntando si el usuario ha olvidado su contraseña.	X		
Mostrar una opción preguntando si el usuario no tiene una cuenta y que le permita acceder al formulario de registro.	X		
Acceder al menú de la Aplicación dependiendo del rol del usuario autenticado.	X		

Tabla CXIV. Gestionar cuentas de usuarios

Caso de prueba de Caja Negra 3			
Historia de Usuario 3: Gestionar Cuentas de Usuarios			
Criterios de Aceptación	Cumple		Observación
	Si	No	
En el panel del administrador, en la sección Arrendador y Arrendatario, podrá el administrador visualizar los usuarios registrados.	X		
Debe permitir filtrar los usuarios por medio del estado de los mismos, para ello debe contener 3 opciones (Activados, Desactivados, Todos).	X		
Al seleccionar el filtro, se deberán mostrar los usuarios que cumplan con esa condición.	X		
En cada ítem se mostrará un botón para activar o desactivar el usuario, también la información de todos los usuarios arrendadores (estado, imagen, nombre, número de celular y teléfono, correo electrónico) y tres botones (Inmuebles, Visitas, Contratos).	X		
Al dar clic en el botón Activar/Desactivar el usuario, se presentará una advertencia indicando lo siguiente: "¿Está seguro de realizar la siguiente acción?"	X		
Al dar clic en el botón Aceptar, se notificará mediante una advertencia que el usuario ha sido Activado o Desactivado	X		
Al desactivar un usuario las opciones de los botones (Inmuebles, Visitas, Contratos) quedarán inhabilitadas.	X		
Al dar clic en el botón Inmuebles del usuario "ACTIVADO", se mostrarán los inmuebles de ese usuario en una nueva sección.	X		
Al dar clic en el botón Visitas del usuario "ACTIVADO", se mostrarán las visitas de ese usuario en una nueva sección.	X		
Al dar clic en el botón Contratos del usuario "ACTIVADO", se mostrarán los contratos de ese usuario en una nueva sección.	X		

Tabla CXV. Gestionar perfil de usuario

Caso de prueba de Caja Negra 4			
Historia de Usuario 4: Gestionar perfil de usuario			
Criterios de Aceptación	Cumple		Observación
	Si	No	
Mostrar el formulario de perfil de usuario a través de la opción "Mi perfil" desde el panel administrativo.	X		
Mostrar información (Nombres, Apellidos, Cédula, Celular, Convencional, foto) del usuario autenticado	X		
El campo cédula será obligatorio y debe validar el número de cédula ingresado, en caso de ser incorrecto deberá indicar que el número de cédula es inválido.	X		
El campo convencional será opcional.	X		
Al completar todos los campos y dar clic en el botón "Actualizar Información", se desplegará un mensaje de aviso, indicando lo siguiente: "Actualización exitosa. Se ha actualizado su información correctamente".	X		
Existirá una sección para actualizar la foto de perfil, donde se podrá buscar una foto en el directorio del equipo del usuario y se habilitará el botón para actualizar la misma.	X		

Tabla CXVI. Resetear contraseña

Caso de prueba de Caja Negra 5			
Historia de Usuario 5: Resetear Contraseña			
Criterios de Aceptación	Cumple		Observación
	Si	No	
Mostrar el formulario de recuperar contraseña a través de la opción ¿Olvidaste tu contraseña?	X		
Validar que se haya ingresado el correo electrónico.	X		
Validar que la cuenta de usuario exista a través del correo electrónico; en caso de que no exista se mostrará un mensaje de aviso.	X		
Generar una contraseña temporal.	X		
Notificar la contraseña al correo electrónico del usuario.	X		
Mostrar un mensaje de aviso indicando que su contraseña ha sido actualizada y se ha enviado al correo electrónico.	X		

Tabla CXVII. Cambiar contraseña

Caso de prueba de Caja Negra 6			
Historia de Usuario 6: Cambiar Contraseña			
Criterios de Aceptación	Cumple		Observación
	Si	No	
Mostrar una acción al usuario autenticado que le permita actualizar su contraseña.	X		
Mostrar un formulario de cambio de contraseña.	X		
Validar que los campos nueva contraseña y confirmación de contraseña no estén vacíos; en caso de estarlos presentar un mensaje indicando que están vacíos.	X		
Debe validar que la nueva contraseña cumpla con la siguientes seguridades mínimas: <ul style="list-style-type: none"> • Tamaño mínimo 6 caracteres. • Debe contener al menos una letra mayúscula. • Debe contener al menos un dígito. 	X		
Validar que la contraseña nueva y la confirmación de la contraseña coincidan.	X		
Mostrar un mensaje de aviso indicando que alguna de las validaciones no se cumple.	X		
Mostrar un mensaje de aviso indicando que su contraseña ha sido actualizada.	X		

Tabla CXVIII. Publicar anuncio

Caso de prueba de Caja Negra 7			
Historia de Usuario 7: Publicar anuncio			
Criterios de Aceptación	Cumple		Observación
	Si	No	
Mostrar el listado de bienes inmuebles registrados por el usuario arrendador.	X		
Publicar el bien inmueble que desee a través del botón publicar.	X		
Mostrar una advertencia indicando lo siguiente: “¿Está seguro de realizar la siguiente acción? El inmueble estará: PÚBLICO”.	X		
Cambiar el estado del bien inmueble ha publicado.	X		
Inhabilitar el botón “publicar” y habilitar el botón “dejar de publicar” el inmueble.	X		
Mostrar un mensaje de aviso indicando lo siguiente: “inmueble PÚBLICO”.	X		
Mostrar el bien inmueble publicado en la pantalla principal de la Aplicación Informática.	X		

Tabla CXIX. Gestión de bienes inmuebles

Caso de prueba de Caja Negra 8			
Historia de Usuario 8: Gestión de bienes inmuebles			
Criterios de Aceptación	Cumple		Observación
	Si	No	
Acceder a la sección de gestión de bienes inmuebles a través de la opción Gestionar Inmuebles del menú del usuario arrendador.	X		
Visualizar el listado de bienes inmuebles registrados.	X		
Buscar bienes inmuebles según el nombre.	X		
Filtrar inmuebles según el estado: "Disponibles", "Ocupados" o "Todos".	X		
Mostrar los bienes inmuebles según el criterio de búsqueda, cuya información a mostrar es: estado, imágenes, nombre, precio, dirección, y acciones (editar, eliminar, publicar, dejar de publicar).	X		

Tabla CXX. Registrar bienes inmuebles

Caso de prueba de Caja Negra 9			
Historia de Usuario 9: Registrar bienes inmuebles			
Criterios de Aceptación	Cumple		Observación
	Si	No	
Mostrar el formulario de registro de un nuevo inmueble a través del botón Nuevo Inmueble de la pantalla gestión de inmuebles.	X		
Cargar las opciones de tipo de inmueble como: casa, departamento, cuarto, mini-departamento en la pantalla de registro de nuevos inmuebles.	X		
Seleccionarlos servicios que incluye un bien inmueble.	X		
Validar campos obligatorios.	X		
Mostrar un mensaje indicando que hay campos obligatorios vacíos.	X		
Controlar si el bien inmueble posee garantía, si no se ingresa un monto, por defecto será 0.	X		
Guardar el inmueble al dar clic en el botón "Guardar inmueble".	X		
Registrar el bien inmueble y mostrar un mensaje notificando al usuario que se ha registrado su inmueble.	X		
Redirigir a otra pantalla donde se podrá subir las imágenes del inmueble.	X		
Cargar imágenes al bien inmueble.	X		
Quitar imágenes (de ser necesario) del bien inmueble.	X		
Mostrar un mensaje indicando lo siguiente: "Imágenes cargadas exitosamente".	X		

Tabla CXXI. Editar bienes inmuebles

Caso de prueba de Caja Negra 10			
Historia de Usuario 10: Editar bienes inmuebles			
Criterios de Aceptación	Cumple		Observación
	Si	No	
Mostrar el formulario de actualizar un inmueble a través de la opción Editar del listado de bienes Inmuebles.	X		
Mostrar los datos del inmueble seleccionado.	X		
Seleccionar o quitar imágenes seleccionadas (de ser necesario) antes de actualizar los datos del bien inmueble.	X		
Cargar las opciones de tipo de inmueble como: casa, departamento, cuarto, mini-departamento.	X		
Seleccionarlos servicios que incluye un bien inmueble.	X		
Validar campos obligatorios.	X		
Mostrar un mensaje indicando que hay campos obligatorios vacíos.	X		
Controlar si el bien inmueble posee garantía, si no se ingresa un monto, por defecto será 0.	X		
Actualizar el inmueble al dar clic en el botón "Guardar inmueble".	X		
Registrar el bien inmueble y mostrar un mensaje notificando al usuario que se ha actualizado su inmueble.	X		
Al dar clic en el botón cargar imágenes se deberá redirigir a otra pantalla donde se podrá subir las imágenes del inmueble.	X		
Cargar imágenes al bien inmueble.	X		
Quitar imágenes (de ser necesario) del bien inmueble.	X		
Mostrar un mensaje indicando lo siguiente: "Imágenes cargadas exitosamente".	X		

Tabla CXXII. Gestión de solicitudes de reserva de alquiler recibidas

Caso de prueba de Caja Negra 11			
Historia de Usuario 11: Gestión de solicitudes de reserva de alquiler recibidas.			
Criterios de Aceptación	Cumple		Observación
	Si	No	
Acceder a la sección de Gestión de Visitas a través de la opción en el menú del usuario arrendador.	X		
Visualizar el listado de solicitud de visitas.	X		
Filtrar las solicitudes de visitas de acuerdo al estado (PENDIENTE, ACEPTADA, ATENDIDA, TODAS).	X		
Mostrar información de la solicitud: imagen del usuario arrendatario, nombre del usuario arrendatario, la descripción de la visita, el inmueble a visitar, fecha de la 55visita,el estado del mismo y 4 acciones (ACEPTAR, RECHAZAR, CONTRATO, VER).	X		

Al seleccionar la opción ACEPTAR se notificará al usuario arrendatario que su solicitud de visita será atendida en la fecha solicitada.	X		
Al seleccionar la opción RECHAZAR se notificará al usuario arrendatario que su solicitud de visita no será atendida por retraso en la fecha solicitada.	X		
Mostrarla notificación en la aplicación del usuario arrendatario.	X		
Al seleccionar la opción VER, se mostrará la solicitud con la siguiente información (la imagen del usuario arrendatario, el nombre, número de celular, el inmueble a visitar, fecha de visita y el estado de la visita).	X		
Se deberá mostrar la información de la solicitud de la visita elegida.	X		
Al seleccionar la opción CONTRATO se redirigirá a la sección de generar un contrato de alquiler.	X		

Tabla CXXIII. Gestionar contratos de alquiler

Caso de prueba de Caja Negra 12			
Historia de Usuario 12: Gestionar contratos de alquiler			
Criterios de Aceptación	Cumple		Observación
	Si	No	
Acceder a la sección de Gestión de Contratos de Alquiler a través de la opción en el menú del usuario arrendador.	X		
Visualizar el listado contratos de alquiler generados.	X		
Buscar contratos según el nombre del contrato.			
Mostrar los contratos de alquiler, según el criterio de búsqueda en una tabla cuyas columnas debe contener: Nombre, Inicio, Finalización, Meses de alquiler, Estado y Acciones (ver).	X		
En la columna Contrato, si el estado está vigente o en borrador, se debe visualizar un texto que diga: "Sin acciones".	X		
En la columna Contrato, si el estado ha terminado, se debe visualizar dos botones: "Terminar el contrato y Publicar el inmueble".	X		
El botón "Terminar el contrato" estará habilitado y el arrendador deberá dar clic y se desplegará una advertencia indicando: "¿Está seguro de realizar la siguiente acción? El contrato será: TERMINADO", al dar clic en aceptar, el contrato se dará por terminado y se deshabilitará el botón.	X		
Una vez terminado el contrato, se habilitará el botón "Publicar el inmueble", al dar clic se desplegará una advertencia indicando: "¿Está seguro de realizar la siguiente acción? El inmueble estará: DISPONIBLE", al dar clic en aceptar, el inmueble estará disponible para de nuevo ser publicado y se deshabilitará el botón.	X		

Al dar clic en el botón de la columna “Acciones” se desplegará el contrato generado, donde se podrá visualizarlo o descargarlo.	X		
---	---	--	--

Tabla CXXIV. Alquilar inmueble

Caso de prueba de Caja Negra 13			
Historia de Usuario 13: Alquilar inmueble			
Criterios de Aceptación	Cumple		Observación
	Si	No	
Mostrar el listado de bienes inmuebles públicos registrados por el usuario arrendador.	X		
Alquilar el bien inmueble que desee a través del botón “Alquilar”.	X		
Redirigir a una nueva sección de un formulario de registro.	X		
Buscar el usuario arrendatario mediante su correo electrónico.	X		
Verificar que el usuario exista, en caso de no existir se deberá mostrar un mensaje al usuario arrendador que “El usuario con el correo no existe”.	X		
Verificar que no se pueda alquilar el bien inmueble a un usuario arrendador ni administrador, para ello se deberá desplegar una advertencia indicando lo siguiente: “No se puede asignar el contrato al usuario”.	X		
El usuario arrendador podrá registrar un usuario arrendatario en caso de no estar registrado en la Aplicación Informática para lo cual deberá ingresar los siguientes datos: Nombres, Apellidos, Celular, Correo electrónico.	X		
Se validarán los campos vacíos, en caso de estarlos, indicar que se deben completar.	X		
Al dar clic en el botón “Registrar arrendatario”, quedará registrado el nuevo usuario con el rol predefinido como ARRENDATARIO.	X		
Se deberá mostrar el nombre y apellido del usuario arrendatario en un campo de texto.	X		
Se podrá seleccionar la fecha de inicio y finalización del contrato mediante un calendario.	X		
Validar que se haya seleccionado una fecha de inicio y finalización del contrato, caso contrario indicar mediante una advertencia, lo siguiente: “Debe ingresar la fecha de finalización del contrato”.	X		
Validar que se seleccione por lo menos un mes en el tiempo del contrato, caso contrario indicar mediante una advertencia, lo siguiente: “El contrato debe por lo menos ser de un mes”.	X		
Se podrá modificar el monto mensual de alquiler y la garantía.	X		
Al dar clic en el botón “Generar contrato”, se mostrará una advertencia, indicando lo siguiente: “¿Está seguro de generar el siguiente contrato de alquiler? Recuerde que	X		

no podrá modificar este contrato, por favor revise bien la información del contrato antes de continuar”.			
Al dar clic en el botón “Generar contrato” con los datos correctamente ingresados, se mostrará en otra página el contrato generado.	X		

Tabla CXXV. Visualizar información de un bien inmueble

Caso de prueba de Caja Negra 14			
Historia de Usuario 14: Visualizar información de un bien inmueble			
Criterios de Aceptación	Cumple		Observación
	Si	No	
Los usuarios podrán visualizar una lista de bienes inmuebles públicos en la pantalla principal con la siguiente información: imagen, nombre, ubicación, descripción del inmueble, el precio de alquiler.	X		
Al dar clic en el botón “Ver inmueble” del anuncio publicado, podrán acceder a la información más detallada de dicho bien inmueble seleccionado.	X		
Mostrar la información de dicho inmueble por separado en una nueva sección.	X		

Tabla CXXVI. Gestión de solicitudes de reserva de alquiler solicitadas

Caso de prueba de Caja Negra 15			
Historia de Usuario 15: Gestionar solicitudes de reserva de alquiler solicitadas			
Criterios de Aceptación	Cumple		Observación
	Si	No	
Al seleccionar el inmueble a visitar, se mostrará un botón que permita solicitar una visita a dicho bien inmueble.	X		
Al dar clic en el botón solicitar una visitase deberá verificar si el usuario está autenticado, caso contrario mostrar una advertencia indicando lo siguiente: “No estás logueado”.	X		
Al dar clic en el botón solicitar una visita y el usuario está logueado, se mostrará la pantalla donde podrá solicitar una visita a dicho inmueble previamente seleccionado.	X		
Se mostrará una sección para solicitar la visita donde se deberá ingresar la fecha de visita y la descripción de la misma.	X		
Deberá existir una descripción predefinida indicando lo siguiente: “Hola, estoy interesado en su inmueble “nombre del inmueble”, y me gustaría estar en contacto con usted para poder llegar a un acuerdo. Muchas gracias por su tiempo, hasta luego”.	X		
Validar que los campos de texto estén completados, en caso de no estar, se deberá advertir al usuario que debe completarlos.	X		
Al dar clic en el botón “Solicitar visita”, se registrará la visita y redirigirá al usuario a la sección de visitas solicitadas.	X		

Mostrar una tabla con las visitas realizadas por el usuario con los siguientes campos: Estado, Fecha solicitada, Ver inmueble.	X		
En el campo ver inmueble existirá un botón que permita ver la visita a más detalle en una nueva página.	X		
En el campo ver inmueble existirá un botón que permita eliminar o dar de baja la visita.	X		

Tabla CXXVII. Seleccionar filtros de búsqueda

Caso de prueba de Caja Negra 16			
Historia de Usuario 16: Seleccionar filtros de búsqueda			
Criterios de Aceptación	Cumple		Observación
	Si	No	
En la página principal se mostrará 3 opciones para seleccionar el tipo de inmueble, precio y ubicación para poder filtrar los inmuebles en base a dichos criterios de búsqueda.	X		
El primer criterio de búsqueda "Tipo de inmueble" constará de todos los tipos de inmuebles que se hayan definido.	X		
En el segundo criterio tendrá rangos de precios desde 5-100, 100-150, 150-200 y >200.	X		
En el tercer criterio de búsqueda contendrá todas las ubicaciones previamente cargadas.	X		
Al dar clic en el botón "Buscar" se filtrarán los inmuebles que cumplan con todos los filtros seleccionados.	X		
Mostrar inmuebles que hayan cumplido el filtro de búsqueda.	X		

Tabla CXXVIII. Enviar mensajes

Caso de prueba de Caja Negra 17			
Historia de Usuario 17: Enviar mensajes			
Criterios de Aceptación	Cumple		Observación
	Si	No	
En la página principal se mostrará una opción "Contáctanos" la cual permitirá al usuario enviar un mensaje al administrador.	X		
Mostrar un formulario solicitando la siguiente información: Título, Asunto y correo.	X		
Se deberá validar que se ingrese la información en el formulario.	X		
Indicar al usuario que se deben completar los campos de texto.	X		
Verificar que el correo del usuario exista, en caso de no existir, indicarle que el correo no existe.	X		
Si todos los datos ingresados son correctos, al dar clic en el botón "Enviar mensaje" se enviará el mensaje al administrador.	X		

Notificar al usuario que su mensaje ha sido enviado.	X		
--	---	--	--

Tabla CXXIX. Gestionar servicios básicos

Caso de prueba de Caja Negra 18			
Historia de Usuario 18: Gestionar Servicios Básicos			
Criterios de Aceptación	Cumple		Observación
	Si	No	
En el panel administrativo en la sección de Configuraciones, en la opción Servicios, al seleccionar, se mostrará todos los Servicios Básicos registrados.	X		
Debe permitir registrar nuevos servicios básicos, a través de un botón "Registrar Servicio".	X		
Se debe validar que el usuario ingrese el nombre del servicio básico, en caso de no ingresar el nombre del servicio básico, deberá indicar mediante una advertencia lo siguiente: "El nombre del servicio es obligatorio".	X		
Se debe validar que el nombre del servicio básico no se repita, en caso de repetirse, deberá indicar mediante una advertencia lo siguiente: "El servicio ya existe, ingresa uno diferente".	X		
Guardar el servicio básico al dar clic en el botón "Guardar servicio".	X		
Registrar el servicio básico y mostrar un mensaje notificando al usuario que se ha registrado su servicio básico.	X		
Al dar clic en el botón de editar, del campo Acciones, se deberá mostrar en una nueva sección el servicio básico seleccionado.	X		
Cambiar el nombre del servicio básico, en caso de no existir el nombre, se deberá indicar mediante una advertencia, lo siguiente: "El nombre del servicio es obligatorio".	X		
Actualizar el servicio básico al dar clic en el botón "Guardar servicio".	X		
Registrar el servicio básico y mostrar un mensaje notificando al usuario que se ha actualizado su servicio básico.	X		
Al dar clic en el botón eliminar, se deberá indicar mediante una advertencia, lo siguiente: "¿Está seguro de borrar el servicio? Está a punto de borrar el servicio".	X		
Al dar clic en el botón eliminar de la advertencia, se eliminará el servicio básico y se notificará al usuario mediante una advertencia.	X		
Mostrar los servicios básicos actualizados en la tabla.	X		
Se podrá buscar los servicios básicos mediante el nombre del mismo.	X		

Con la ejecución de las pruebas realizadas por parte del ing. Wilman Patricio Chamba Zaragocín, encargado de la revisión y aprobación del documento de pruebas se identificaron funcionalidades que se habían omitido y mediante las mismas se lograron completar, concluyendo de esta manera con el cumplimiento de todos los criterios de aceptación descritos en los diferentes casos de prueba de caja negra.

Anexo 10: Pruebas de Carga y Estrés

Pruebas de Carga y Estrés

Proyecto: Desarrollo de una Aplicación Informática para la búsqueda de oferta de alquiler de inmuebles en la ciudad de Loja.

VALIDACIÓN DE LAS PRUEBAS DE CARGA Y ESTRÉS PARA LA APLICACIÓN LOJAHOUSE.

E = Encargado de la elaboración del documento.

R = Encargado de la revisión del documento.

A = Encargado de la aprobación del documento de pruebas de carga y estrés.

Nombre	Cargo	E	R	A	Firma
Steeven Michael Armijos Bravo	Tesista	X			
Wilman Patricio Chamba Zaragocín	Revisor Técnico		X	X	

1. Introducción

El presente documento muestra las pruebas de carga y estrés de la aplicación LojaHouse, para determinar en qué nivel de carga falla la aplicación, además permitirá conocer el límite que puede soportar la aplicación antes de perder rendimiento.

2. Propósito

Conocer el nivel de carga que puede soportar la aplicación LojaHouse antes de perder rendimiento, es decir, el número de usuarios que pueden utilizar simultáneamente la aplicación, para lo cual se utilizó la herramienta de software Apache JMeter.

3. Referencias

Referencia	Sitio Web
Apache JMeter	https://jmeter.apache.org/

4. Pruebas de Carga y Estrés

La aplicación LojaHouse se encuentra alojada en el servidor gratuito de Firebase Hosting que provee 10 GB de almacenamiento y una transferencia de datos de 360MB por día.

Para la creación de la prueba se configuró la herramienta JMeter como se muestra en la Figura 53, definiendo el nombre del proyecto para la prueba de la aplicación LojaHouse.

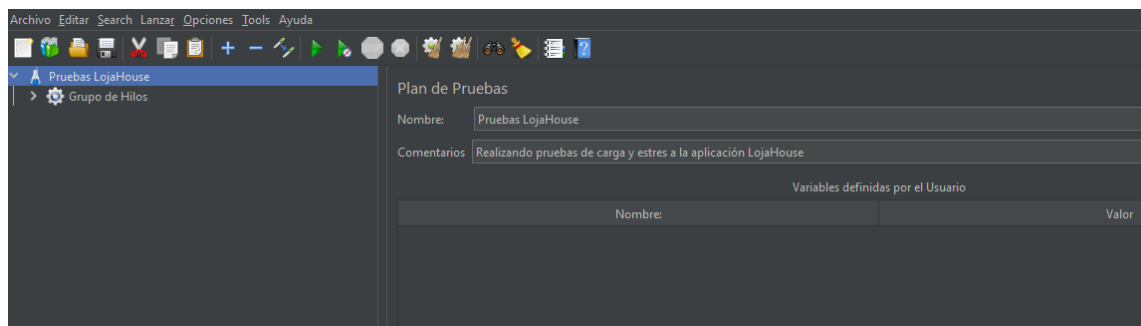


Figura 53. Creación de la prueba en apache Jmeter

En la Figura 54 se muestra la configuración del grupo de hilos, donde se define las peticiones de tipo GET y POST que se estima se podrían producir cuellos de botella, configurando 1000 peticiones en un periodo de un segundo que se ejecutarán de manera concurrente.

