



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN SISTEMAS

**DESARROLLO DE UN SISTEMA INFORMÁTICO
MULTIPLATAFORMA DEDICADO A LA ADOPCIÓN Y
RESCATE DE MASCOTAS PARA LA CORPORACIÓN LADRA
EMPLEANDO LA TECNOLOGÍA ANGULAR-IONIC &
CODEIGNITER**

Trabajo de Integración Curricular

Tipo: Proyecto Técnico

Presentando para optar al grado académico de:

INGENIERO EN SISTEMAS INFORMÁTICOS

AUTOR: DANNY ALEXANDER USCA FARINANGO

DIRECTOR: Ing. DIEGO FERNANDO ÁVILA PESANTEZ

Riobamba – Ecuador

2021

© 2021, **Danny Alexander Usca Farinango**

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

Yo, Danny Alexander Usca Farinango, declaro que el presente trabajo de integración curricular es de mi autoría y los resultados de este son auténticos. Los textos en el documento que provienen de otras fuentes están debidamente citados y referenciados.

Como autor asumo la responsabilidad legal y académica de los contenidos de este trabajo de integración curricular; el patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

Riobamba, 19 de marzo de 2021



Danny Alexander Usca Farinango

171918194-1

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN SISTEMAS

El Tribunal del Trabajo de Integración Curricular certifica que: El trabajo de integración curricular; tipo: Proyecto Técnico: **DESARROLLO DE UN SISTEMA INFORMÁTICO MULTIPLATAFORMA DEDICADO A LA ADOPCIÓN Y RESCATE DE MASCOTAS PARA LA CORPORACIÓN LADRA EMPLEANDO LA TECNOLOGÍA ANGULAR- IONIC & CODEIGNITER**, realizado por el señor **DANNY ALEXANDER USCA FARINANGO**, ha sido minuciosamente revisado por los Miembros del Tribunal del Trabajo de Integración Curricular, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal Autoriza su presentación.

	FIRMA	FECHA
Ing. Danilo Mauricio Pastor Ramírez PRESIDENTE DEL TRIBUNAL	 Firmado digitalmente por DANILO MAURICIO PASTOR RAMIREZ Fecha: 2021.05.03 18:15:29 -05'00'	19/03/2021
Ing. Diego Fernando Ávila Pesantez DIRECTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR	 Digitally signed by DIEGO FERNANDO AVILA PESANTEZ Date: 2021.05.10 15:12:47 -05'00'	19/03/2021
Ing. Raúl Hernán Rosero Miranda MIEMBRO DEL TRIBUNAL	 RAUL HERNAN ROSERO MIRANDA	19/03/2021

DEDICATORIA

A mi familia, principalmente a mis padres Blanca y César, gracias a su apoyo incondicional, amor y sabiduría lograron hacer de mí una persona correcta, enseñándome que la vida está llena más de triunfos que de fracasos; a mis hermanas Fernanda y Anabel por sus consejos y paciencia en todo momento. A mis amigos a quienes aprecio bastante, a todas esas personas que han estado desde el comienzo y siguen hasta hoy les dedico este pequeño logro que para mí representa la culminación de mucho esfuerzo y dedicación.

Danny

AGRADECIMIENTO

Mi más sincero agradecimiento a la Escuela Superior Politécnica de Chimborazo, institución que me brindó la oportunidad para poder continuar con mi educación superior, a la Escuela de Ingeniería en Sistemas en la cual logré adquirir el conocimiento para alcanzar mis metas, a todos los docentes quienes con paciencia y mucha dedicación lograron transmitir sus enseñanzas de la mejor manera, a mi familia por apoyarme en toda mi etapa estudiantil y un agradecimiento especial a los Ingenieros Diego Ávila y Raúl Rosero por sus enseñanzas, conocimientos, apoyo, paciencia y guía brindada en este trabajo de Integración Curricular.

Danny

TABLA DE CONTENIDO

ÍNDICE DE TABLAS.....	xi
ÍNDICE DE FIGURAS.....	xiii
ÍNDICE DE GRÁFICOS.....	xiv
ÍNDICE DE ANEXOS.....	xv
RESUMEN.....	xvi
SUMMARY.....	xvii
INTRODUCCIÓN.....	1

CAPÍTULO I

1. DIAGNÓSTICO DEL PROBLEMA.....	3
1.1. Antecedentes.....	3
1.2. Formulación del problema.....	5
1.2.1. Sistematización del problema.....	5
1.3. Justificación.....	5
1.3.1. Justificación teórica.....	5
1.3.2. Justificación aplicativa.....	7
1.4. Objetivos.....	9
1.4.1. Objetivo general.....	9
1.4.2. Objetivos específicos.....	9

CAPÍTULO II

2. MARCO TEÓRICO REFERENCIAL.....	10
2.1. Aplicaciones Informáticas.....	10
2.1.1. Aplicaciones Web.....	10
2.1.1.1. <i>Ventajas</i>	10
2.1.1.2. <i>Desventajas</i>	10
2.1.2. Aplicaciones Web Progresivas (PWA).....	10
2.1.2.1. Características.....	11
2.1.3. Aplicaciones de Pagina Única (SPA).....	12
2.1.3.1. <i>Arquitectura</i>	12
2.1.3.2. <i>Optimización de una SPA</i>	13
2.1.4. Aplicaciones Móviles Nativas.....	13

2.1.4.1.	<i>Ventajas</i>	14
2.1.4.2.	<i>Desventajas</i>	14
2.1.5.	Aplicaciones Móviles Híbridas	14
2.1.5.1.	<i>Ventajas</i>	15
2.1.5.2.	<i>Desventajas</i>	15
2.1.6.	Comparativa entre aplicaciones	16
2.2.	Servicios Web RESTFUL	16
2.2.1.	REST	16
2.2.1.1.	<i>Ventajas y desventajas</i>	17
2.2.2.	CodeIgniter	18
2.2.3.	Symfony	18
2.2.4.	Laravel	18
2.2.5.	Comparativa entre frameworks	18
2.3.	Framework para desarrollo de Aplicaciones Multiplataforma	19
2.3.1.	Angular	20
2.3.2.	Ionic	20
2.3.3.	Xamarin	20
2.3.4.	Flutter	21
2.3.5.	React Native	21
2.3.6.	Comparativa entre frameworks	21
2.4.	Herramientas de Desarrollo para Aplicaciones Multiplataforma	22
2.4.1.	Visual Studio Code	22
2.4.2.	XAMPP Server	22
2.4.3.	Bootstrap Studio	23
2.4.4.	Postman	23
2.4.5.	Android Studio	23
2.4.6.	Xcode	23
2.4.7.	VMware Workstation	24
2.5.	Servidor Web APACHE	24
2.5.1.	Características	24
2.5.2.	Ventajas y desventajas	24
2.6.	MySQL	25
2.6.1.	Características	25
2.6.2.	Comparativa entre bases de datos	25
2.7.	Patrón de Arquitectura Modelo, Vista, Controlador (MVC)	26
2.7.1.	Niveles	26

2.7.2.	Funcionamiento	27
2.7.3.	Ventajas	27
2.7.4.	Desventajas	28
2.8.	Gestión de Proyectos Ágiles – Metodología SCRUM	28
2.8.1.	Flujo de proceso de SCRUM	29
2.8.2.	Comparación con otras metodologías	29
2.9.	TestLink	30
2.9.1.	Características	30
2.9.2.	Plan de Pruebas	31
2.9.2.1.	<i>Caso de Prueba</i>	31
2.9.3.	TestLink y la metodología SCRUM	31
2.10.	NORMA ISO/IEC 25010	33
2.10.1.	Características	33
2.10.2.	Usabilidad	34
2.11.	Trabajos similares	35

CAPITULO III

3.	MARCO METODOLÓGICO	36
3.1.	Actividades de la metodología	36
3.1.1.	Tipo de Investigación	36
3.1.2.	Métodos de investigación	36
3.1.2.1.	<i>Método Deductivo</i>	36
3.1.2.2.	<i>Método Inductivo</i>	37
3.1.2.3.	<i>Método Analítico</i>	37
3.1.3.	Técnicas de investigación	37
3.1.3.1.	<i>Observación</i>	37
3.1.3.2.	<i>Entrevista</i>	37
3.1.3.3.	<i>Cuestionario</i>	37
3.1.3.4.	<i>TestLink como plan de pruebas</i>	38
3.2.	Fase de Planificación	38
3.2.1.	Personas y Roles involucrados en el proyecto	39
3.2.2.	Tipo de roles de usuarios del sistema	39
3.2.3.	Product Backlog	39
3.2.4.	Historia de usuario	42
3.2.5.	Sprint backlog	43

3.3.	Fase de desarrollo.....	45
3.3.1.	Arquitectura del sistema.....	45
3.3.2.	Diagramas UML.....	45
3.3.3.	Estándar de codificación.....	48
3.3.4.	Diseño de la base de datos	48
3.3.4.1.	<i>Diseño físico.....</i>	48
3.3.4.2.	<i>Diccionario de datos</i>	49
3.3.5.	Plan de pruebas	50
3.3.5.1.	<i>Instalación del software TestLink</i>	50
3.3.5.2.	<i>Creación de un Plan de Pruebas</i>	54
3.3.5.3.	<i>Creación de una Suite de Pruebas</i>	54
3.3.5.4.	<i>Creación de los Casos de Pruebas.....</i>	56
3.3.5.5.	<i>Ejecución de un Caso de Prueba</i>	58
3.3.6.	Diseño de interfaces.....	60
3.3.6.1.	<i>Aplicativo web.....</i>	60
3.3.6.2.	<i>Aplicativo móvil</i>	61
3.3.7.	Reuniones y entregables	62
3.3.8.	Recursos para utilizar	63
3.3.8.1.	<i>Recurso Software</i>	63
3.3.8.2.	<i>Recurso Hardware</i>	64
3.4.	Fase de finalización	65
3.4.1.	Gestión del proyecto.....	65
3.5.	Método utilizado para evaluar la usabilidad de la plataforma	66

CAPITULO IV

4.	MARCO DE RESULTADOS Y DISCUSIÓN.....	67
4.1.	Usabilidad: subcaracterísticas y propósito para evaluar	67
4.2.	Escalas de evaluación.....	68
4.3.	Tamaño de la muestra	68
4.3.1.	Tipo de muestreo probabilístico	69
4.4.	Adecuación del Cuestionario USE de Arnold Lund para la validación	70
4.5.	Análisis de los resultados obtenidos al realizar la evaluación de la plataforma..	71
4.5.1.	Análisis descriptivo	71
4.5.1.1.	<i>Inteligibilidad.....</i>	71
4.5.1.2.	<i>Operabilidad</i>	74

4.5.1.3. <i>Estética</i>	77
4.5.2. Análisis Inferencial	79
4.5.2.1. <i>Inteligibilidad</i>	80
4.5.2.2. <i>Operabilidad</i>	81
4.5.2.3. <i>Estética</i>	83
CONCLUSIONES	85
RECOMENDACIONES	86
GLOSARIO	
BIBLIOGRAFIA	
ANEXOS	

ÍNDICE DE TABLAS

Tabla 1-2:	Características de PWA.....	11
Tabla 2-2:	Plataformas de desarrollo	13
Tabla 3-2:	Comparativa entre aplicaciones informáticas.....	16
Tabla 4-2:	Ventajas de la arquitectura REST.....	17
Tabla 5-2:	Desventajas de la arquitectura REST	17
Tabla 6-2:	Comparativa entre frameworks basados en PHP.....	19
Tabla 7-2:	Diferentes frameworks para enfoques multiplataforma	19
Tabla 8-2:	Comparativa entre frameworks para desarrollo multiplataforma.....	21
Tabla 9-2:	Comparación de funciones de bases de datos relacionales.....	25
Tabla 10-2:	Comparación de SCRUM con otras metodologías.....	29
Tabla 11-2:	Forma de trabajo de Scrum y TestLink	33
Tabla 12-2:	Modelos de calidad, características y subcaracterísticas	33
Tabla 1-3:	Roles y Personas.....	39
Tabla 2-3:	Tipos de roles de usuarios	39
Tabla 3-3:	Puntos estimados T-Shirt.....	40
Tabla 4-3:	Product Backlog	40
Tabla 5-3:	Historia de Usuario.....	42
Tabla 6-3:	Sprint backlog.....	43
Tabla 7-3:	Definición del estándar de codificación	48
Tabla 8-3:	Diccionario de datos	50
Tabla 9-3:	Reuniones y entregables	63
Tabla 10-3:	Recursos software.....	64
Tabla 11-3:	Recurso hardware	64
Tabla 1-4:	Características, subcaracterísticas y propósito a evaluar.....	67
Tabla 2-4:	Escala cualitativa (escala de Likert) a usar.....	68
Tabla 3-4:	Muestreo probabilístico estratificado	69
Tabla 4-4:	Adaptación del cuestionario USE para evaluar la Usabilidad	70
Tabla 5-4:	Parámetros utilizados para obtener resultados de la tabulación	71
Tabla 6-4:	Tabulación de las preguntas de la subcaracterística de Inteligibilidad.....	72
Tabla 7-4:	Resultados de la subcaracterística de Inteligibilidad.....	73
Tabla 8-4:	Tabulación de las preguntas de la subcaracterística de Operabilidad.....	74
Tabla 9-4:	Resultados de la subcaracterística de Operabilidad.....	76
Tabla 10-4:	Tabulación de las preguntas de la subcaracterística de Estética.....	77
Tabla 11-4:	Resultados de la subcaracterística de Estética.....	78

Tabla 12-4: Valor de mu para cada subcaracterística de la usabilidad 80

ÍNDICE DE FIGURAS

Figura 1-2:	Componentes de una PWA	11
Figura 2-2:	Arquitectura SPA	12
Figura 3-2:	Estructura de una aplicación móvil híbrida.....	14
Figura 4-2:	Arquitectura de un complemento	15
Figura 5-2:	Diagrama de secuencia de MVC.....	27
Figura 6-2:	Metodología SCRUM	28
Figura 7-2:	Forma de trabajo simplificada con SCRUM.....	32
Figura 8-2:	Forma de trabajo con TestLink	32
Figura 1-3:	Diagrama de Gantt del Plan de Trabajo	38
Figura 2-3:	Pantalla de instalación de TestLink 1.9	51
Figura 3-3:	Términos y aceptación de licencia	51
Figura 4-3:	Verificación de los requerimientos del sistema	52
Figura 5-3:	Definición del acceso a la base de datos	53
Figura 6-3:	Pantalla de autenticación de TestLink	53
Figura 7-3:	Creación de un plan de pruebas en TestLink	54
Figura 8-3:	Plan de pruebas creado para la Corporación LADRA	54
Figura 9-3:	Creación de una Suite de Pruebas	55
Figura 10-3:	Suites de pruebas para la Corporación LADRA	55
Figura 11-3:	Suites de pruebas para la Corporación LADRA - Continuación	56
Figura 12-3:	Creación de un Caso de Prueba.....	56
Figura 13-3:	Caso de Prueba para la Corporación LADRA	57
Figura 14-3:	Ejecución de un caso de prueba para la Corporación LADRA.....	58
Figura 15-3:	Reporte generado al ejecutar un caso de prueba	59
Figura 16-3:	Pantalla de inicio de sesión, aplicativo web.....	60
Figura 17-3:	Pantalla principal, aplicativo web	61
Figura 18-3:	Pantalla login, aplicativo móvil	61
Figura 19-3:	Pantalla principal, aplicativo móvil	62
Figura 1-4:	Cálculo de la normalidad de la inteligibilidad con la función shapiro.test	80
Figura 2-4:	Test de Wilcoxon para validar la subcaracterística de inteligibilidad.....	81
Figura 3-4:	Cálculo de la normalidad de la operabilidad con la función shapiro.test.....	82
Figura 4-4:	Test de Wilcoxon para validar la subcaracterística de operabilidad	82
Figura 5-4:	Calculo de la normalidad de la estética con la función shapiro.test.....	83
Figura 6-4:	Test de Wilcoxon para validar la subcaracterística de estética	84

ÍNDICE DE GRÁFICOS

Gráfico 1-1:	Flujo de trabajo manual	4
Gráfico 2-1:	Flujo de trabajo automatizado	5
Gráfico 1-3:	Arquitectura del sistema - Patrón MVC	45
Gráfico 2-3:	Diagrama de clases	46
Gráfico 3-3:	Diagrama de secuencia	47
Gráfico 4-3:	Diseño físico de la base de datos	49
Gráfico 5-3:	Burn Down Chart	65
Gráfico 1-4:	Comparación entre puntuaciones obtenidas de las respuestas - Inteligibilidad..	74
Gráfico 2-4:	Comparación entre puntuaciones obtenidas de las respuestas - Operabilidad ...	76
Gráfico 3-4:	Comparación entre las puntuaciones obtenidas de las respuestas - Estética	79

ÍNDICE DE ANEXOS

ANEXO A: Diccionario de datos

ANEXO B: Diagramas de secuencia

ANEXO C: Historias Técnicas

ANEXO D: Historias de Usuario

ANEXO E: Cuestionario USE de Arnold Lund usado para medir la Usabilidad

ANEXO F: Cuestionario USE de Arnold Lund usado para medir la Usabilidad (español)

ANEXO G: Cuestionario final adecuado para obtener los resultados

ANEXO H: Reportes generados al ejecutar el plan de pruebas en el software TestLink

RESUMEN

El presente trabajo de integración curricular tuvo como objetivo el desarrollo de una aplicación multiplataforma dedicada a la adopción y rescate de mascotas para la corporación “LADRA” de la ciudad de Riobamba. Aplicando la metodología de desarrollo ágil SCRUM; se obtuvo información esencial para el correcto desarrollo del aplicativo web y móvil, como fue las reuniones establecidas con el cliente en donde se pudo definir un total de 37 requerimientos funcionales y 8 requerimientos no funcionales. Por medio de un estudio comparativo de las diferentes tecnologías de desarrollo multiplataforma, se escogió los frameworks Angular-Ionic y CodeIgniter basados en TypeScript y PHP respectivamente; a la par del desarrollo se definió un plan de pruebas para el proyecto que garantizó la creación de un software eficiente. Finalizada la fase de desarrollo se evaluó la calidad del producto software, específicamente la característica Usabilidad de la Norma ISO/IEC 25010 aplicando una encuesta de 13 preguntas basadas en el cuestionario USE de Arnold Lund; logrando así evaluar las subcaracterísticas de inteligibilidad, operabilidad y estética de la plataforma. Al tabular los resultados se obtuvo para la subcaracterística de inteligibilidad un valor de 25,66 sobre 28, operabilidad un valor de 32,06 sobre 35 y estética un valor de 19,56 sobre 21, concluyendo que la plataforma web y móvil tiene una interfaz agradable, es útil, fácil de usar, satisface los requerimientos funcionales planteados y les ahorra tiempo de proceso, siendo más productivos. No obstante, se recomienda el uso de la herramienta de administración de proyectos Jira por su rápida adaptación con la metodología de desarrollo ágil y plan de pruebas.

Palabras claves: <INGENIERÍA DE SOFTWARE>, <TECNOLOGÍA Y CIENCIAS DE LA INGENIERÍA>, <APLICACIONES MULTIPLATAFORMA>, <SCRUM>, <TESTLINK>, <FRAMEWORK ANGULAR-IONIC>, <FRAMEWORK CODEIGNITER>, <USABILIDAD>.

LUIS
ALBERTO
CAMINOS
VARGAS

Firmado digitalmente por
LUIS ALBERTO CAMINOS
VARGAS
Nombre de
reconocimiento (DN):
c=EC, l=RIOBAMBA,
serialNumber=+60276697
, o=LUIS ALBERTO
CAMINOSVARGAS
Fecha: 2021.04.22
12:10:21 -05'00'



1055-DBRA-UTP-2021

SUMMARY

The objective of the current curriculum integration work was to develop a cross-platform application dedicated to adopt and rescue pets for the “LADRA” company from the city of Riobamba. We obtained the essential information using the agile software development methodology SCRUM for the proper development of the web and mobile application like the established meetings with the client where we defined a total of 37 functional requirements and 8 non-functional requirements. We did a comparative study of the different technologies of cross-platform development to choose the Angular-Ionic y CodeIgniter frameworks based on TypeScript and PHP and we also defined a test plan for the project guarantying the creation of an efficient software. Once the development phase was finished, we evaluated the software product quality focusing on the usability characteristics of the ISO/IEC 25010 standard applying a survey of 13 questions based on the USE test by Arnold Lund, thus managing to evaluate the sub-characteristics of appropriateness recognizability, operability and user interface aesthetics of the platform. When tabulating the results, we obtained a 25,66 value out of 28 for the sub-characteristics of appropriateness recognizability, a 32,06 value out of 35 for operability and a 19,56 value out of 21 for user interface aesthetics. It was also concluded that the web and mobile platform has a user-friendly interface because it is useful, easy to use, satisfies functional requirements established and saves process time, being more productive. Nevertheless, the use of the agile project management tool Jira is recommended because of its rapid adaptation with the agile development methodology and test plan.

Key words: < SOFTWARE ENGINEERING >, <TECHNOLOGY AND ENGINEERING SCIENCES>, <CROSS-PLATFORM APPLICATION>, <SCRUM (AGILE SOFTWARE DEVELOPMENT METHODOLOGY)>, <TESTLINK>, <ANGULAR-IONIC FRAMEWORK>, <CODEIGNITER FRAMEWORK >, <USABILITY>.

INTRODUCCIÓN

La forma exponencial en que va creciendo la tecnología, hace que la sociedad comience a depender más de ella y la acople en su diario vivir como una herramienta importante para realizar sus actividades cotidianas. Específicamente el uso de plataformas web y móviles constituye que exista este gran crecimiento pues, es utilizada para la mayoría de actividades cotidianas, desde acceder a entretenimiento, estudio, noticias, libros, comercio electrónico hasta usarlas como herramientas indispensables para facilitar el trabajo.

En la actualidad se puede observar un gran número de plataformas web y móviles destinadas a la búsqueda y adopción de mascotas mediante publicaciones, sin embargo, la mayoría no está centralizada en brindar una solución a un ciudad y entidad en concreto, por lo cual se ha creado la plataforma “LADRA” para la Legión Activista por la Defensa y Respeto Animal.

La finalidad de la plataforma web y móvil “LADRA”, es compartir publicaciones de mascotas que se encuentren extraviadas, como también publicaciones de adopciones y rescate de perros y gatos, en la ciudad de Riobamba. Para la creación del aplicativo, su interfaz de usuario se la realizó empleando los frameworks Angular-Ionic, la comunicación con el servidor se complementó con el uso del framework CodeIgniter y para almacenar la información se empleó el gestor de base de datos MySQL.

En el desarrollo del trabajo de integración curricular se usó la metodología de desarrollo ágil SCRUM, que nos brinda un ambiente de buenas prácticas y garantiza mejores resultados, juntamente para el desarrollo de un software eficiente se utilizó un plan de pruebas. Finalmente, como parte de análisis de los resultados se utilizó el modelo de Calidad del Producto Software (ISO/IEC 25010) para evaluar la usabilidad de la plataforma.

El documento se encuentra dividido en cuatro capítulos de tal forma se tiene:

Capítulo I: Diagnostico del problema, antecedentes, justificación y objetivos a cumplir.

Capítulo II: Marco teórico con la descripción de conceptos, herramientas y metodología que se empleó en el proyecto.

Capítulo III: Detalla la metodología SCRUM y el desarrollo de la plataforma web y móvil.

Capítulo IV: Corresponde al análisis de resultados de Usabilidad al aplicar la Calidad del Producto Software (ISO/IEC 25010).

CAPÍTULO I

1. DIAGNÓSTICO DEL PROBLEMA

1.1. Antecedentes

Una de las características de Riobamba es ser una ciudad que posee varias fundaciones dedicadas totalmente a la adopción y rescate de animales domésticos o callejeros. De varias fundaciones que existen, se ve la necesidad de resaltar una de ellas que responde al nombre de “Corporación LADRA” (Legión Activista por la Defensa y Respeto Animal); LADRA tiene como función principal brindar una mejor calidad de vida para los animales callejeros de la ciudad, enfocándose en la adopción y rescate de los mismos.

Tomando en cuenta dos aspectos fundamentales que intervienen en las actividades de la corporación que son: el rescate de animales que recorren las calles de la ciudad y la adopción de animales que son rescatados; se plantea mejorar la forma de trabajo mediante el uso de la tecnología y de esta manera reducir este inconveniente.

Por medio del presente trabajo de integración curricular se propone dar solución a un problema social que encontramos desde muchos años atrás e involucra tanto a personas como a los animales. El desarrollo de la plataforma se encuentra encaminado, para facilitar el proceso a las distintas personas de la localidad que se comunican cada día a la institución a reportar una pérdida o adoptar una mascota, y a los funcionarios de la corporación que recorren las calles de la ciudad en busca de animales que necesitan ser rescatados. De esta manera se busca mejorar el proceso de pérdida o adopción de mascotas por parte de la ciudadanía riobambeña y por la parte administrativa se desea acelerar y precisar la búsqueda de mascotas que necesiten ser rescatadas.

Tal como afirma (Heredia, 2017) en la actualidad se registra que más del 79% de la población ecuatoriana tiene acceso a internet por diferentes medios electrónicos como: smartphone, Tablets, laptops, computadores de escritorio u otro dispositivo inteligente; del porcentaje antes referido el 56% de la ciudadanía accede a través de un smartphone, por último y un punto importante se agrega que estas métricas se encuentran en un cambio continuo.

Expresado de otra manera, a las personas a quienes va dirigido este proyecto no tendrán inconvenientes a la hora de usar una plataforma web o móvil. Buscando de esta manera que el usuario obtenga los beneficios que brinda la tecnología como es la adopción de mascotas de una

manera virtual y la geolocalización de forma oportuna de animales perdidos o abandonados. Es de conocimiento que un porcentaje considerable de la ciudadanía riobambeña que encuentra un animal abandonado; desconoce la información o procedimiento a seguir para ayudar a este ser vivo y muchas veces en el peor de los escenarios opta por dejarlo sin amparo.

El flujo de trabajo el cual ha manejado la corporación está contemplado en el Gráfico 1-1, donde se puede apreciar el trabajo manual que realizan los funcionarios.

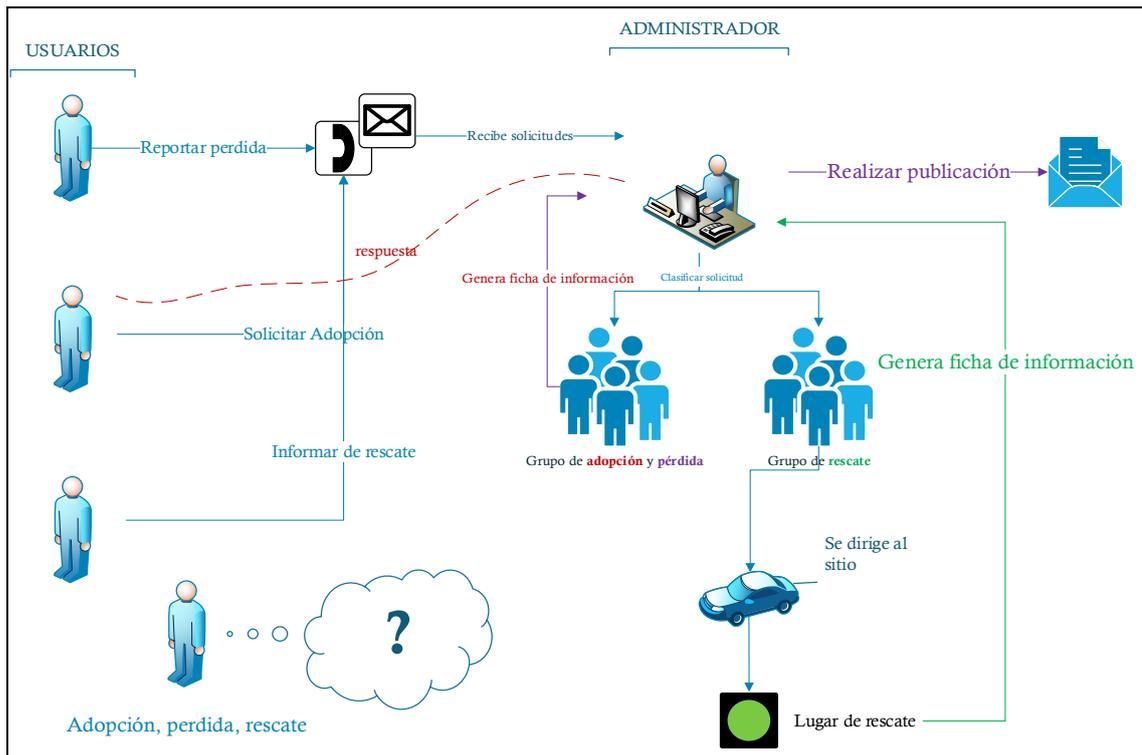


Gráfico 1-1: Flujo de trabajo manual

Realizado por: Usca Danny, 2021

Por los antecedentes expuestos se propone el desarrollo e implementación de una plataforma informática, como propuesta para mejorar el proceso de adopción, rescate que ha llevado a cabo la corporación en ayuda a los animales de la ciudad de Riobamba en los últimos años, pretendiendo generar el siguiente flujo de trabajo automatizado, Gráfico 2-1.

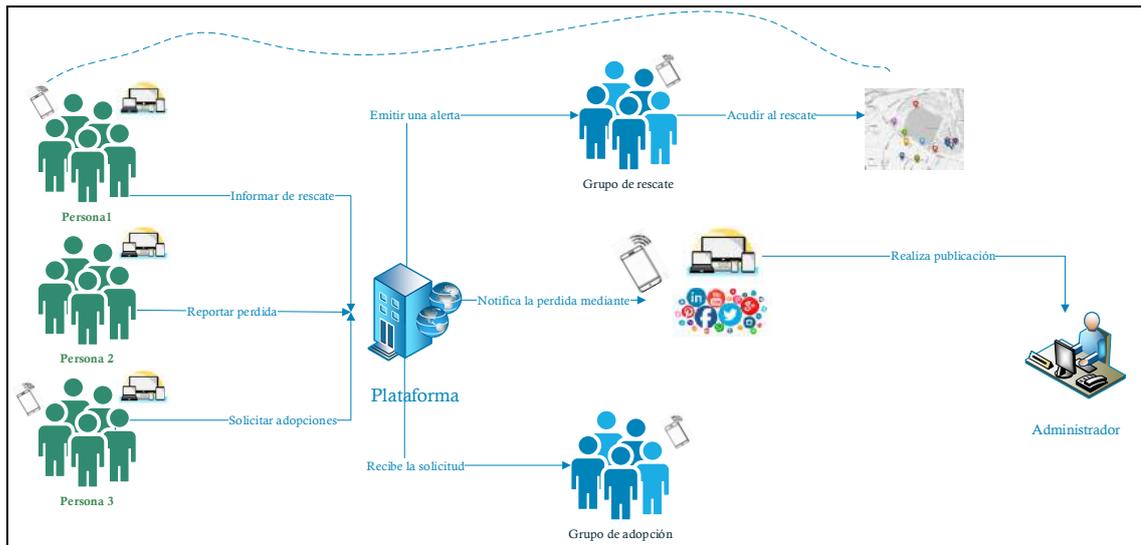


Gráfico 2-1: Flujo de trabajo automatizado
Realizado por: Usca Danny, 2021

1.2. Formulación del problema

¿Cómo reducir el tiempo empleado en la adopción y rescate de mascotas de la Corporación LADRA - Riobamba, utilizando un sistema informático multiplataforma?

1.2.1. Sistematización del problema

- ¿Qué inconvenientes relacionados con la adopción de mascotas enfrentan los ciudadanos que llegan a las instalaciones de la Corporación LADRA - Riobamba?
- ¿Cómo es la forma en que los rescatistas socorren a los animales abandonados en la ciudad de Riobamba?
- ¿Qué características, ventajas y desventajas al momento de desarrollo poseen los framework Angular, Ionic y CodeIgniter?
- ¿Qué medios informativos suelen usar las personas para buscar una mascota en adopción dentro de la ciudad de Riobamba?
- ¿El uso de las Tics puede ayudar con la adopción y rescate de mascotas de manera más rápida?

1.3. Justificación

1.3.1. Justificación teórica

Este proyecto está inclinado en un factor tecnológico debido al uso de dispositivos inteligentes, para mostrar los diferentes lugares, en donde se necesita la ayuda de la Corporación “LADRA”;

además de la información de los usuarios que deseen dar en adopción a diferentes mascotas, por lo tanto, aquí es clave el uso de geolocalización mediante GPS (Sistema de Posicionamiento Global) y notificaciones de forma que se pueda cumplir con este requerimiento.

En el desarrollo de la aplicación móvil se utilizará el framework Ionic, definido como un marco para el desarrollo móvil en HTML5 brinda la facilidad de crear aplicaciones web progresivas e híbridas. Las aplicaciones web progresivas están recibiendo más atención debido a que su mayor parte está basado en lo mejor de varias tecnologías híbridas: por un lado, poseen lo mejor de la web y por otro, lo esencial de aplicaciones nativas. Podemos llegar a la conclusión que este tipo de plataformas están reduciendo el vacío que existía entre las aplicaciones móviles nativas y las webs.

Conforme al desarrollo web se usará el framework Angular, para la creación de una plataforma administrativa que será usado principalmente por el personal de la Corporación “LADRA”, a diferencia de Ionic; Angular es usado en su totalidad para la creación de aplicaciones web de página única por sus siglas denominada SPA, su mayor ventaja frente a una aplicación web convencional es debido a que la navegación entre diferentes paginas o secciones la trabaja en una sola, por medio de peticiones al servidor la carga de información se realiza de forma dinámica, instantánea y asíncronamente, sobre todo sin refrescar la página en ningún momento.

Para complementar el desarrollo en la capa de negocio y datos, se utilizará CodeIgniter que es un framework basado en el lenguaje de programación PHP (*HyperText Preprocessor*); el mismo que proporciona una interfaz de programación para aplicaciones de tipo REST (Transferencia de Estado Representacional); internamente CodeIgniter funciona con una arquitectura MVC (Modelo Vista Controlador) y con ella se propone una mejor forma de trabajar además de dar una mayor velocidad a la hora de crear servicios web.

El proyecto se dividirá en dos partes, una aplicación web y móvil. En la plataforma web esta contará con un panel administrativo con información de la institución y donde los trabajadores de la corporación podrán interactuar con diferentes módulos como: estadísticas de usuario, publicaciones, reportes y noticias o eventos que se vayan a realizar.

El aplicativo móvil pretende contar con la facilidad de poder ubicar a mascotas que necesiten ayuda; a su vez los usuarios podrán visualizar y publicar mascotas perdidas, encontradas o en adopción; parte esencial también son las redes sociales, su uso es vital pues podrán generar avisos si así lo desea el usuario en su perfil.

El marco de trabajo para el desarrollo ágil de la plataforma será SCRUM, la cual es descrita como un proceso de gestión que disminuye la dificultad en el desarrollo de productos, satisfaciendo las necesidades de los clientes; pues de manera continua el equipo de trabajo y el director de proyecto realizan entregas de software funcional en periodos cortos de una hasta cuatro semanas.

Es vital que el código tenga una excelente cobertura de pruebas pues también se pretende medir su usabilidad y TestLink es una herramienta de diseño que ayuda al desarrollador de software a crear un plan de pruebas general e irlos dividiendo en casos de pruebas; dando como resultado que el desarrollador ejecute cada uno de los test a fin de almacenar resultados de forma dinámica, presentar informes, estadísticas y avances de cada requerimiento planteado y en caso de ser necesario dar prioridad a diferentes tareas.

Tomando como base, en las líneas de investigación de la Escuela Superior Politécnica de Chimborazo el respectivo proyecto está contemplado dentro de las Tecnologías de la Información, Comunicación, Procesos Industriales y Biotecnológicos dentro del apartado 3 en lo que se refiere a un programa para el desarrollo de aplicaciones de software para procesos de gestión y administración pública y privada. Además, la Escuela de Ingeniería en Sistemas Informáticos cuenta también con líneas de investigación; en el eje trasversal destaca el Desarrollo de Aplicaciones Informáticas, e inmerso en esta se encuentra el desarrollo de aplicaciones para dispositivos móviles y desarrollo de aplicaciones web.

En conclusión, la solución tecnológica que se va a implementar está sujeta a las líneas de investigación de la EIS (Escuela de Ingeniería en Sistemas). El proyecto se encuentra inmerso en el Proceso de Desarrollo de Software en el ámbito del Análisis y diseño de Proyectos de Software. En el ámbito nacional según el Plan Nacional del Buen Vivir la población debe tener acceso a TICS en específico también en la Generación de sistemas nacionales de patentes. La Línea de Investigación de ACM-IEEE justifica el proyecto en el apartado de Sistemas de Información exactamente en las Aplicaciones de Sistemas de Información.

1.3.2. Justificación aplicativa

La constante evolución de la tecnología abre las puertas para el uso de aplicaciones web y móviles, las cuales se encuentran en auge y, un gran porcentaje de la población de estudio a la cual está enfocada se encuentra familiarizada; por lo que utilizar sus beneficios en este proyecto, permite brindar soluciones realmente efectivas.

Otro punto a tomar en cuenta es que la mayoría de los habitantes a diario descargan aplicaciones en sus dispositivos móviles y por esta razón el propósito es que la aplicación se encuentre también disponible en una tienda virtual como Google Play Store.

Un teléfono inteligente en nuestro día a día y como una fuente de información y de localización es muy importante; partiendo desde su tamaño cómodo en relación a otros dispositivos y la transmisión de información resulta sencillamente rápido.

El manejo de una gran cantidad de información como reportes, estadísticas, informes es muy esencial y a su vez una plataforma web sigue siendo indispensable para acceder a estos datos, por ello una aplicación web de página única (SPA) es la forma más óptima para manejar un panel administrativo.

El proyecto está dirigido en su mayoría a los usuarios de la “Corporación LADRA”, quienes podrán utilizar sus dispositivos web o móviles, como medio para la adopción y rescate de animales dentro del Cantón Riobamba.

La estructura de módulos que conforman la plataforma web se encuentra detallada como:

- Módulo de autenticación
- Módulo de administración de noticias y cuidados
- Módulo de listados
- Módulo de reportes
- Módulo de notificaciones
- Módulo de estadísticas

La estructura de módulos que conforman la plataforma móvil se encuentra detallada como:

- Módulo de autenticación
- Módulo de noticias y cuidados
- Módulo de publicaciones
- Módulo de gestión geolocalización
- Módulo de notificaciones
- Módulo de imágenes
- Modulo para el consumo de APIs de terceras personas

1.4. Objetivos

1.4.1. Objetivo general

Desarrollar una aplicación multiplataforma dedicada a la adopción y rescate de mascotas para la “Corporación LADRA” empleando los framework Angular-Ionic & CodeIgniter.

1.4.2. Objetivos específicos

- Analizar las herramientas a utilizar para el desarrollo de la multiplataforma de manera que se pueda reducir el tiempo de desarrollo de las aplicaciones.
- Desarrollar una aplicación web y móvil para la adopción y rescate de las mascotas de la Corporación LADRA mediante el uso de la metodología SCRUM y de los framework Angular-Ionic & CodeIgniter.
- Aplicar un plan de pruebas empleando el software TestLink para el eficiente desarrollo de la plataforma web y móvil.
- Utilizar el modelo de Calidad del Producto Software (ISO/IEC 25010) para evaluar la usabilidad de la plataforma web y móvil.

CAPÍTULO II

2. MARCO TEÓRICO REFERENCIAL

2.1. Aplicaciones Informáticas

2.1.1. *Aplicaciones Web*

De acuerdo con (Luján, 2002) puntualiza que una aplicación web está basado en tres jerarquías: consideradas como el cliente, el servidor, y el almacenamiento de los datos. Es por ello, que para la comunicación entre componentes se utiliza el protocolo HTTP. Además, los entornos web donde las aplicaciones web se ejecutan son: Internet, Intranet y Extranet.

(Hadley, 2006) puntualiza que las aplicaciones web se enfocan en el protocolo HTTP para la interacción con los diferentes componentes: como lo es el navegador y el servidor.

2.1.1.1. *Ventajas*

- Administración del código fuente es realizado por lado del servidor.
- Costes reducidos.
- El internet es la piedra angular de toda una aplicación web.
- Facilidad en el aprendizaje y uso.

2.1.1.2. *Desventajas*

- Limitaciones en el lenguaje HTML
- Las funcionalidades offline no están inmersas (Luján,2002,p:84-85).

2.1.2. *Aplicaciones Web Progresivas (PWA)*

Frances Berriman y Alex Russell establecieron que una PWA es similar a una aplicación web, pero utiliza web modernas y herramientas del dispositivo, es por ello por lo que se denomina: una aplicación para todas las plataformas. Además, una PWA tiene aspectos importantes que resaltan su uso, como: tamaño de instalación, tiempo de inicio y tiempo de renderizado de la barra de herramientas (Khan et al., 2019).

2.1.2.1. Características

En la Tabla 1-2, se describen las características de las aplicaciones web progresivas (PWA), resaltando las más importantes.

Tabla 1-2: Características de PWA

Características	Descripción
Progresivo	Basadas en la mejora constante, su funcionamiento no tiene limitaciones sin importar el dispositivo.
Responsive	Adaptación a cualquier tamaño de pantalla.
Usabilidad	La sensación de uso es igual a una aplicación nativa.
Independiente de la conectividad	Los Service Workers permiten trabajar sin conexión a internet.
Instalación	Instalación inmediata, sin la necesidad de ir a la tienda de aplicaciones.
Reconocible	Denominadas como aplicaciones, la búsqueda es rápida mediante: el manifiesto W3C y los Service Workers.
Adaptabilidad	Iconos, mensajes, notificaciones Push se incorporan a cualquier dispositivo.
Seguridad	Basadas en el blindaje del protocolo HTTPS, sin autenticación no pueden ser manipuladas.
Actualización	Las PWA siempre están actualizadas, debido a los Service Workers.