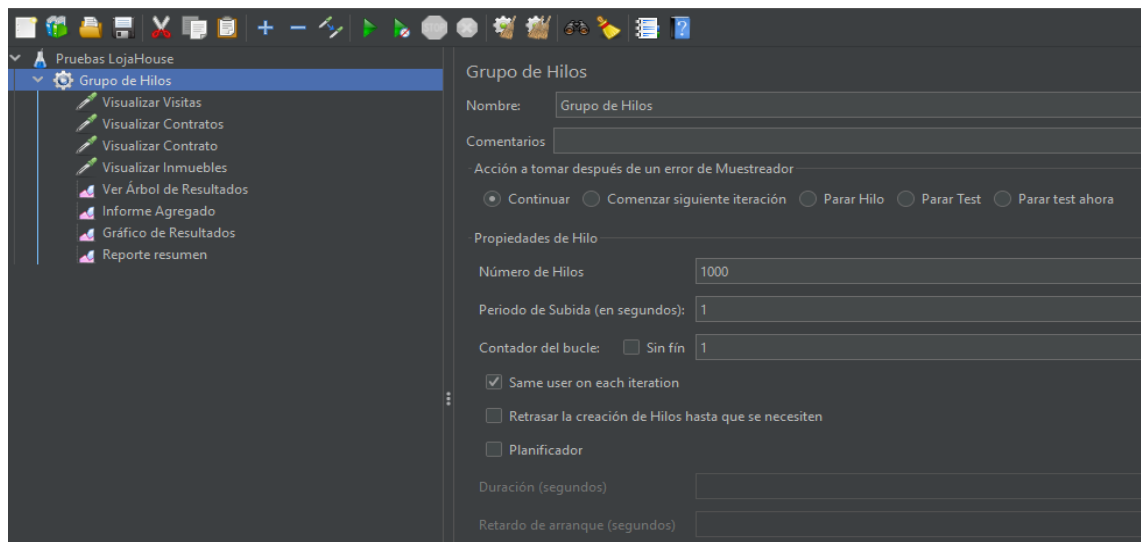


Figura 54. Configuración de peticiones

Al finalizar la prueba se obtuvo un gráfico de resultados donde se puede observar como la aplicación LojaHouse al procesar 1000 peticiones por segundo logra mantener un rendimiento estable a lo largo de la ejecución de la prueba de carga y estrés como se muestra en la Figura 55.

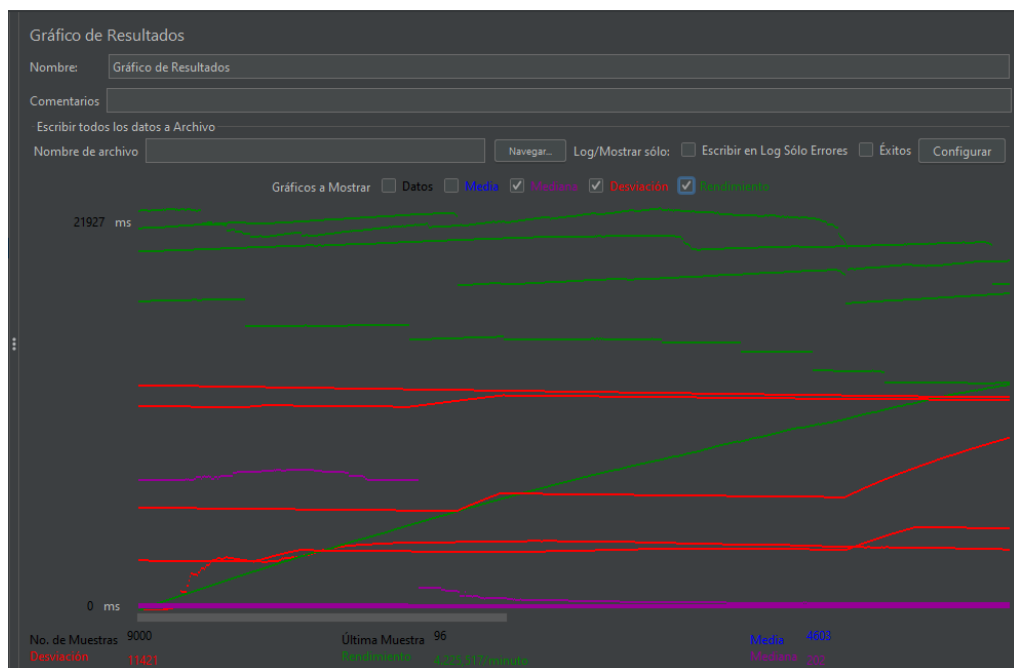


Figura 55. Gráfico de resultados de la prueba de carga y estrés

Por otro lado, en la Figura 56 se puede observar el resultado en forma de resumen, destacando el error obtenido en la prueba, en este caso se obtuvo un porcentaje de 6.36%, es decir, la aplicación LojaHouse tiende a fallar más al realizar peticiones concurrentes en la página principal, por otro lado, la página de login tiende a fallar en menor medida. Aun así, se puede evidenciar que la aplicación soportó 1000 peticiones concurrentes presentando un rendimiento estable y solo en las peticiones que se han generado error ha presentado variación, dando un total de 70,4% de rendimiento, por lo que se concluye que la aplicación funciona de forma correcta con tiempos de respuesta mínimos de 79 milisegundos y como máximo de 42418 segundos.

Reporte resumen

Nombre:

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 Byt...
Principal	1000	29284	7079	42418	14185,40	47,90%	23,6/sec	70,72	1,48	3072,2
Login	1000	9151	278	42124	13774,98	9,20%	12,9/sec	40,66	2,37	3220,2
Buscar inmu...	1000	1540	80	42081	5417,05	0,10%	8,4/sec	26,45	1,01	3222,2
Visualizar De...	1000	221	80	4717	312,69	0,00%	8,4/sec	26,40	1,01	3222,5
Solicitar visita	1000	472	278	4274	331,49	0,00%	8,4/sec	26,59	4,38	3252,0
Obtener con...	1000	197	80	3047	255,77	0,00%	8,4/sec	26,44	1,01	3222,5
Visualizar Co...	1000	212	79	2825	267,28	0,00%	8,4/sec	26,45	1,01	3222,6
Obtener inm...	1000	183	81	3141	195,45	0,00%	8,4/sec	26,46	1,01	3222,6
Obtener visitas	1000	168	80	2357	182,83	0,00%	8,4/sec	26,47	1,01	3222,6
Total	9000	4603	79	42418	11421,85	6,36%	70,4/sec	220,69	11,65	3208,8

Figura 56. Reporte de resumen de resultados de la ejecución de la prueba de carga y estrés

Anexo 11: Arquitectura de Software

Proyecto: Desarrollo de una Aplicación Informática para la búsqueda de oferta de alquiler de inmuebles en la ciudad de Loja.

Documento Arquitectura de Software

4. Introducción

El presente documento contiene la arquitectura de software de la Aplicación Informática LojaHouse, la cual es muy importante para que se pueda desarrollar de manera exitosa el producto de software, ya que permite establecer el comportamiento e interacción del sistema y tener una visión clara de lo que se espera construir.

5. Propósito

El presente documento tiene como propósito definir la arquitectura de software de la Aplicación Informática LojaHouse empleando el modelo arquitectónico 4+1, donde se detalla mediante diversas vistas el comportamiento del sistema.

6. Alcance

El presente documento detalla la arquitectura de software de la Aplicación Informática LojaHouse, a través del modelo 4+1 [41], el mismo que propone describir la arquitectura de software por medio de diferentes vistas, como son: vista lógica, vista de despliegue, vista de escenarios, vista física y vista de procesos.

7. Referencias

Tabla Anexo 1. REFERENCIA ARQUITECTURA DE SOFTWARE

Referencia	Título del Documento
Anexo I	Especificación de Requerimientos
Modelo	Arquitectura de Software 4+1 [41]

8. Vista Global

El presente documento detalla cómo se encuentra organizada la arquitectura de la Aplicación Informática LojaHouse mediante el uso del modelo 4+1 [41], el cual es un modelo para describir la arquitectura de sistemas de software, basándose en el uso de múltiples vistas concurrentes; la vista lógica que detalla la estructura y funcionalidad; la vista de despliegue que representa cómo interactúan los componentes del software del sistema; la vista de escenarios que son los casos de uso del sistema; la vista física que describe los componentes físicos con los que interactúa el sistema y la vista de procesos que describe los procesos del sistema y su forma de comunicación.

9. Representación de la Arquitectura

La Aplicación Informática utiliza la arquitectura cliente servidor, a través del modelo 4+1 [41], mismo que propone cinco vistas, las cuales se detallan en la Tabla CXXX.

Tabla CXXX. Arquitectura 4+1

Vista	Elemento Modelado	Descripción
Vista de Escenarios	Caso de Uso	Muestra la interacción que tienen los actores en el sistema.
Vista Lógica	Modelo Conceptual Diagrama de Clases	Representa las funcionalidades y el servicio que proporciona a los usuarios.
Vista Física	Diagrama de Despliegue	Muestra los componentes físicos del sistema.
Vista de Despliegue	Diagrama de Componentes	Describe los componentes del sistema con el fin que el programador entienda las interacciones que existen.
Vista de Procesos	Diagrama de Actividad	Describe los procesos de la funcionalidad del sistema.

10. Objetivos de la Arquitectura

La Aplicación Informática cumplirá con lo siguiente:

- **Rendimiento:** El tiempo de respuesta en las operaciones debe ser inferior o igual a 5 segundos.
- **Seguridad:**
 - La Aplicación Informática deberá encriptar la contraseña de los usuarios registrados mediante el método de encriptación bcrypt.
 - Para controlar el ingreso a las diferentes funcionalidades de la Aplicación Informática se lo realizará mediante el uso de JWT.
 - La Aplicación Informática funcionará bajo el protocolo HTTP.
 - La Aplicación Informática deberá restringir el acceso a los usuarios dependiendo de su rol dentro de la Aplicación Informática.
- **Fiabilidad:** La Aplicación Informática deberá registrar los datos ingresados por parte de los usuarios de manera correcta y al producirse un error de cualquier índole, se deberá emitir un mensaje dentro de la Aplicación.

- **Disponibilidad:** La disponibilidad de la Aplicación Informática será de los 365 días del año, cuya afección producida dependerá del servicio de hosting que se contrate.
- **Portabilidad:** La Aplicación Informática podrá ser utilizada tanto en navegadores web y Smartphone con sistema operativo Android.
- **Usabilidad:** La Aplicación Informática debe ser desarrollada con una interfaz amigable para el usuario y de fácil comprensión, con la finalidad que pueda acceder a las diferentes funcionalidades de forma rápida e intuitiva.

11. Vistas de Escenarios

En esta vista se presenta los diagramas de caso de uso, donde se muestra la interacción del usuario y la aplicación móvil y web.

11.1. Diagramas de Caso de Uso

El diagrama de caso de uso se compone de tres actores: Administrador, Arrendador y Arrendatario; cada actor realiza actividades específicas dependiendo del rol, para ello deberán estar autenticados, ver Figuras 57-60.

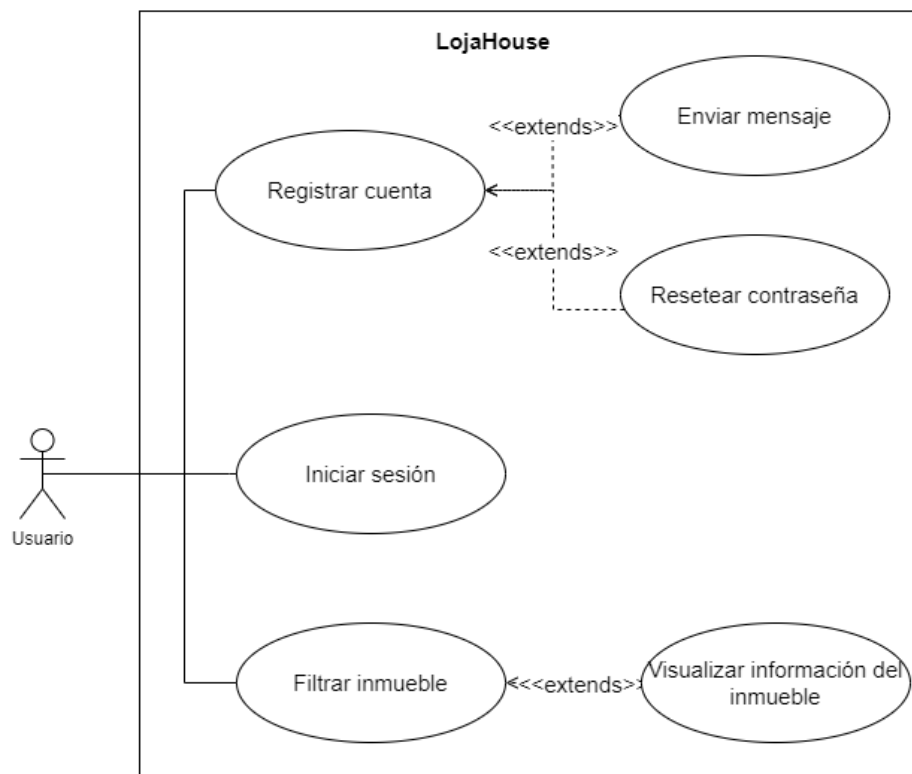


Figura 57. Diagrama de caso de uso -Usuario no registrado.

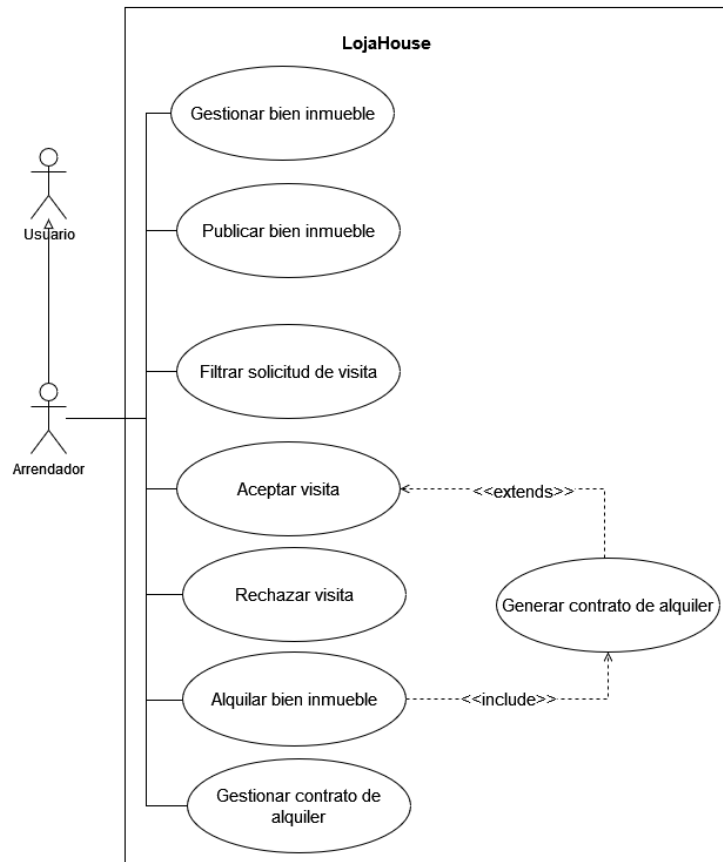


Figura 58. Diagrama de caso de uso -Usuario Arrendador

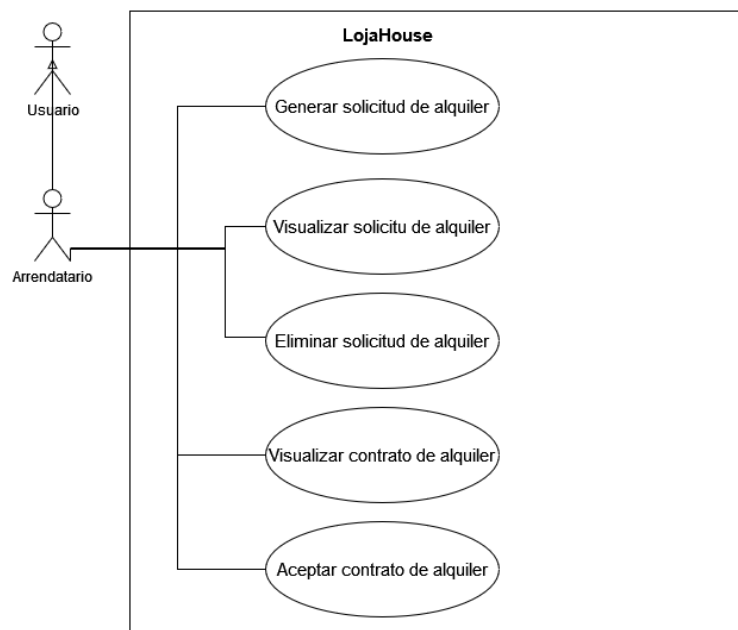


Figura 59. Diagrama de caso de uso -Usuario Arrendatario

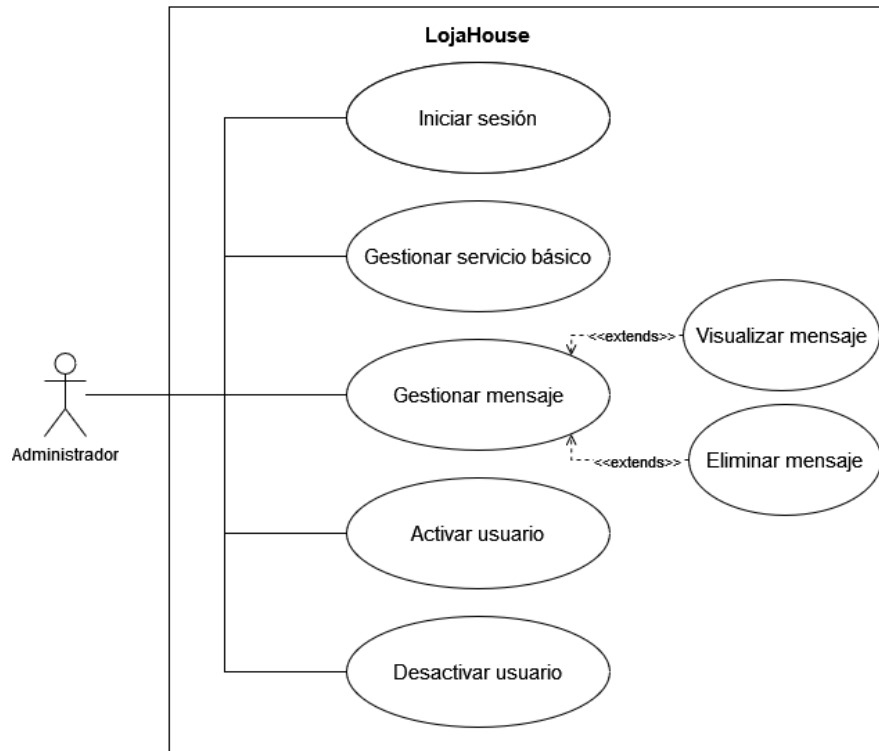


Figura 60. Diagrama de caso de uso -Usuario Administrador

El usuario Administrador puede realizar las funciones de:

- Iniciar sesión
- Gestionar servicios básicos.
- Gestionar mensaje.
- Activar usuario.
- Desactivar usuario.
- Gestionar perfil de usuario.

El usuario Arrendador puede realizar las funciones de:

- Iniciar sesión.
- Gestionar bien inmueble.
- Publicar bien inmueble.
- Filtrar solicitud de visita.
- Aceptar visita.
- Rechazar visita.
- Alquilar bien inmueble.

- Gestionar contrato de alquiler.
- Gestionar perfil de usuario.

El usuario Arrendatario puede realizar las funciones de:

- Iniciar sesión.
- Generar solicitud de visita de alquiler.
- Visualizar solicitud de visita de alquiler.
- Eliminar solicitud de visita de alquiler.
- Visualizar contrato de alquiler.
- Aceptar contrato de alquiler.
- Gestionar perfil de usuario.

12. Vista Lógica

En esta vista se hace referencia a los requerimientos funcionales de la aplicación LojaHouse, para ello se incluirá el Modelo Conceptual y el Diagrama de Clases.

12.1. Modelo Conceptual

En la Figura 61 se puede observar el modelo estático de objetos que componen la solución a la búsqueda de alquiler de inmuebles.

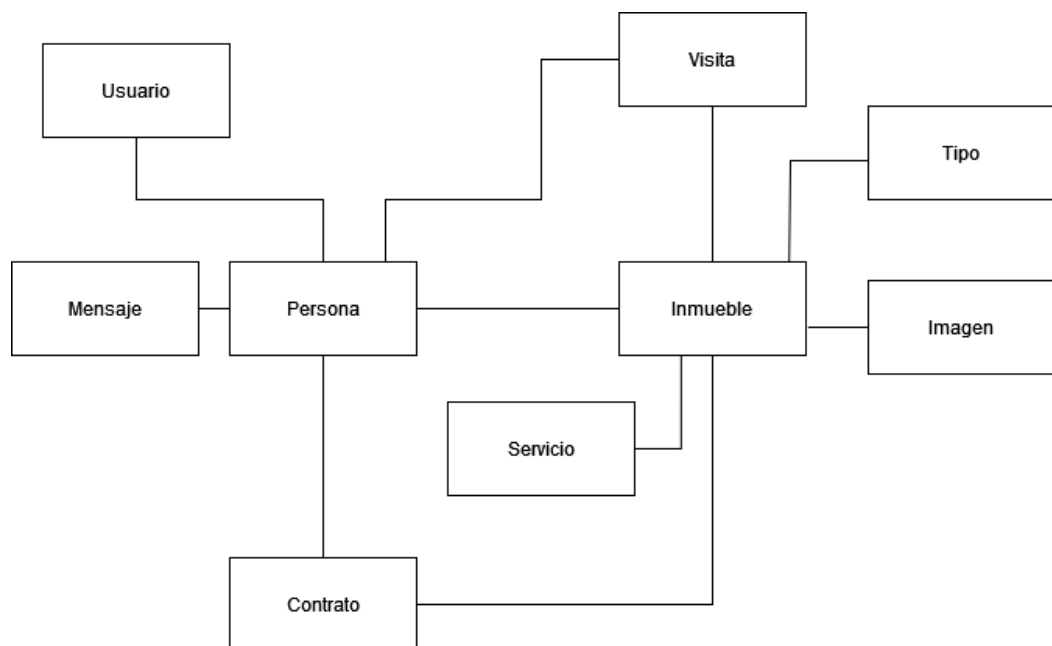


Figura 61. Modelo conceptual

12.2. Diagrama de clases

La Figura 62 muestra el diagrama de clases donde se identifica las diferentes clases con los atributos que se utilizaron para modelar la base de datos.

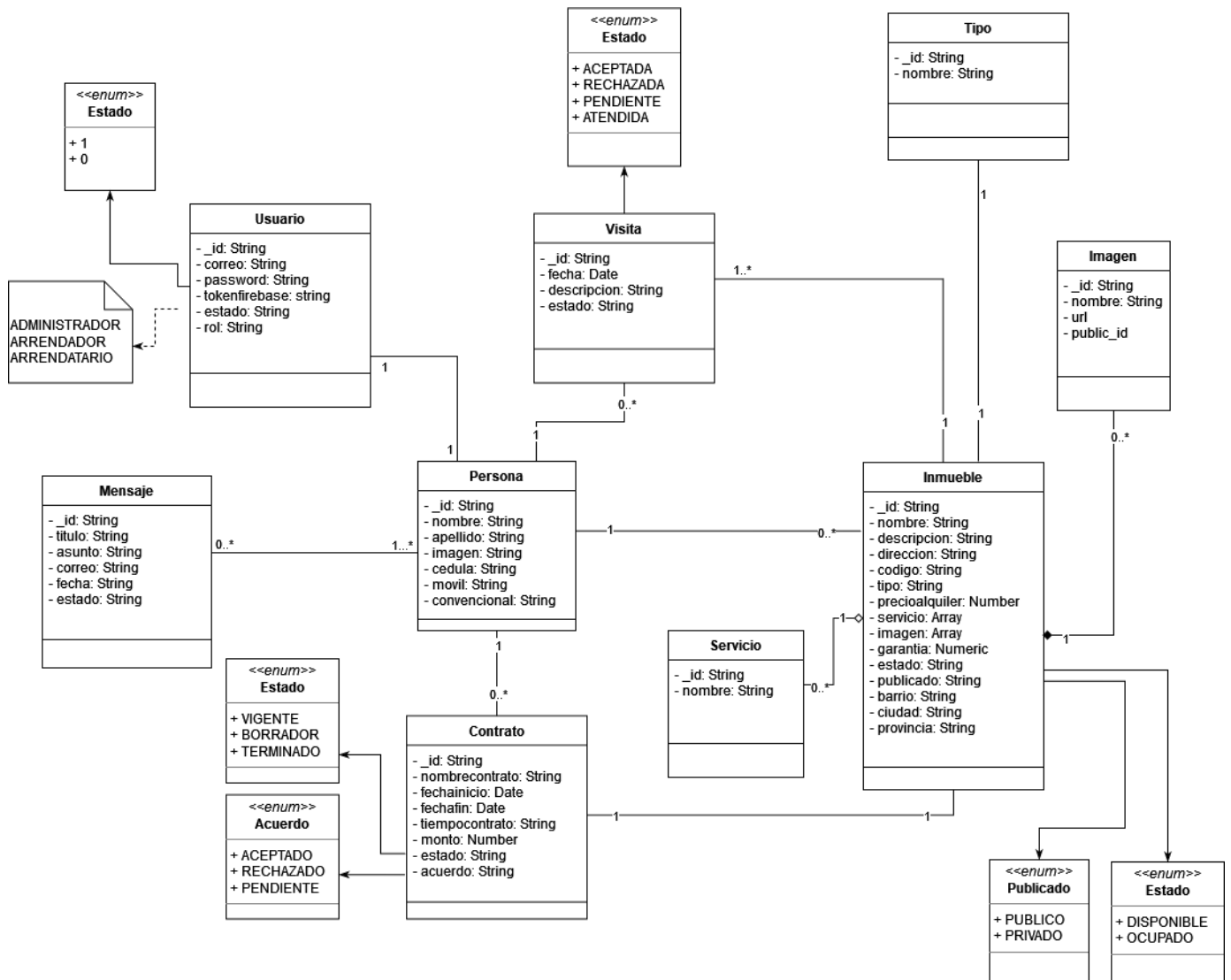


Figura 62. Diagrama de clases

13. Vista de Procesos

En esta vista se detalla el diagrama de actividades de la aplicación.