Fuente: (Tandel et al., 2018)

Realizado por: Usca Danny, 2021

Componentes de una PWA: El desarrollo de una PWA requiere menos costos y esfuerzo, es por ello por lo que está integrado de una serie de componentes como se observa en la Figura 1-2.

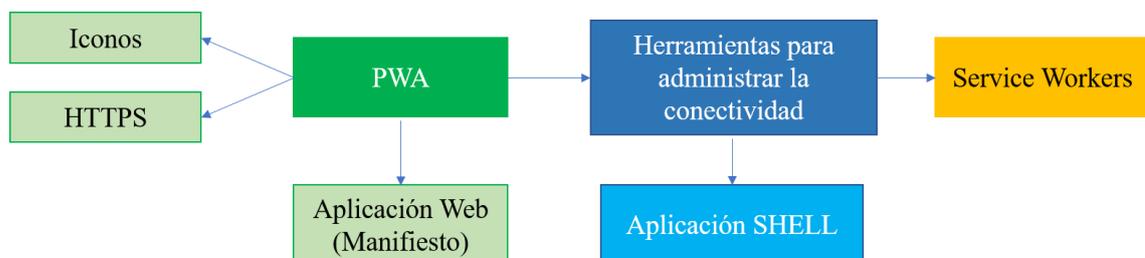


Figura 1-2: Componentes de una PWA

Fuente: (Khan et al., 2019)

- **Service Workers:** Herramienta que permite el acceso sin conexión, notificaciones push y el almacenamiento de la cache.
- **Aplicación SHELL:** Permite la separación lógica de los componentes de la aplicación web progresiva como la interfaz, contenido.
- **Manifiesto de la aplicación web:** Archivo de texto JSON el cual contiene la información de la aplicación antes de ser iniciada, además de las tecnologías incorporadas (Tandel et al., 2018).

2.1.3. Aplicaciones de Pagina Única (SPA)

(Puskaric et al., 2019) apunta que SPA se denomina al español como Aplicación de una Sola Página, se establece dentro de las aplicaciones web por lo que carga todos los recursos necesarios y no vuelve a actualizar la página mientras este en uso. El trabajo de todo este proceso se lo atribuye al navegador, por lo que utiliza el internet para su conexión y el servidor para almacenar información.

(Stepniak y Nowak, 2017) puntualiza que una aplicación de página única (SPA) se carga dentro del navegador (aplicación del cliente), la aplicación web maneja una sola página HTML como base para interactuar con las demás paginas para ello utiliza JavaScript, HTML y CSS. El funcionamiento de todo el proceso de la interacción con el usuario se lo realiza en el dispositivo y por ello no es necesario volver a cargar la página.

2.1.3.1. Arquitectura

La comunicación entre las dos aplicaciones cliente y servidor se basa en la forma asincrónica, trabajando en módulos separando así el código por lo cual permite gestionar de manera dinámica. De la misma manera, los requerimientos realizados por el cliente son devueltos en formato tipo JSON (Puskaric et al., 2019). En la figura 2-2 se muestra la arquitectura SPA:



Figura 2-2: Arquitectura SPA

Fuente: (Puskaric et al., 2019)

2.1.3.2. Optimización de una SPA

Como (Stepniak y Nowak, 2017) pone a consideración los siguientes métodos para mejorar el proceso de carga de una aplicación de página única:

- **Concatenación de recursos:** Reducir el número de peticiones por medio de la combinación de múltiples archivos.
- **Compresión de datos:** Guardar la información mediante el archivo .gzip en los archivos HTML, JavaScript y CSS.
- **Eliminación de reglas innecesarias CSS:** Las reglas enviadas al navegador son innecesarias no es factible su envío, para detectar se puede utilizar grunt-uncss y el propio navegador Google Chrome.
- **Minimizar archivos JavaScript:** Realizar refactorización del código que se encuentra en los archivos JavaScript.
- **Protocolo HTTP/2.0:** Utilizar las cabeceras HTTP y trabajar mediante el uso de prioridades.
- **PUSH Promise:** Envío de recursos por parte del servidor a través del protocolo HTTP.

2.1.4. Aplicaciones Móviles Nativas

(Min Tun, 2014) argumenta que las aplicaciones móviles nativas ofrecen una experiencia de usuario de mayor calidad, debido a que son instaladas en el sistema operativo del dispositivo y no poseen herramientas intermediarias para su funcionamiento. Además, poseen el acceso a las funciones integradas del dispositivo como: cámara, ubicación, SMS, etc. Su desarrollo requiere un conocimiento profundo, debido a que son desarrolladas para un sistema operativo móvil determinado.

(Heard y Ardila, 2019) describe que las aplicaciones móviles nativas utilizan marcos de desarrollos proporcionados por los dispositivos como son: iOS y Android mediante Xcode y Android Studio respectivamente. Por lo que la gran demanda por las aplicaciones móviles es alta. Para cada sistema operativo existen funciones específicas y la realización está evidenciada en la Tabla 2-2.

Tabla 2-2: Plataformas de desarrollo

	Android	iOS	Windows Phone
Lenguaje	Java	Objective-C	C#, VB.NET y otros
Herramientas	Android SDK	Xcode	Visual Studio
Complejidad	Media	Media	Media

	Android	iOS	Windows Phone
Formato	.apk	.app	.xap

Fuente: (Min Tun, 2014)

Realizado por: Usca Danny, 2021

(Min Tun, 2014) señala que las principales ventajas y desventajas de las aplicaciones nativas, sin hacer hincapié de las diferentes plataformas que se encuentran en el mercado, son:

2.1.4.1. Ventajas

- Compatibilidad con APIs y actualizaciones permanentes
- Utiliza determinadas características del software y hardware
- Integración con aplicaciones propias del dispositivo
- Se encuentran en la tienda de aplicaciones

2.1.4.2. Desventajas

- Tiempo, desarrollo y costos elevados
- SDK específico para cada plataforma móvil

2.1.5. Aplicaciones Móviles Híbridas

(Khandeparkar et al., 2015) argumenta que las aplicaciones móviles híbridas se caracterizan por utilizar principalmente el motor del navegador debido a que es construido en base a tecnologías web como: HTML, CSS y JavaScript por lo que mientras su desarrollo se aprecia en el navegador (escritorio). Las diferentes capacidades del dispositivo son aprovechadas por las aplicaciones híbridas a través de una capa de abstracción. En la Figura 3-2 se muestra la estructura de una aplicación móvil híbrida.

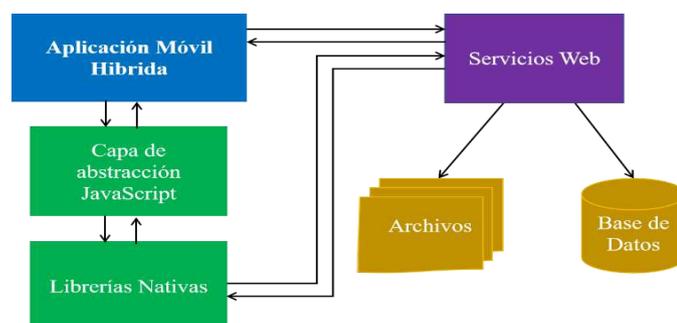


Figura 3-2: Estructura de una aplicación móvil híbrida

Fuente: (Khandeparkar et al., 2015)

(Bosnic et al., 2016) expone que una aplicación híbrida o multiplataforma su uso no está limitado del tipo de dispositivo, y es una solución factible. Para el desarrollo menciona la herramienta de Apache Cordova y utiliza CSS3 para la interfaz de usuario y JavaScript en la lógica. Por otro lado, Apache Cordova nos permite crear complementos personalizados y su arquitectura se muestra en la Figura 4-2.

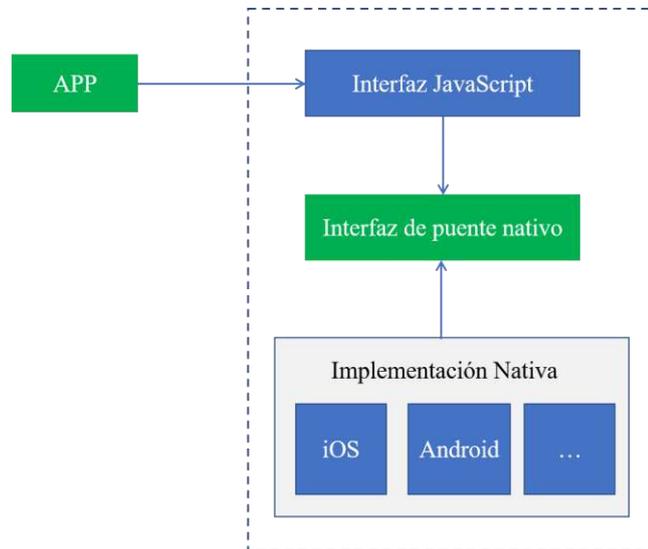


Figura 4-2: Arquitectura de un complemento

Fuente: (Bosnic et al., 2016)

(Heard y Ardila, 2019) menciona ventajas y desventajas relevantes con respecto al utilizar una aplicación móvil híbrida.

2.1.5.1. Ventajas

- Presenta una rentabilidad alta con respecto al tiempo de comercialización.
- Ahorro de tiempo y dinero en el desarrollo son bajos.
- Interfaz de usuario permite tener una mejor experiencia.
- La aplicación híbrida son livianas con respecto al tamaño de espacio.
- Integración con los componentes del dispositivo físico.
- Modo offline.
- Versionamiento mejorado para realizar mantenimientos.

2.1.5.2. Desventajas

- Enfrentar desafíos al utilizar una aplicación híbrida: introducción de nuevas funciones, reemplazo y eliminación.
- Agregación de código para interactuar entre los componentes de la aplicación y el dispositivo

- Corrección de errores mantienen una tendencia de requerir más tiempo de lo previsto
- Integración con componentes específicos de un hardware

2.1.6. Comparativa entre aplicaciones

En la Tabla 3-2 se muestra las características principales y puestas en comparación con los diferentes tipos de aplicaciones; seleccionando para la construcción de la plataforma el uso de una aplicación web progresiva (PWA) en la parte de desarrollo móvil, y una aplicación de página única (SPA) en el desarrollo web, pues en comparación con las características de las aplicaciones presentes en la tabla estas presentan rasgos sobresalientes que facilitarán un óptimo desarrollo.

Tabla 3-2: Comparativa entre aplicaciones informáticas

Características	Aplicación				
	Nativa	Híbrida	Web	PWA	SPA
Presencia en la tienda de aplicaciones	Si	Si	No	No	No
Velocidad	Muy rápida	Rápida	Rápida	Muy rápida	Muy rápida
Tamaño	Pesada	Pesada	Poco pesada	Ligera	Ligera
Disponible offline	Si	Si	No	Si	No
Interfaz de usuario	Buena	Buena	Regular	Buena	Buena
Multiplataforma	No	Si	Si	Si	Si
Posicionamiento en buscadores (SEO)	No	No	Si	Si	Si

Fuente: Basado en el trabajo de (Jobe, 2013)

Realizado por: Usca Danny, 2021

2.2. Servicios Web RESTFUL

(Valverde y Pastor, 2009) explica que un servicio REST es el acceso a múltiples recursos a través de una URL, por medio del uso de la tecnología Web 2.0 y el protocolo HTTP. Además, son documentados como archivos HTML y adquiere el uso de modelos conceptuales para describir los Servicios REST, mejorando de esta manera la abstracción de la complejidad para los desarrolladores y la introducción de un lenguaje legible para el desarrollo de las aplicaciones web.

2.2.1. REST

(Halili y Ramadani, 2018) REST denominado en español Transferencia de estado representacional, el cual determina la relación entre el cliente y servidor (arquitectura) y el almacenamiento del estado.

Es ligera en comparación a SOAP, maneja el formato de texto JSON para el intercambio de información y los recursos son identificados por medio de URI.

(Xinyang et al., 2009) indica que REST trabaja mediante restricciones que permiten la escalabilidad de los componentes, interfaces, seguridad, encapsular y la independencia. A continuación, se muestran las restricciones relevantes:

- Recursos
- Identificar recursos por medio de URL
- Interfaz concisa
- Determinar recursos por medio de representaciones
- Mensajes
- Interacción sin guardar estado
- Hipermedia

2.2.1.1. Ventajas y desventajas

La arquitectura REST tiene varios aspectos que lo resaltan como un recurso ideal para interactuar con el sistema, es por ello en las siguientes tablas están plasmadas las diferentes ventajas (Tabla 4-2) y desventajas (Tabla 5-2).

Tabla 4-2: Ventajas de la arquitectura REST

Ventajas	
Formato de mensaje JSON	Utiliza múltiples formatos
Curva de aprendizaje reducida	Uso y seguridad mediante estándares HTTP
Compatibilidad con la comunicación sin estado	Acceso y consumo por cualquier cliente
Fácil de aprender e implementación	Datos disponibles como recurso
Ancho de banda ligero	Notación de objetos JavaScript

Fuente: (Halili y Ramadani, 2018)

Realizado por: Usca Danny, 2021

Tabla 5-2: Desventajas de la arquitectura REST

Desventajas
No es adecuado para una gran cantidad de datos
No confiable

Desventajas
Solicitud Get no es conveniente en muchos datos
Latencia en tiempos de solicitud y ancho de banda
La API REST siempre dependen de los encabezados para el estado

Fuente: (Halili y Ramadani, 2018)

Realizado por: Usca Danny, 2021

2.2.2. *CodeIgniter*

CodeIgniter es una framework simple y potente diseñado especialmente para el lenguaje de programación PHP, el cual permite crear aplicaciones web usando la arquitectura Modelo-Vista-Controlador. Una de sus características principales es su amplia gama de librerías, debido a ello lo hace ideal para configurar un servidor web RESTFUL en cuestión de minutos a diferencia de otros frameworks basados en PHP. (CodeIgniter, 2006).

2.2.3. *Symfony*

Se trata de un framework basado en PHP, el cual hace uso de la arquitectura Modelo-Vista-Controlador, su funcionamiento modular permite una mejor mantenibilidad del código, proporciona un motor de plantillas llamado Twig, permite el rápido desarrollo de interfaces. (Eguiluz, 2013).

2.2.4. *Laravel*

Según (Antón, 2016), Laravel es un marco de trabajo relativamente nuevo en el desarrollo web con el lenguaje de programación PHP. No obstante, el framework permite implementar soluciones rápidas, eficientes y seguras, al estar basado en la arquitectura Modelo-Vista-Controlador resulta muy familiar para la comunidad desarrolladora.

2.2.5. *Comparativa entre frameworks*

Contar con una correcta comunicación entre el FrontEnd y Backend es crucial al momento de desarrollar un aplicativo web o móvil, más aún es pieza fundamental seleccionar un marco de trabajo, que se acople a las necesidades de desarrollo, en la Tabla 6-2, se muestra las características principales de frameworks basados PHP, y más utilizados por los desarrolladores en la creación de servicios web RESTFUL. Como parte de la comparativa se resalta el framework

CodeIgniter para ser usado dentro del proyecto por contar con características positivas que facilitarán una creación de servicios web más potente.

Tabla 6-2: Comparativa entre frameworks basados en PHP

Características	Framework		
	Laravel	Symfony	CodeIgniter
Documentación	Amplia	Amplia	Amplia
Curva de aprendizaje	Alta	Media	Baja
Flexibilidad	No	Si	Si
Escalabilidad	Si	Si	Si
Seguridad Integrada	Si	No	Si
Arquitectura MVC	Si	Si	Si
Ideal para pequeños proyectos	No	No	Si
Tamaño	Pesado	Pesado	Ligero

Fuente: Basado en el trabajo de (Laaziri et al., 2019)

Realizado por: Usca Danny, 2021

2.3. Framework para desarrollo de Aplicaciones Multiplataforma

(Ciman y Gaggi, 2017) describe el framework como un marco de desarrollo para aplicaciones web y móviles. Por lo que se divide en clases como: web, híbrido, interpretado y compilado cruzado (Tabla 7-2). Cada enfoque, se basa un lenguaje de programación para su funcionamiento, seguridad y rendimiento óptimo, con el fin de ofrecer una experiencia de usuario vívida.

Tabla 7-2: Diferentes frameworks para enfoques multiplataforma

Enfoque	Lenguaje de programación	Plataformas soportadas	Características	Framework
Web	HTML, CSS, JavaScript, Dart, C#	Android, iOS, Navegadores	Actualizaciones constantes No necesita instalación Misma interfaz de usuario	Angular- Ionic, React Native, Flutter, Xamarin
Híbrido	HTML, CSS, JavaScript	Android, iOS, Windows	Se ejecuta dentro de un WebView	PhoneGap, jQuery mobile, Framework 7

Enfoque	Lenguaje de programación	Plataformas soportadas	Características	Framework
Interpretado	JavaScript	Android, iOS	Experiencia de usuario igual al nativo	Titanium
Compilación cruzada	C#, C++, JavaScript	Android, iOS, Symbian	Se ejecuta bajo el propio sistema operativo	Mono, MoSync

Fuente: Basado en el trabajo de (Ciman y Gaggi, 2017)

Realizado por: Usca Danny, 2021

2.3.1. *Angular*

(Poulter et al., 2015) puntualiza que angular es un marco de desarrollo de código abierto, especialmente para construir aplicaciones web de una sola página (SPA) mediante el modelo MVC. Con el uso de Bootstrap, permite el diseño de contenido web potente y funcional, es provisto por Google. (Elrom, 2016) añade que Angular está basada para crear aplicaciones en el Front-End para brindar una experiencia de usuario sean excelentes, del mismo modo trabaja con bibliotecas como JavaScript. Posee también la división de código para efectuar una reutilización y su versionamiento. (Karanjit, 2016) sostiene que el enlace de datos bidireccional, patrón MVC y creación de plantillas son las principales fortalezas.

2.3.2. *Ionic*

(Yang et al., 2018) sostiene que el framework Ionic está basado en HTML5, CSS, JavaScript para crear aplicaciones multiplataformas, móviles progresivos e híbridas que brindan una experiencia similar a una aplicación móvil nativa. Proporciona el desarrollo de interfaces y aplicaciones complejas en un corto de tiempo reducido. Cordova es un marco de desarrollo simple, rápido y código libre por lo que utiliza tecnologías web. Su compatibilidad esta para casi todos los sistemas operativos móviles denominado, así como un framework multiplataforma de Código abierto. (Justin y Jude, 2017) identifica al framework Ionic uno de los mejores para el desarrollo de aplicaciones multiplataforma, gracias a su amplia comunidad.

2.3.3. *Xamarin*

(Radi, 2016) describe a Xamarin como un framework que entrega una solución óptima para la codificación de aplicaciones móviles y de escritorio de manera ágil, mediante el uso del lenguaje de programación C#, se construye un código único, una de las características principales es su

interfaz muy asemejada al de una interfaz nativa, no obstante, su uso exclusivo del lenguaje de programación C# complica acoplar ciertas características nativas.

2.3.4. *Flutter*

(Dagne, 2019) expone el uso del framework Flutter como marco de desarrollo completo con la capacidad de crear aplicaciones para dispositivos de escritorios, web y móviles por medio del uso de un único lenguaje de programación (Dart). Su rendimiento a nivel móvil lo hace destacar por acercarse lo más parecido a una aplicación nativa debido al uso flexible de widgets de alto nivel. Pero, debido a su flexibilidad hace que el framework presente en ciertos casos incompatibilidades o limitación en el desarrollo de aplicaciones móviles para iOS o Android.

2.3.5. *React Native*

Es un framework desarrollado por la empresa Facebook, se encuentra basado en el lenguaje de programación JavaScript y permite crear aplicaciones web e híbridas. Mediante la aplicación de la tecnología conocida como construcción de bloques de interfaz de usuario, permite la creación de una aplicación casi nativa, su objetivo es simplificar el desarrollo y diseño de forma exponencial, (Blanes, 2019).

2.3.6. *Comparativa entre frameworks*

Es indispensable contar con un framework que nos facilite la codificación de aplicaciones multiplataforma, en la Tabla 8-2 se muestra características sobresalientes y puestas en comparación con los diferentes marcos de trabajo especializados en el desarrollo multiplataforma. Al analizar los datos expuestos en la Tabla 8-2, se observa que sobresale el framework Angular-Ionic, de esta manera su uso, como marco de trabajo principal para el desarrollo multiplataforma repercutirá de manera positiva en la codificación.

Tabla 8-2: Comparativa entre frameworks para desarrollo multiplataforma

Característica	Framework			
	React Native	Flutter	Xamarin	Angular -Ionic
Curva de aprendizaje	Media	Alta	Alta	Media
Documentación	Amplia	Poca	Amplia	Muy amplia
Comunidad	Media	Baja	Media	Alta

Característica	Framework			
	React Native	Flutter	Xamarin	Angular -Ionic
Reusabilidad de código	Hasta un 90%	De 50% al 90%	Hasta un 96%	Hasta un 98%
Rendimiento móvil	Bueno	Bueno	Bueno	Muy bueno
Rendimiento web	Muy bueno	Pobre	Bueno	Excelente
Precio	Código abierto	Código abierto	Código abierto	Código abierto

Fuente: Basado en el trabajo de (Isitan y Koklu, 2020)

Realizado por: Usca Danny, 2020

2.4. Herramientas de Desarrollo para Aplicaciones Multiplataforma

Para el desarrollo de las aplicaciones web y móvil está basado en el lenguaje de programación JavaScript para el Front-End. Y el Back-End está construido en PHP. A continuación, se muestran las diferentes herramientas de desarrollo.

2.4.1. Visual Studio Code

Visual Studio Code es un editor de código que posee una infinidad de recursos para escribir y desarrollar código flexible y mantenible. Bajo el desarrollo de Microsoft es una herramienta gratuita y de código abierto. Además, tiene la ayuda integrada y el versionamiento incorporado mediante Git (Velasco, 2020).

2.4.2. XAMPP Server

XAMPP es una aplicación que permite emular un servidor apache dentro de una computadora personal especialmente para el desarrollo de aplicaciones web. Desarrollado con PHP y su instalación es sencilla. Su nombre está compuesto por el acrónimo de los programas que contiene: Servidor Apache, Base de datos MySQL y MariaDB por ultimo los lenguajes de programación Perl y PHP. La singularidad de X es por los sistemas en lo que se usa: Linux, Windows y MacOS X.

- **Apache:** Servidor web.
- **MySQL/MariaDB:** Base de Datos
- **PHP:** Lenguaje de programación, para crear aplicaciones web

- **Perl:** Lenguaje de programación, para administrar el sistema, desarrollo web y red (García, 2020).

2.4.3. *Bootstrap Studio*

Bootstrap especializado en el desarrollo de Front-End, es un conjunto de herramientas para realizar maquetados de un sitio web con un diseño adaptativo y receptivo para cualquier tipo de pantalla. Posee un sistema de mallas el cual permite un diseño limpio y rápido, basado en CSS (Ruiz, 2014).

2.4.4. *Postman*

Postman permite hacer pruebas de las APIs (Interfaz de programación de aplicaciones), cuenta con herramientas y utilidades gratuitas. Por medio de esta herramienta se puede depurar y escribir pruebas automatizadas, en base a: Creación de peticiones, elaboración de test para validaciones, entornos de trabajo y permite realizar documentación para ser exportada en formato JSON (López, 2019).

2.4.5. *Android Studio*

Android Studio es un IDE (Entorno de Desarrollo Integrado) que permite crear aplicaciones para Android. De la misma manera posee otras funcionalidades: Emulador, Analizar y Ejecutar el cual se lo realiza por medio de la depuración USB (Ferreño, 2020). Además, resalta las siguientes características:

- Sistema de compilación basado en Gradle
- Versionamiento por medio de GitHub
- Múltiples marcos de trabajo
- Compatibilidad con C++ y Google (Android Studio, 2020)

2.4.6. *Xcode*

Xcode es un entorno de desarrollo integrado para diseñar y construir aplicaciones iOS. Tiene funcionalidades para el diseño de la interfaz de usuario, pruebas, depuración y subir a la App Store. Tiene un excelente soporte para la documentación y ejemplos de códigos que pueden ser implementados, su sistema de seguridad es alto (Alvarado, 2014).

2.4.7. VMware Workstation

VMware es un programa que permite crear máquinas virtuales y posee un amplio repertorio de plataformas que soporta. Se caracteriza por su seguridad, compatibilidad y su independencia cuando se instala un sistema operativo dentro de una máquina virtual sin afectar al Host (Sistema operativo principal) (Velasco, 2020)

2.5. Servidor Web APACHE

(Hu et al., 1999) expone, APACHE es un servidor Web basado completamente en UNIX, su enfoque está priorizado por la funcionalidad, eficiencia y velocidad. Además, es el más utilizado en los dominios web del internet debido a que su código fuente es personalizable.

Es un servidor web HTTP de código abierto y gratuito. Es multiplataforma, encargado de recibir peticiones de un gran flujo de clientes. De la misma manera restringir, manejar errores y gestionar la información a ser proporcionada. Es uno de los servidores web más utilizados (Ramírez, 2017).

2.5.1. Características

El servidor web Apache permite alojar n dominios, y el soporte para Virtual Host. Partiendo de estas características se agregan funcionalidades que lo hacen uno de los más populares a continuación, se listan las principales (De León, 2019):

- Modularidad extensible y adaptable
- Gestión de usuarios
- Configuración e instalación sencillas
- Lenguajes soportados: Perl, PHP y Python

2.5.2. Ventajas y desventajas

El motivo por el cual Apache es el servidor más utilizado (Díaz et al., 2002), es por lo siguiente:

- **Licencia:** Código abierto, tipo BSD
- **Comunidad:** Desarrolladores y ayuda permanente
- **Arquitectura modular:** Personalización para cada host
- **Portabilidad:** Funcionamiento sobre Unix, Linux, Windows, Mac OS y mainframes
- **Seguridad:** Robusto

Básicamente, la desventaja que posee el servidor web Apache es cuando se crea un protocolo personalizado puede comprometer la seguridad y presentar errores desconocidos si no se manipula adecuadamente las variables (Rodríguez, 2019).

2.6. MySQL

Es un sistema de gestión de base de datos relacional (RDBMS siglas en inglés) basada en código abierto y posee un modelo cliente-servidor. Básicamente, permite crear y gestionar base de datos (Robledano, 2019).

2.6.1. Características

Basada en el soporte a continuación, se listan las características de usar MySQL (Robledano, 2019):

- **Compatibilidad:** Permite trabajar con el lenguaje SQL
- **Vistas:** Realizar vistas personalizadas para las bases de datos extensas
- **Procedimientos almacenados:** Incrementar la eficiencia al no procesar las tablas directamente
- **Desencadenantes:** Automatizar tareas dentro de la Base de Datos
- **Transacciones:** Resguardo de información por medio de la ejecución de diversas operaciones

2.6.2. Comparativa entre bases de datos

Una de las funcionalidades que caracteriza a MySQL es la velocidad y confiabilidad. Las características expuestas en la Tabla 9-2 frente a otras bases de datos hacen que el sistema gestor de base de datos MySQL sea el más óptimo para el almacenamiento de la información.

Tabla 9-2: Comparación de funciones de bases de datos relacionales

Función/ Motor BD		MySQL	PostgreSQL	Oracle	MSSQL
Almacenamiento de datos	Modelos de almacenamiento	MyISAM, InnoDB, Berkeley DB, full text	Postgres	Bitmapped, B-tree, IOT, function-based	Clustered, no clustered
	Fiabilidad	Alta/Muy alta	Alta	Alta/Muy alta	Alta/Muy alta
	Escalabilidad	Grande/Muy grande	Grande	Grande/Muy grande	Grande/Muy grande

Función/ Motor BD		MySQL	PostgreSQL	Oracle	MSSQL
Índices	Columnas, clave principal y texto completo	Si	Si	Si	Si
Integridad de datos	Cumplimiento de ACID, bloqueo a nivel de fila, copia de seguridad en caliente y reversión parcial	Si	Si	Si	Si
Replicación	Patrón Simple	Si	Si	Si	Si
	Patrón múltiple	Si	Si/No (Licencia)	Si	Si
	Clustering	Si	Si	Si	Si
Métodos de interfaz	ODBC/JDBC, C/C++, y Java	Si	Si	Si	Si
Características avanzadas	Procedimientos almacenados, vistas, disparadores, secuencias y cursores	Si (a partir versión 5.x)	Si	Si	Si

Fuente: (Di Giacomo, 2005)

Realizado por: Usca Danny, 2021

2.7. Patrón de Arquitectura Modelo, Vista, Controlador (MVC)

MVC es un patrón de arquitectura utilizado para desarrollar aplicaciones web, el cual separa en distintos componentes: interfaz de usuario, modelo de negocio y la lógica de control (Castejón, 2004). Permite tener independencia de los lenguajes de programación en el mismo código (Bahit, 2011).

2.7.1. Niveles

MVC divide en tres niveles de abstracción a las aplicaciones web y móviles (Fernandez y Diaz, 2012):

- **Modelo:** Este encargado del acceso a la base de datos, por medio de las reglas del negocio por el cual se involucra a la vista para representar datos.
- **Controlador:** Es un observador, el cual recibe los eventos de entrada(vista) mediante el uso de reglas de gestión de eventos. Los eventos son emitidos por el modelo.
- **Vista:** Mostrar al usuario la información enviada por el controlador o el modelo

2.7.2. Funcionamiento

En la Figura 5-2 se muestra la secuencia de la arquitectura MVC y sus diferentes interacciones entre sus componentes.

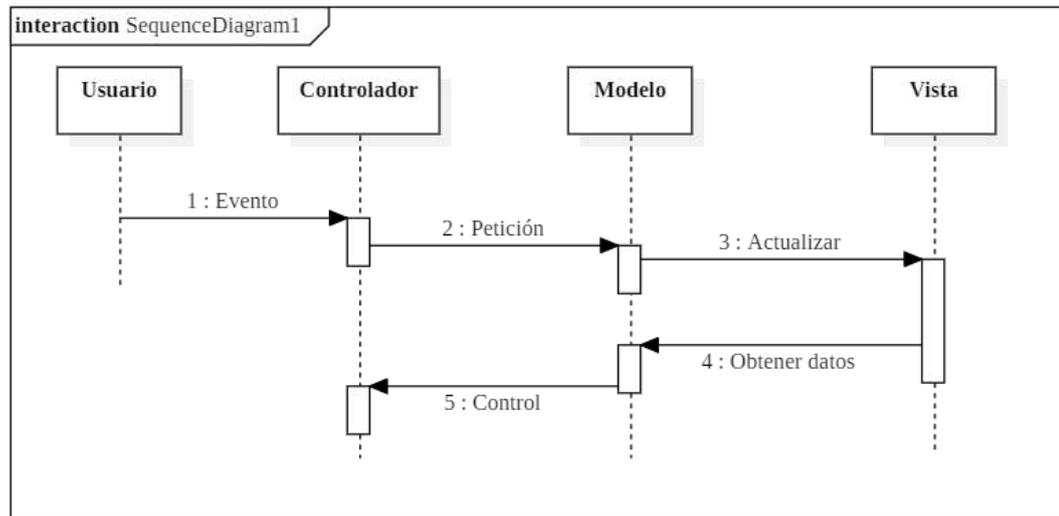


Figura 5-2: Diagrama de secuencia de MVC

Fuente: (Fernandez y Diaz, 2012)

Pasos replicados de la Figura 5-2:

1. Usuario introduce un evento.
2. Controlador recibe el evento y envía una petición al modelo (Tiene la opción de enviarlo a la vista).
3. El modelo notifica a la vista para su actualización.
4. Para la actualización la Vista solicita datos al Modelo.
5. Controlador recibe el control.

Priorizado para las aplicaciones web con un alto nivel de complejidad en su desarrollo, a continuación se listan los puntos fuertes y la deficientes de utilizar el patrón MVC (Aguilar, 2019):

2.7.3. Ventajas

- Velocidad de desarrollo por parte del equipo de trabajo, ideal en aplicaciones grandes
- Múltiples vistas para asegurar consistencia
- Facilidad en realizar pruebas unitarias
- Implementación en forma modular

2.7.4. Desventajas

- Limitaciones al usar el patrón, utilizar muchos archivos para tareas simples
- Curva de aprendizaje baja

2.8. Gestión de Proyectos Ágiles – Metodología SCRUM

(Srivastava et al., 2017) señala a SCRUM como una metodología ágil permitiendo así trabajar de forma flexible y colaborativamente en proyectos que poseen gran variedad de requisitos a través de Sprints para su solución. Realiza entregas parciales para buscar el beneficio del cliente del proyecto.

(Schwaber, 1997) describe un enfoque empírico a SCRUM debido a que los procesos de análisis, diseño, desarrollo en la fase Sprint son impredecibles (Figura 6-2). A continuación, están descritas características de la Metodología SCRUM:

- Los procesos explícitos son la Planificación y Cierre, debido que son definidos.
- Un proceso empírico es el Sprint de cada iteración, por lo que se ayuda con la Gestión de Riesgos.
- Mediante ensayo y error se maneja los Sprint.
- Los entregables pueden ser alterados en la fase de planificación y Sprint del proyecto
- A medida del desarrollo del proyecto, se puede determinar el Entregable Final.

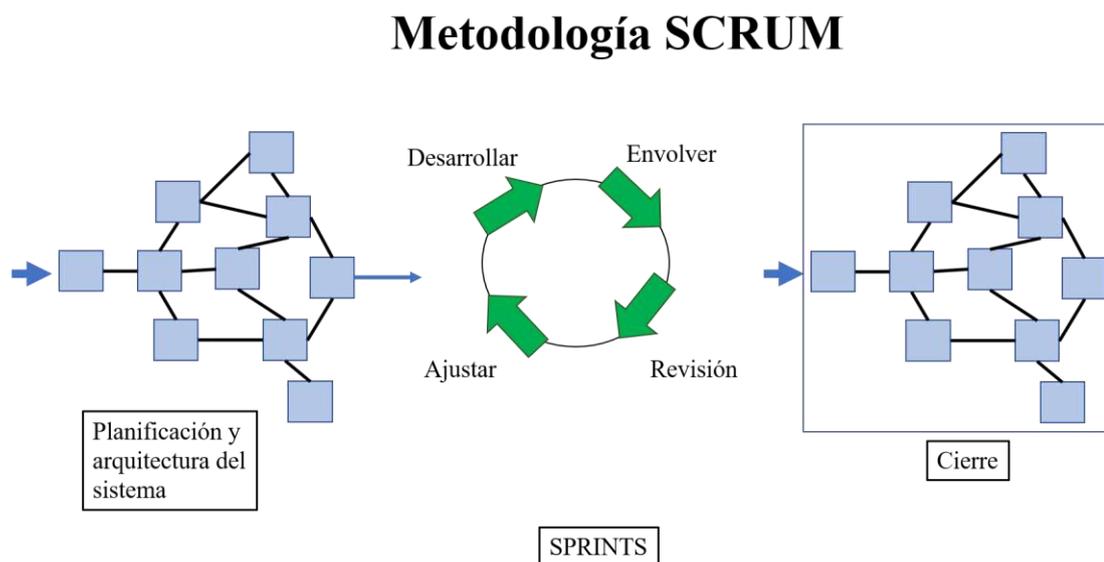


Figura 6-2: Metodología SCRUM

Fuente: (Schwaber, 1997)

2.8.1. Flujo de proceso de SCRUM

(Schwaber, 1997) expone que SCRUM tiene el siguiente flujo de proceso:

Antes del juego

- **Planificación:** Si es nuevo proyecto, se realiza análisis y conceptualización caso contrario se está mejorando un sistema está limitada a un análisis.
- **Arquitectura:** Creación o modificación de la arquitectura y el diseño de alto nivel.

En el juego

- **Sprint de desarrollo:** Enfocada en el desarrollo del proyecto en base al tiempo mediante la planificación realizada. Se realizan múltiples sprints de desarrollo que permiten dar valor mediante funciones realizadas para el sistema.

Después del juego

- **Cierre:** Preparación para el lanzamiento, documentación final, pruebas antes y después del lanzamiento.

2.8.2. Comparación con otras metodologías

La metodología SCRUM se destaca por el desarrollo del proyecto y reuniones constantes con el cliente, esto aumenta la calidad. En la Tabla 10-2 se muestra funciones que sobresalen frente a otras metodologías.

Tabla 10-2: Comparación de SCRUM con otras metodologías

SCRUM	OTROS
Estructuras de reuniones formales, roles e iteraciones bien definidos	Kanban: Sin reuniones formales, roles e iteraciones indefinidos
Prioridad en la productividad para entregables al cliente efectivos	XP: El desarrollo no tiene prioridad, por lo tanto, menos flexible
Comunicación entre los miembros del equipo, disminuye la complejidad	FDD: Procedimientos complejos y menos comunicación

SCRUM	OTROS
Mejor comunicación con los miembros del equipo	DSDM: Menos comunicación entre los miembros del equipo
Facilidad para resolver procedimientos complejos	ASD: Estructura de procesos demasiado compleja
Los requisitos del usuario permiten realizar la planificación y desarrollo con estimaciones de tiempo	Crystal: No se enfoca demasiado en los requisitos del usuario e inconvenientes al revisar el desarrollo del proyecto

Fuente: (Srivastava et al., 2017)

Realizado por: Usca Danny, 2021

2.9. TestLink

(Rodríguez, 2013) describe a TestLink como una herramienta que permite crear y gestionar casos de pruebas mediante planes de prueba basado en la web y posee licencia *Open Source*. Mediante los planes permite al grupo de trabajo ejecutar casos de test y registrarlos dinámicamente, de la misma manera generar informes y mantener el objetivo de los requerimientos. Permite la gestión de actividades o proyectos de testing con la integración de herramientas como: bugtrakers, JIRA, etc. (Terra, 2019).

2.9.1. Características

Es importante manejar una herramienta de gestión de pruebas en proyectos, en este caso TestLink posee características (Garzas, 2013) sobresalientes listadas a continuación:

- Soporte pruebas automatizadas y manuales
- Formatos en los que importa/Exporta datos
- Soporte tipo de pruebas
- Generación de informes
- Gestión de Requisitos
- Planes de prueba
- Grupos de trabajo
- Añadir archivos adicionales
- Bases de datos: MySQL, PostgreSQL, MS-SQL.
- Sistemas de seguimiento de errores: Bugzilla, Mantis, Jira, TrackPlus, Eventum, Trac, Seapine, Redmine

2.9.2. Plan de Pruebas

El plan de prueba es la base para llevar a cabo la ejecución de los casos de prueba, también se crean builds que son versiones específicas de un software. Es necesario tener construcciones o builds para la ejecución de los planes de prueba (TestLink Community, 2004).

Un plan de prueba contiene nombre, descripción, colección de casos de prueba elegidos, compilaciones, resultados de prueba, hitos, asignación de probador y definición de prioridad. La descripción debe incluir la siguiente información con respecto a los procesos de la compañía:

- Resumen / Alcance
- Características para probar y no ser probadas
- Criterios de prueba (para aprobar el producto probado)
- Entorno de prueba, Infraestructura
- Herramientas de prueba
- Riesgos
- Referencias (plan de producto o solicitud de cambio, documentos de calidad, etc.)