13.1. Diagrama de Actividades

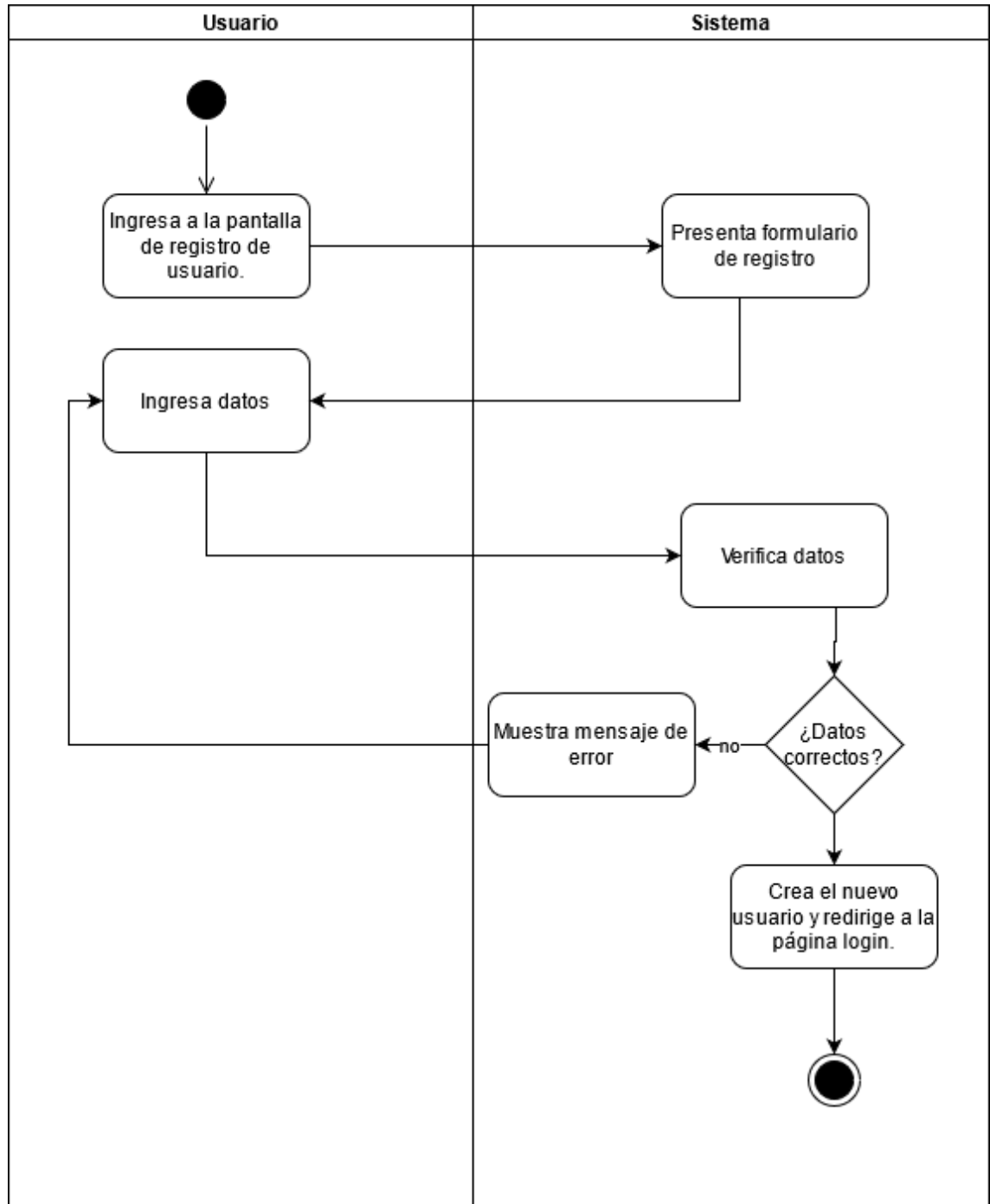


Figura 63. Diagrama de actividades - Registrar cuenta

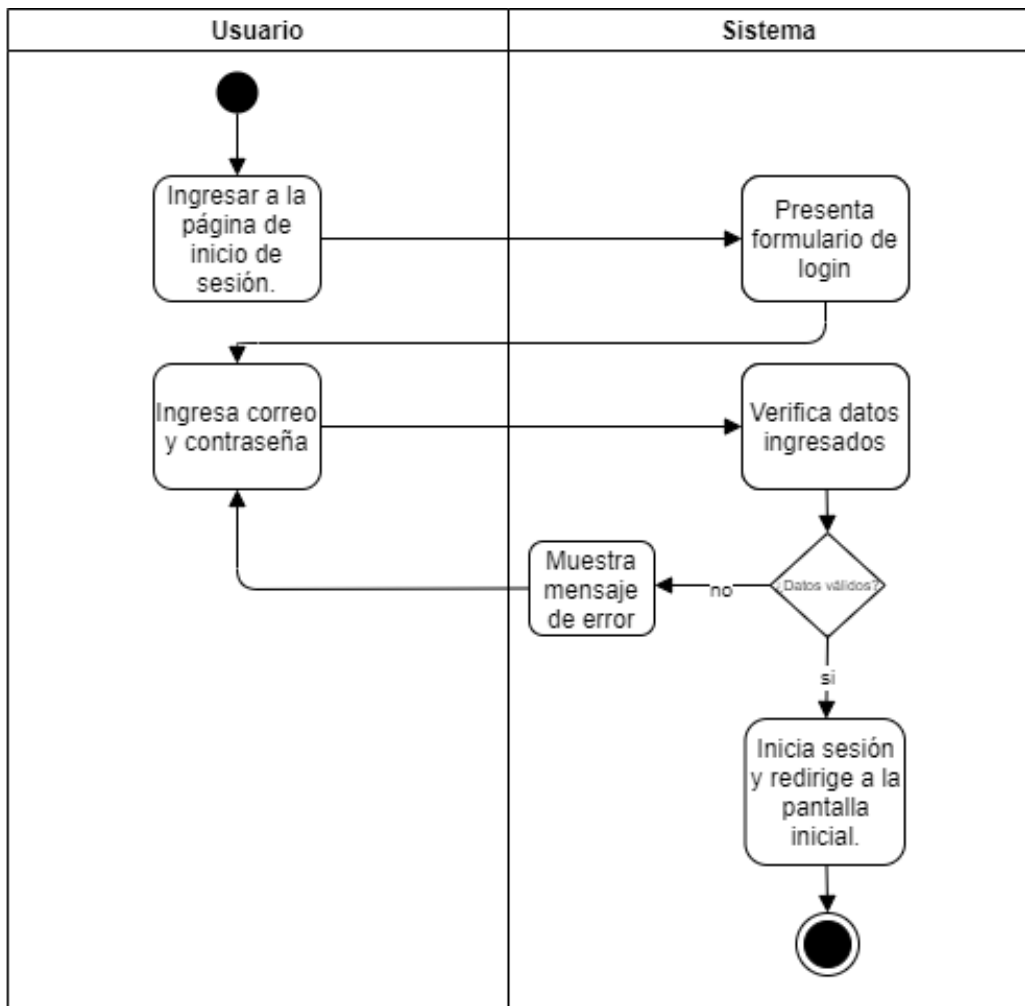


Figura 64. Diagrama de actividades - Inicio de sesión

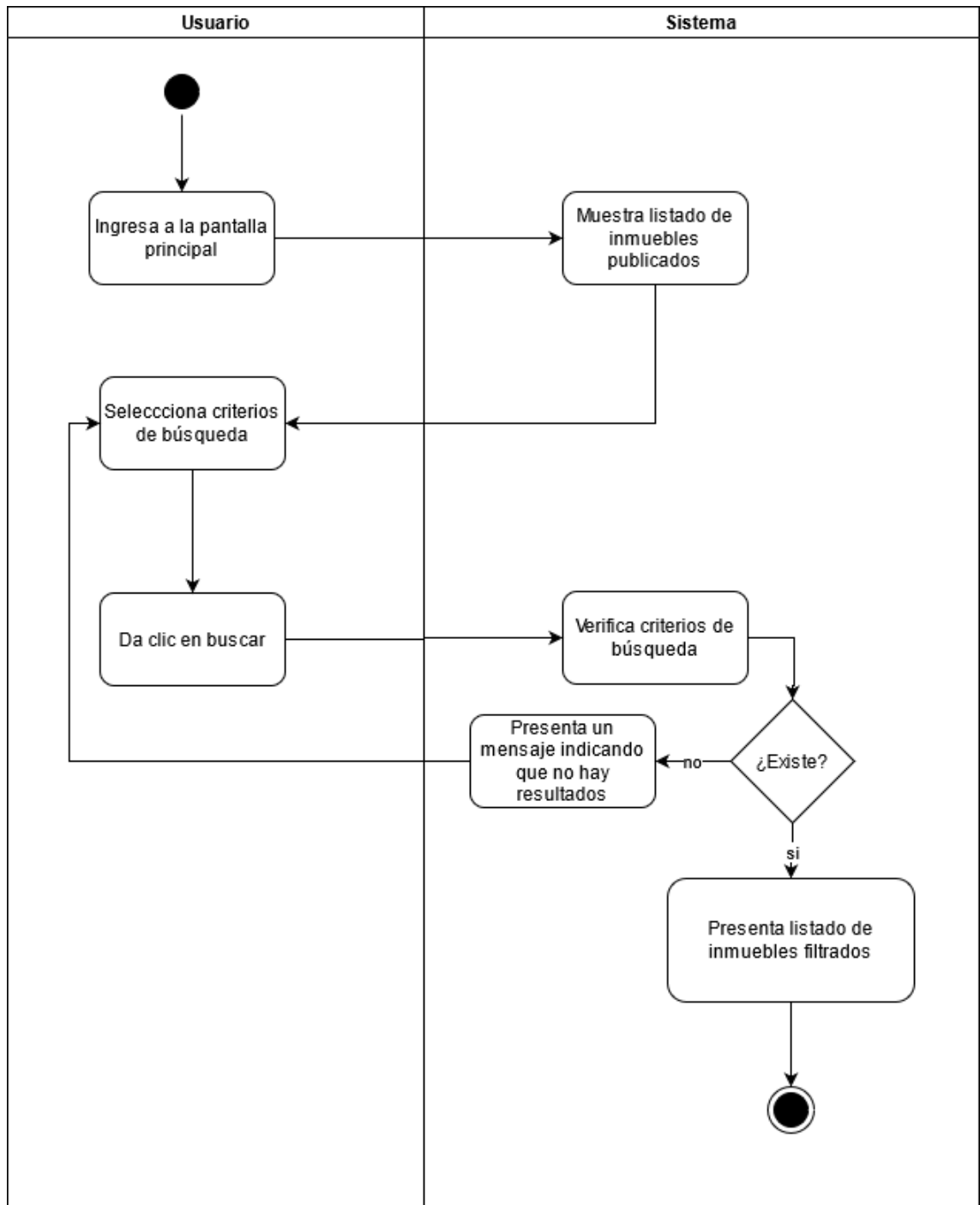


Figura 65. Diagrama de actividades - Filtrar inmuebles

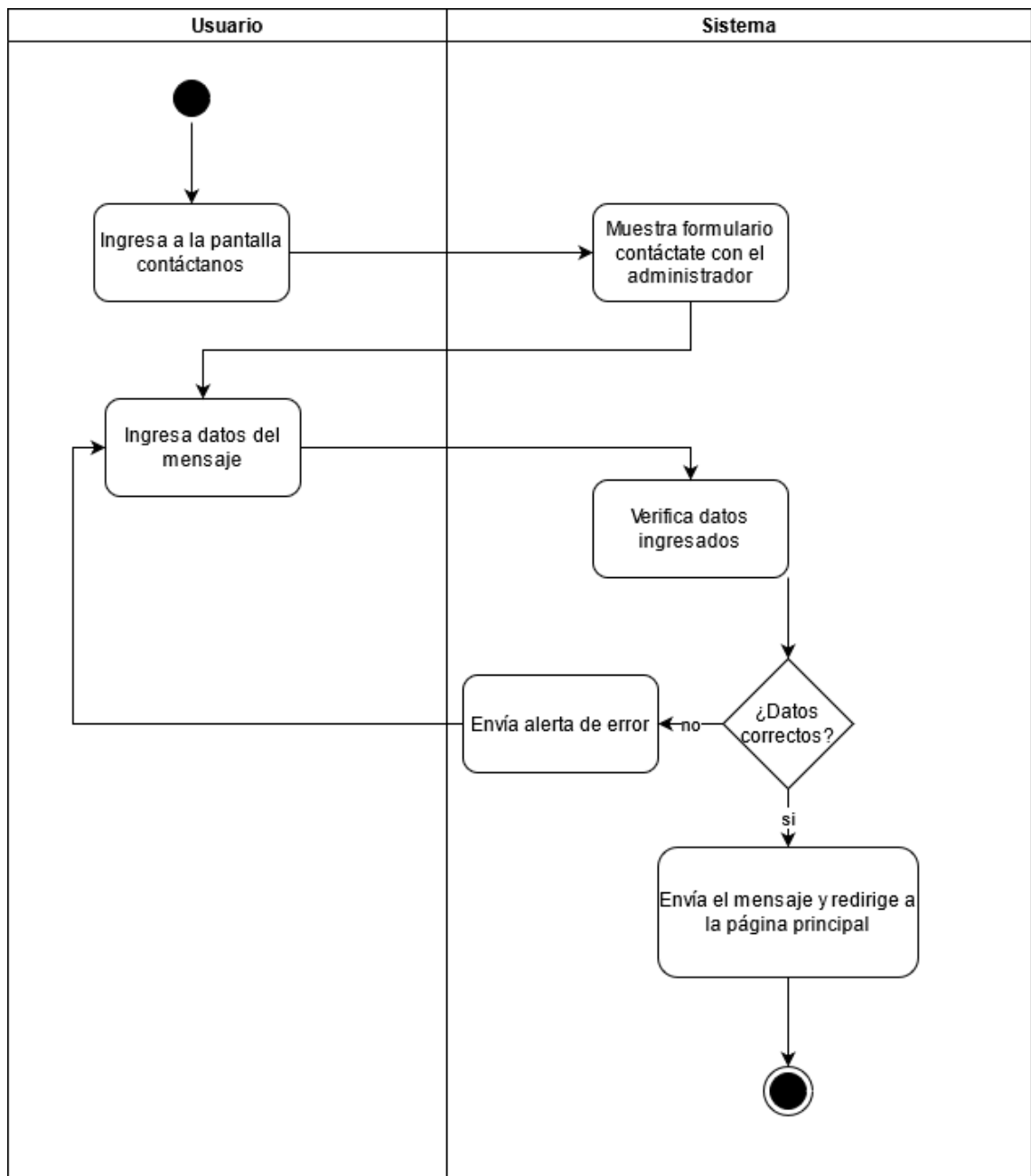


Figura 66. Diagrama de actividades - Enviar mensaje

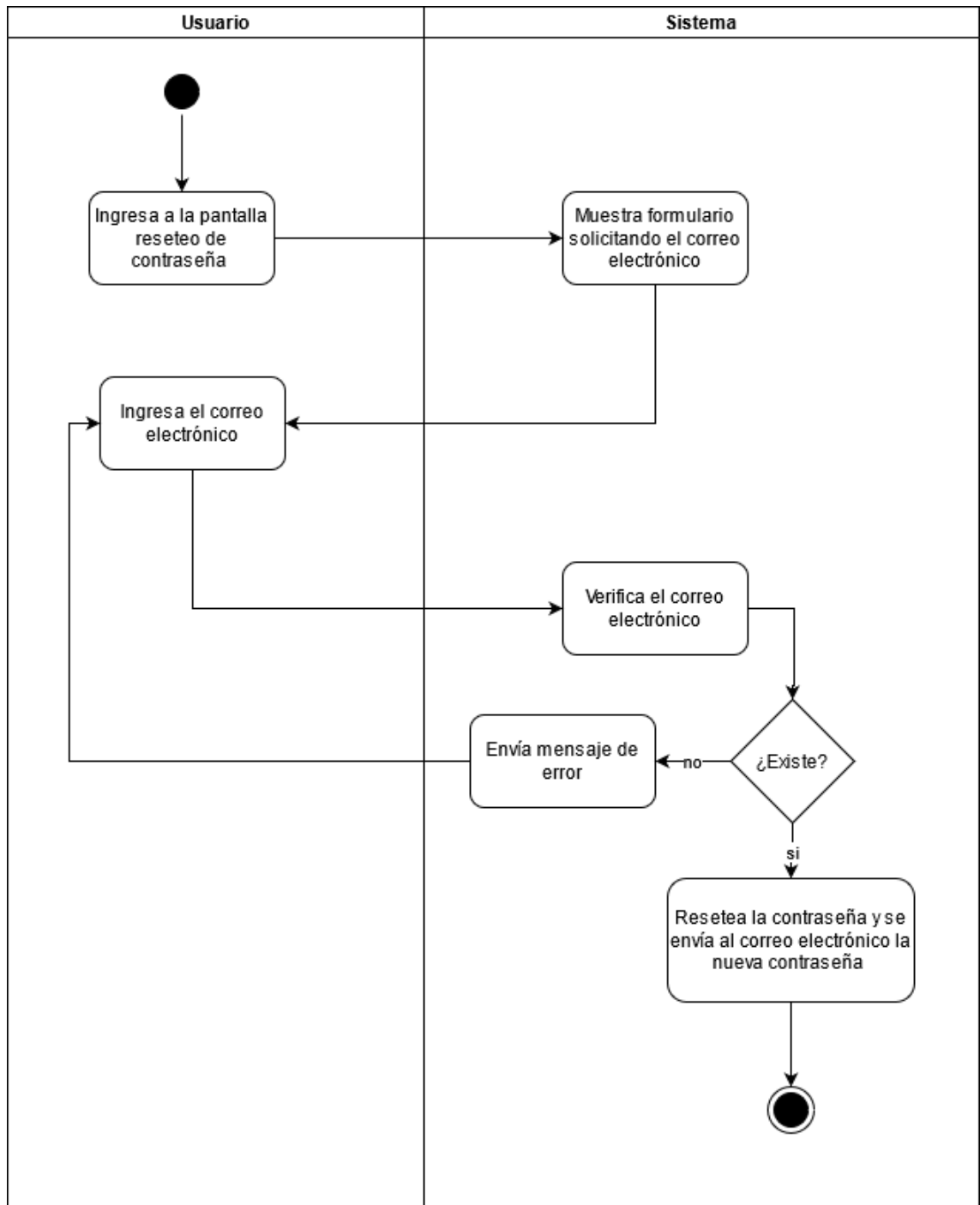


Figura 67. Diagrama de actividades - Resetear contraseña

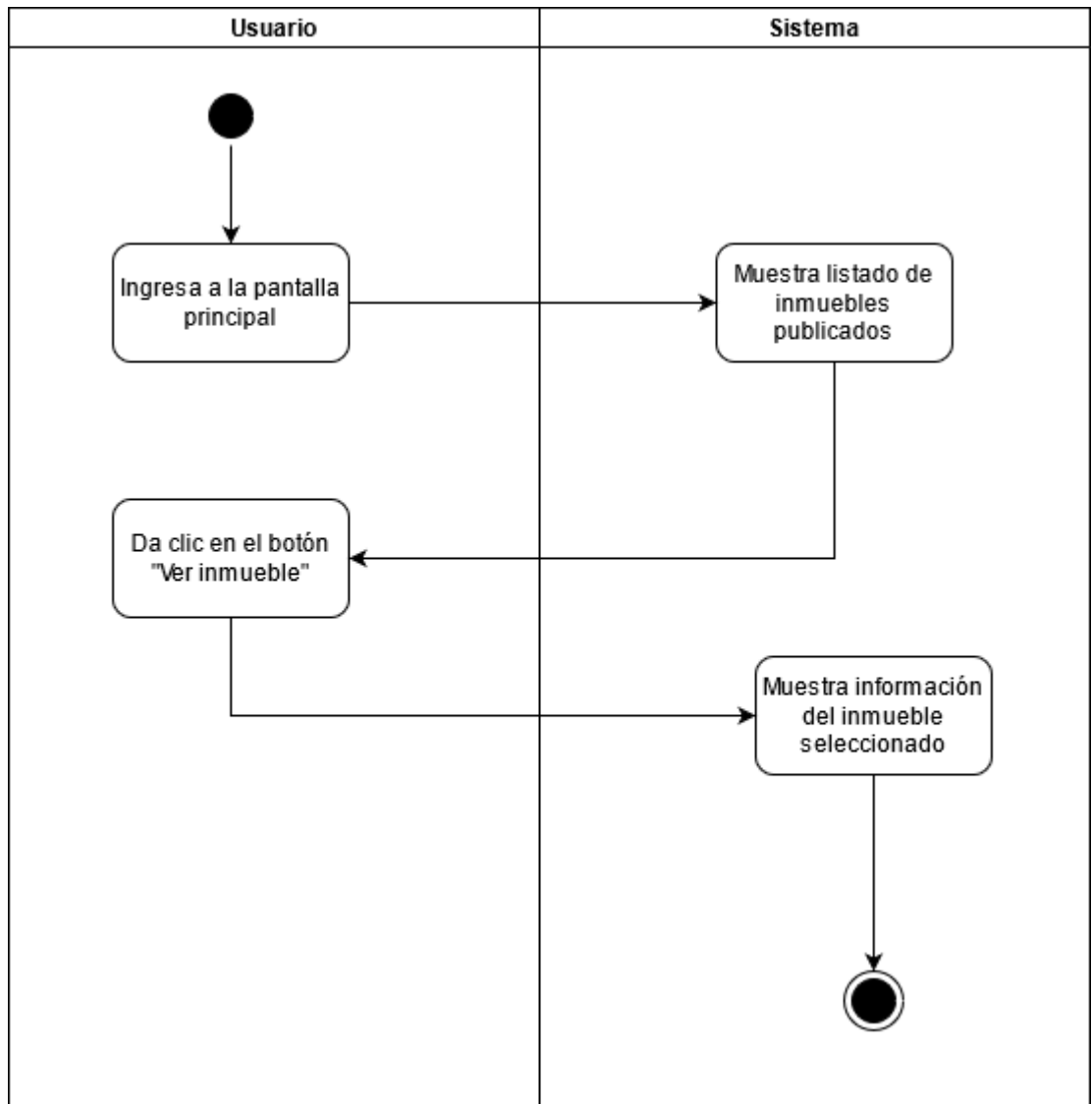


Figura 68. Diagrama de actividades - Ver información del inmueble

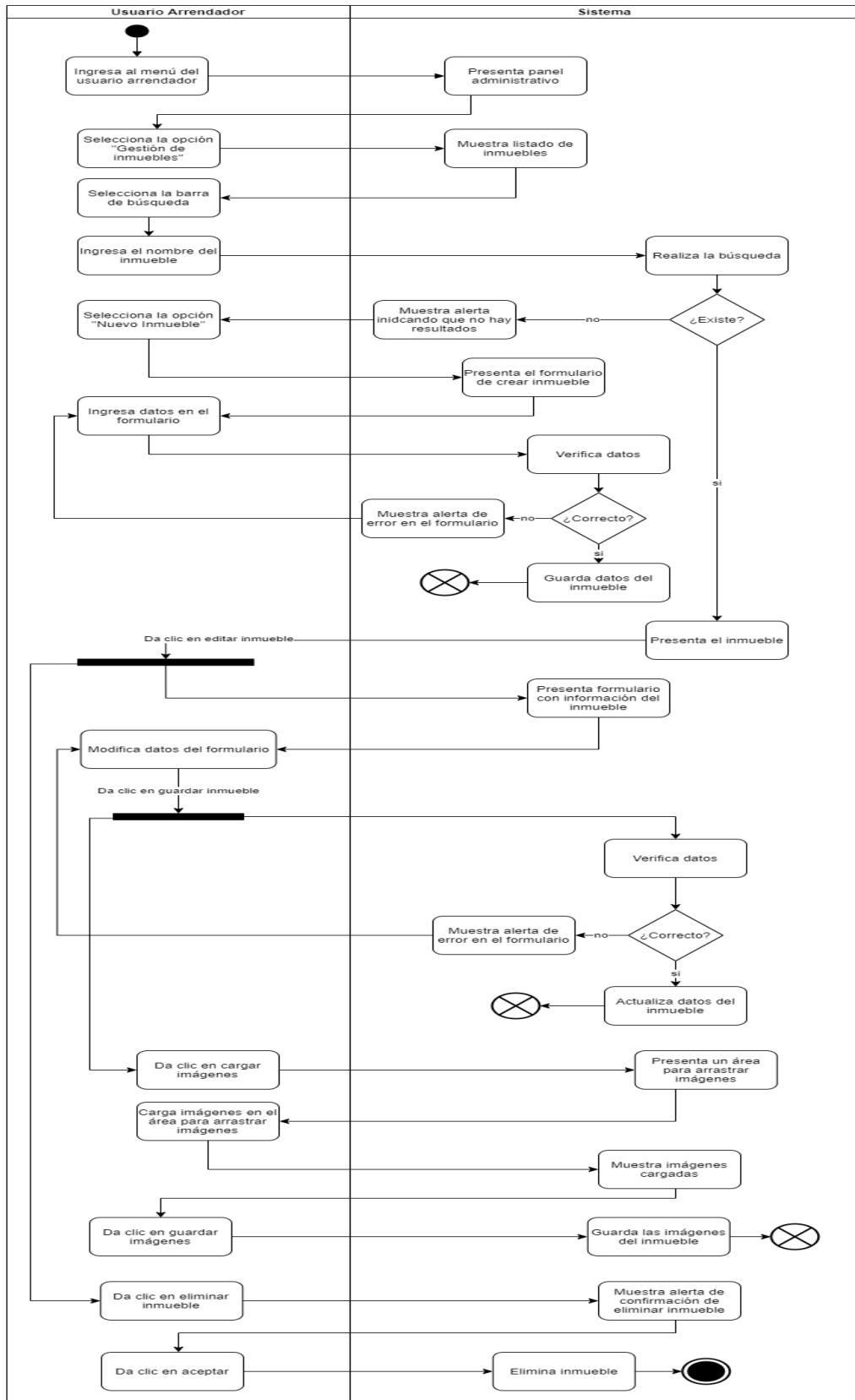


Figura 69. Diagrama de actividades - Gestionar bien inmueble

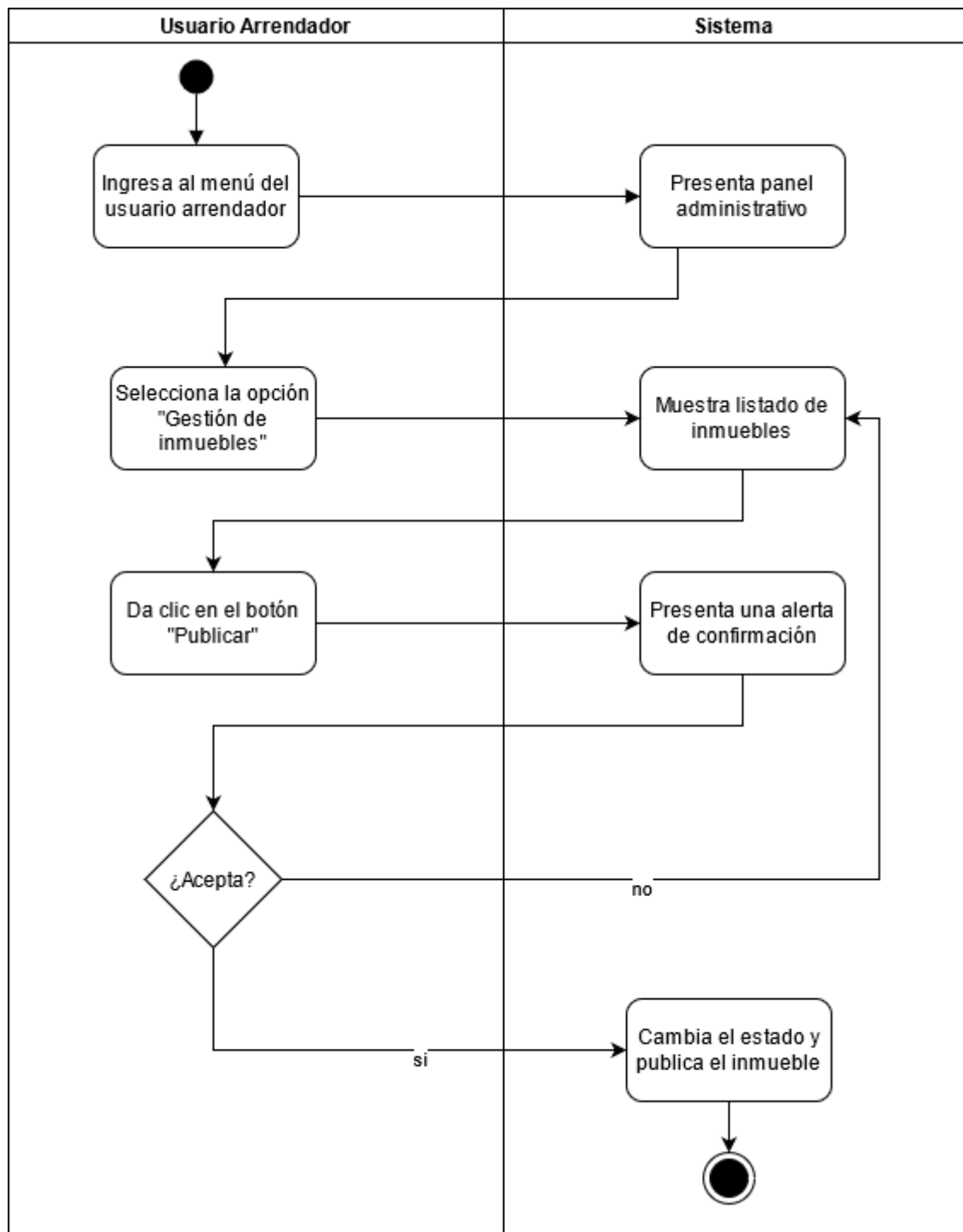


Figura 70. Diagrama de actividades - Publicar bien inmueble

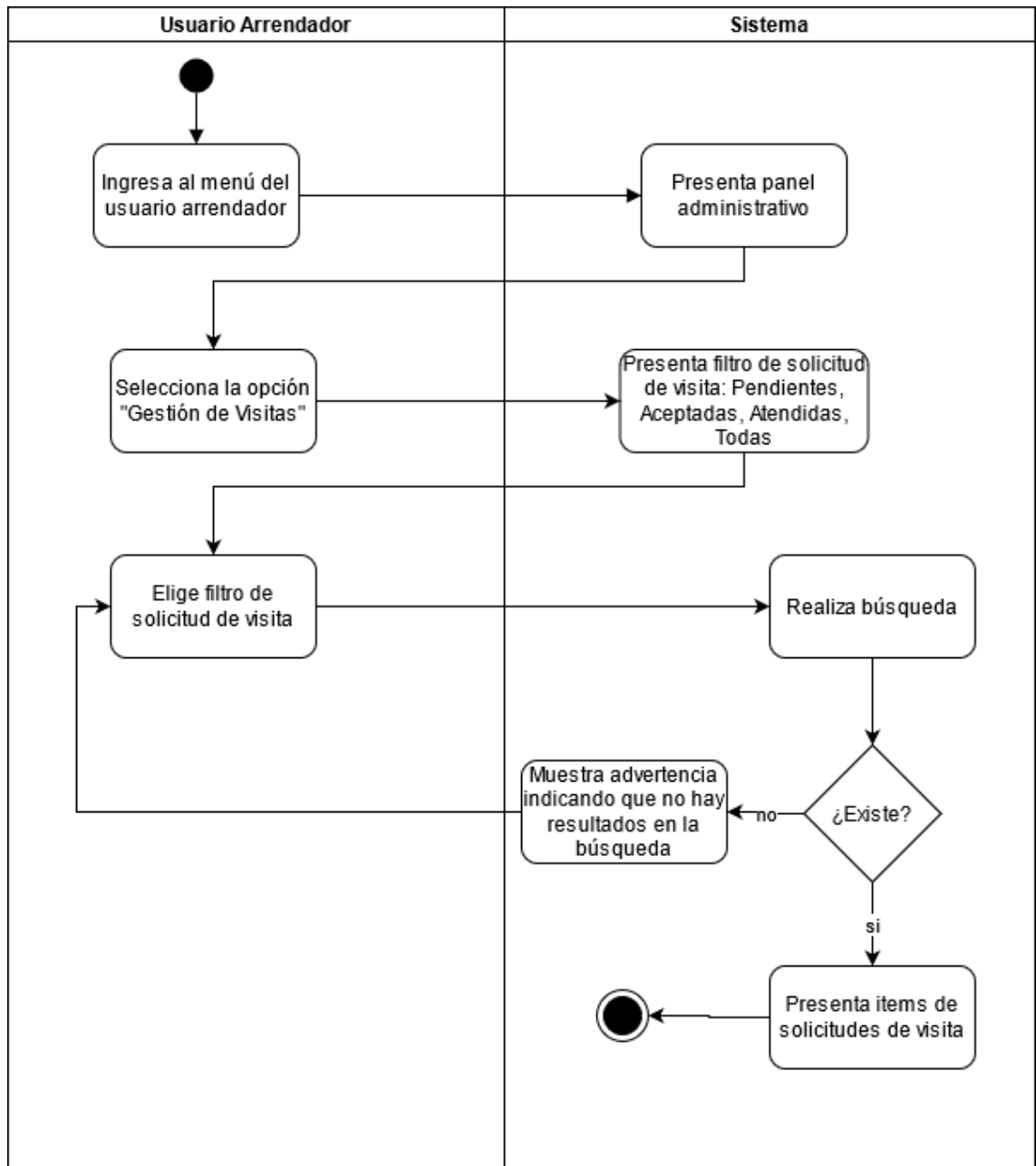


Figura 71. Diagrama de actividades - Filtrar solicitud de visita

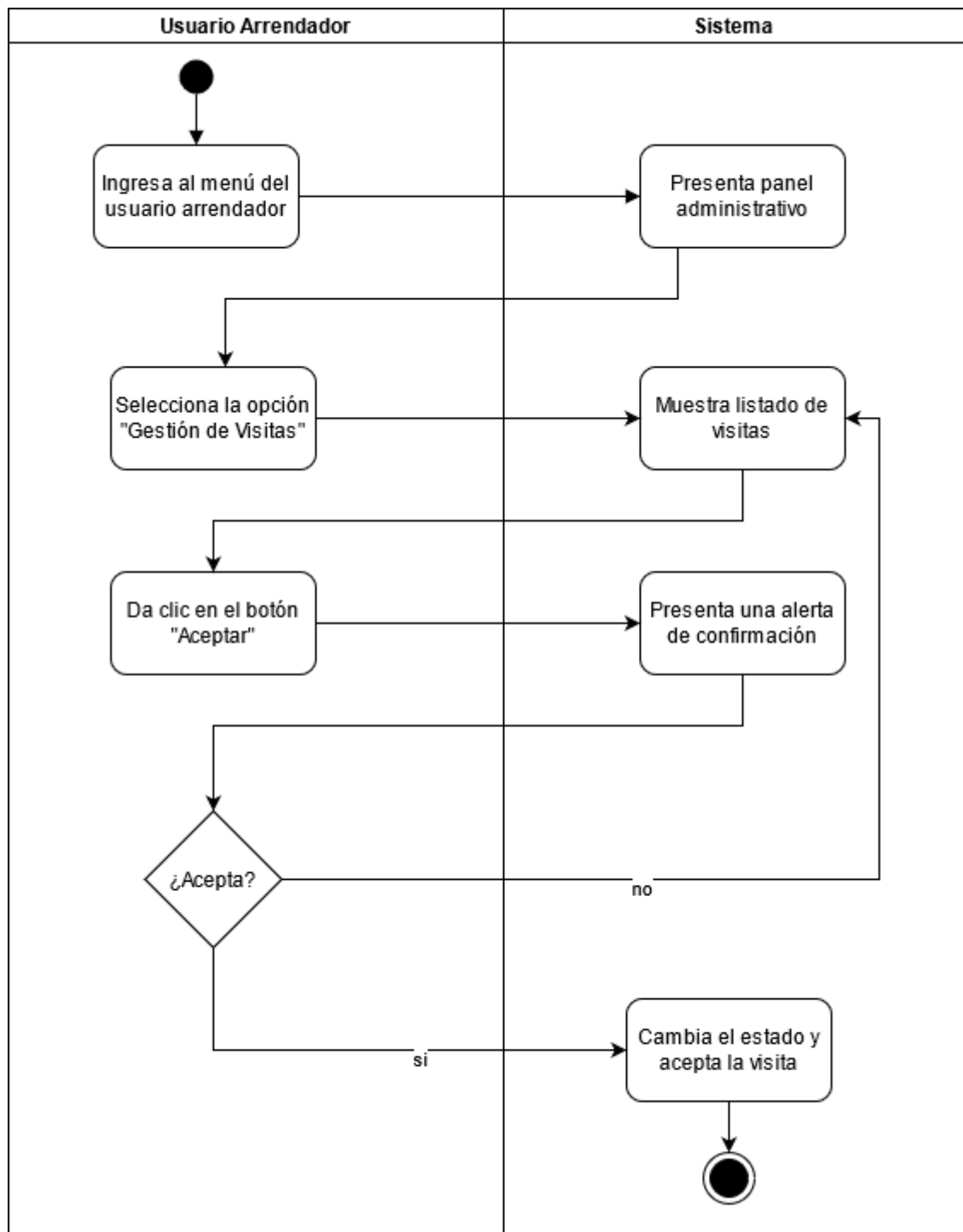


Figura 72. Diagrama de actividades - Aceptar visita

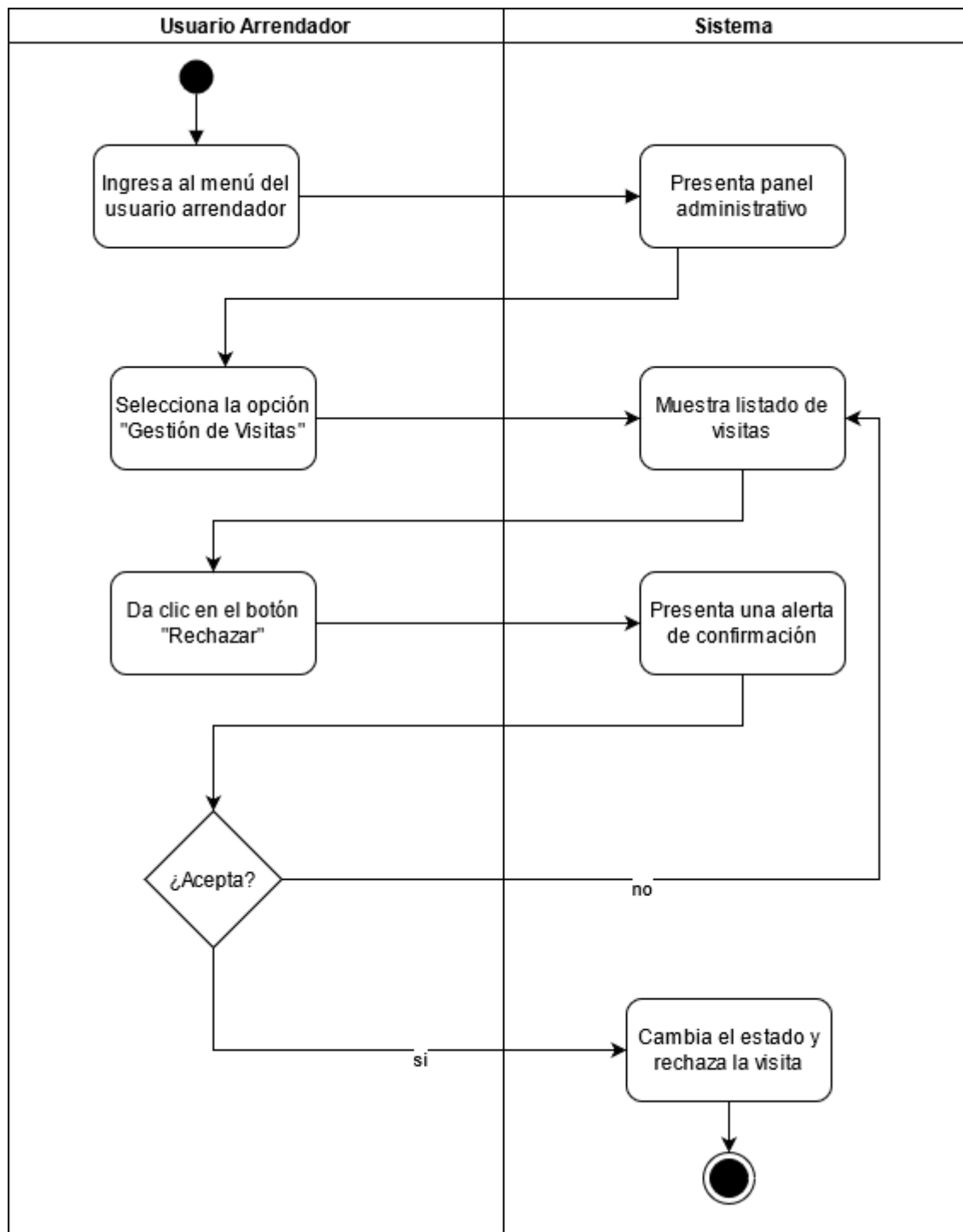


Figura 73. Diagrama de actividades - Rechazar visita

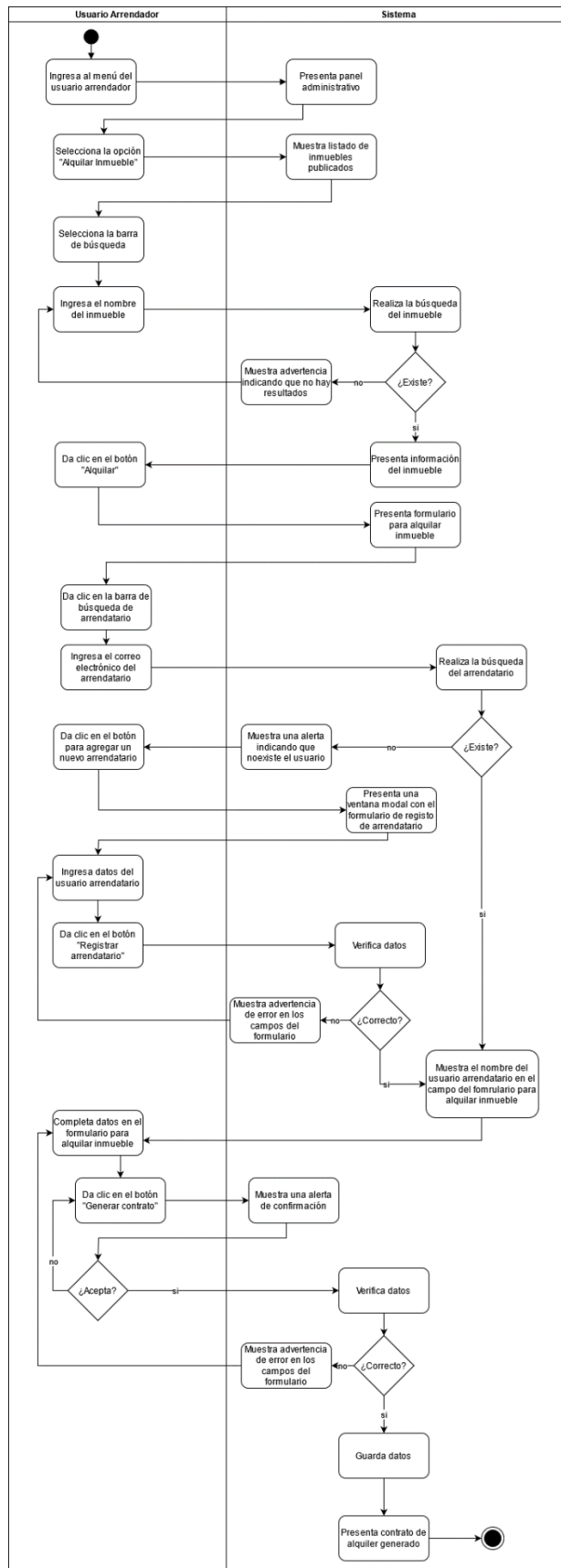


Figura 74. Diagrama de actividades - Alquilar bien inmueble

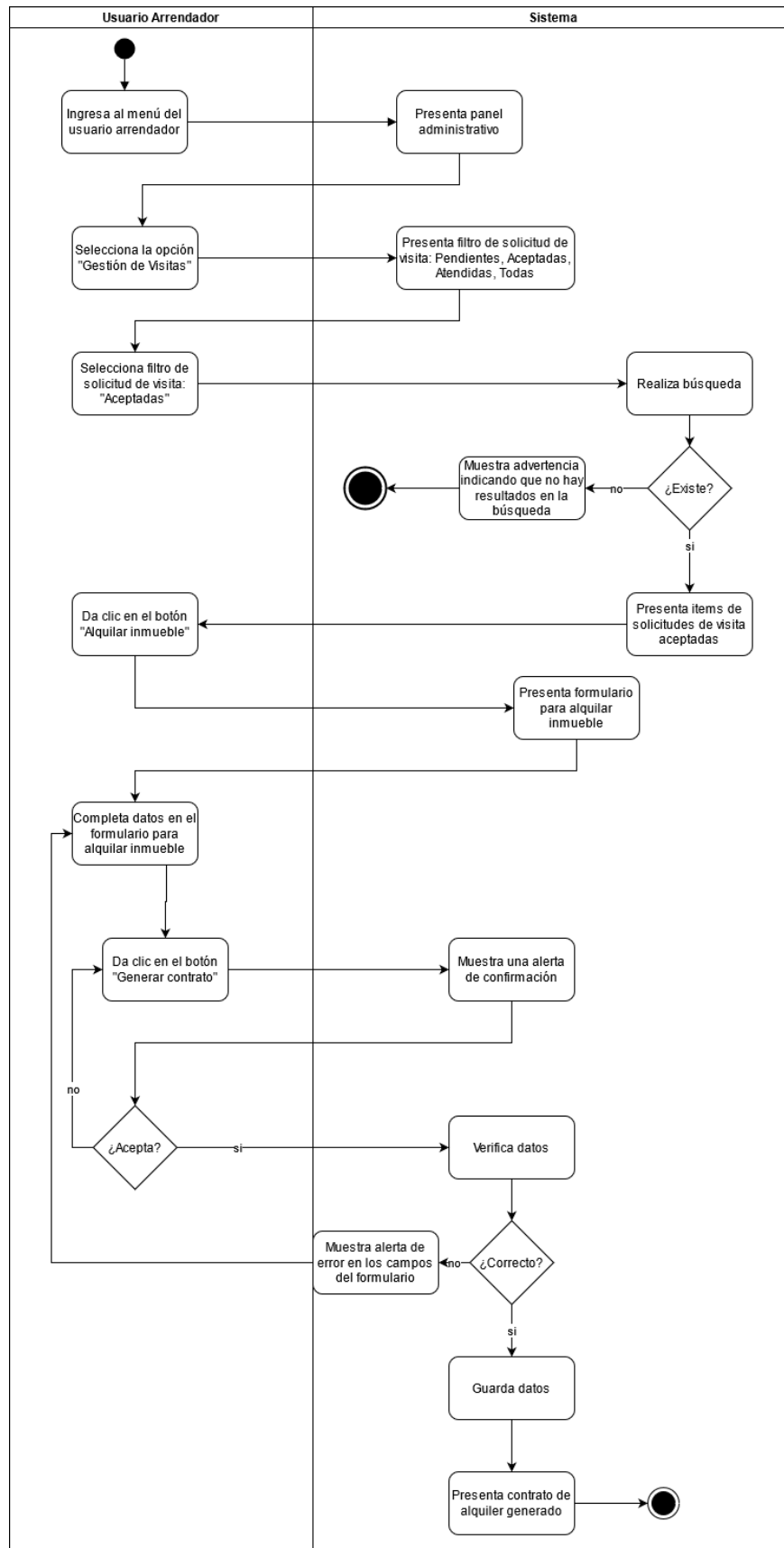


Figura 75. Diagrama de actividades - Generar contrato de alquiler

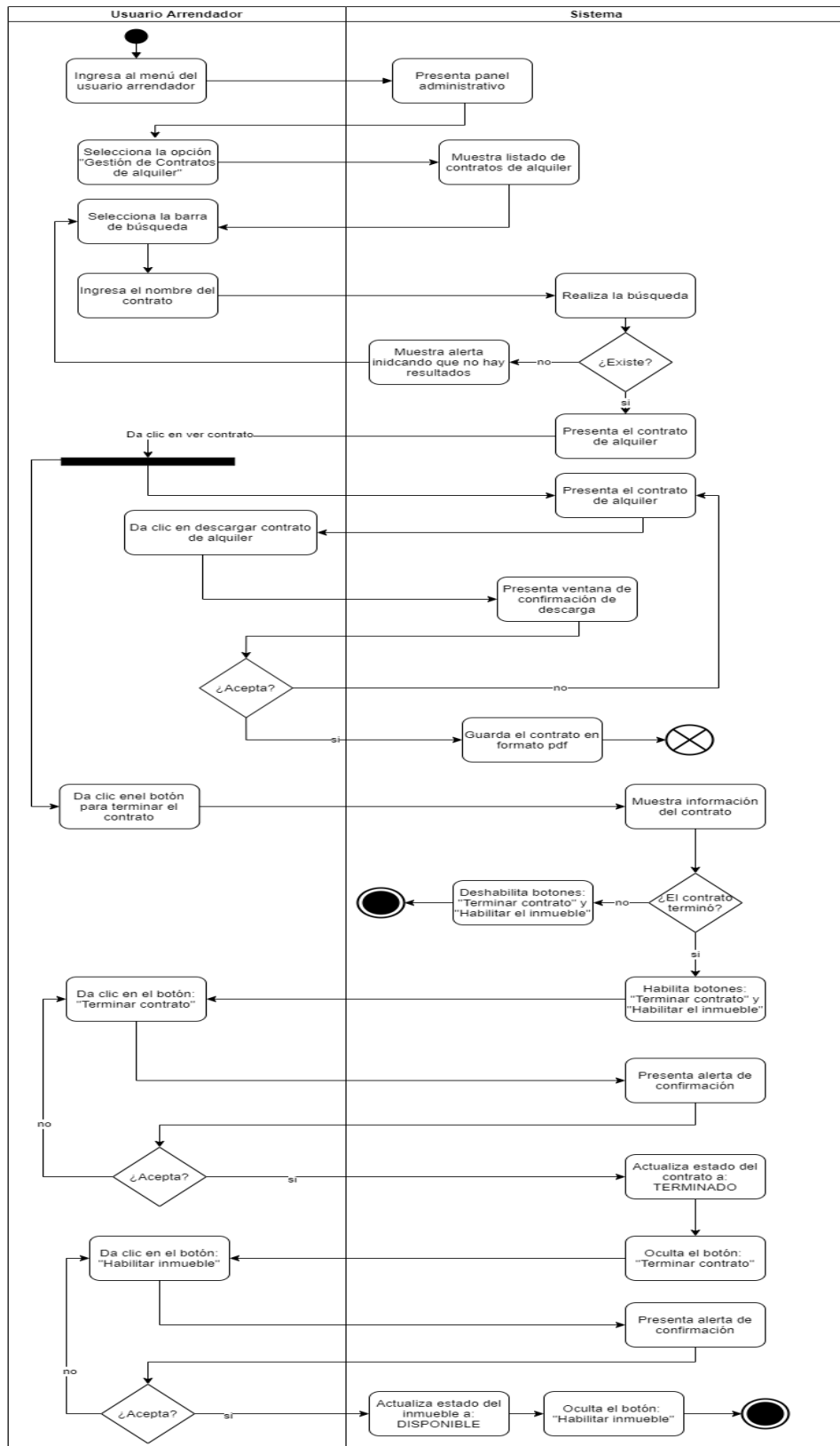


Figura 76. Diagrama de actividades - Gestionar contrato de alquiler

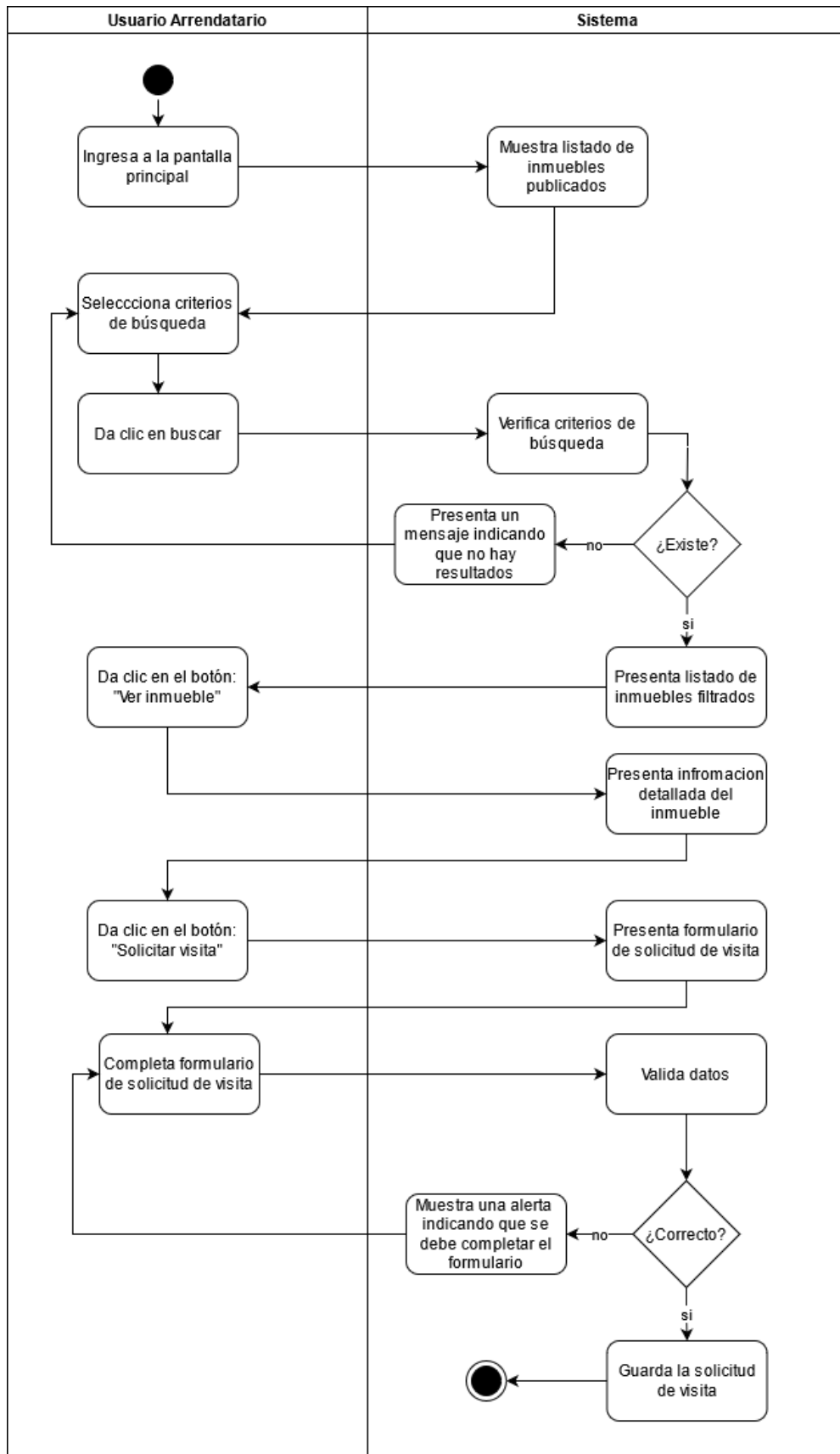


Figura 77. Diagrama de actividades - Generar solicitud de visita de alquiler

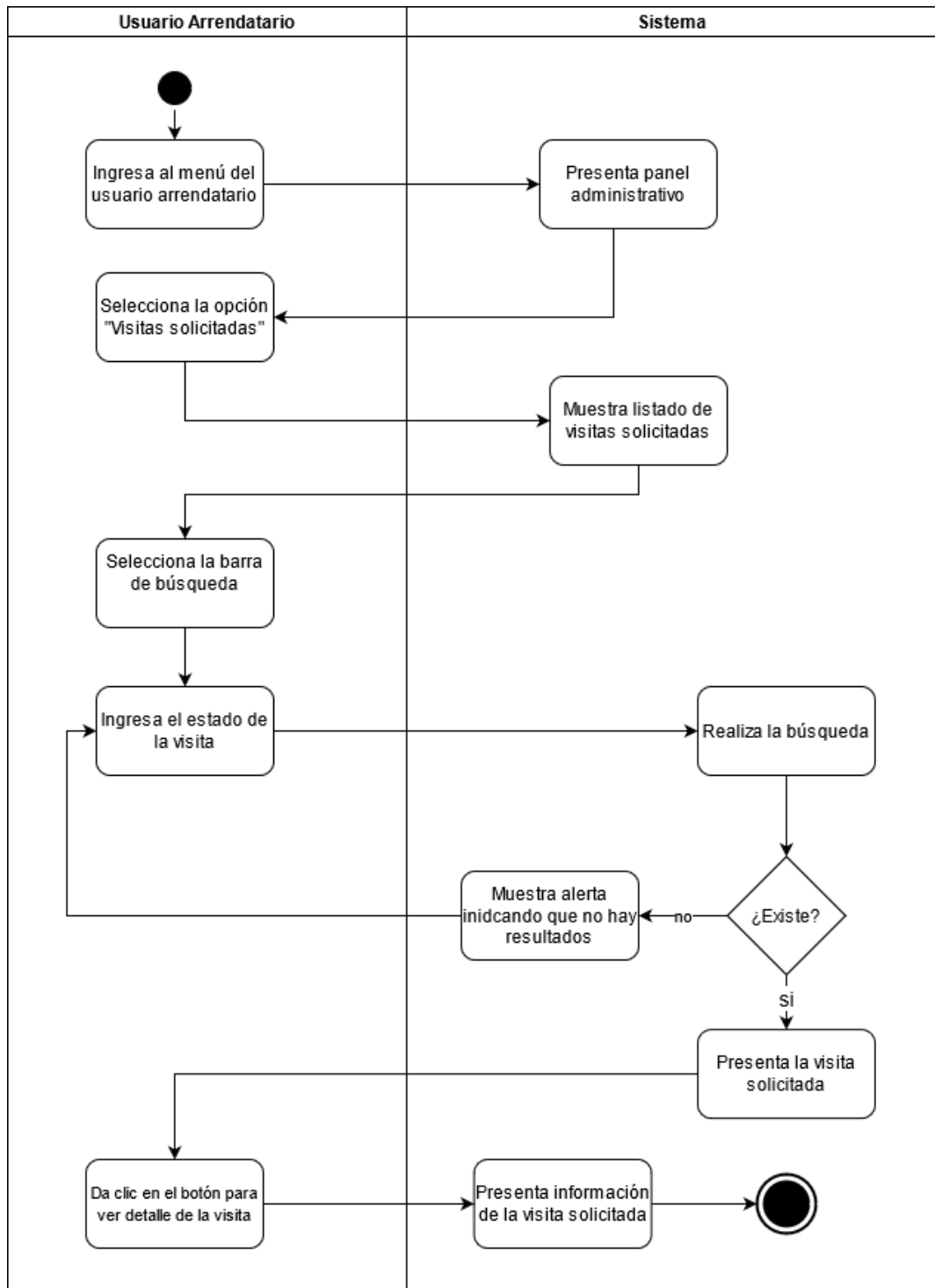