2.9.2.1. Caso de Prueba

Es un conjunto de entradas, precondiciones y resultados esperados permitiendo así verificar si un requerimiento cumple con lo establecido (TestLink, 2018). Los casos de prueba están conformados por:

- **Título:** Descripción corta o larga
- **Resumen:** Corto y para ser visto de forma general
- **Pasos:** Detalla el escenario de pruebas e incluye precondiciones con la información de depuración
- **Resultados esperados:** Especifica los puntos de verificación y el comportamiento esperado.
- **ID numérico:** Asignado automáticamente
- **Anexos:** Agregados para respaldar todo el proceso

2.9.3. TestLink y la metodología SCRUM

El gran potencial que presenta el software TestLink lo hace apto para ser gestionado con una metodología ágil como SCRUM. Estructurando el flujo de trabajo de forma simplificada con

SCRUM se puede observar que, dentro de un proyecto se encuentran inmersas diferentes historias de usuario o historias técnicas que contienen un escenario donde se desarrollan cada una de ellas; formando así en un lapso específico de tiempo que toma entre una a cuatro semanas un Sprint, finalmente el conjunto de sprints generan un Release o versión del producto, tal como se detalla en la Figura 7-2:

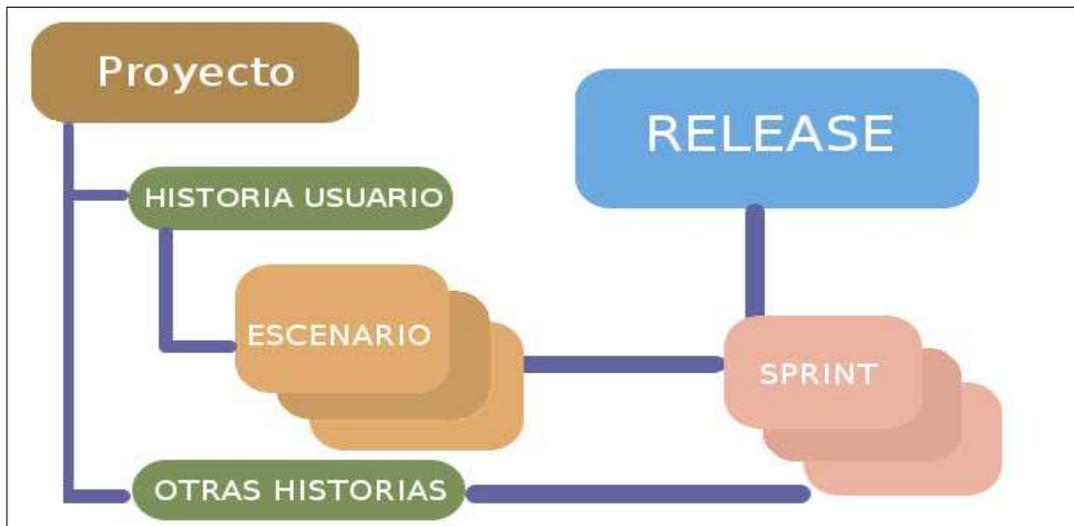


Figura 7-2: Forma de trabajo simplificada con SCRUM

Fuente: (Macías, 2013)

Para monitorear el proyecto con TestLink se conoce que un plan de pruebas (test plan) está compuesto por varias builds que internamente contienen una suite de pruebas (test suite) con sus diferentes casos de prueba (test case). Una forma de estructurar el concepto de un plan de pruebas se muestra en la Figura 8-2:

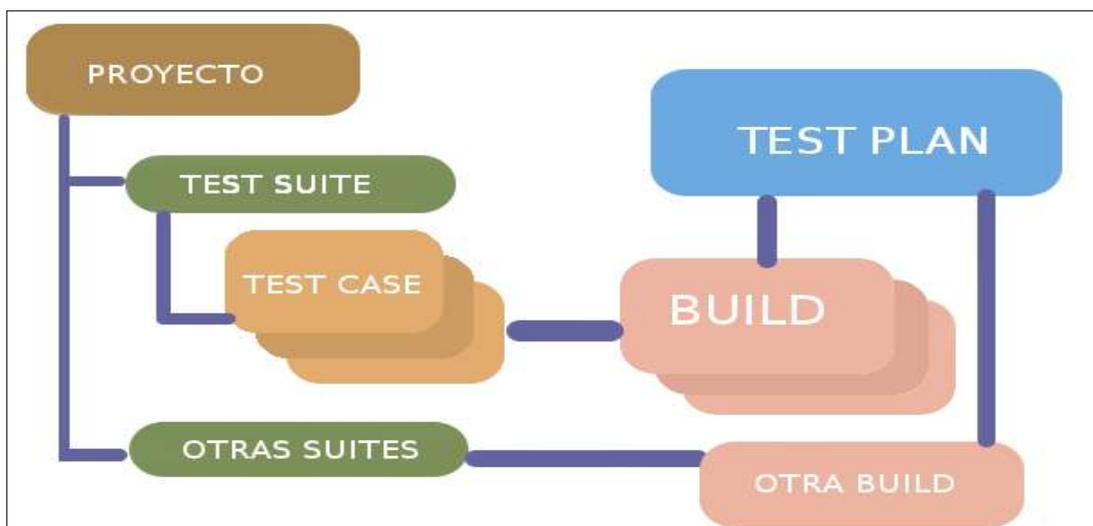


Figura 8-2: Forma de trabajo con TestLink

Fuente: (Macías, 2013)

(Macías, 2013) propone una forma de gestionar un plan de pruebas en TestLink con la metodología ágil SCRUM tal como se muestra en la Tabla 11-2, básicamente, lo que se logra es adaptar sus propiedades de forma que no pierda relevancia tanto la metodología ágil como el plan de pruebas.

Tabla 11-2: Forma de trabajo de Scrum y TestLink

En Scrum	En Test
Release	Test Plan
Sprint	Build
Historia de usuario	Test Suite
Escenario	Test Case

Fuente: (Macías, 2013)

Realizado por: Usca Danny, 2021

2.10. NORMA ISO/IEC 25010

NORMA ISO/IEC 25010 (SQuaRE - Requisitos y evaluación de la calidad del software y del sistema) Es un modelo de calidad para la evaluación de la calidad del producto software. Se maneja a través de características para evaluar las propiedades y los requisitos de los usuarios (ISO2500, 2019).

2.10.1. Características

(Sivaji et al., 2014) Este estándar tiene como característica fundamental la combinación del estándar: ISO/IEC 9126:2001 e ISO 9241-11:1997 que eran utilizados para medir la interacción del Hombre-Computador. A continuación, en la Tabla 12-2 se listan las características de este nuevo estándar:

Tabla 12-2: Modelos de calidad, características y subcaracterísticas

Calidad en uso	Calidad del producto	Calidad del producto
Eficacia	Adecuación funcional <ul style="list-style-type: none"> • Integridad funcional • Corrección funcional • Idoneidad funcional 	Fiabilidad <ul style="list-style-type: none"> • Madurez • Disponibilidad • Tolerancia a fallos • Recuperabilidad
Eficiencia	Eficiencia en el desempeño <ul style="list-style-type: none"> • Comportamiento temporal • Utilización de recursos • Capacidad 	Seguridad <ul style="list-style-type: none"> • Confidencialidad • Integridad • No repudio • Responsabilidad • Autenticidad

Calidad en uso	Calidad del producto	Calidad del producto
Satisfacción <ul style="list-style-type: none"> • Utilidad • Confianza • Placer • Comodidad 	Compatibilidad <ul style="list-style-type: none"> • Coexistencia • Interoperabilidad 	Mantenibilidad <ul style="list-style-type: none"> • Modularidad • Reutilización • Analizabilidad • Modificabilidad • Testabilidad
Libertad de riesgo <ul style="list-style-type: none"> • Mitigación del riesgo económico • Mitigación de riesgos de salud y seguridad • Mitigación de riesgos ambientales 	Usabilidad <ul style="list-style-type: none"> • Adecuación reconocible • Capacidad de aprendizaje • Operabilidad • Protección contra errores del usuario • Estética de la interfaz de usuario • Accesibilidad 	Portabilidad <ul style="list-style-type: none"> • Adaptabilidad • Instalabilidad • Reemplazabilidad
Cobertura de contexto <ul style="list-style-type: none"> • Integridad del contexto • Flexibilidad 		

Fuente: (Estdale y Georgiadou, 2018)

Realizado por: Usca Danny, 2020

2.10.2. Usabilidad

(Speicher, 2015) describe a la usabilidad como la medida en que un producto puede ser utilizado por usuarios específicos para lograr objetivos con eficacia, eficiencia y satisfacción en un contexto de uso. Al igual hace referencia a Nielsen como la capacidad de comprender: Capacidad de aprendizaje, eficiencia, memorabilidad, errores y satisfacción. A continuación, se listan las subcaracterísticas de usabilidad (ISO2500, 2019):

- **Capacidad para reconocer su adecuación:** Capacidad del producto que permite al usuario entender si el software cumple con los requisitos del usuario.
- **Capacidad de aprendizaje:** Capacidad del producto que permite al usuario aprender y entender la aplicación.
- **Capacidad para ser usado:** Capacidad del producto que permite al usuario operarlo y controlarlo con facilidad.
- **Protección contra errores de usuario:** Capacidad del sistema para proteger a los usuarios de realizar errores.
- **Estética de la interfaz de usuario:** Capacidad de la interfaz de usuario de brindar una experiencia de usuario alta.

- **Accesibilidad:** Capacidad del producto que permite que sea utilizado por usuarios con determinadas características y discapacidades.

2.11. Trabajos similares

Revisando propuestas similares en el DSpace ESPOCH al trabajo propuesto presentado, se encontraron casos que contenían determinadas características, las cuales son expuestas a continuación:

(Díaz, 2015) hace hincapié en el trazo de rutas es una de las funciones que serán implementadas en el proyecto, haciendo referencia al título: Creación de los Módulos de Localización de Personas, Emisión de Alertas y Definición de Rutas en el Sistema Skillbox, con Signalr de la Tecnología Web Asp. Net. En el siguiente proyecto: Desarrollo de un portal web y una aplicación móvil para la ubicación de los sitios turísticos del cantón Guano menciona la ubicación como enfoque para las aplicaciones web y móvil (Olmedo y Nacimba, 2017).

El uso de la herramienta GPS y trazar rutas con el uso del servicio de Google Maps está enfocada en el proyecto: Diseño e implementación de un sistema de localización, medición de velocidad y aceleración de un vehículo para determinar rutas alternas, utilizando tecnologías GPS Y GPRS (Cherez, 2017).

En las diferentes soluciones tecnológicas realizadas por estudiantes de la Escuela Superior Politécnica de Chimborazo, es por ello por lo que se evidencia la ubicación/localización de los lugares y sitios. El proyecto que se plantea es la ubicación de mascotas a ser adoptadas o que necesiten ser rescatadas, por lo que la aplicación será diferente y los beneficiados directamente serán los ciudadanos riobambeños y la corporación LADRA. Además, la comunicación constante brindará un mejor servicio. Cabe mencionar que el framework Angular, IONIC, CodeIgniter no ha sido aplicada en ninguno de los trabajos relacionados al ser consideradas tecnologías nuevas y en constante desarrollo.

CAPITULO III

3. MARCO METODOLÓGICO

En el presente capítulo se describe la metodología y el proceso usado para elaborar la plataforma web y móvil destinada a la Corporación “LADRA”, sintetizando se desea lograr el resultado más óptimo al culminar el proyecto propuesto.

En el desarrollo de este proyecto se utilizará la metodología ágil SCRUM tiene como base la idea de creación de ciclos breves para el desarrollo, que comúnmente se llaman iteraciones conocidos como Sprints, de esta manera se trabajara con el cliente directamente y los entregables pasarán por un plan de pruebas mediante el uso del software TestLink.

3.1. Actividades de la metodología

3.1.1. *Tipo de Investigación*

(Lozada, 2014) menciona: La investigación aplicada recalca en generar conocimiento por medio de la aplicación de este en los problemas relacionados con la sociedad. De la misma manera el seguimiento y el uso de estrategias para alcanzar un nivel óptimo a la solución del problema planteado.

Para llevar a cabo el desarrollo de este proyecto se utilizará la investigación aplicada, a través de esto podemos hacer uso de nuestros conocimientos adquiridos. Este proyecto permitirá el desarrollo de aplicaciones web y móvil para la Corporación “LADRA” especializada en adopción y rescate de mascotas.

3.1.2. *Métodos de investigación*

3.1.2.1. *Método Deductivo*

Con el uso de este método se obtendrá el proceso general de la adopción, rescate de una mascota. Esto permitirá el manejo de toda la información de la Corporación LADRA, la relación entre todos los actores y las acciones que realizan.

3.1.2.2. Método Inductivo

Concluida la recopilación de toda la información, mediante el análisis de las características de cada proceso que maneja la Corporación “LADRA”. Se busca implementar y automatizar la gestión del rescate y adopción de cada mascota, es por ello que se lleva a cabo el desarrollo de la plataforma web y móvil. La primera tendrá la función administrativa y el aplicativo móvil estará en uso por medio de los clientes.

3.1.2.3. Método Analítico

Por medio de este método, se enfocará el estudio en el proceso de la adopción es el principal problema para resolver, debido al tiempo en dar a conocer las mascotas que se encuentran en espera. Además, de brindar soluciones para usuarios y administradores en el rescate de mascotas.

3.1.3. Técnicas de investigación

En la recopilación de información para el presente proyecto, se basará en el uso de las siguientes técnicas de investigación:

3.1.3.1. Observación

Por medio de esta técnica, se detecta y asimila las diferentes funciones que realiza la Corporación “LADRA” a través de la notificación de una nueva mascota rescatada y el proceso que este conlleva, es necesario recalcar que la logística es primordial.

3.1.3.2. Entrevista

El cliente es fundamental en todo el proceso de la construcción de las aplicaciones web y móvil, mediante la entrevista se extrajo las diferentes funcionalidades. Enfocándose, en los problemas que existen y mediante la conclusión de este proyecto, permite brindar soluciones.

3.1.3.3. Cuestionario

La retroalimentación estará enfocada en el uso de cuestionarios para ir evidenciando el cumplimiento de los requerimientos solicitados, y permitir crear o mejorar funciones necesarias o complementarias en las diferentes aplicaciones. La Corporación “LADRA” se basa en la adopción y rescate, también busca obtener el mayor porcentaje de usabilidad en sus aplicaciones.

3.1.3.4. TestLink como plan de pruebas

Mediante el uso de la herramienta TestLink se ejecutará un plan de pruebas e inmerso en este se creará casos de pruebas para la codificación de un software de calidad. De esta manera, se busca cumplir el desarrollo de los requerimientos y tengan un valor positivo frente a la Norma IEC/ISO 25010 - Usabilidad.

3.2. Fase de Planificación

En esta fase se realiza la recopilación de toda la información referente a los procesos que serán implementados en el desarrollo de las aplicaciones, mediante la realización de reuniones con la Corporación “LADRA”. De esta manera se fijará el alcance del proyecto, requerimientos y funcionalidades de las aplicaciones.

En la Figura 1-3 está representado las diferentes fases del plan de trabajo mediante el diagrama de Gantt fijando así los tiempos de entregable para cada requerimiento establecido.

Modo de tarea	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
	SISTEMA "LADRA" - RIOBAMBA	150 días	lun 23/9/19	vie 17/4/20	
	INICIO	0 días	lun 23/9/19	lun 23/9/19	
	FASE I: INVESTIGACION	27 días	lun 23/9/19	mar 29/10/19	
	Asignación de Tareas	4 días	lun 23/9/19	jue 26/9/19	
	Revisión de bibliografía	3 días	vie 27/9/19	mar 1/10/19	4
	Realizar entrevistas	4 días	mié 2/10/19	lun 7/10/19	5
	Recolección de información	3 días	mar 8/10/19	jue 10/10/19	6
	Análisis de la información	6 días	vie 11/10/19	vie 18/10/19	7
	Selección de información	5 días	lun 21/10/19	vie 25/10/19	8
	Investigar y definir las herramientas de desarrollo	2 días	lun 28/10/19	mar 29/10/19	9
	FASE II: ANÁLISIS Y PLANIFICACIÓN	10 días	mié 30/10/19	mar 12/11/19	10
	Análisis de las herramientas de desarrollo	4 días	mié 30/10/19	lun 4/11/19	
	Planificación del desarrollo del proyecto.	4 días	mar 5/11/19	vie 8/11/19	12
	Estándar de codificación	2 días	lun 11/11/19	mar 12/11/19	13
	FASE III: DESARROLLO Y PRUEBAS	70 días	mié 13/11/19	mar 18/2/20	14
	Detalle procedimental	2 días	mié 13/11/19	jue 14/11/19	
	Desarrollo mediante plan de pruebas	12 días	vie 15/11/19	lun 2/12/19	16
	Arquitectura de software	4 días	mar 3/12/19	vie 6/12/19	17
	Diseño de interfaces	8 días	lun 9/12/19	mié 18/12/19	18
	Codificación de la aplicación	44 días	jue 19/12/19	mar 18/2/20	19
	FASE IV: DOCUMENTACION E IMPLANTACION	43 días	mié 19/2/20	vie 17/4/20	20
	Elaboración y presentación del borrador de la tesis	30 días	mié 19/2/20	mar 31/3/20	
	Corrección de cambios en el borrador	10 días	mié 1/4/20	mar 14/4/20	22
	Despliegue de la aplicación	2 días	mié 15/4/20	jue 16/4/20	23
	Capacitación a los usuarios	1 día	vie 17/4/20	vie 17/4/20	24

Figura 1-3: Diagrama de Gantt del Plan de Trabajo

Realizado por: Usca Danny, 2021

3.2.1. Personas y Roles involucrados en el proyecto

En el desarrollo del proyecto están inmersas personas que tendrán cargos para ejecutar la planificación prevista de manera concreta, los roles están descritos en la Tabla 1-3.

Tabla 1-3: Roles y Personas

Persona	Contacto	Rol
Sr. Roberto Valle	robert88valle41@gmail.com	Product Owner
Ing. Diego Ávila	diego.avila@epoch.edu.ec	Scrum Master
Danny Usca	danny.usca@epoch.edu.ec	Developer

Realizado por: Usca Danny, 2021

3.2.2. Tipo de roles de usuarios del sistema

Para las aplicaciones web y móvil de la Corporación “LADRA”, están descritos dos tipos de usuario con su respectivo rol en la Tabla 2-3 mencionados continuación:

Tabla 2-3: Tipos de roles de usuarios

Tipo	Responsable	Rol	Aplicación
Administrador	Sr. Roberto Valle robert88valle41@gmail.com	Gestión de usuarios, noticias, estadísticas y reportes.	Web
Cliente	Danny Usca danny.usca@epoch.edu.ec	Gestión publicaciones, adopción, seguimiento, notificaciones. Visualizar consejos, noticias	Móvil

Realizado por: Usca Danny, 2021

3.2.3. Product Backlog

Esta sección muestra la recolección de los requerimientos que serán llamados como historias de usuario para el proceso de adopción y rescate de mascotas, mediante reuniones mantenidas con la Corporación “LADRA”. Para dar valor de prioridad a las historias de usuario se asignan puntos de estimación, en este proyecto se hará el uso de la técnica T-Shirt para la estimación de tiempos en el desarrollo de cada historia de usuario. Las horas de trabajo están referenciadas a los puntos estimados en este caso: un punto estimado es igual a una hora de trabajo, y la jornada de trabajo establecida es de lunes a viernes en la cual se trabaja 40 horas semanales. En la Tabla 3-3 se encuentran descritas:

Tabla 3-3: Puntos estimados T-Shirt

Talla	Puntos Estimados	Horas – Trabajo
XS	5	5
S	10	10
M	12	12
L	20	20
XL	40	40

Realizado por: Usca Danny, 2021

En la Tabla 4-3 se listan las diferentes historias de usuarios “HU” y las historias técnicas “HT” del proyecto para la Corporación LADRA. El tiempo de entrega de los sprints es de dos semanas laborables, dando un total de 80 puntos estimados utilizando la metodología de desarrollo ágil SCRUM.

Tabla 4-3: Product Backlog

Id	Descripción	Prioridad	Puntos Estimados
HT-01	Definir los requerimientos del sistema	Alta	32
HT-02	Definir la arquitectura del sistema	Alta	32
HT-03	Definir el estándar de codificación	Alta	16
HT-04	Diseño de la base de datos	Alta	40
HT-05	Definir el estándar de interfaces	Alta	40
HT-06	Documentación	Alta	32
HT-07	Despliegue de la plataforma	Alta	32
HT-08	Capacitación de los usuarios	Alta	16
HU-01	Crear una cuenta de usuario en la aplicación móvil	Alta	24
HU-02	Recuperar credenciales mediante su correo electrónico	Media	24
HU-03	Iniciar sesión en la aplicación móvil	Alta	24
HU-04	Modificar información personal del usuario	Baja	8
HU-05	Visualizar ayuda de cuidados responsables	Media	16
HU-06	Visualizar las noticias publicadas por la corporación LADRA	Media	16
HU-07	Crear un anuncio de mascota extraviada, en adopción o encontrada	Alta	48
HU-08	Visualizar los anuncios de mascotas extraviadas, en adopción o encontradas	Alta	40
HU-09	Modificar los anuncios de mascotas extraviadas, en adopción o encontradas	Media	40
HU-10	Eliminar los anuncios creados de mascotas extraviadas, en adopción o encontradas	Baja	16
HU-11	Mediante el uso de GPS filtrar los anuncios de los diferentes usuarios por categorías en un mapa o listado	Alta	40

Id	Descripción	Prioridad	Puntos Estimados
HU-12	Mediante un marcador en un mapa visualizar la ubicación de los anuncios por categorías	Alta	24
HU-13	Enviar un mensaje de texto a WhatsApp o realizar una llamada telefónica desde la aplicación móvil	Alta	32
HU-14	Compartir contenido de la aplicación móvil mediante redes sociales	Alta	32
HU-15	Guardar como favorito un anuncio existente en la aplicación	Media	16
HU-16	Visualizar anuncios guardados como favoritos	Media	24
HU-17	Eliminar de favorito un anuncio existente en la aplicación	Baja	16
HU-18	Visualizar notificaciones Push de un anuncio	Alta	40
HU-19	Iniciar sesión en la aplicación web	Alta	32
HU-20	Visualizar la cantidad de anuncios creados por categorías (Extraviados, Adopciones, Encontrado)	Media	24
HU-21	Visualizar la cantidad de usuarios registrados en el sistema	Media	24
HU-22	Crear una noticia	Alta	32
HU-23	Visualizar una noticia	Alta	16
HU-24	Modificar una noticia	Media	16
HU-25	Eliminar una noticia	Baja	16
HU-26	Crear una publicación de cuidados responsables	Alta	40
HU-27	Visualizar una publicación de cuidados responsables	Media	24
HU-28	Modificar una publicación de cuidados responsables	Media	16
HU-29	Eliminar una publicación de cuidados responsables	Baja	16
HU-30	Listar los usuarios del sistema	Media	24
HU-31	Crear catálogo de raza de animales	Alta	40
HU-32	Visualizar catálogo de raza de animales	Alta	16
HU-33	Modificar catálogo de raza de animales	Media	16
HU-34	Eliminar catálogo de raza de animales	Baja	8
HU-35	Crear notificaciones Push con noticias, novedades y promociones relevantes	Alta	40
HU-36	Generar un reporte en Excel de los usuarios que usan la aplicación web y móvil	Media	40
HU-37	Generar un reporte en Excel de las publicaciones por categoría	Media	40

Realizado por: Usca Danny, 2021

3.2.4. Historia de usuario

En este proyecto técnico está proyectado el uso de la metodología ágil SCRUM, por lo que para la representación de las funcionalidades se realiza por medio de las Historias de Usuarios (HU), maneja una regla para escribirla: Como (rol), quiero (evento), para (funcionalidad). Además, tiene los criterios de aceptación de esta manera se da como fallida o éxito a una historia de usuario. De esta manera, permiten tener una perspectiva final del cliente sobre cada funcionalidad. A continuación, en la Tabla 5-3 se muestra una tarjeta de usuario.

Tabla 5-3: Historia de Usuario

01: Historia Usuario			
ID: HU-01	Nombre de la Historia: Crear una cuenta de usuario en la aplicación móvil		
Usuario: Usuario		Sprint: 3	
Prioridad en el Negocio: Alta (Alta/Media/Baja)		Puntos Estimados: 24	
		Puntos Reales: 24	
Fecha Inicio: 21/10/2019		Fecha Fin: 23/10/2019	
Descripción: Como usuario quiero crear la interfaz y el servicio web para crear un usuario de tipo cliente en la aplicación móvil.			
Pruebas de Aceptación			
ID_PA	Descripción	Estado	Responsable
HU01 PA01	Verificar que el usuario sea único	Exitosa	Danny Usca
HU01 PA02	Verificar el correcto almacenamiento del usuario en la base de datos	Exitosa	Danny Usca
Tareas de Ingeniería			
ID_TI	Descripción TI	Esfuerzo	
HU01 TI01	Desarrollar la interfaz de la pantalla de login	12	
HU01 TI02	Desarrollar el servicio web RESTful para interactuar entre la aplicación móvil y la base de datos.	12	

Realizado por: Usca Danny, 2021

En conclusión, las historias de usuario y las historias técnicas se encuentran con más detalle en el Anexo C y Anexo D respectivamente.

3.2.5. *Sprint backlog*

El Sprint Backlog detalla las tareas de cada iteración sobre el sistema, en la cual se encuentran las historias de usuario y las historias técnicas. En el proyecto se obtuvo un total de 37 historias de usuario y 8 historias técnicas divididas en los diferentes sprints, para ello está contabilizado un total de 15 sprints y cada sprint tiene un plazo de dos semanas para su entrega trabajando un total de 40 horas de trabajo. A continuación, en la Tabla 6-3 se listan los diferentes sprints con sus historias de usuario asignadas.

Tabla 6-3: Sprint backlog

SPRINT BACKLOG				
Nro.	ID	FECHAS		Módulo/HU/HT
		Inicio	Fin	
1	HT-01	lun 23/9/19	jue 26/9/19	Definir los requerimientos del sistema
1	HT-02	vie 27/9/19	mié 2/10/19	Definir la arquitectura del sistema
1	HT-03	jue 3/10/19	vie 4/10/19	Definir el estándar de codificación
2	HT-04	lun 7/10/19	vie 11/10/19	Diseño de la base de datos
2	HT-05	lun 14/10/19	vie 18/10/19	Definir el estándar de interfaces
3	HU-01	lun 21/10/19	mié 23/10/19	Crear una cuenta de usuario en la aplicación móvil
3	HU-02	jue 24/10/19	lun 28/10/19	Recuperar credenciales mediante su correo electrónico
3	HU-03	mar 29/10/19	jue 31/10/19	Iniciar sesión en la aplicación móvil
3	HU-04	vie 1/11/19	vie 1/11/19	Modificar información personal del usuario
4	HU-05	lun 4/11/19	mar 5/11/19	Visualizar ayuda de cuidados responsables
4	HU-06	mié 6/11/19	jue 7/11/19	Visualizar las noticias publicadas por la corporación LADRA
4	HU-07	vie 8/11/19	vie 15/11/19	Crear un anuncio de mascota extraviada, en adopción o encontrada
5	HU-08	lun 18/11/19	vie 22/11/19	Visualizar los anuncios de mascotas extraviadas, en adopción o encontradas
5	HU-09	lun 25/11/19	vie 29/11/19	Modificar los anuncios de mascotas extraviadas, en adopción o encontradas
6	HU-10	lun 2/12/19	mar 3/12/19	Eliminar los anuncios creados de mascotas extraviadas, en adopción o encontradas
6	HU-11	mié 4/12/19	mar 10/12/19	Mediante el uso de GPS filtrar los anuncios de los diferentes usuarios por categorías en un mapa o listado
6	HU-12	mié 11/12/19	vie 13/12/19	Mediante un marcador en un mapa visualizar la ubicación de los anuncios por categorías

SPRINT BACKLOG				
Nro.	ID	FECHAS		Módulo/HU/HT
		Inicio	Fin	
7	HU-13	lun 16/12/19	jue 19/12/19	Enviar un mensaje de texto a WhatsApp o realizar una llamada telefónica desde la aplicación móvil
7	HU-14	vie 20/12/19	mié 25/12/19	Compartir contenido de la aplicación móvil mediante redes sociales
7	HU-15	jue 26/12/19	vie 27/12/19	Guardar como favorito un anuncio existente en la aplicación
8	HU-16	lun 30/12/19	mié 1/1/20	Visualizar anuncios guardados como favoritos
8	HU-17	jue 2/1/20	vie 3/1/20	Eliminar de favorito un anuncio existente en la aplicación
8	HU-18	lun 6/1/20	vie 10/1/20	Visualizar notificaciones Push de un anuncio
9	HU-19	lun 13/1/20	jue 16/1/20	Iniciar sesión en la aplicación web
9	HU-20	vie 17/1/20	mar 21/1/20	Visualizar la cantidad de anuncios creados por categorías (Extraviados, Adopciones, Encontrado)
9	HU-21	mié 22/1/20	vie 24/1/20	Visualizar la cantidad de usuarios registrados en el sistema
10	HU-22	lun 27/1/20	jue 30/1/20	Crear una noticia
10	HU-23	vie 31/1/20	lun 3/2/20	Visualizar una noticia
10	HU-24	mar 4/2/20	mié 5/2/20	Modificar una noticia
10	HU-25	jue 6/2/20	vie 7/2/20	Eliminar una noticia
11	HU-26	lun 10/2/20	vie 14/2/20	Crear una publicación de cuidados responsables
11	HU-27	lun 17/2/20	mié 19/2/20	Visualizar una publicación de cuidados responsables
11	HU-28	jue 20/2/20	vie 21/2/20	Modificar una publicación de cuidados responsables
12	HU-29	lun 24/2/20	mar 25/2/20	Eliminar una publicación de cuidados responsables
12	HU-30	mié 26/2/20	vie 28/2/20	Listar los usuarios del sistema
12	HU-31	lun 2/3/20	vie 6/3/20	Crear catálogo de raza de animales
13	HU-32	lun 9/3/20	mar 10/3/20	Visualizar catálogo de raza de animales
13	HU-33	mié 11/3/20	jue 12/3/20	Modificar catálogo de raza de animales
13	HU-34	vie 13/3/20	vie 13/3/20	Eliminar catálogo de raza de animales
13	HU-35	lun 16/3/20	vie 20/3/20	Crear notificaciones Push con noticias, novedades y promociones relevantes
14	HU-36	lun 23/3/20	vie 27/3/20	Generar un reporte en Excel de los usuarios que usan la aplicación web y móvil
14	HU-37	lun 30/3/20	vie 3/4/20	Generar un reporte en Excel de las publicaciones por categoría

SPRINT BACKLOG				
Nro.	ID	FECHAS		Módulo/HU/HT
		Inicio	Fin	
15	HT-06	lun 6/4/20	jue 9/4/20	Documentación
15	HU-07	vie 10/4/20	mié 15/4/20	Despliegue de la plataforma
15	HU-08	jue 16/4/20	vie 17/4/20	Capacitación de los usuarios

Realizado por: Usca Danny, 2021

3.3. Fase de desarrollo

3.3.1. Arquitectura del sistema

La arquitectura del sistema propuesta está basada en la comunicación de los componentes (web y móvil), se usa el patrón MVC permitiendo de esta manera separa la interfaz de usuario, la lógica de negocio y el acceso a datos para tener ventajas en su desarrollo y estructura. Este proyecto está enfocado en el uso de framework en su back-end y front-end debido a que comparten el lenguaje de programación JavaScript y PHP. En el siguiente Grafico 1-3 se muestra el diagrama de componentes del sistema.

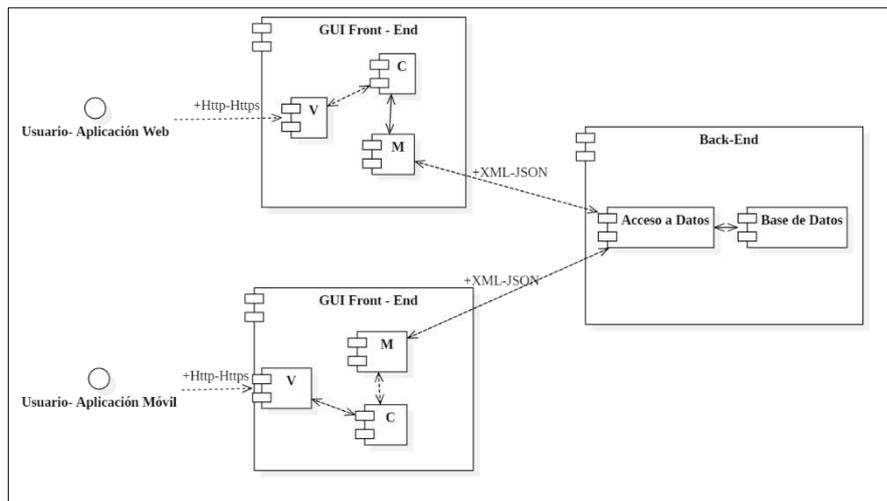


Gráfico 1-3: Arquitectura del sistema - Patrón MVC

Realizado por: Usca Danny, 2021

3.3.2. Diagramas UML

Los diagramas UML permiten entender como está construido un software o como funciona en un campo específico, es por ello por lo que en el desarrollo de este proyecto se plantea el uso de los diagramas de Casos de Uso que representan los requerimientos funcionales. El diagrama de clases posee los atributos y métodos utilizados dentro de la codificación de los diferentes módulos, el

diagrama de secuencia muestra los pasos para ejecutar una funcionalidad dentro de los aplicativos, por lo que muestra las acciones, mensajes y ayuda cuando están en ejecución. Por último, se plantea el uso del diagrama de componentes para representar el patrón arquitectónico MVC para tener una vista de alto nivel de los diferentes componentes para el funcionamiento de la aplicación web y móvil.

Diagrama de clases

El diagrama de clases permite visualizar los atributos, métodos y la relación entre los objetos. En el Grafico 2-3 se muestra un total de nueve clases, el mismo tendrá uso para la aplicación web y móvil, debido a que comparten el backend del proyecto.

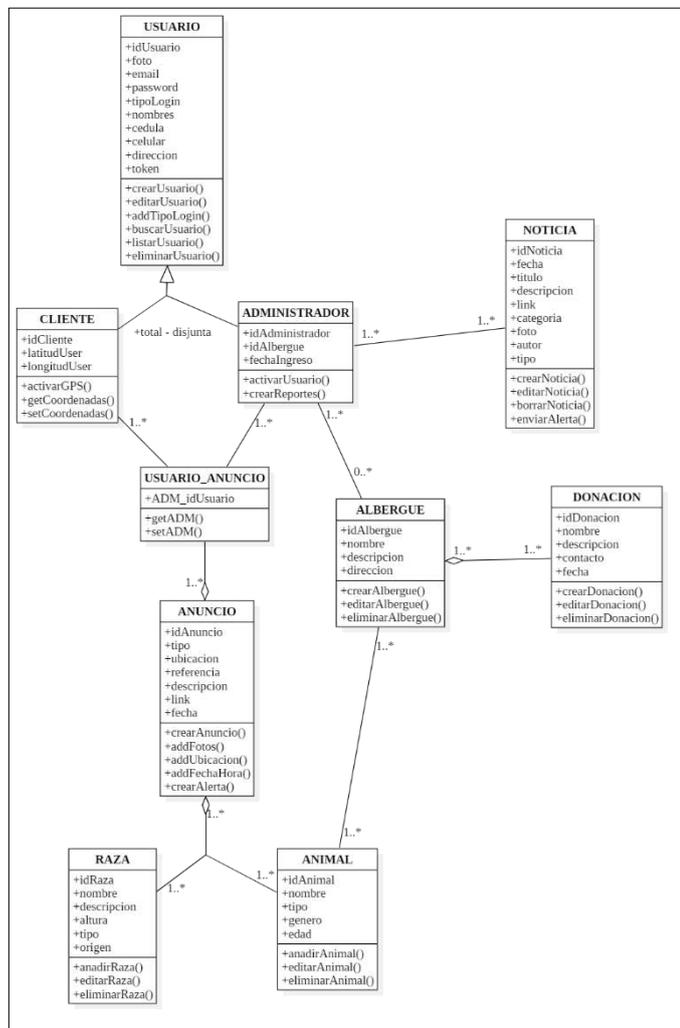


Gráfico 2-3: Diagrama de clases

Realizado por: Usca Danny, 2021

Diagrama de secuencia

El diagrama de secuencia modela la interacción de los objetos de un sistema. En el Grafico 3-3 se muestra la secuencia y la interacción entre los diferentes actores con objetos, del diagrama de caso de uso: Crear una cuenta mediante el uso de sus datos personales.

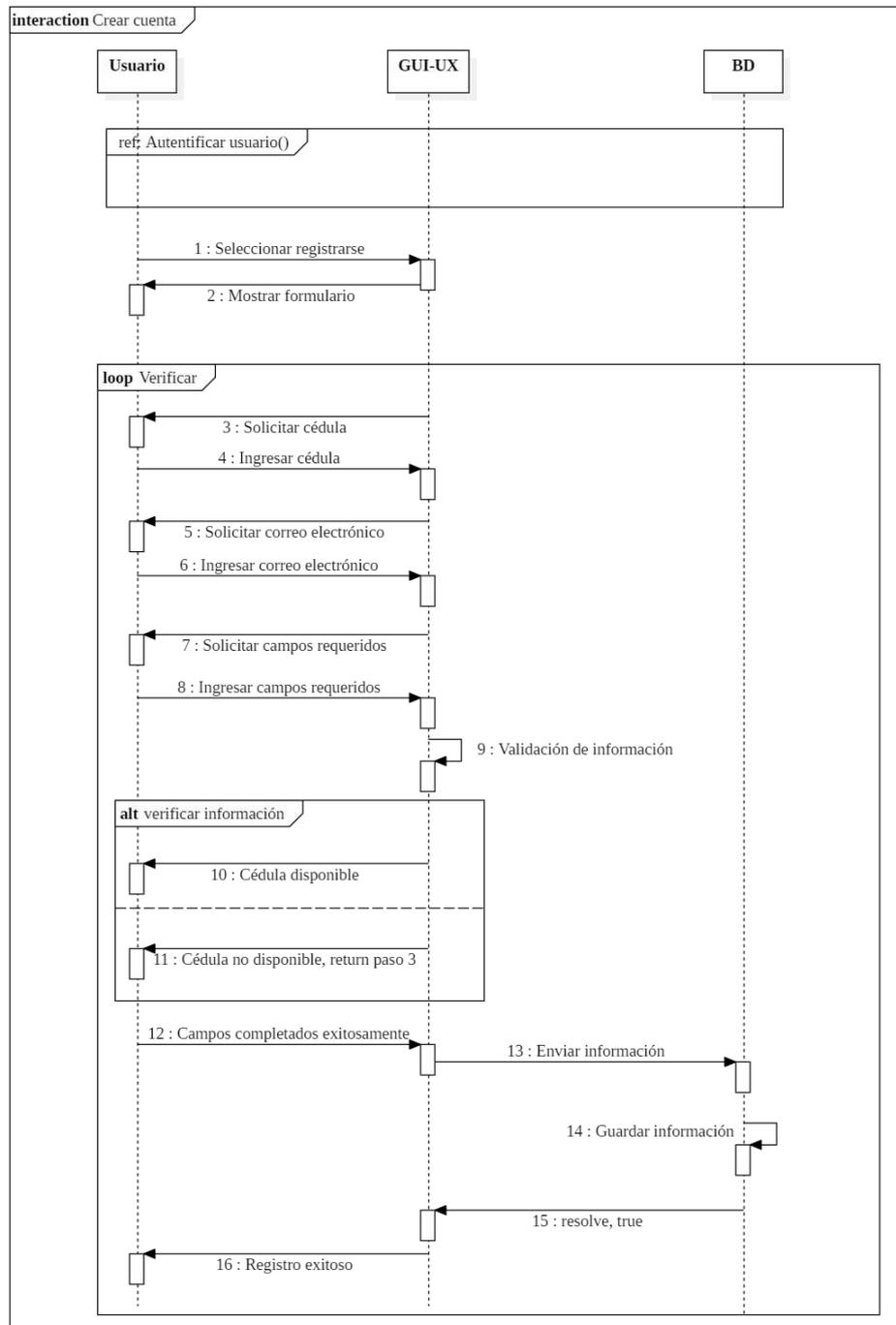


Gráfico 3-3: Diagrama de secuencia

Realizado por: Usca Danny, 2021

En el Anexo B, se encuentran los correspondientes diagramas UML del proyecto.

3.3.3. Estándar de codificación

El estándar de codificación para el desarrollo de los aplicativos es Camel Case y el tipo seleccionado es lowerCamelCase de esta manera se obtiene consistencia en la definición de métodos, atributos, variables, clases, etc. Al utilizar un estándar de codificación permite facilitar la lectura de código y su mantenimiento considerado, así como una buena práctica de programación a continuación, en la Tabla 7-3 se muestran ejemplos del uso de lowerCamelCase en el proyecto.

Tabla 7-3: Definición del estándar de codificación

Componente	Ejemplo	Descripción
Paquetes/Plugin	ionic-native/camera	Los paquetes y plugins serán escritos en minúscula, además todo será unido por el signo “_”.
Clases	Home Page SearchGlobalComponent	Se utilizará UpperCamelCase para identificar las paginas, clases y componentes.
Métodos	ngOnInit(); ionViewWillLeave();	Los métodos inician en minúscula, se utilizará lowerCamelCase.
Variables	usuarioData:any[] = []; name: string;	Identifica a la operación que será asignada en minúsculas, si la variable es un nombre complejo utilizar lowerCamelCase.

Realizado por: Usca Danny, 2021

3.3.4. Diseño de la base de datos

El diseño de la base de datos está enfocado en la persistencia de los datos y la información de los aplicativos web y móvil de la Corporación “LADRA” en este proyecto se utilizará MySQL en su versión más reciente, por lo que la base de datos esta normalizada hasta la 3FN.

3.3.4.1. Diseño físico

Con el fin de asegurar la integridad de los datos y optimizar el rendimiento de la base de datos, en el Grafico 4-3 se muestra el diseño físico, con las principales tablas y sus relaciones. Además, están incorporadas los atributos, las claves foráneas y las claves primarias.

descrita la entidad Cliente. En el Anexo A se encuentra el diccionario de datos de las tablas del modelo físico de la base de datos.

Tabla 8-3: Diccionario de datos

Campo	Tipo de dato	Obligatorio	Restricción
idCliente	int(11)	si	
latitudCliente	char(50)	si	
longitudCliente	char(50)	si	
foto	text	no	
email	text	si	
password	text	si	md5: encriptado
tipoLogin	text	si	
nombres	text	si	
cedula	text	si	cedula.lenght==10
celular	text	si	
direccion	text	no	
token	text	si	
tokenPush	text	si	

Realizado por: Usca Danny, 2021

3.3.5. Plan de pruebas

Para el eficiente desarrollo de la plataforma web y móvil se ha optado por utilizar la herramienta TestLink; en la cual se ha diseñado un plan de pruebas que será utilizado para todo el proyecto.