Figura 78. Diagrama de actividades - Visualizar solicitud de visitas de alquiler

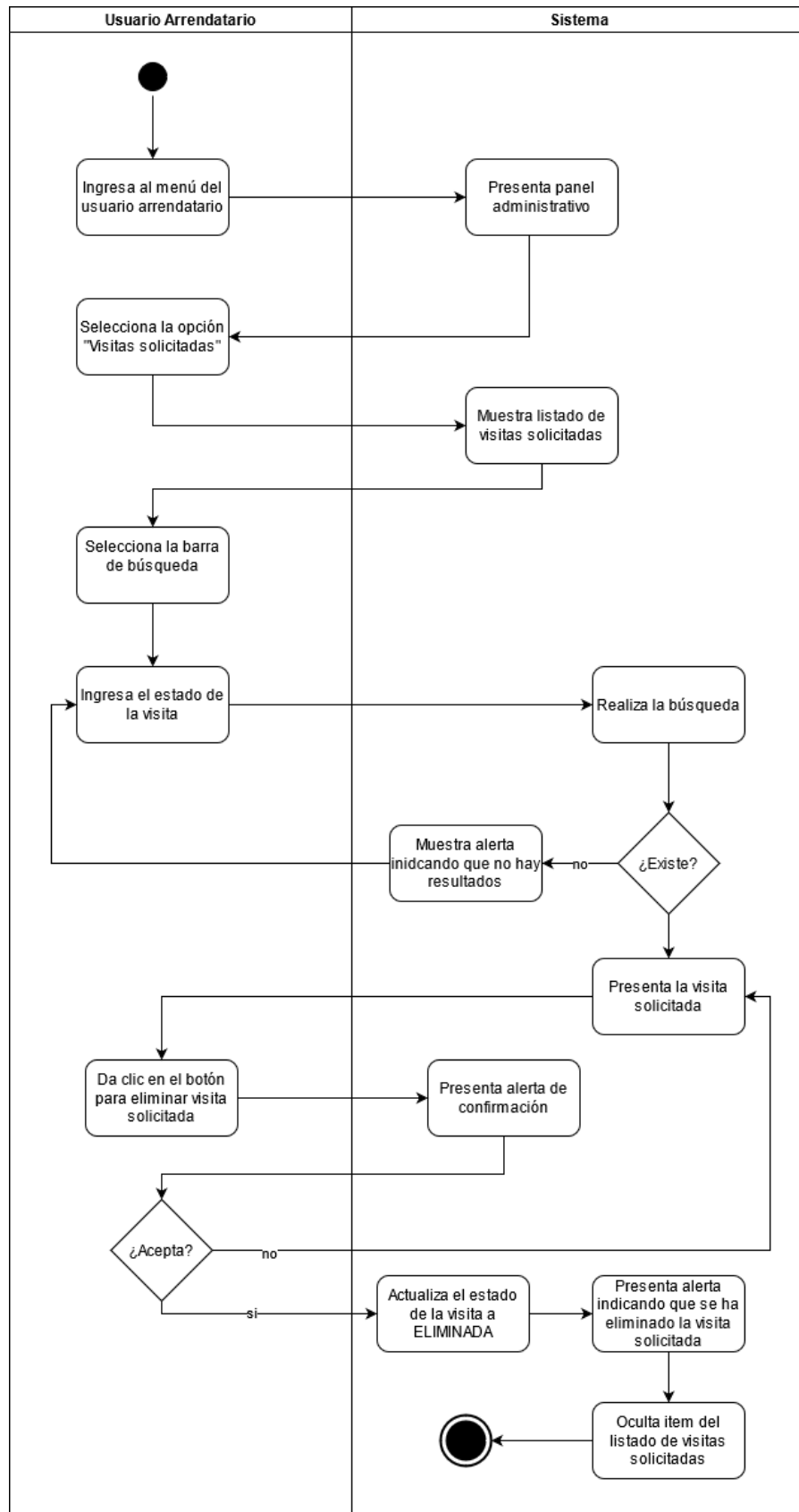


Figura 79. Diagrama de actividades - Eliminar solicitud de visitas de alquiler

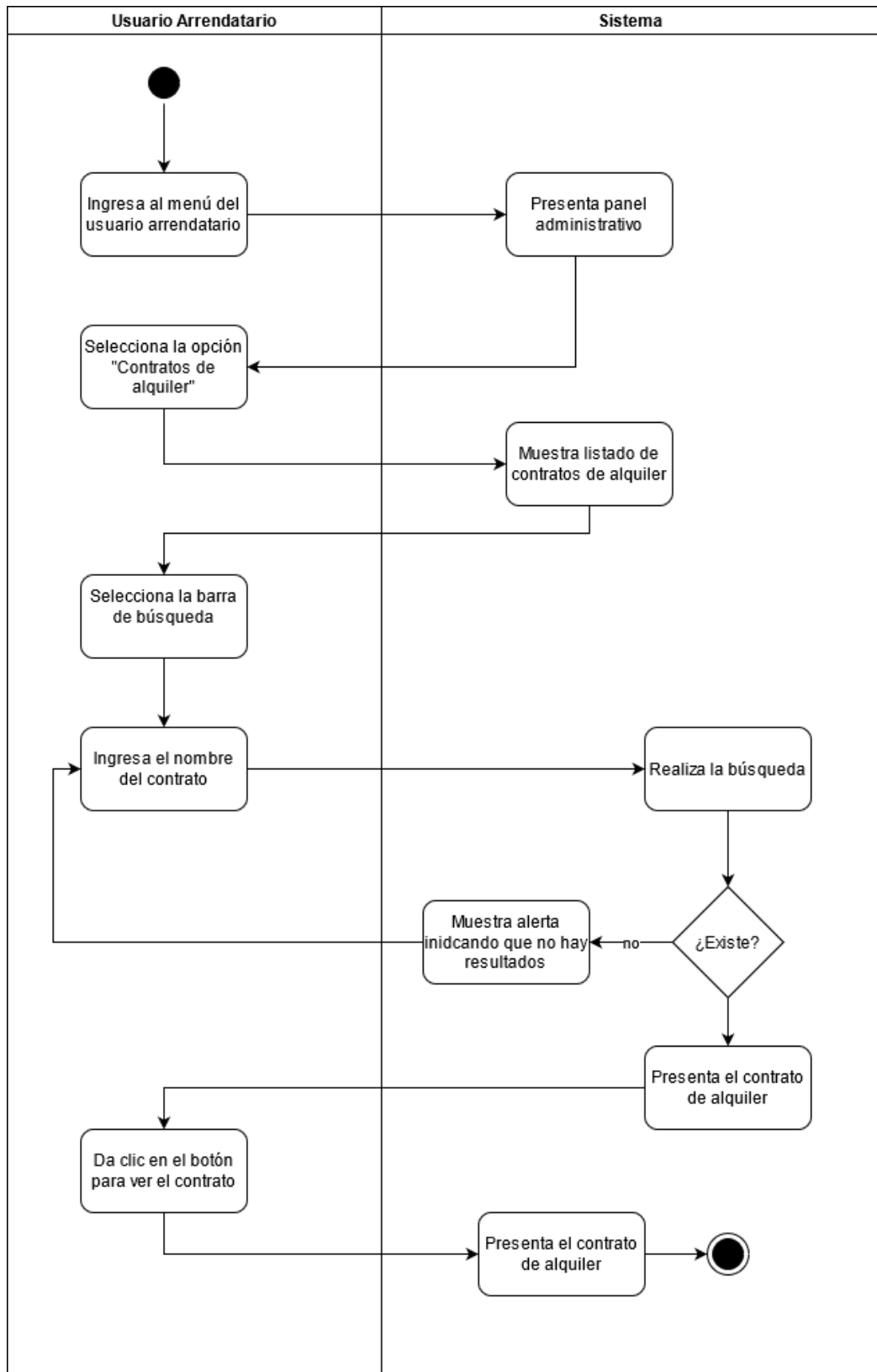


Figura 80. Diagrama de actividades - Visualizar contrato de alquiler

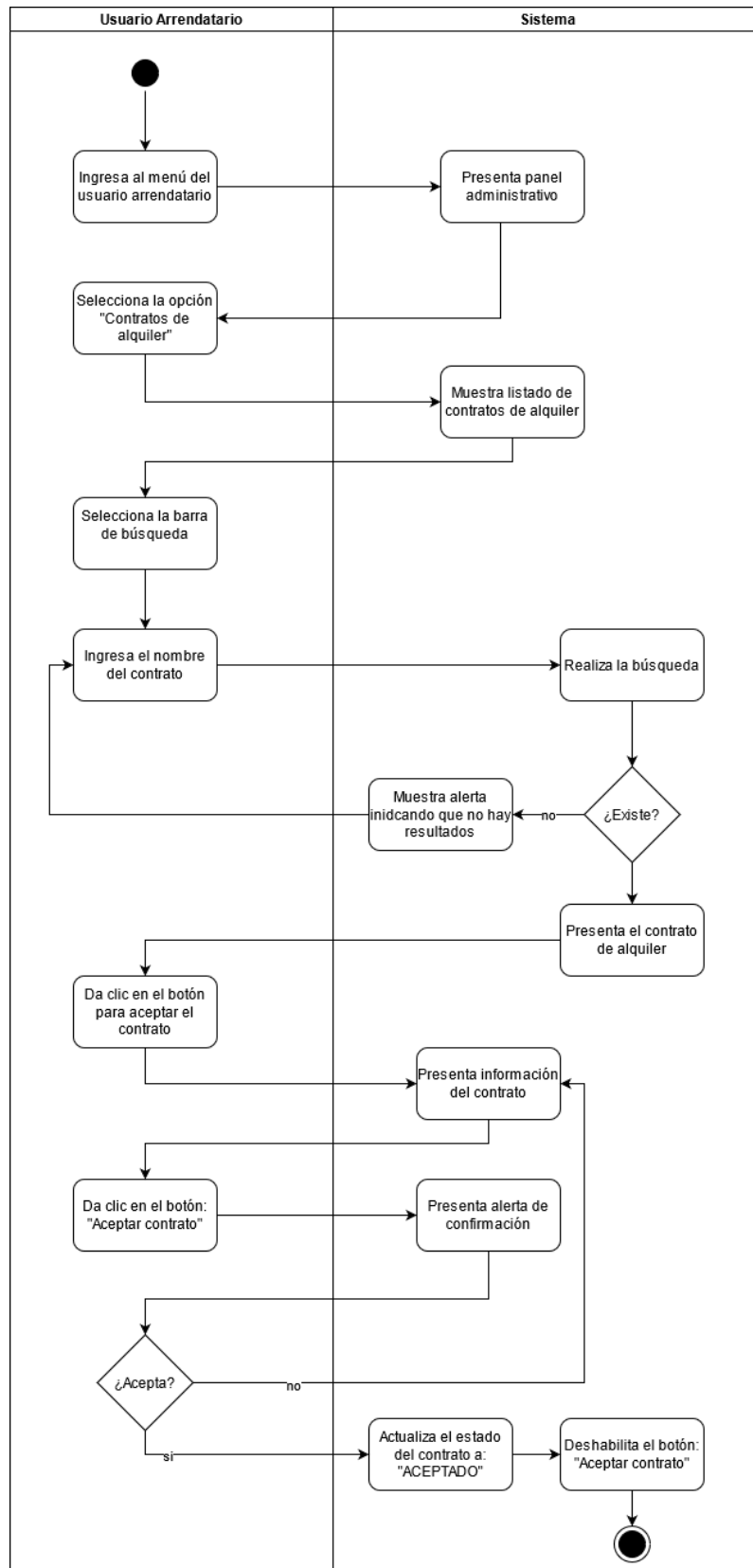


Figura 81. Diagrama de actividades - Aceptar contrato de alquiler

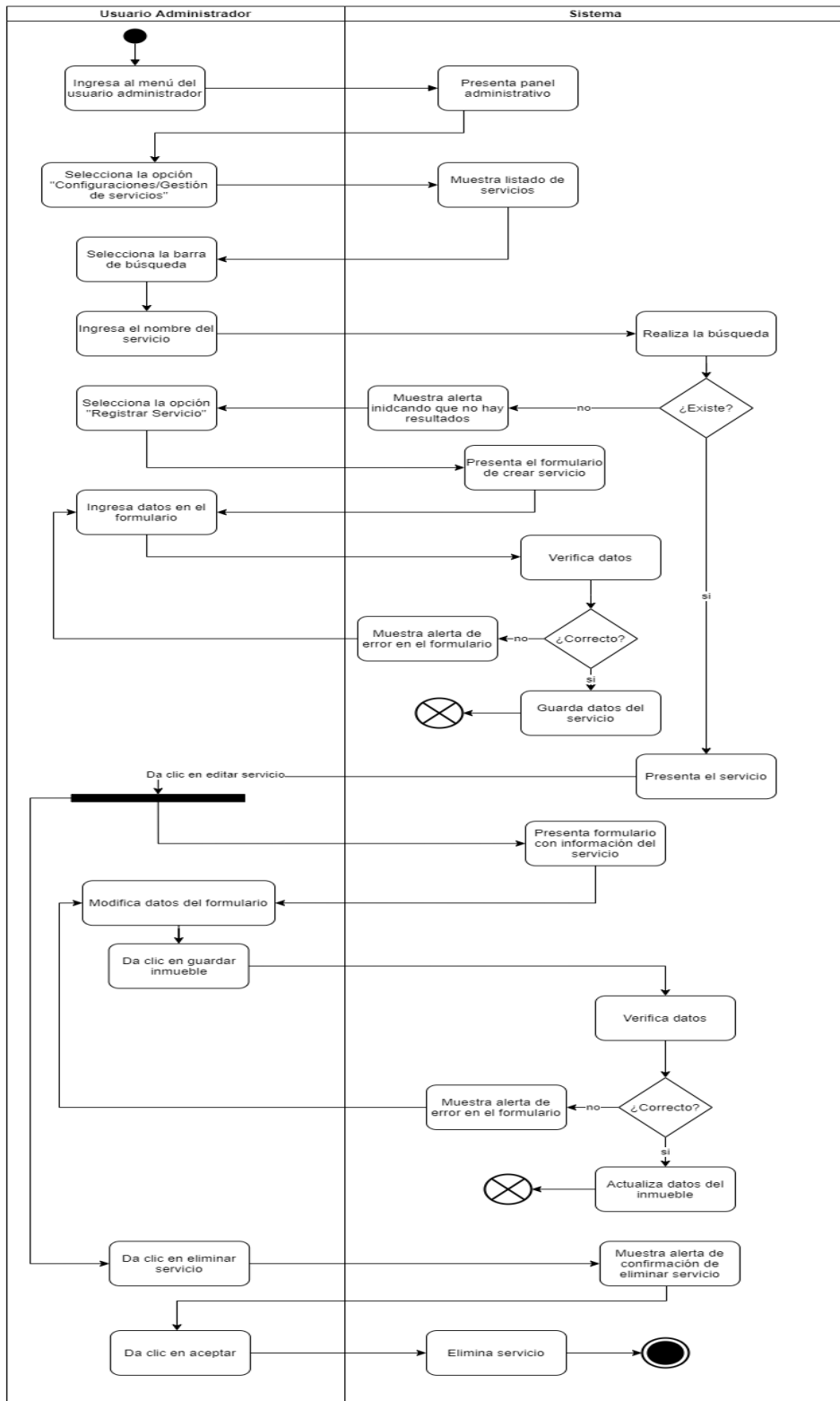


Figura 82. Diagrama de actividades - Gestionar servicio básico

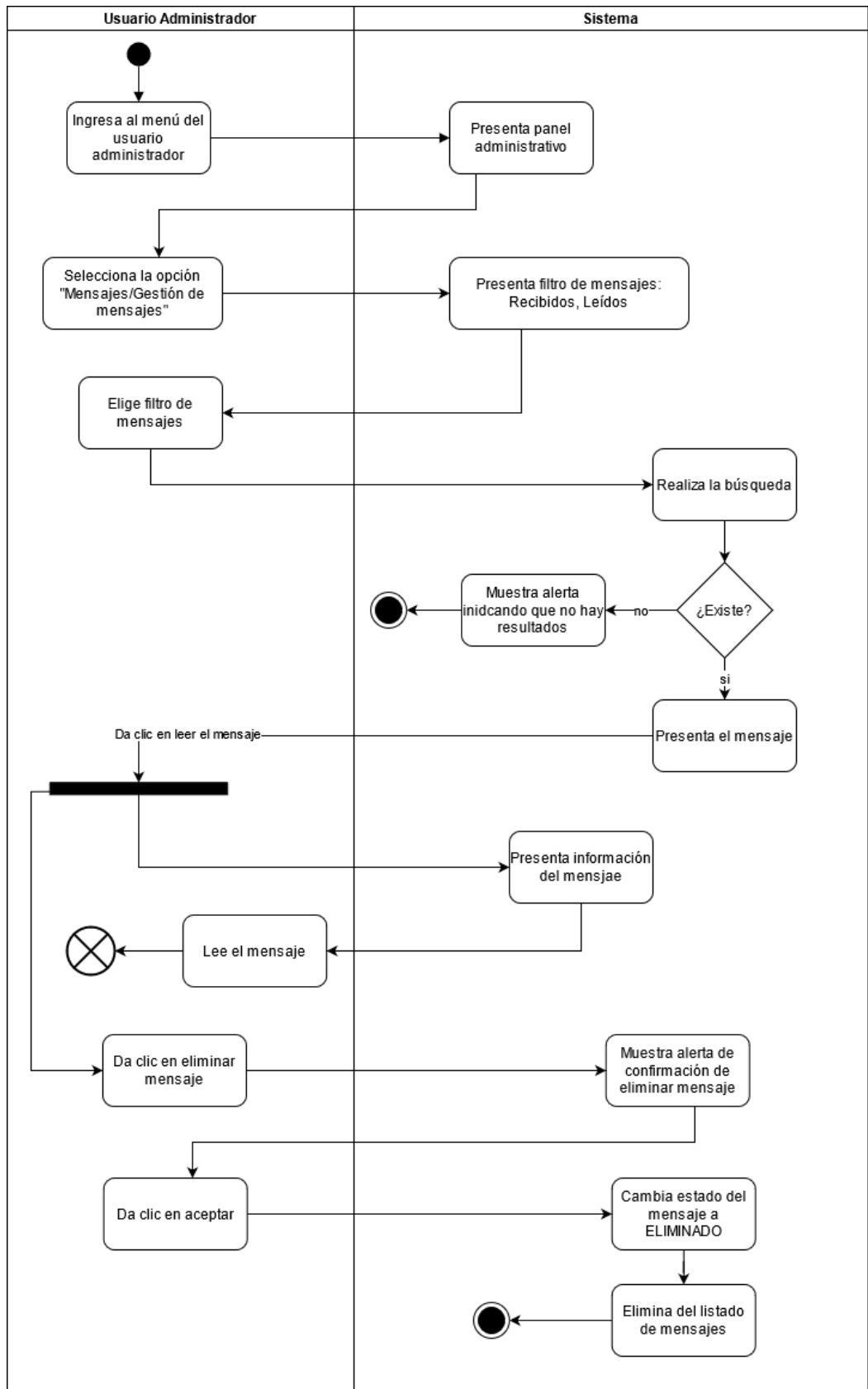


Figura 83. Diagrama de actividades - Gestionar mensaje

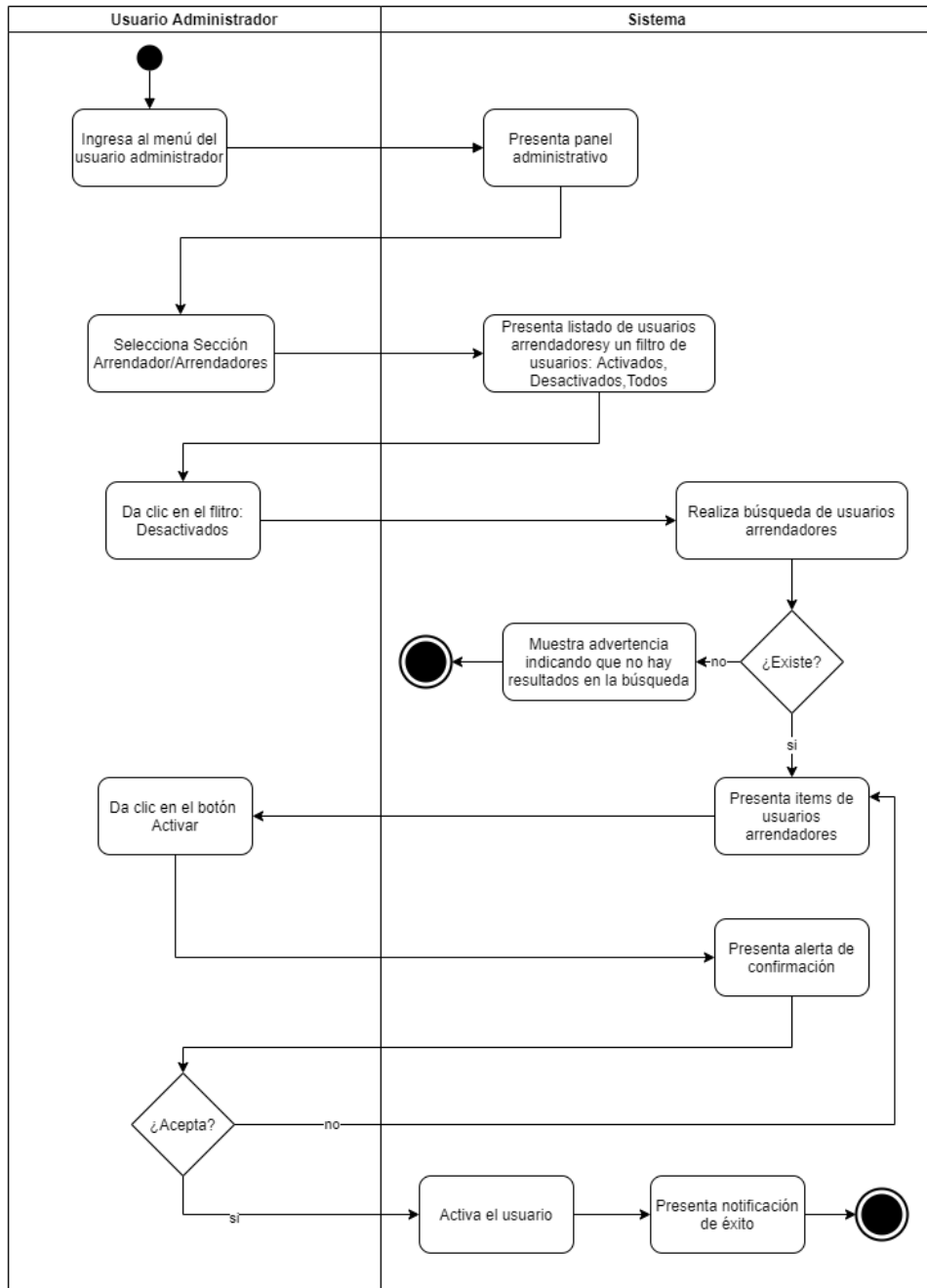


Figura 84. Diagrama de actividades - Activar usuario

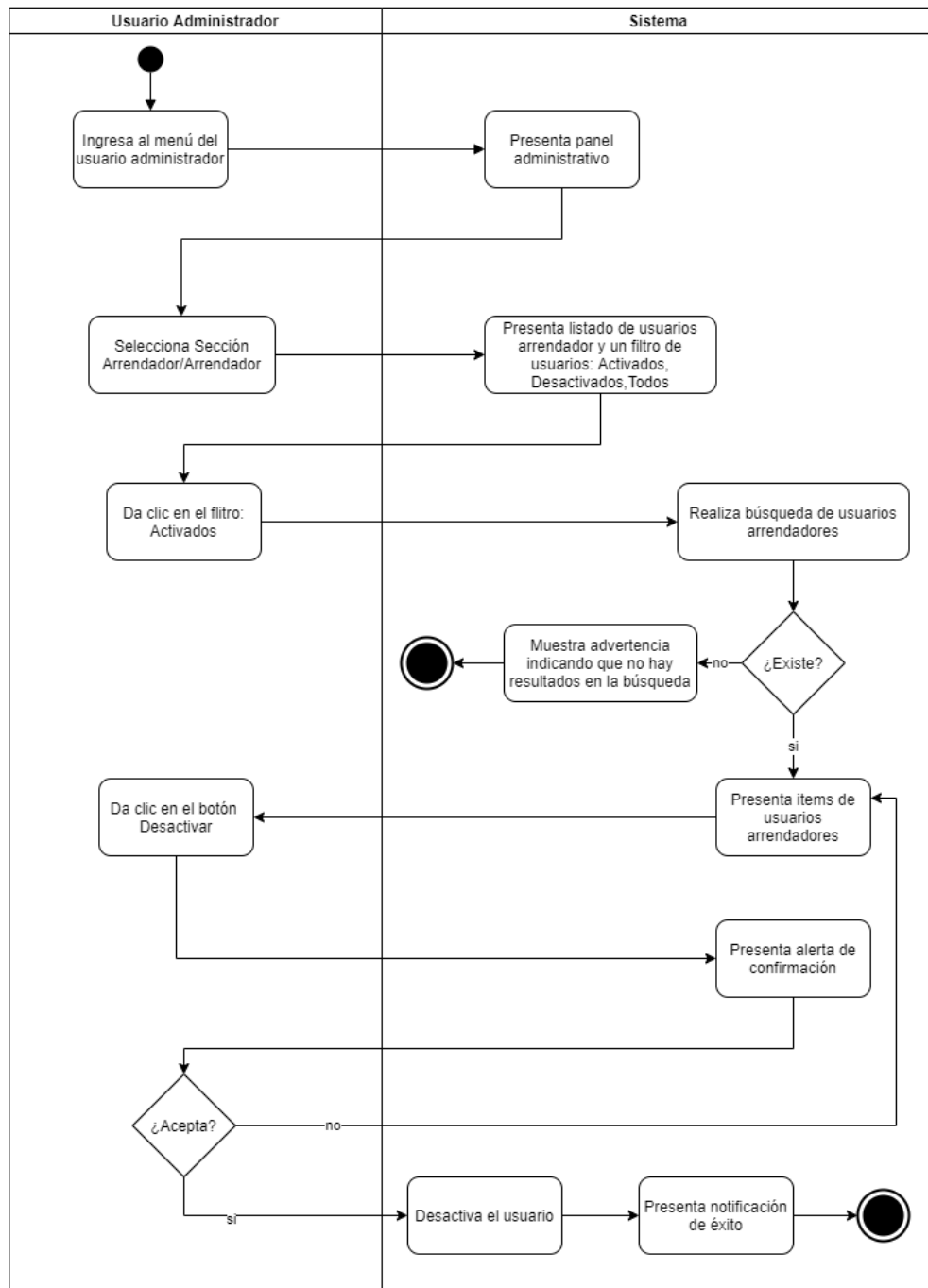


Figura 85. Diagrama de actividades - Desactivar usuario

14. Vista de Despliegue

14.1. Diagrama de componentes

En esta sección se define la organización de los diferentes componentes de la Aplicación Informática LojaHouse, la estructura de la Aplicación Informática se muestra en la Figura 86 y se detalla a continuación:

Servidor de Web: en esta sección se encuentra desarrollado el backend de la Aplicación LojaHouse, el cual se utilizó NodeJS y el framework Express, en el servidor web se encuentran definidas las rutas y métodos que se comunican con la base de datos MongoDB. Cada ruta está vinculada a un controlador, y estos a su vez responden a las peticiones realizadas por el cliente mediante el protocolo http.

Clientes:

- **Navegador Web:** en esta sección se desarrolló la parte web de la aplicación, la misma que interactúa directamente con el usuario, para lo cual se desarrolló utilizando el framework Angular, es aquí donde realizan las peticiones hacia el servidor web (API-REST) mediante el protocolo http.
- **Aplicación Móvil:** en esta sección se desarrolló utilizando el framework Flutter, el mismo que se comunica con el servicio web (API-REST) mediante el uso del protocolo http.

Base de Datos: este componente se comunica con el servidor web (API-REST) a través del ODM (Modelo de datos de objetos) Mongoose que realiza las consultas a la base de datos MongoDB.

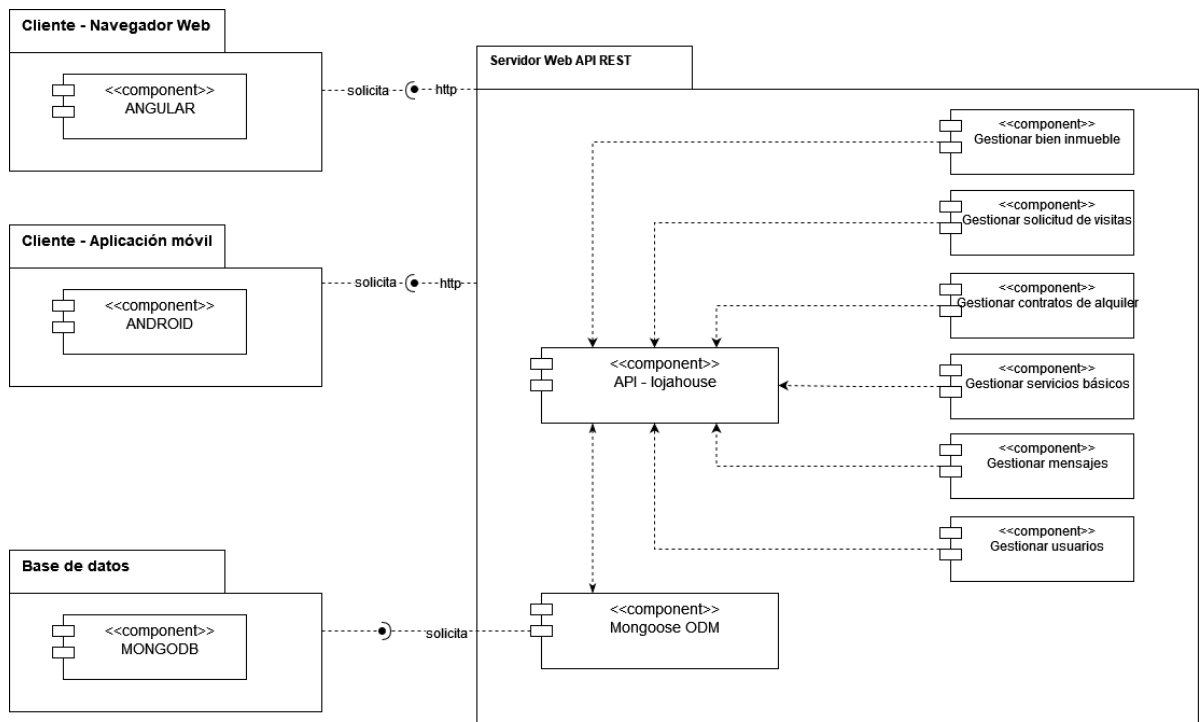


Figura 86. Diagrama de componentes

15. Vista Física

15.1. Diagrama de Despliegue

Se definen los nodos que conforman la solución y se define la arquitectura Cliente/Servidor multicapa.

En la Figura 87 se puede observar el diagrama de despliegue de la aplicación LojaHouse, el cual consta de dos clientes, el navegador web y el dispositivo móvil, mismos que se comunican con el servidor web donde se encuentra el servicio web (API-REST) desarrollado en NodeJS utilizando el framework Express conjuntamente con Mongoose, y así poder interactuar con el servidor de base de datos donde se encuentra alojada la base de datos MongoDB para el procesamiento de la información.

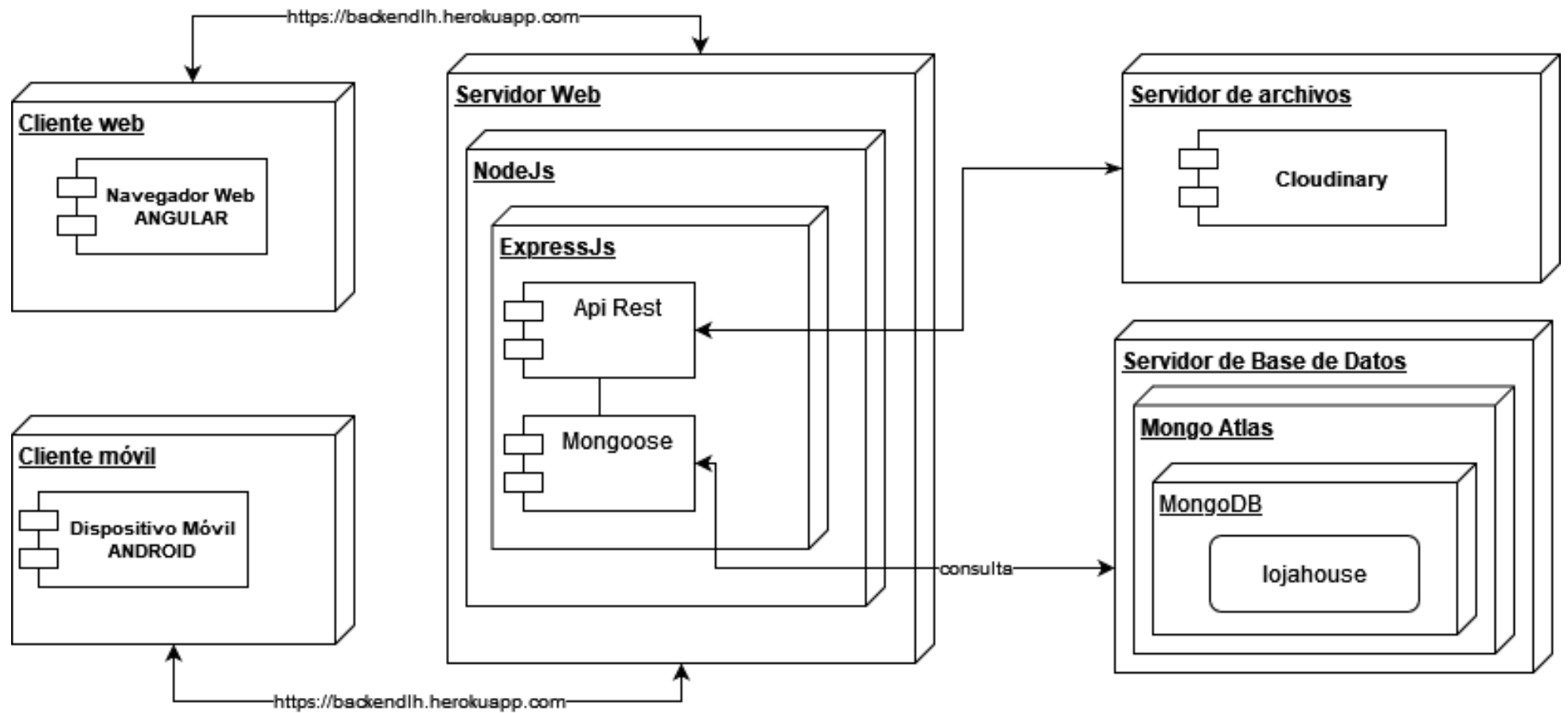


Figura 87. Vista física – Diagrama de despliegue

16. Arquitectura de la Aplicación

En Figura 88 se detalla la arquitectura de la Aplicación LojaHouse tanto del cliente web Angular, móvil Android y el servicio web (API-REST).

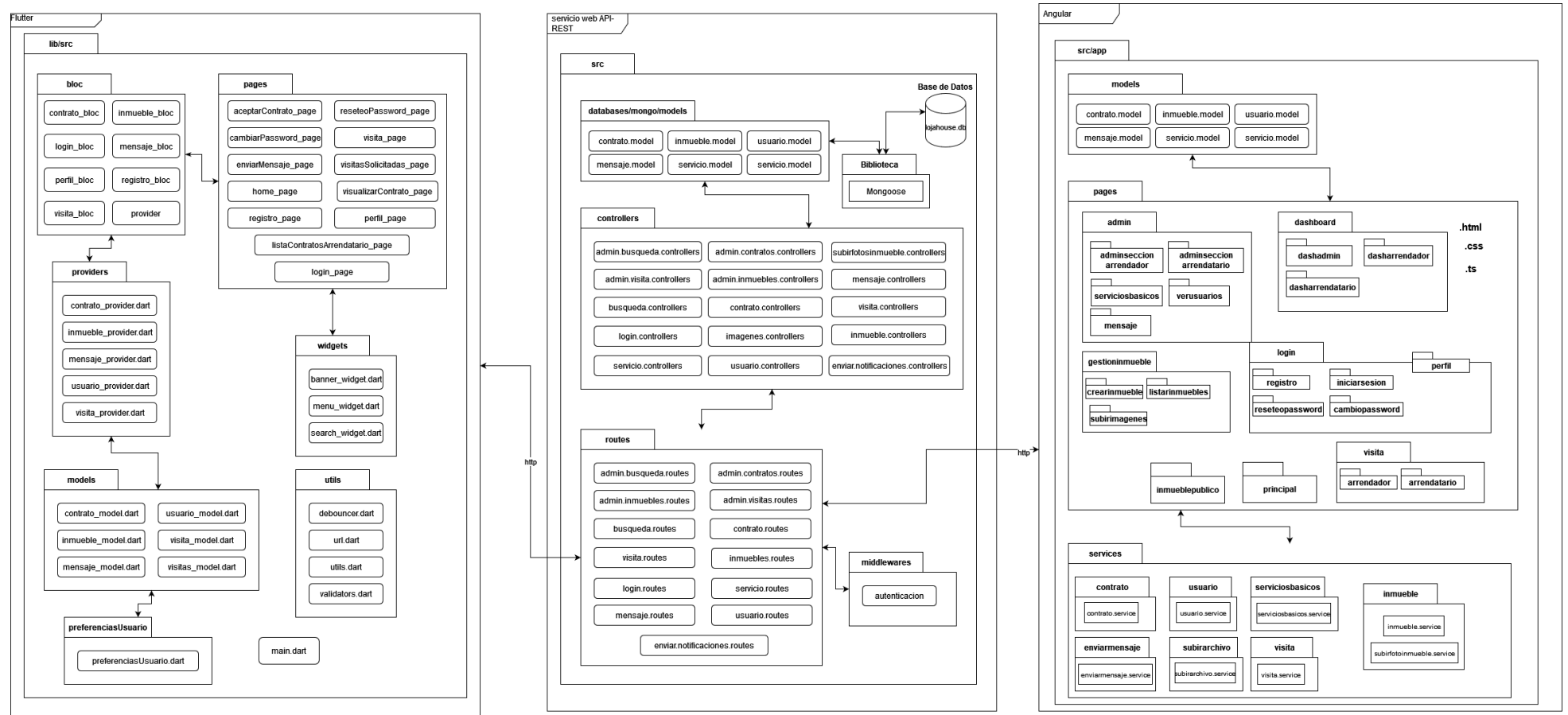


Figura 88. Arquitectura de la Aplicación

Anexo 12: Prototipo final de interfaz

Pantallas de Aceptación de la Aplicación Informática LojaHouse.