3.3.5.1. Instalación del software TestLink

Mediante el acceso a la página oficial de TestLink (<https://testlink.org>) resulta muy ágil comenzar con el proceso de instalación, como requerimientos previos es indispensable contar con un servidor web como apache y un gestor de base de datos MySQL y procedemos a realizar los siguientes pasos:

- Descomprimir el fichero descargado testlink.zip que contiene el módulo de instalación de software.
- Copiar la carpeta descomprimida en el servidor local de apache, para el proyecto se utilizó XAMPP.
- Accedemos mediante un navegador web al siguiente enlace para realizar la instalación de manera automática: <http://localhost/testlink/install/index.php> y se visualizará la pantalla de instalación como se muestra en la Figura 2-3, para continuar se debe seleccionar la opción “Nueva instalación”:

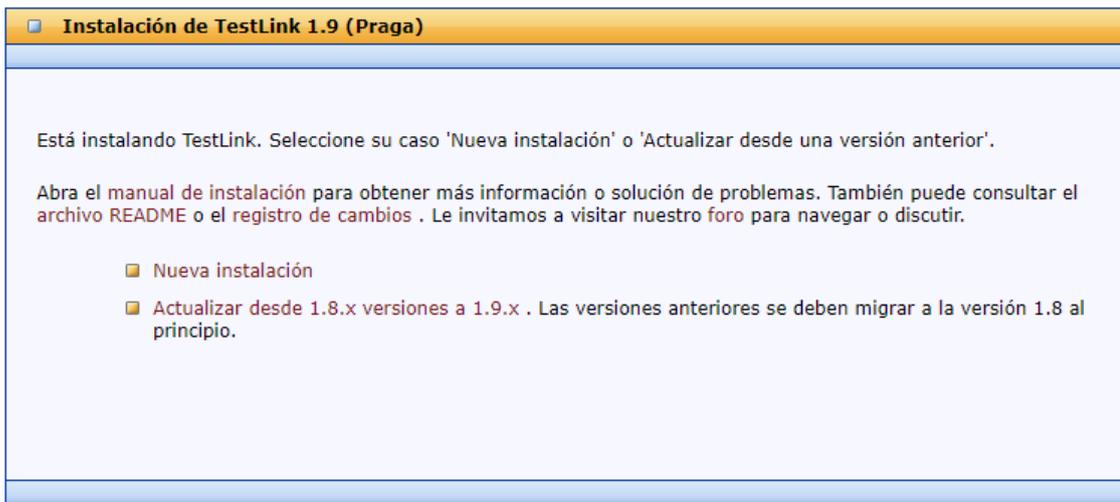


Figura 2-3: Pantalla de instalación de TestLink 1.9

Realizado por: Usca Danny, 2021

- Acto seguido en la Figura 3-3 se detallan los términos establecidos que se deben aceptar para continuar con la instalación y para poder hacer uso del software:

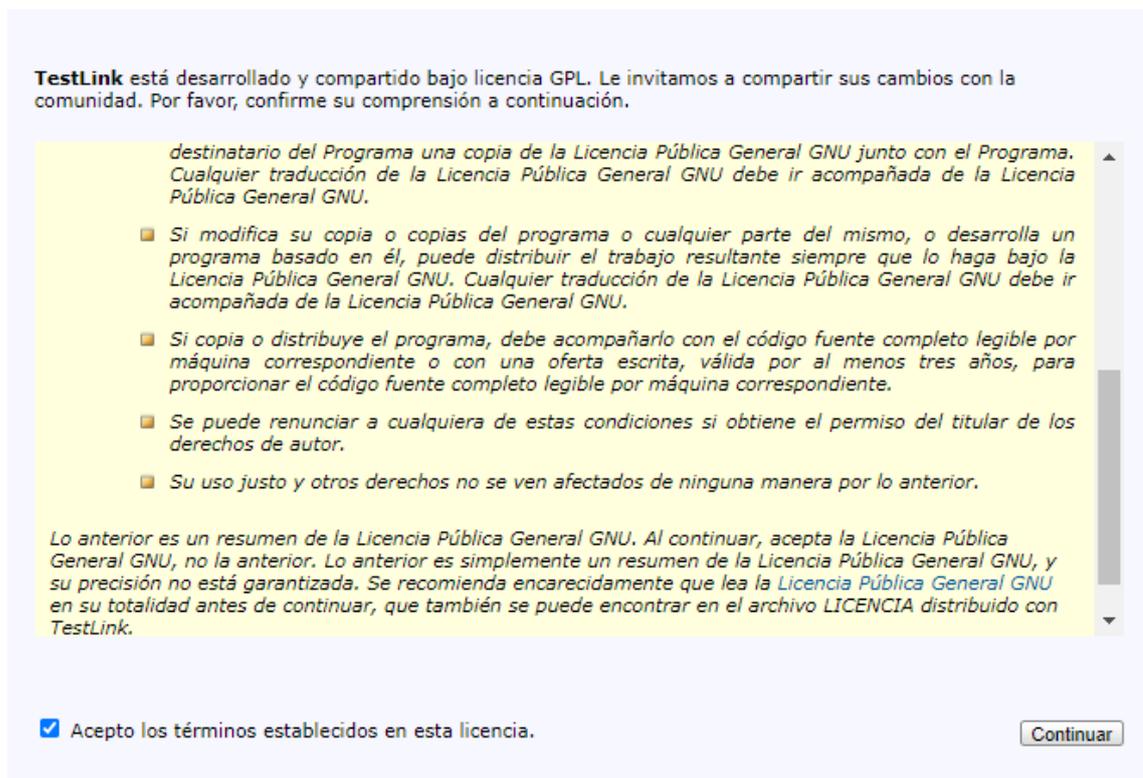


Figura 3-3: Términos y aceptación de licencia

Realizado por: Usca Danny, 2021

- En la Figura 4-3 se debe verificar los requerimientos de configuración del sistema y permisos de lectura y escritura, si el instalador no muestra ningún problema fatal se prosigue a seleccionar la opción de Continuar:

Requisitos del sistema

Sistema operativo del servidor (sin restricciones)	GANAR
Versión PHP	OK (5.2.0 [versión mínima] <= 7.3.27 [su versión])

Configuración web y PHP

Tiempo máximo de inactividad de la sesión antes del tiempo de espera	24 minutos y 0 segundos - (Corto. Considere ampliar).
Comprobando máx. tiempo de ejecución (parámetro max_execution_time)	OK (120 segundos)
Comprobación de la memoria máxima permitida (parámetro memory_limit)	Correcto (512 megabytes)
Comprobando si Register Globals está deshabilitado	OK
Comprobación de la base de datos MySQL	iHa fallado! No se puede utilizar la base de datos MySQL.
Comprobación de la base de datos de Postgres	iHa fallado! No se puede utilizar la base de datos de Postgres.
Comprobación de la biblioteca gráfica GD	OK
Comprobando la biblioteca LDAP	iHa fallado! Biblioteca LDAP no habilitada. No se puede utilizar la autenticación LDAP. (la autenticación interna predeterminada funcionará).
Comprobando la biblioteca JSON	OK

Permisos de lectura / escritura

Verificando si existe el directorio C: \xampp\htdocs\testlink\gui\templates_c	OK
Verificando si el directorio C: \xampp\htdocs\testlink\gui\templates_c se puede escribir	OK
Verificando si existe el directorio C: \xampp\htdocs\testlink\logs	OK
Comprobando si el directorio C: \xampp\htdocs\testlink\logs se puede escribir	OK
Verificando si existe el directorio C: \xampp\htdocs\testlink\upload_area	OK
Verificando si el directorio C: \xampp\htdocs\testlink\upload_area se puede escribir	OK

Su sistema está preparado para la configuración de TestLink (no se encontró ningún problema fatal). Continuar

Figura 4-3: Verificación de los requerimientos del sistema

Realizado por: Usca Danny, 2021

- Como último paso para la instalación se observa en la Figura 5-3 que se debe establecer un usuario y una contraseña para dar acceso a la base de datos y así finalizar el Proceso de configuración de TestLink:

Establezca un usuario de base de datos existente con derechos administrativos (raíz):

Inicio de sesión de administrador de base de datos **Contraseña de administrador de la base de datos**

Este usuario requiere permiso para crear bases de datos y usuarios en el servidor de bases de datos. Estos valores se utilizan solo para estos procedimientos de instalación y no se guardan.

Definir el usuario de la base de datos para el acceso a Testlink:

Inicio de sesión en TestLink DB **TestLink DB contraseña**

Este usuario tendrá permiso solo para trabajar en la base de datos de TestLink y se almacenará en la configuración de TestLink. Todas las solicitudes de TestLink a la base de datos se realizarán con este usuario.

Después de la instalación exitosa, tendrá el siguiente inicio de sesión para TestLink Administrator:
 nombre de inicio de sesión:
 contraseña de administrador: admin

Figura 5-3: Definición del acceso a la base de datos

Realizado por: Usca Danny, 2021

- Cuando el proceso de configuración haya finalizado de manera correcta, automáticamente se redireccionará a una nueva ventana de autenticación como se muestra en la Figura 6-3, si los valores se eligieron por defecto, el usuario y contraseña para la autenticación será admin respectivamente:

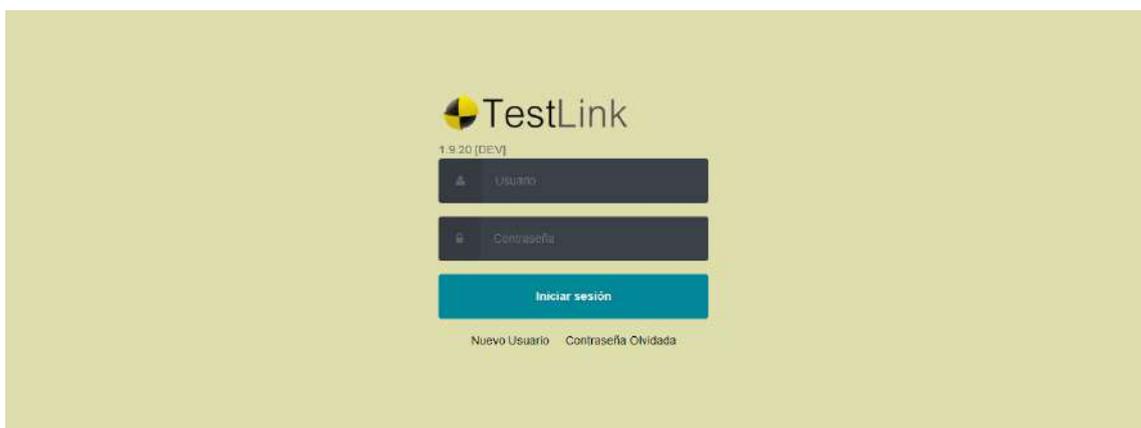
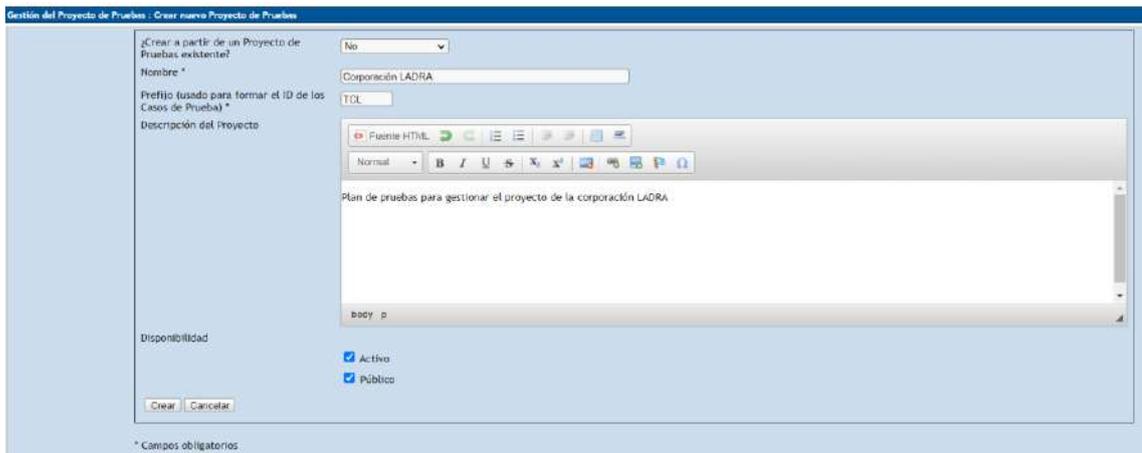


Figura 6-3: Pantalla de autenticación de TestLink

Realizado por: Usca Danny, 2021

3.3.5.2. Creación de un Plan de Pruebas

El primer paso para gestionar las diferentes suites de prueba (Test Suite) de la Corporación LADRA es la creación de un plan de pruebas (Test Plan), el cual englobará estos parámetros; la Figura 7-3 detalla la forma de crear un proyecto, para ello se define un nombre, prefijo y una pequeña descripción como campos obligatorios.



El formulario muestra los siguientes campos y opciones:

- ¿Crear a partir de un Proyecto de Pruebas existente?: No
- Nombre *: Corporación LADRA
- Prefijo (usado para formar el ID de los Casos de Prueba) *: TDL
- Descripción del Proyecto: Plan de pruebas para gestionar el proyecto de la corporación LADRA
- Disponibilidad: Activo, Público
- Botones: Crear, Cancelar

Figura 7-3: Creación de un plan de pruebas en TestLink

Realizado por: Usca Danny, 2021

En la Figura 8-3 se visualiza una tabla con el proyecto activo que se acaba de crear, bajo el cual se va a gestionar la información correspondiente a cada una de las Suites de pruebas (Historias de Usuario o Historias Técnicas) y sus diferentes casos de prueba (Pruebas de aceptación).



Nombre	Descripción	Número de Casos de Prueba	Build N°	Activo	Público
Plan de Pruebas Corporación LADRA	Plan de pruebas para la Corporación LADRA	45	1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Figura 8-3: Plan de pruebas creado para la Corporación LADRA

Realizado por: Usca Danny, 2021

3.3.5.3. Creación de una Suite de Pruebas

Una Suite de Pruebas es la representación de una historia de usuario o historia técnica; dentro del proyecto se obtuvo un total de 45 suites de pruebas. En la Figura 9-3 se detalla el proceso para su creación, para lo cual se debe seleccionar la opción “Crear Suite de Pruebas”, en los campos obligatorios se debe ingresar un nombre y detalles y adicional se puede seleccionar palabras claves, para continuar.

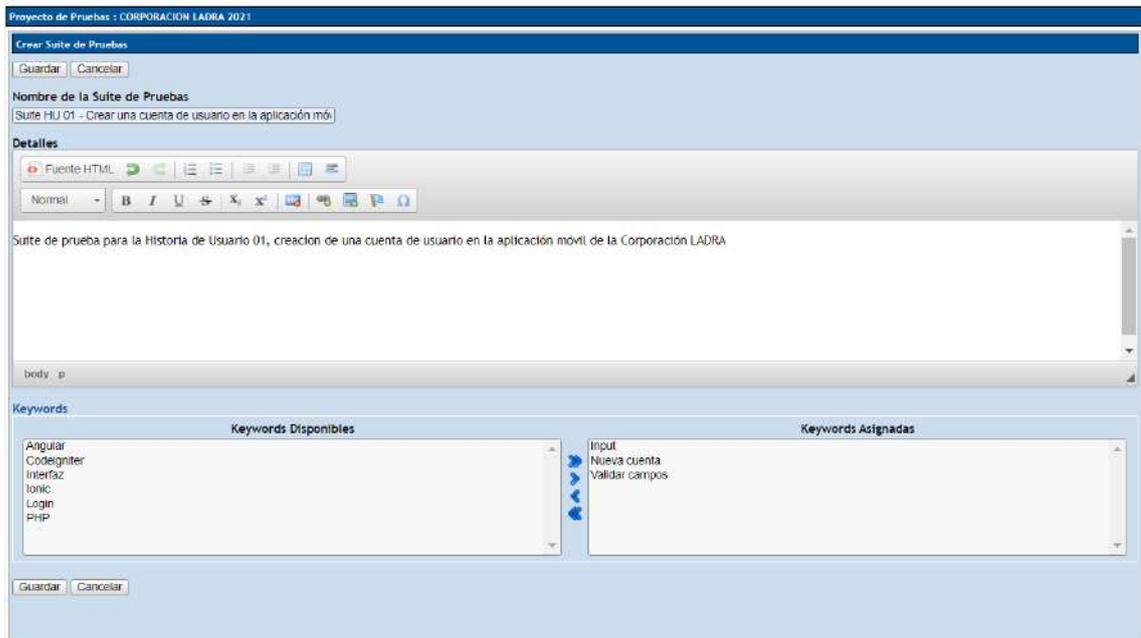


Figura 9-3: Creación de una Suite de Pruebas

Realizado por: Usca Danny, 2021

A continuación, en la Figura 10-3 y Figura 11-3 se listan las diferentes Suites de pruebas creadas para el desarrollo del proyecto.

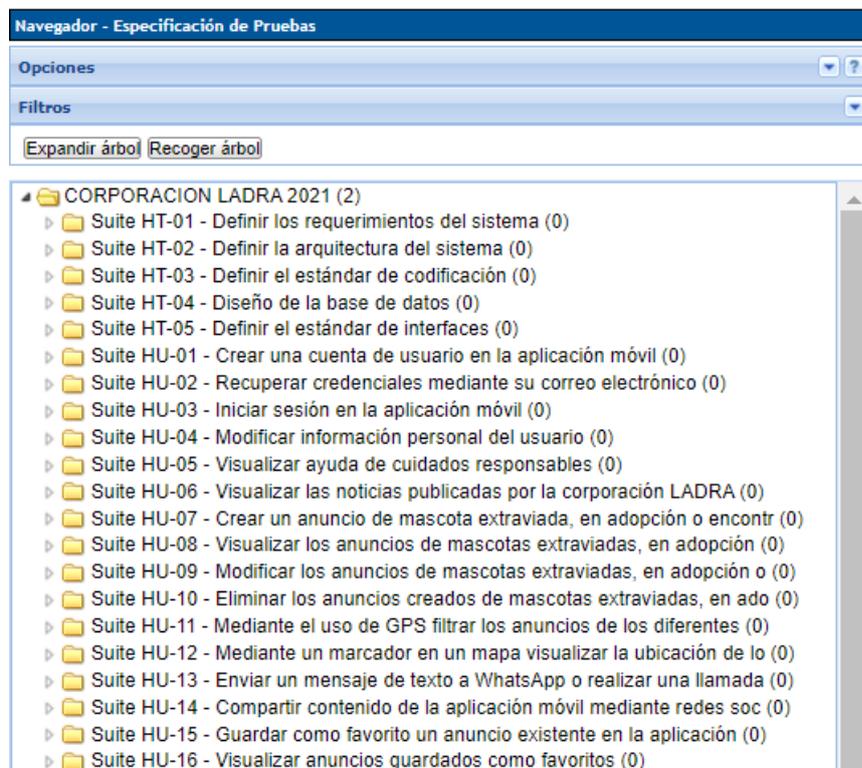


Figura 10-3: Suites de pruebas para la Corporación LADRA

Realizado por: Usca Danny, 2021

- ▶ Suite HU-17 - Eliminar de favorito un anuncio existente en la aplicación (0)
- ▶ Suite HU-18 - Visualizar notificaciones Push de un anuncio (0)
- ▶ Suite HU-19 - Iniciar sesión en la aplicación web (0)
- ▶ Suite HU-20 - Visualizar la cantidad de anuncios creados por categorías (Ex) (0)
- ▶ Suite HU-21 - Visualizar la cantidad de usuarios registrados en el sistema (0)
- ▶ Suite HU-22 - Crear una noticia (0)
- ▶ Suite HU-23 - Visualizar una noticia (0)
- ▶ Suite HU-24 - Modificar una noticia (0)
- ▶ Suite HU-25 - Eliminar una noticia (0)
- ▶ Suite HU-26 - Crear una publicación de cuidados responsables (0)
- ▶ Suite HU-27 - Visualizar una publicación de cuidados responsables (0)
- ▶ Suite HU-28 - Modificar una publicación de cuidados responsables (0)
- ▶ Suite HU-29 - Eliminar una publicación de cuidados responsables (0)
- ▶ Suite HU-30 - Listar los usuarios del sistema (0)
- ▶ Suite HU-31 - Crear catálogo de raza de animales (0)
- ▶ Suite HU-32 - Visualizar catálogo de raza de animales (0)
- ▶ Suite HU-33 - Modificar catálogo de raza de animales (0)
- ▶ Suite HU-34 - Eliminar catálogo de raza de animales (0)
- ▶ Suite HU-35 - Crear notificaciones Push con noticias, novedades y promoción (0)
- ▶ Suite HU-36 - Generar un reporte en Excel de los usuarios que usan la aplic (0)
- ▶ Suite HU-37 - Generar un reporte en Excel de las publicaciones por categoría (0)
- ▶ Suite HT-06 - Documentación (0)
- ▶ Suite HU-07 - Despliegue de la plataforma (0)
- ▶ Suite HU-08 - Capacitación de los usuarios (0)

Figura 11-3: Suites de pruebas para la Corporación LADRA - Continuación

Realizado por: Usca Danny, 2021

3.3.5.4. Creación de los Casos de Pruebas

Para ejecutar el plan de pruebas se debe asignar a cada suite sus casos de prueba para ello con la ayuda de las Pruebas de Aceptación de cada Historia Técnica o de Usuario se procede a su construcción como se detalla en la Figura 12-3, en donde se establece un título, precondiciones, pasos a ejecutar, resumen, estado, nivel de importancia y tiempo estimado de ejecución de la suite actual:

Figura 12-3: Creación de un Caso de Prueba

Realizado por: Usca Danny, 2021

En la Figura 13-3 se aprecia el resultado de la creación de un caso de prueba listo para ser ejecutado. Dentro del proyecto cabe mencionar que se han creado en total 59 casos de prueba, distribuidos en cada una de sus suites correspondientes.