Proyecto: Desarrollo de una aplicación informática para la búsqueda de oferta de alquiler de inmuebles en la ciudad de Loja.

LEGALIZACIÓN DE APROBACIÓN DE LAS INTERFACES O PANTALLAS DE LA APLICACIÓN LOJAHOUSE.

E = Encargado que realizó la elaboración de la aplicación LojaHouse.

R = Encargado que revisó el Prototipo de la aplicación LojaHouse.

A = Encargado de la Aprobación del Prototipo de la aplicación LojaHouse

Nombre	Cargo	E	R	A	Firma
Steeven Michael Armijos Bravo	Tesista	X			
Wilman Patricio Chamba Zaragocín	Revisor Técnico		X	X	

PROTOTIPO FINAL DE INTERFACES DE USUARIO DE LA APLICACIÓN WEB LOJAHOUSE.

El presente documento contiene el diseño final de la aplicación, la cual fue validada por el director del presente TT concluyendo así la versión final de las pantallas de la aplicación LojaHouse.

Tabla CXXXI. Diseño de la pantalla principal

Pantalla principal de la aplicación LojaHose		
Nro., de pantalla	001	
Estado de prueba	Aceptado sin cambios	X
	Aceptado con cambios	
	Rediseñar	
Observaciones		

Tabla CXXXII. Diseño de la pantalla gestión de inmuebles

Pantalla gestión de inmuebles		
Nro., de pantalla	002	
Estado de prueba	Aceptado sin cambios	x
	Aceptado con cambios	
	Rediseñar	
Observaciones		

Buscar inmuebles...

Nuevo inmueble

Inmuebles registrados(3)

inmueble publicado



\$150

Lujoso departamento

Ac. Universitaria
Perfecto para estudiantes

✓
✗

Dejar de publicar

inmueble publicado



\$150

Casa en Mirador Natural...

yJhKcKtLutlNlaaB5CwllqVlChy...
yJhGcOUlZlNlshR5C5lkoVlC...

✓
✗

Dejar de publicar

inmueble publicado



\$148

Bosques de Armaru - Coll...

Ac. Nueva York
La optimización del código siemp...

✓
✗

Dejar de publicar

Tabla CXXXIII. Diseño de la pantalla crear inmueble

Crear inmueble		
Nro., de pantalla	003	
Estado de prueba	Aceptado sin cambios	X
	Aceptado con cambios	
	Rediseñar	
Observaciones		

[Regresar](#)

Nombre

Tipo de inmueble

Ingrese los servicios que incluye el bien inmueble

garaje X

+ Servicio básico

Dirección

Descripción

Provincia

Ciudad

Barrio

Código del inmueble

¿Solicita garantía? Ingrese el valor

Precio de alquiler

Guardar inmueble

Tabla CXXXIV. Diseño de la pantalla actualizar inmueble

Actualizar inmueble		
Nro., de pantalla	004	
Estado de prueba	Aceptado sin cambios	X
	Aceptado con cambios	
	Rediseñar	
Observaciones		

[Regresar](#)

Nombre **Tipo de inmueble**

Lujoso departamento Departamento

ingrese los servicios que incluye el bien inmueble

garaje X
luz X
agua X
+ Servicio básico

Descripción

Perfecto para estudiantes

Dirección **Ciudad**

Av. Universitaria Loja

Provincia **Barrio**

Loja Gran Colombia

Código del inmueble **¿Solicita garantía? Ingrese el valor**

18-34 0

Precio de alquiler

150

Guardar inmueble

Tabla CXXXV. Diseño de la pantalla de inmuebles publicados


Listar inmuebles disponibles para alquilar			
Nro., de pantalla	005		
Estado de prueba	Aceptado sin cambios	X	
	Aceptado con cambios		
	Rediseñar		
Observaciones			
<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <input type="text" value="Buscar inmuebles..."/> </div> <div style="text-align: center; margin-bottom: 10px;"> <h3 style="color: #0070C0;">Lujoso departamento</h3> </div> <div style="text-align: center; margin-bottom: 10px;">  </div> <div style="text-align: right; margin-bottom: 10px;"> <p>\$150</p> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>Código del inmueble: 18-34</p> <p>Dirección: 📍 Av. Universitaria</p> </div> <div style="width: 45%;"> <p>Servicios incluidos: garaje, luz, agua</p> </div> </div> <div style="text-align: center; background-color: #00A0C0; color: white; padding: 10px; margin-top: 10px;"> <p>ALQUILAR</p> </div>			

Tabla CXXXVI. Diseño de la pantalla generar contrato de alquiler

Generar contrato de alquiler			
Nro., de pantalla	006		
Estado de prueba	Aceptado sin cambios	X	
	Aceptado con cambios		
	Rediseñar		
Observaciones			
<div style="background-color: #e6f2ff; padding: 10px;"> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>Usuario arrendatario</p> <input type="text" value="José Andrés Contreras Portiño"/> </div> <div style="width: 45%;"> <p>Inmueble</p> <input type="text" value="DEPARTAMENTO"/> </div> </div> <div style="display: flex; justify-content: space-between; margin-top: 10px;"> <div style="width: 45%;"> <p>Fecha de inicio</p> <input type="text" value="dd/mm/aaaa"/> </div> <div style="width: 45%;"> <p>Fecha de finalización</p> <input type="text" value="dd/mm/aaaa"/> </div> </div> <div style="display: flex; justify-content: space-between; margin-top: 10px;"> <div style="width: 45%;"> <p>Monto mensual de alquiler</p> <input type="text" value="150"/> </div> <div style="width: 45%;"> <p>Garantía</p> <input type="text" value="0"/> </div> </div> <div style="text-align: center; margin-top: 10px;"> <input type="button" value="Generar contrato"/> </div> </div>			

Tabla CXXXVII. Diseño de la pantalla gestión de visitas

Gestión de visitas		
Nro., de pantalla	007	
Estado de prueba	Aceptado sin cambios	X
	Aceptado con cambios	
	Rediseñar	
Observaciones		
<p>Filtrar visitas.</p> <p> <input type="radio"/> Pendientes <input checked="" type="radio"/> Aceptadas <input type="radio"/> Atendidas <input type="radio"/> Todas </p> <p>Han solicitado (2) visitas</p> <p>La visita está: ACEPTADA</p> <p>José Andrés Contreras Portiño</p> <p>Descripción: Hola, estoy interesado en su inmueble "Lujoso departamento" y me gustaría estar en contacto con usted para poder llegar a un acuerdo. Muchas gracias por su tiempo, hasta luego.</p> <p>Inmueble de interés: DEPARTAMENTO</p> <p>Fecha solicitada para la visita: 09-06-2021</p> <p> <input type="button" value="✓"/> <input type="button" value="✗"/> <input type="button" value="📧"/> </p>		

Tabla CXXXVIII. Diseño de la pantalla listado de contratos de alquiler

Listar contratos de alquiler															
Nro., de pantalla	008														
Estado de prueba	Aceptado sin cambios	X													
	Aceptado con cambios														
	Rediseñar														
Observaciones															
<div style="border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;"> <input type="text" value="Buscar contratos..."/> </div> <div style="border: 1px solid #ccc; padding: 10px;"> <p>Contratos registrados (1)</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">Nombre</th> <th style="text-align: left;">Inicio</th> <th style="text-align: left;">Finalización</th> <th style="text-align: left;">Meses de alquiler</th> <th style="text-align: left;">Estado</th> <th style="text-align: left;">Acciones</th> </tr> </thead> <tbody> <tr> <td>Mite (CASA)</td> <td>16-05-2021</td> <td>16-06-2021</td> <td>1</td> <td>VIGENTE</td> <td> </td> </tr> </tbody> </table> <div style="margin-top: 10px;"> </div> </div>				Nombre	Inicio	Finalización	Meses de alquiler	Estado	Acciones	Mite (CASA)	16-05-2021	16-06-2021	1	VIGENTE	
Nombre	Inicio	Finalización	Meses de alquiler	Estado	Acciones										
Mite (CASA)	16-05-2021	16-06-2021	1	VIGENTE											

Tabla CXXXIX. Diseño de la pantalla solicitar visita de alquiler

Solicitar visita de alquiler			
Nro., de pantalla	009		
Estado de prueba	Aceptado sin cambios	X	
	Aceptado con cambios		
	Rediseñar		
Observaciones			
<p>DEPARTAMENTO</p>  <p style="text-align: right;">DISPONIBLE</p> <p>Lujoso departamento \$150</p> <p>Código del inmueble: 18-34</p> <p>Dirección: Av. Universitaria</p> <p>Servicios incluidos: garaje,luz,agua</p> <p>Descripción: Perfecto para estudiantes</p> <p style="text-align: center;">Solicitar Visita</p>			

Tabla CXL. Diseño de la pantalla editar perfil de usuario

Editar perfil de usuario			
Nro., de pantalla	010		
Estado de prueba	Aceptado sin cambios	X	
	Aceptado con cambios		
	Rediseñar		
Observaciones			

PROTOTIPO FINAL DE INTERFACES DE USUARIO DE LA APLICACIÓN MÓVIL LOJAHOUSE.

A continuación se detalla el diseño final de la aplicación móvil que está dirigida específicamente para los usuarios arrendatarios, la cual fue validada por el director del presente TT concluyendo así la versión final de las pantallas de la aplicación LojaHouse.

Tabla CXLI. Diseño de pantalla principal de la aplicación móvil

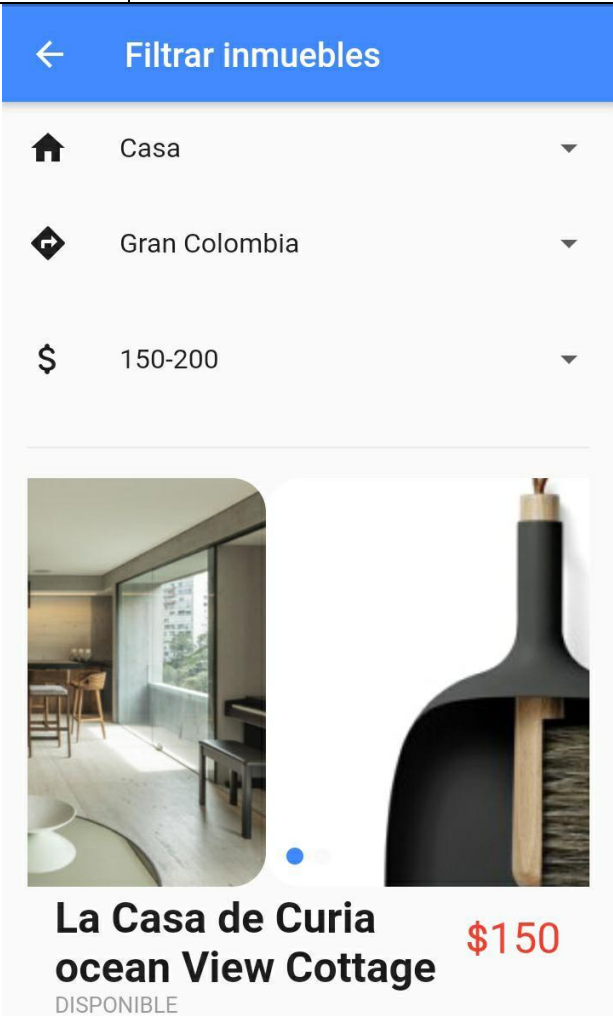
Pantalla principal de la aplicación LojaHose		
Nro., de pantalla	001	
Estado de prueba	Aceptado sin cambios	X
	Aceptado con cambios	
	Rediseñar	
Observaciones		
		

Tabla CXLII. Diseño de la pantalla solicitar visita

Pantalla solicitar visita		
Nro., de pantalla	002	
Estado de prueba	Aceptado sin cambios	X
	Aceptado con cambios	
	Rediseñar	
Observaciones		

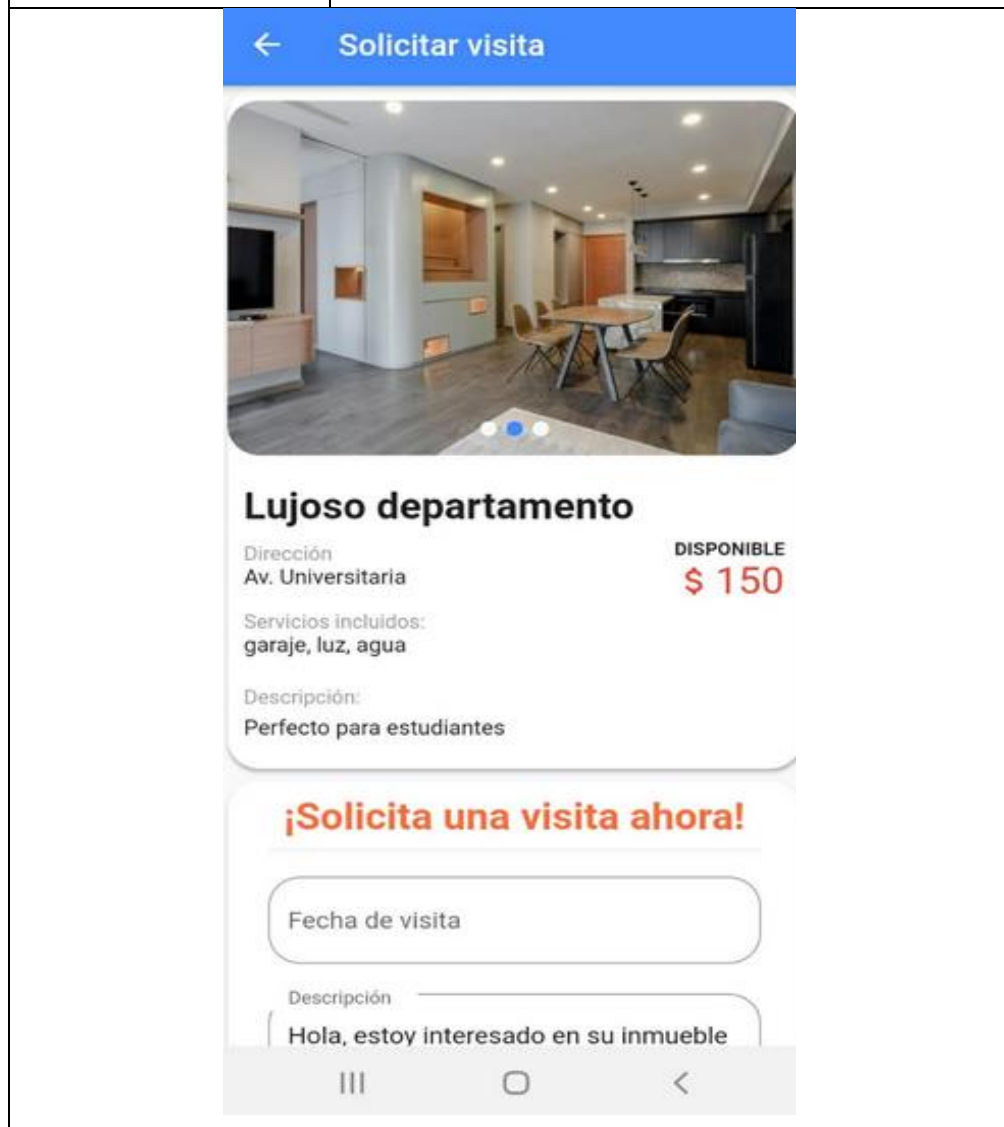


Tabla CXLIII. Diseño de la pantalla enviar mensaje

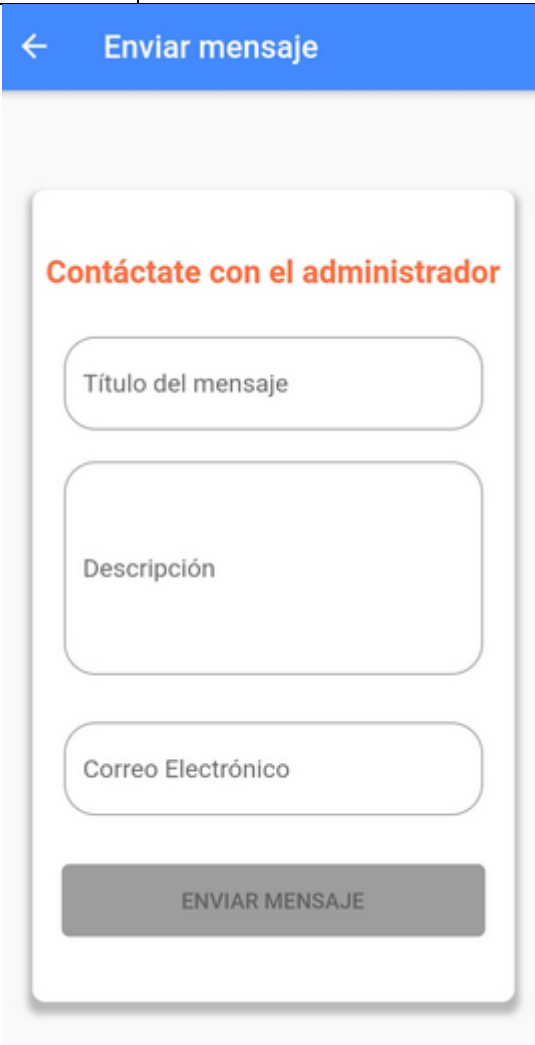
Pantalla enviar mensaje		
Nro., de pantalla	003	
Estado de prueba	Aceptado sin cambios	X
	Aceptado con cambios	
	Rediseñar	
Observaciones		
		

Tabla CXLIV. Diseño de pantalla listado de solicitudes de visita

Pantalla listado de solicitudes de visita		
Nro., de pantalla	004	
Estado de prueba	Aceptado sin cambios	X
	Aceptado con cambios	
	Rediseñar	
Observaciones		

←
Solicitudes de visita Arrendatario

Solicitud realizada al inmueble:
Lujoso departamento

Estado de la solicitud
ATENDIDA

Fecha de visita
10-06-2021

→
Ver más

Solicitud realizada al inmueble:
Bosques de Amaru · Calle Rumichaca y Amaru

Estado de la solicitud
ATENDIDA

Fecha de visita
03-07-2021

→
Ver más

Solicitud realizada al inmueble:
La Casa de Curia ocean View Cottage

Estado de la solicitud
PENDIENTE

Fecha de visita

→
Ver más

Tabla CXLV. Diseño de pantalla listado de contratos

Pantalla listado de contratos		
Nro., de pantalla	005	
Estado de prueba	Aceptado sin cambios	X
	Aceptado con cambios	
	Rediseñar	
Observaciones		

←
Contratos del arrendatario

nombre del contrato
Contreras Portiño(DEPARTAMENTO) →

Meses de alquiler
1 El contrato está ACEPTADO

nombre del contrato
Contreras Portiño(CASA) →

Meses de alquiler
1 El contrato está PENDIENTE

Anexo 13. Desarrollo del experimento

1. Definición del alcance

Mediante el uso de la plantilla del método GQM (Goal Question Metric) se definió el siguiente objetivo: Evaluar la usabilidad de la aplicación LojaHouse mediante el uso e interacción del prototipo.

2. Planificación

Se realizó la planificación del mismo para tener claro cómo se llevará a cabo la experimentación, la misma que se la dividió en tareas las cuales se presentan a continuación.

a. Selección del contexto

La selección del contexto consiste en determinar el entorno en el que se ejecutará el experimento, el cual se lo ha caracterizado de acuerdo a las siguientes dimensiones:

- ✓ **On-line:** debido a que la aplicación LojaHouse, en su versión web está alojada en un servidor y la versión móvil tendrá que realizar peticiones por lo que necesitará una conexión a internet.
- ✓ **Usuarios:** debido a que la encuesta y la aplicación LojaHouse serán aplicados a usuarios arrendatarios y arrendadores.
- ✓ **Aplicación:** LojaHouse es la aplicación que va hacer manipulada, la misma que tiene distintas funcionalidades para la búsqueda de alquiler de inmuebles, entre ellos: gestionar inmuebles, realizar solicitudes de alquiler, enviar mensajes al administrador, aceptar contrato.

b. Selección de sujetos

En el experimento participaron personas en rol de arrendadores, las personas de la ciudad de Loja y en rol de arrendatarios, los estudiantes de primer ciclo de la carrera Ingeniería en Sistemas/Computación de la UNL. Los mismos que participaron de forma voluntaria y fueron seleccionados por conveniencia, debido a su accesibilidad y proximidad para llevar a cabo este proceso.

c. Selección de variables

Se considera como variable independiente a la aplicación LojaHouse y como variables dependientes: la usabilidad y aceptación de la aplicación.

- ✓ **Facilidad de uso de la aplicación (FU-App):** medida relacionada al nivel de facilidad para la utilización de la aplicación.

d. Formulación de hipótesis

Se formularon las siguientes hipótesis relacionadas con las dos variables dependientes propuestas:

$H_{1,0}$: No se puede afirmar que la aplicación es fácil de usar para los sujetos.

$H_{1,1}:\neg H_{1,0}$

e. Elección del diseño

El diseño seleccionado es experimental ya que nos permite determinar los efectos que se producen al modificar la variable dependiente, es decir al poner a prueba la aplicación y permite ver el impacto que éste produce frente a los sujetos.

f. Instrumentación

Se empleó el instrumento de post test, donde previamente se prepararon las actividades que deberían realizar los sujetos, con el fin de poder contrastar con las hipótesis formuladas. Para llevar a cabo la experimentación se consideró la utilización de las siguientes instrucciones (Ver Tabla CXLIX, CL):

Tabla CXLVI. Instrucciones para el arrendador

Tareas	Descripción de las tareas.
Usuario Arrendador.	
T1	Ingresar al sitio web: https://frontendlh.web.app/#/principal
T2	Dar clic en registrarse.
T3	Completar los campos y crear cuenta.
T4	Revisar contraseña enviada a su correo electrónico.
T5	Iniciar sesión.
Gestión de inmuebles	
T6	Dar clic en el menú, seleccionar: gestión de inmuebles.
T7	Dar clic en el botón: nuevo inmueble.
T8	Completar formulario.
T9	Subir imágenes del inmueble.
T10	Dar clic en el botón para editar inmuebles.
T11	Modificar información de los campos de texto.
T12	Modificar imágenes.
T13	Dar clic en eliminar inmueble.
Gestión de visitas	

T14	Dar clic en el menú, seleccionar: gestión de visitas.
T15	Dar clic en los botones para filtrar visitas por: pendientes, aceptadas, atendidas o todas.
T16	Dar clic en el botón para aceptar visita.
T17	Dar clic en el botón para rechazar visita.
T18	Dar clic en el botón para generar contrato de alquiler.
	Gestión de contratos de alquiler
T19	Dar clic en el menú, seleccionar: gestión de contratos de alquiler.
T20	Dar clic en el botón de ver contrato
T21	Dar clic en el botón de descargar el contrato
T22	Dar clic en el botón para regresar
T23	Dar clic en el botón para terminar un contrato
T24	Dar clic en terminar contrato
T25	Dar clic en habilitar inmueble
	Alquilar inmueble
T26	Dar clic en el menú, seleccionar: alquilar inmueble.
T27	Dar clic en el botón: alquilar
T28	Completar campos de texto
T29	Buscar o registrar un usuario
T30	Dar clic en generar contrato

Tabla CXLVII. Instrucciones para el arrendatario

Tareas	Descripción de las tareas.
	Usuario Arrendatario.
T1	Ingresar a la PlayStore buscar, descargar e instalar la aplicación LojaHouse
T2	Ejecutar o abrir la aplicación
T3	Dar clic en el botón login.
T4	Dar clic en la opción: regístrate
T5	Completar los campos y crear cuenta.
T6	Revisar contraseña enviada a su correo electrónico.
T7	Iniciar sesión.
	Generar solicitud de alquiler
T8	Dar clic en el inmueble publicado.

T9	Visualizar detalle del bien inmueble.
T10	Completar campos de texto
T11	Dar clic en el botón solicitar visita
	Visualizar/eliminar solicitud de alquiler
T12	Dar clic en el menú de opciones.
T13	Seleccionar la opción de visitas solicitadas.
T14	Dar clic en el botón de eliminar solicitud.
	Visualizar/aceptar contrato de alquiler
T15	Dar clic en el menú de opciones.
T16	Seleccionar la opción de contratos de alquiler.
T17	Dar clic en el contrato.
T18	Visualizar los datos del contrato.
T19	Dar clic en el botón: leer contrato.
T20	Dar clic en el botón de la casilla y aceptar contrato.

3. Operación

Durante esta actividad se realizaron tres tareas: preparación, ejecución y validación de datos, mismas que se detallan a continuación.

a. Preparación

Antes de iniciar con la experimentación, se pidió a los participantes su consentimiento para la ejecución del mismo, luego indicó las tareas que debían realizar en la aplicación durante la ejecución del experimento. Finalmente se procedió a notificar que al finalizar la experimentación tendrían que evaluar al prototipo a través de una encuesta.

b. Ejecución

Para llevar a cabo las pruebas se procedió a invitar de manera voluntaria a los participantes en rol de arrendadores por medio de las Aplicaciones Facebook y WhatsApp; para ello se envió la dirección web desde el cual accedían a la aplicación LojaHouse, para el caso de los participantes de la carrera de Ingeniería en Sistemas/Computación, en su rol de arrendatarios, se solicitó descargar la aplicación móvil de la Play Store de Google para llevar a cabo las pruebas, las mismas que fueron efectuadas en el lapso de 27 días obteniendo un total de 63 personas encuestadas que probaron la aplicación LojaHouse. Es importante resaltar que, por motivos de la pandemia, COVID-19, no se pudo realizar esta fase con más sujetos de prueba.

Para evitar posibles sesgos no se les mencionó a los individuos las hipótesis bajo estudio, solamente se les manifestó que se estaba evaluando la aplicación “LojaHouse”. Además se les informó que los datos recogidos en la encuesta serían en forma anónima y confidencial.

c. Validación de datos

Posterior a la ejecución del experimento, para la validación de los datos se aplicó el método de la encuesta, la misma que fue realizada en base a un cuestionario estandarizado PSSUQ (Cuestionario de Usabilidad del Sistema Posterior al Estudio) utilizado en [41]. Este cuestionario permite obtener respuestas en base a la escala de Likert la cual se aplicó luego que el usuario interactuó con la Aplicación Informática LojaHouse (ver Anexo 8: Desarrollo de la encuesta para estudiantes); la cual permitió la recolección de información empírica; dicha información fue almacenada y tabulada por el investigador en un documento Excel.

4. Análisis e interpretación

El resultado de cada pregunta de la encuesta realizada acerca de las pruebas de usabilidad de la aplicación LojaHouse es la siguiente:

1. En general, estoy satisfecho con lo fácil que es utilizar este sistema

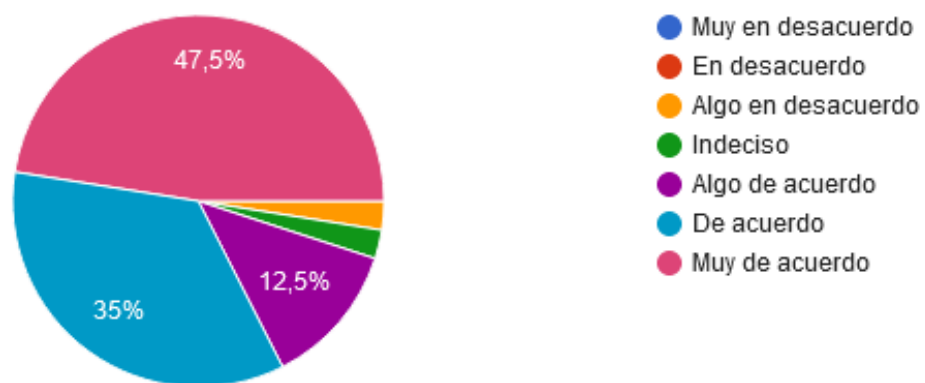


Figura 89. Prueba de Usabilidad - Pregunta 1

2. Fue sencillo utilizar este sistema

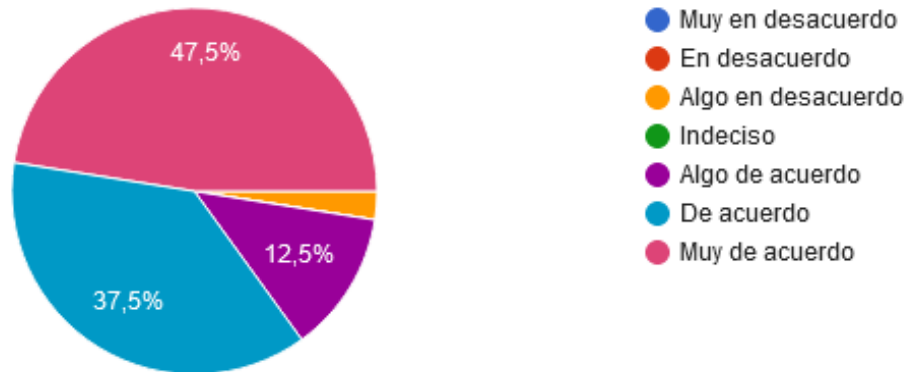


Figura 90. Prueba de Usabilidad - Pregunta 2

3. Pude completar las tareas y escenarios rápidamente usando este sistema

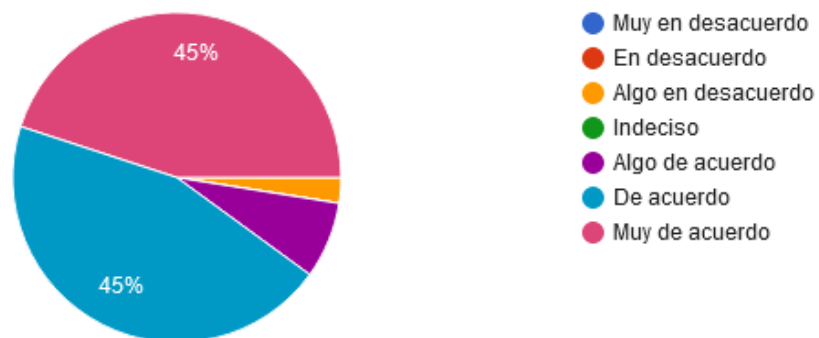


Figura 91. Prueba de Usabilidad - Pregunta 3

4. Me sentí cómodo usando este sistema

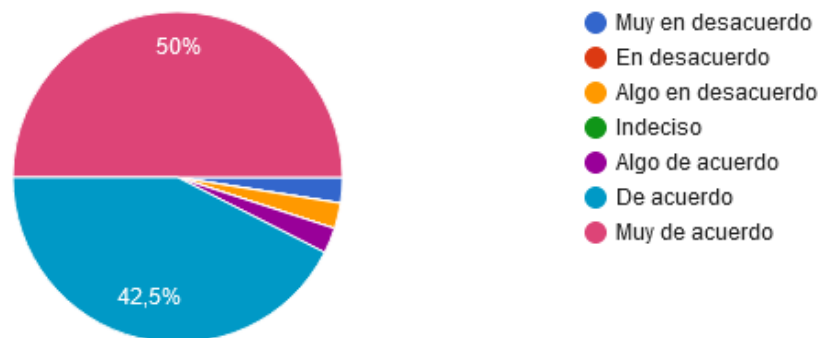


Figura 92. Prueba de Usabilidad - Pregunta 4

5. Fue fácil aprender a usar este sistema

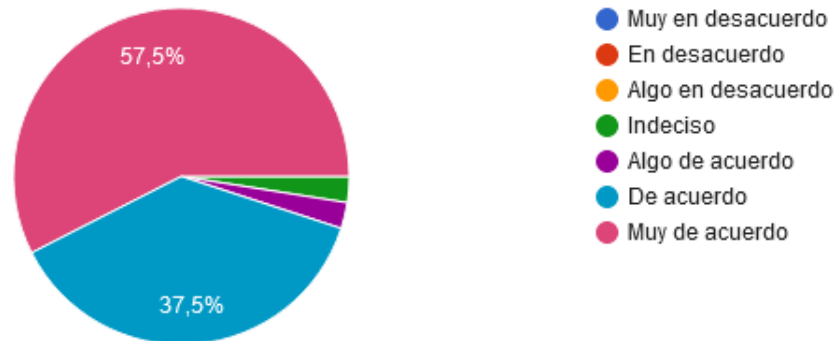


Figura 93. Prueba de Usabilidad - Pregunta 5

6. Creo que podría volverme productivo rápidamente usando este sistema

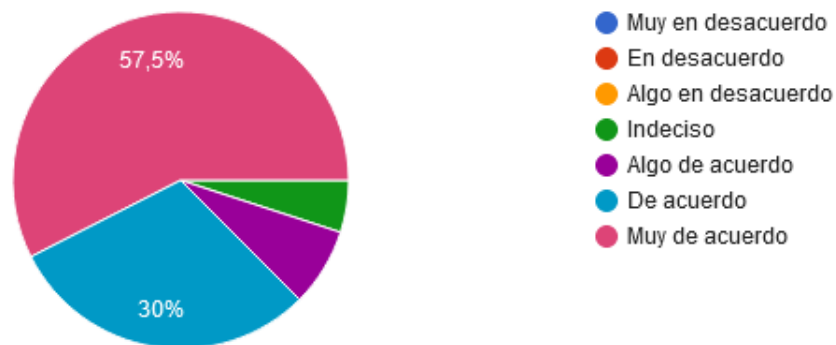


Figura 94. Prueba de Usabilidad - Pregunta 6

7. El sistema dio mensajes de error que me indicaban claramente cómo solucionar problemas

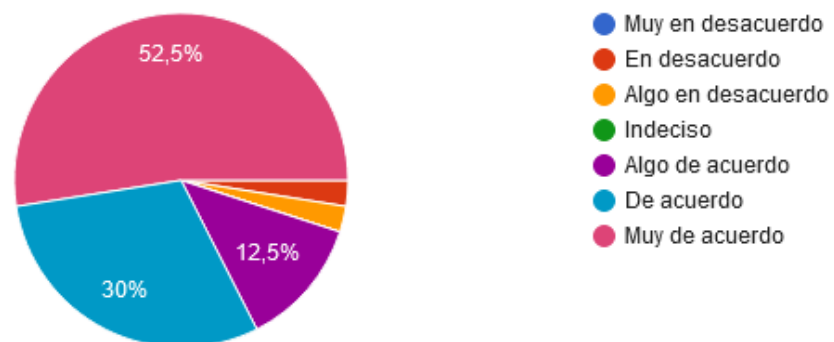


Figura 95. Prueba de Usabilidad - Pregunta 7

8. Siempre que cometía un error al utilizar el sistema, podía recuperarme fácil y rápidamente.

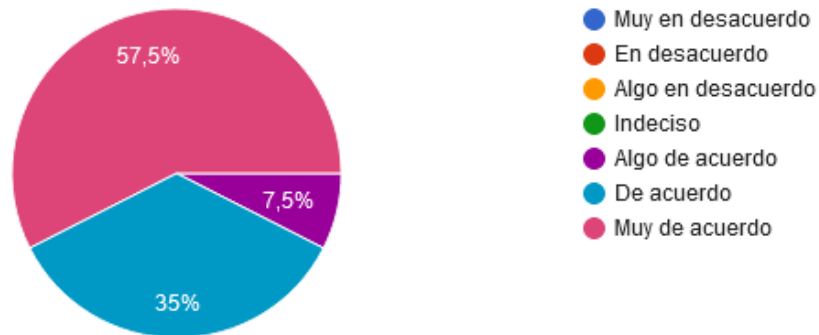


Figura 96. Prueba de Usabilidad - Pregunta 8

9. La información (como ayuda en línea, mensajes en pantalla y otra documentación) proporcionada con este sistema era clara.

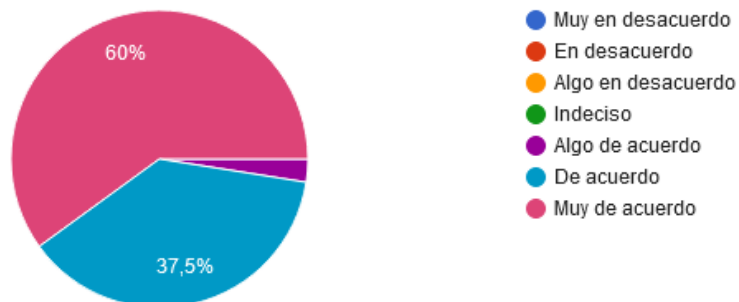


Figura 97. Prueba de Usabilidad - Pregunta 9

10. Fue fácil encontrar la información que necesitaba.

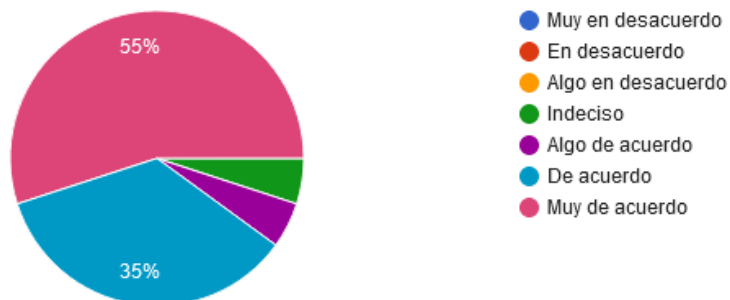


Figura 98. Prueba de Usabilidad - Pregunta 10

11. La información fue eficaz para ayudarme a completar las tareas y los escenarios.

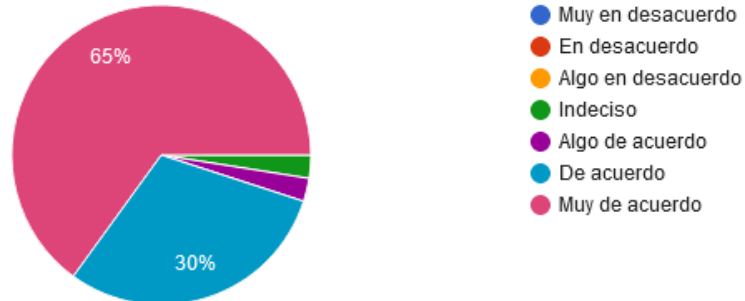


Figura 99. Prueba de Usabilidad - Pregunta 11

12. La organización de la información en las pantallas del sistema fue clara.

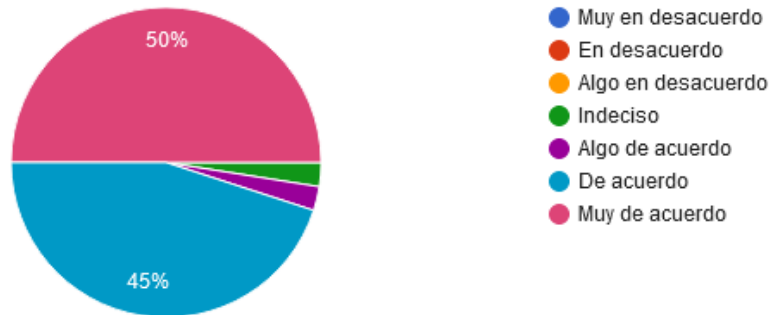


Figura 100. Prueba de Usabilidad - Pregunta 12

13. La interfaz de este sistema fue agradable.

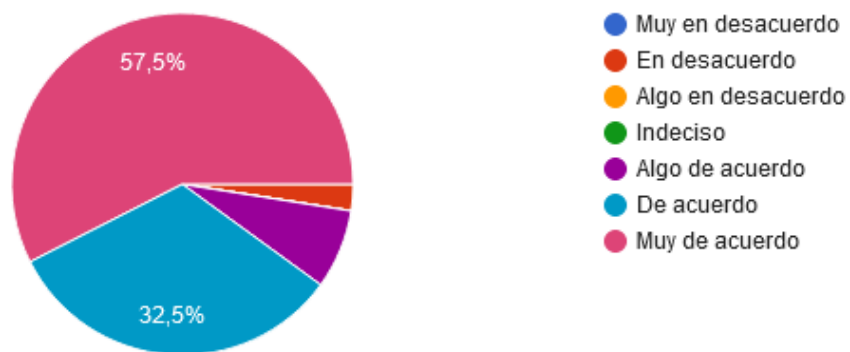


Figura 101. Prueba de Usabilidad - Pregunta 13

14. Me gustó usar la interfaz de este sistema.

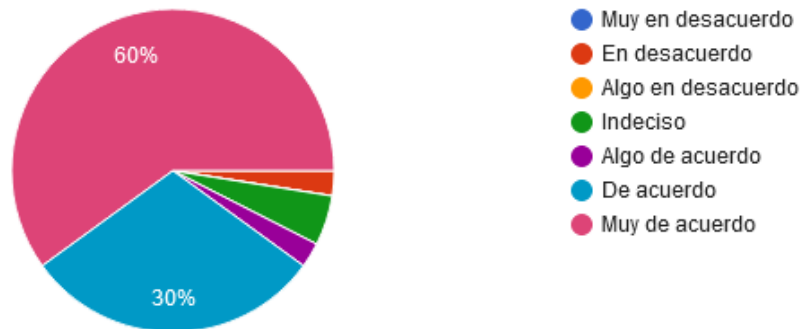


Figura 102. Prueba de Usabilidad - Pregunta 14

15. Este sistema tiene todas las funciones y capacidades que espero que tenga.

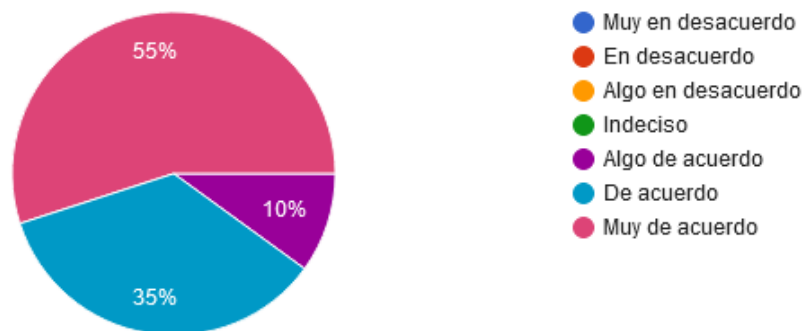


Figura 103. Prueba de Usabilidad - Pregunta 15

16. En general, estoy satisfecho con este sistema.

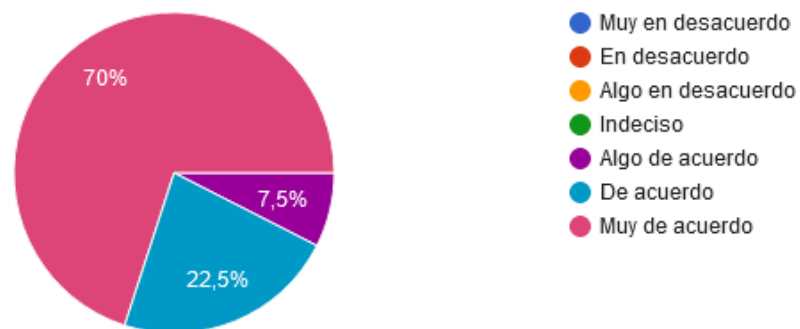


Figura 104. Prueba de Usabilidad - Pregunta 16

Para el análisis de las respuestas obtenidas, se asignó una puntuación o nivel a cada ítem como se muestra en la Tabla CL para una mejor manipulación de los resultados.

Tabla CXLVIII. Puntuación de cada ítem

Grado de usabilidad	Niveles
Muy en desacuerdo	1
En desacuerdo	2
Algo en desacuerdo	3
Indeciso	4
Algo de acuerdo	5
De acuerdo	6
Muy de acuerdo	7

A cada pregunta se identificó como ítem para mayor facilidad de manipulación de los datos (ver Tabla CLI).

Tabla CXLIX. Identificación de ítems

Pregunta	Ítem
En general, estoy satisfecho con lo fácil que es utilizar este sistema	1
Fue sencillo utilizar este sistema	2
Pude completar las tareas y escenarios rápidamente usando este sistema	3
Me sentí cómodo usando este sistema	4
Fue fácil aprender a usar este sistema	5
Creo que podría volverme productivo rápidamente usando este sistema	6
El sistema dio mensajes de error que me indicaban claramente cómo solucionar problemas	7
Siempre que cometía un error al utilizar el sistema, podía recuperarme fácil y rápidamente.	8
La información (como ayuda en línea, mensajes en pantalla y otra documentación) proporcionada con este sistema era clara.	9
Fue fácil encontrar la información que necesitaba.	10
La información fue eficaz para ayudarme a completar las tareas y los escenarios.	11
La organización de la información en las pantallas del sistema fue clara.	12
La interfaz de este sistema fue agradable.	13
Me gustó usar la interfaz de este sistema.	14
Este sistema tiene todas las funciones y capacidades que espero que tenga.	15
En general, estoy satisfecho con este sistema.	16

Para la asignación de cada puntuación se realizó en base a las respuestas de cada encuesta realizada a los 40 usuarios como se muestra en la Tabla CLII.

Tabla CL. Respuestas obtenidas de la encuesta

Encuestados	Ítem															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
2	7	6	6	7	7	7	7	6	7	7	7	7	6	6	7	7
3	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
4	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
5	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
6	6	6	7	7	6	6	7	6	7	7	7	6	7	7	6	7
7	7	6	6	7	7	6	7	7	7	6	7	6	7	7	6	7
8	7	5	6	7	7	7	7	7	7	7	7	7	7	7	7	7
9	7	7	6	6	5	7	7	7	7	7	7	7	7	7	7	7
10	5	5	5	6	6	7	7	7	7	7	7	7	7	7	7	7
11	5	6	7	7	6	7	7	7	7	7	7	7	7	7	7	7
12	7	5	7	7	7	7	7	7	7	7	7	7	7	7	7	7
13	7	5	6	7	7	7	7	7	7	7	7	7	7	7	7	7
14	6	7	6	6	7	6	5	7	7	7	7	7	7	6	7	7
15	7	7	7	1	7	7	6	7	7	7	7	7	7	7	7	7
16	5	6	6	6	7	6	6	6	6	6	6	6	6	6	6	6
17	6	6	6	6	6	6	6	6	6	6	6	6	5	6	6	6
18	6	6	6	6	6	6	5	5	7	7	6	6	5	6	5	6
19	6	6	6	6	4	4	5	6	5	5	6	6	6	4	5	6
20	6	7	6	7	6	7	6	6	6	7	7	7	6	7	6	7
21	6	6	6	6	7	7	7	7	7	6	7	6	6	7	6	7
22	6	7	7	7	7	7	3	5	6	5	6	6	6	6	6	6
23	6	6	6	6	6	6	2	7	6	6	6	6	6	6	6	6
24	6	6	7	6	6	7	6	6	7	6	6	7	6	7	6	7
25	7	7	7	7	6	7	6	6	6	6	7	6	7	7	7	6
26	7	7	7	7	7	7	6	7	7	7	7	7	7	7	7	7
27	6	7	6	6	7	5	7	6	7	6	7	6	6	6	6	7
28	7	6	6	7	6	7	5	5	6	7	7	6	7	6	7	6
29	6	7	6	6	6	5	6	6	6	6	6	6	5	6	6	7
30	4	6	5	5	7	6	5	7	6	6	6	6	6	6	6	6
31	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
32	5	5	3	6	6	4	7	7	6	6	6	4	6	4	7	5
33	5	6	5	6	6	6	6	6	6	4	6	6	6	5	5	5
34	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
35	6	7	6	6	6	6	6	6	6	7	6	6	6	6	6	7
36	6	6	7	7	7	7	6	6	6	6	7	6	7	7	6	7
37	7	7	7	7	7	7	7	7	6	4	5	6	7	7	7	7
38	3	3	7	3	6	6	6	5	6	6	4	5	2	2	5	5
39	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7

40	7	7	6	6	6	6	7	7	7	7	7	7	7	7	7	7
----	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Para el análisis de los datos obtenidos se realizó la sumatoria de la frecuencia de las respuestas obtenidas en cada ítem como se muestra en la Tabla CLIII.

Tabla CLI. Frecuencia por ítem

Grado de usabilidad	Ítem																Grado de usabilidad Total	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16		
Muy en desacuerdo	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	
En desacuerdo	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1	0	0	3
Algo en desacuerdo	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	5
Indeciso	1	0	0	0	1	2	0	0	0	2	1	1	0	2	0	0	10	
Algo de acuerdo	5	5	3	1	1	2	5	4	1	2	1	1	3	1	4	3	42	
De acuerdo	14	15	18	16	16	22	22	33	35	33	32	38	33	32	44	29	222	
Muy de acuerdo	19	19	18	21	22	24	21	33	34	33	36	30	33	44	22	28	357	
Total	40	40	40	40	40	40	40	40	40	40	40	40	40	40	40	40	640	

Finalmente se graficó la columna del grado de usabilidad y el grado de usabilidad total contenidos en la Tabla CLIII como se muestra en la Figura 193, donde el 56% de los encuestados están muy de acuerdo en que la aplicación es fácil de usar, el 35% está de acuerdo y el 9% está algo en desacuerdo.

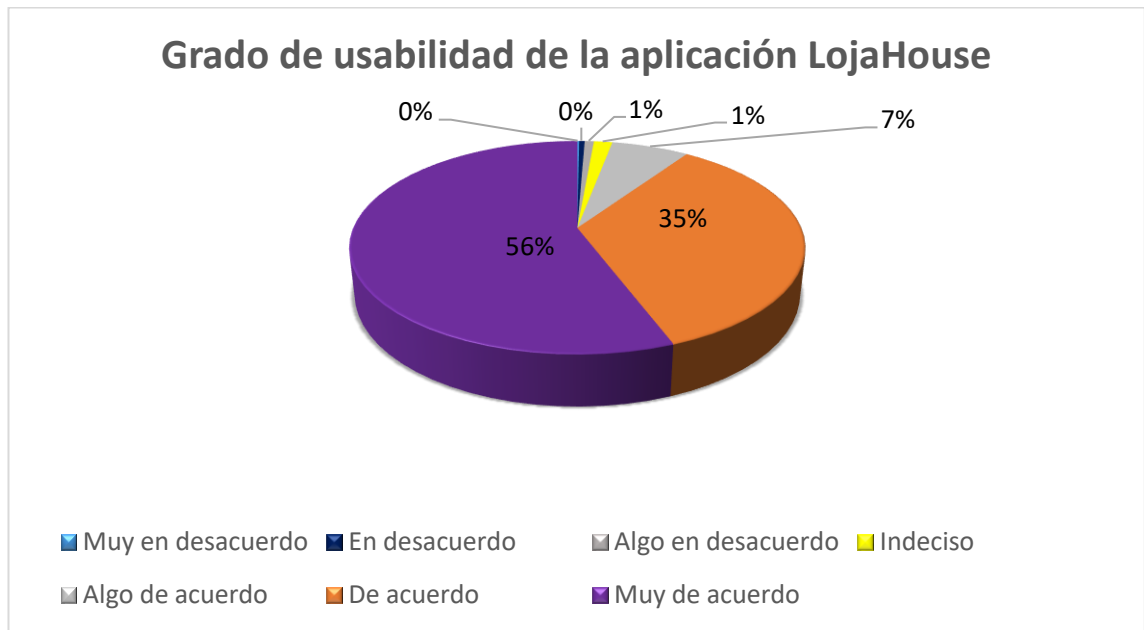


Figura 105. Grado de usabilidad total

5. Resumen y conclusión del análisis de datos

La mayor parte de los sujetos manifestaron que la aplicación es fácil de usar, por lo cual la mayoría no requirió ayuda al realizar tareas de funcionalidad y revelaron que consideran que la aplicación es fácil de usar para cualquier tipo de usuario que tenga un dispositivo móvil, por lo cual la hipótesis nula H_1 se puede negar y aceptar la hipótesis alternativa.

6. Amenazas a la validez

En este apartado se detalla algunos aspectos que pueden haber afectado contra la validez del experimento y la forma en que se intentaron mitigar.

- ✓ **Validez externa:** se relaciona con la posibilidad de la generalización de resultados, se pudo haber afectado con la utilización de estudiantes y los habitantes de la ciudad de Loja como sujetos en la experimentación, aunque en este caso eran idóneos para las tareas que debían realizar. Por ello se considera los resultados como válidos en el ámbito de usuarios arrendadores y arrendatarios.
- ✓ **Validez interna:** se intentaron mitigar con el diseño del experimento, aquí se estableció el proceso, los materiales y el cuestionario a ser aplicado.
- ✓ **Validez de constructo:** se relaciona con las medidas utilizadas en el cuestionario post-test, para este caso se utilizó Cuestionario de Usabilidad del Sistema Posterior al Estudio el cual permite evaluar los datos en la escala de Likert.