Caso de Prueba

CL-1 : Verificar que el usuario sea único - Versión 1

Precondiciones

1. Tener descargado el aplicativo móvil en un smartphone
2. Tener conexión a Internet
3. No estar registrado con el mismo correo
4. No estar registrado con el mismo número telefónico
5. No estar registrado con la misma cédula de identidad

Resumen

Mediante el caso de prueba de verificar que el usuario sea único, se logrará controlar que no se guarde en la base de datos información repetida que pueda afectar al correcto funcionamiento de la plataforma

Nº	Pasos	Resultados Esperados	Ejecución
1	Tener descargado el aplicativo móvil en un smartphone	Visualizar la aplicación desde el dispositivo móvil	Manual <input type="checkbox"/> <input type="checkbox"/>
2	Ingresar a la aplicación móvil LADRA	Visualizar la interfaz principal de login	Manual <input type="checkbox"/> <input type="checkbox"/>
3	Dar clic en la opción de Regístrate aquí	Visualizar la interfaz de Registro	Manual <input type="checkbox"/> <input type="checkbox"/>
4	Ingresar los nombres completos	Visualizar los nombres en la interfaz de registro	Manual <input type="checkbox"/> <input type="checkbox"/>
5	Ingresar la cédula	Visualizar la cédula en la interfaz de registro	Manual <input type="checkbox"/> <input type="checkbox"/>
6	Ingresar el número telefónico	Visualizar el número telefónico en la interfaz de registro	Manual <input type="checkbox"/> <input type="checkbox"/>
7	Ingresar la dirección	Visualizar la dirección en la interfaz de registro	Manual <input type="checkbox"/> <input type="checkbox"/>
8	Ingresar un correo electrónico	Visualizar el correo electrónico en la interfaz de registro	Manual <input type="checkbox"/> <input type="checkbox"/>
9	Ingresar una contraseña	Visualizar la contraseña en la interfaz de registro	Manual <input type="checkbox"/> <input type="checkbox"/>
10	Dar clic en el botón de Crear una Cuenta	Visualizar un mensaje de aprobación en donde la cuenta ya este creada, y se redireccione a la pantalla de autenticación	Manual <input type="checkbox"/> <input type="checkbox"/>

Crear paso Renumerar Pasos

Estado : Borrador Importancia : Alta Tipo de ejecución : Manual Apply To All Steps

Ejec. estimada. (min) : 3.00 Guardar

Figura 13-3: Caso de Prueba para la Corporación LADRA

Realizado por: Usca Danny, 2021

3.3.5.5. Ejecución de un Caso de Prueba

En la Figura 14-3 se representa la forma como se ejecuta un caso de prueba, el éxito o fracaso del mismo depende de las precondiciones y los pasos asignados, una vez ejecutado el caso de prueba se guarda la información ingresada para generar una retroalimentación, de esta manera se asegura dar una solución rápida en caso de ser necesario para obtener un software eficiente.

Historial de Ejecuciones - Build : Build 1 - Sprint 1

No Ejecutado

Aún sin testear

Caso de Prueba CL-2 :: Versión : 1 :: Verificar el correcto almacenamiento del usuario en la base de datos

Duración estimada de la ejec. (min) : 2.00
Tipo de ejecución : Manual
Asignado a : admin

Precondiciones

1. Tener conexión entre el FrontEnd, BackEnd y la Base de Datos

Resumen
Por medio de este caso de prueba, se pretende verificar si la información enviada desde la aplicación móvil se logra almacenar de manera correcta

Nº	Pasos	Resultados Esperados	Ejecución	Notas de la ejecución	Estado de la ejecución
1	Abrir el navegador e ingresar al panel administrativo de Php My Admin	Visualizar el panel administrativo de Php My Admin	Manual	Excelente la conexión al panel administrativo es rápido	Pasado
2	Seleccionar la tabla de usuarios	Visualizar la tabla de usuarios	Manual	Se encontro la tabla rápidamente	Pasado
3	Visualizar el usuario recién creado	Que se encuentre en usuario en el listado de la tabla	Manual	Se visualizó correctamente	Pasado

Save Steps Work In Progress Execution

Figura 14-3: Ejecución de un caso de prueba para la Corporación LADRA

Realizado por: Usca Danny, 2021

Finalmente, el software TestLink genera un reporte automático al ejecutar el caso de prueba como se muestra en la Figura 15-3, esta información se utiliza para llevar a cabo un mejor seguimiento del proyecto en la fase de desarrollo.

Caso de Prueba CL-1: Verificar que el usuario sea único [Versión : 1]				
Autor: admin - 07/04/2021 09:06:52				
Resumen: Mediante el caso de prueba de verificar que el usuario sea único, se logrará controlar que no se guarde en la base de datos información repetida que pueda afectar al correcto funcionamiento de la plataforma				
Precondiciones: <ol style="list-style-type: none"> 1. Tener descargado el aplicativo móvil en un smartphone 2. Tener conexión a internet 3. No estar registrado con el mismo correo 4. No estar registrado con el mismo número telefónico 5. No estar registrado con la misma cédula de identidad 				
N°:	Pasos:	Resultados Esperados:	Notas de la ejecución:	Estado de la ejecución:
1	Tener descargado el aplicativo móvil en un smartphone	Visualizar la aplicación desde el dispositivo móvil	Todo correcto	Pasado
2	Ingresar a la aplicación móvil LADRA	Visualizar la interfaz principal de login	Todo correcto	Pasado
3	Dar clic en la opción de Regístrate aquí	Visualizar la interfaz de Registro	Todo correcto	Pasado
4	Ingresar los nombres completos	Visualizar los nombres en la interfaz de registro	Todo correcto	Pasado
5	Ingresar la cédula	Visualizar la cédula en la interfaz de registro	Todo correcto	Pasado
6	Ingresar el número telefónico	Visualizar el numero telefónico en la interfaz de registro	Todo correcto	Pasado
7	Ingresar la dirección	Visualizar la dirección en la interfaz de registro	Todo correcto	Pasado
8	Ingresar un correo electrónico	Visualizar el correo electrónico en la interfaz de registro	Todo correcto	Pasado
9	Ingresar una contraseña	Visualizar la contraseña en la interfaz de registro	Todo correcto	Pasado
10	Dar clic en el botón de Crear una Cuenta	Visualizar un mensaje de aprobación en donde la cuenta ya este creada, y se redireccione a la pantalla de autenticación	Todo correcto	Pasado
Tipo de ejecución:	Manual			
Duración estimada de la ejec. (min):	3.00			
Importancia:	Alta			
Requisitos:	Ninguno			
Keywords:	Ninguno			
Detalles de la ejecución				
Tester	admin			
Resultado de la Ejecución:	Pasado			
Modo de Ejecución:	Manual			
Duración de la ejecución (min):	3.00			

Figura 15-3: Reporte generado al ejecutar un caso de prueba

Realizado por: Usca Danny, 2021

Los diferentes reportes generados en el plan de pruebas se encuentran con más detalle en el Anexo H.

3.3.6. *Diseño de interfaces*

El diseño de las interfaces de los aplicativos ha sido realizado con el cliente conjuntamente, para llegar a este producto final. En base al logotipo se obtuvo un diseño minimalista para las diferentes pantallas, la maquetación se lo realizo con HTML basado en el sistema de mallado (Grid) y SCSS para los estilos.

3.3.6.1. *Aplicativo web*

En la Figura 16-3, esta implementada los campos (usuario y contraseña) que indican el inicio de sesión de la aplicación web. Esta aplicación está desarrollada con el framework Angular y la incorporación de servicios RESTful.

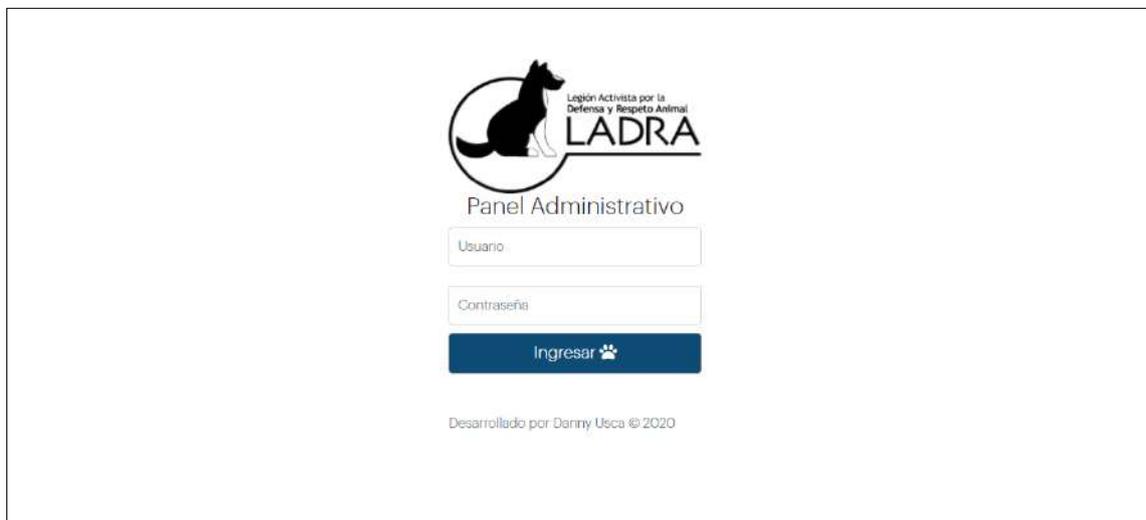


Figura 16-3: Pantalla de inicio de sesión, aplicativo web

Realizado por: Usca Danny, 2021

En la Figura 17-3 es la representación de la pantalla principal de la aplicación web, predominando las pestañas: Panel de Control, Noticias, Usuarios, Reportes, Cuidados y Ajustes. Esto permite el acceso rápido hacia las funcionalidades del administrador. El color blanco y negro predomina en toda la pantalla haciendo referencias a los colores del logotipo de la Corporación LADRA.

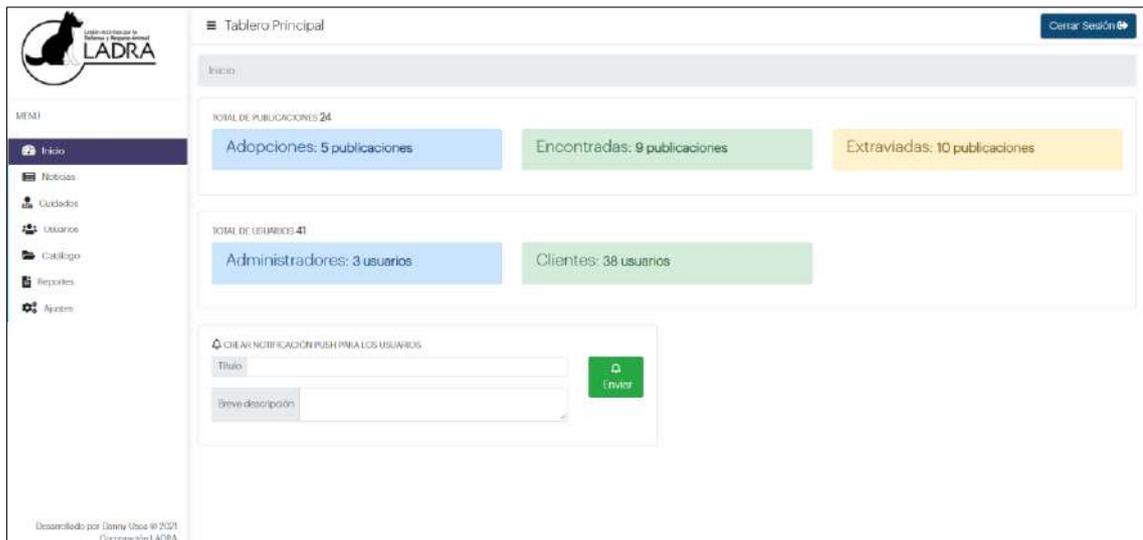


Figura 17-3: Pantalla principal, aplicativo web

Realizado por: Usca Danny, 2021

3.3.6.2. *Aplicativo móvil*

En la Figura 18-3 describe los campos para ingresar a la aplicación móvil, teniendo un apartado para recuperar contraseña a través del correo electrónico. Además, tiene un enlace: “Regístrate aquí” para crear un nuevo usuario de tipo Cliente.



Figura 18-3: Pantalla login, aplicativo móvil

Realizado por: Usca Danny, 2021

En la Figura 19-3 muestra publicaciones, ubicación en el Mapa, acceso a: noticias, anuncios, publicaciones y editar perfil. El mapa incorporado está bajo la licencia Open Source llamado Mapbox, el cual brinda servicios similares al conocido Google Maps. Los colores de cada marker indica la ubicación de una mascota según su categoría: El color rojo denomina a las mascotas extraviadas, el color verde denota a mascotas puestas en adopción y por último el color celeste son mascotas encontradas y notificadas mediante la aplicación móvil.



Figura 19-3: Pantalla principal, aplicativo móvil

Realizado por: Usca Danny, 2021

3.3.7. Reuniones y entregables

Antes y durante el proceso de desarrollo del proyecto se mantienen reunión con el Sr. Roberto Valle encargado de la parte logística de la Corporación LADRA. A fin de obtener los requerimientos y realizar la planificación de las iteraciones para dar valor al producto software final, el cliente estuvo directamente implicado en el proyecto. Al utilizar la metodología de desarrollo ágil SCRUM se denomina al cliente, el Product Owner. En la Tabla 9-3 se muestran las reuniones mantenidas.

Tabla 9-3: Reuniones y entregables

Fecha	Colaboradores	Actividades	Objetivos
16/09/2019	Sr. Roberto Valle Sr. Danny Usca	Análisis del proceso para la adopción, establecer funciones a ser incorporadas en las aplicaciones web y móvil.	Módulos del sistema
23/09/2019	Sr. Roberto Valle Sr. Danny Usca	Establecer el alcance del sistema y limitaciones.	Determinar los requerimientos funcionales
30/09/2019	Ing. Diego Ávila Sr. Roberto Valle Sr. Danny Usca	Determinar los módulos y las funciones a ser incorporadas.	Visto bueno para realizar el desarrollo del proyecto
05/11/2019	Ing. Diego Ávila Sr. Roberto Valle Sr. Danny Usca	Análisis de herramientas de desarrollo y planificación SCRUM	Obtener las historias de usuario y técnicas. Realizar el Sprint Backlog
13/11/2019	Ing. Diego Ávila Sr. Roberto Valle Sr. Danny Usca	Presentación de las interfaces web y móvil	Revisión del capítulo I y pantallas principales de las aplicaciones
19/02/2020	Ing. Diego Ávila Sr. Roberto Valle Sr. Danny Usca	Presentación demos de aplicación web y móvil.	Retroalimentación. Revisión capítulo II y III. Pruebas en dispositivos físicos
01/03/2020	Ing. Diego Ávila Sr. Danny Usca	Desarrollo de pruebas mediante TestLink	Retroalimentación. Test de Usabilidad. Tabulación de resultados obtenidos.

Realizado por: Usca Danny, 2021

3.3.8. Recursos para utilizar

Para el desarrollo de las aplicaciones de la Corporación LADRA desde la recopilación de información hasta la implementación de los servicios, se utilizó diferentes recursos, los cuales son especificados a continuación:

3.3.8.1. Recurso Software

En la Tabla 10-3 están descritos las diferentes aplicaciones informáticas para realizar el proyecto, la mayor parte de ellas poseen licencia Open Source.

Tabla 10-3: Recursos software

Aplicación	Descripción	Implementación
Visual Studio Code	Editor de código	Codificación de las aplicaciones web y móvil
Postman	Gestión de APIs	Pruebas de los Servicios Web RESTful
Microsoft Office 365	Programas de ofimática	Documentación del proyecto
Adobe Ilustrador	Editor de gráficos vectoriales	Desarrollo de interfaces graficas
Zoom	Programa de reuniones virtuales	Presentación de avances y reuniones con el personal
Google Chrome DevTools	Herramientas de creación y depuración web	Depurar aplicación móvil y web
Power Designer 16.6	Herramienta de modelado	Base de datos
XAMPP Server	Paquete de software libre	Servidor local/desarrollo
TestLink	Sistema de gestión de pruebas	Verificar la calidad del producto final (aplicaciones web y móvil)

Realizado por: Usca Danny, 2021

3.3.8.2. Recurso Hardware

En la Tabla 11-3 están listados los recursos hardware para el desarrollo de las aplicaciones web y móvil, cabe recalcar que este proyecto es educativo debido a esto se utiliza los recursos propios existentes del programador.

Tabla 11-3: Recurso hardware

Equipo	Detalle	Estado
Laptop	Procesador: Intel(R) Core™ i7-7700HQ CPU @3.60 GHz. Memoria RAM: 16.0 GB. Disco Duro: 1 TB y SSD M2: 256 GB	Disponible
Smartphone	Procesador: Snapdragon 625 @2 GHZ. Memoria RAM: 4.0 GB. ROM: 64 GB.	Disponible
Disco Duro Externo	TOSHIBA 1TB 5400RPM 2.5" BLACK. USB 3.0 - 5,0 Gbps	Disponible

Monitor	FHD LG 22", 192 x 1080. HDMI: 50 - 75Hz	Disponible
---------	--	------------

Realizado por: Usca Danny, 2021

3.4. Fase de finalización

3.4.1. Gestión del proyecto

La gestión de proyecto está reflejada en el Burn Down Chart (Gráfico 5-3), el cual demuestra un valor de total de 1200 puntos estimados divididos en un total de 15 sprints. Esta representación gráfica destaca el seguimiento del proyecto en realizar las funcionalidades a través del tiempo estimado para su desarrollo.

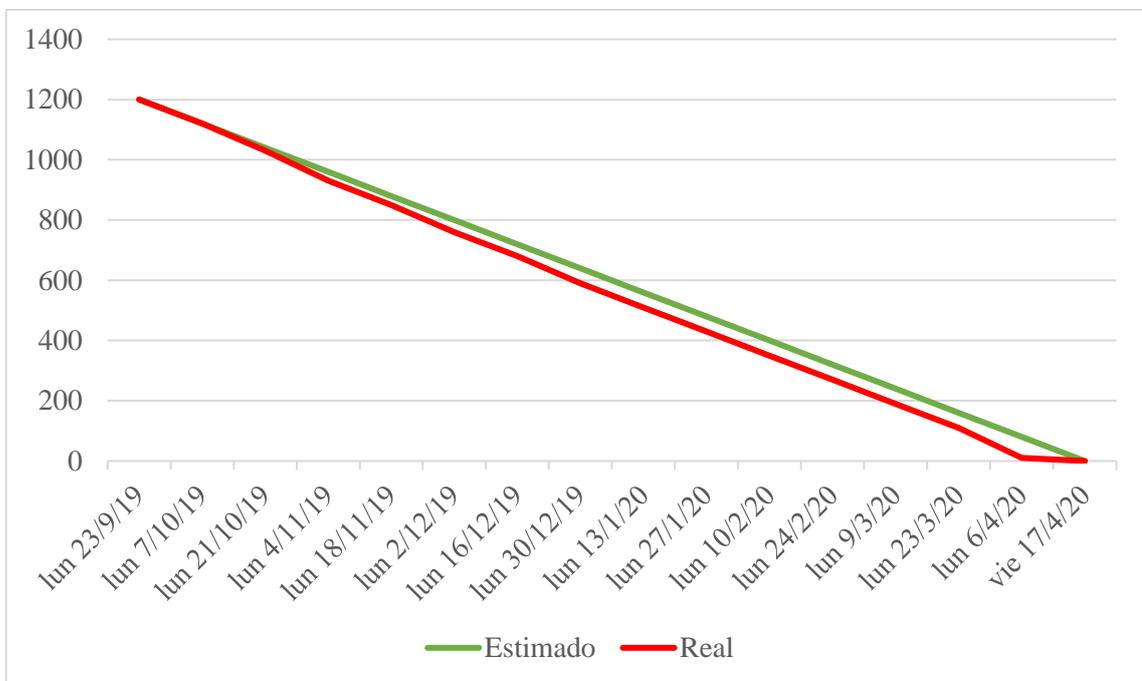


Gráfico 5-3: Burn Down Chart

Realizado por: Usca Danny, 2020

En el Grafico 5-3 está representado el tiempo real (línea color rojo) y el tiempo estimado (línea color verde), el seguimiento del proyecto se detalla a continuación: Sprint 3, requirió más tiempo debido al tener inconvenientes para subir imágenes al servidor por lo que se creó un nuevo servicio web. Sprint 4, problemas con el versionamiento del CLI de Ionic para cumplir con el requerimiento de recuperar las credenciales a través del uso del correo electrónico. Sprint 6 y Sprint 8 presentaron desventajas al realizar pruebas a través de la instalación de la apk en un dispositivo físico. Por último, en el Sprint 15 los despliegues se vieron afectados por la versión

del android-minSdkVersion (Antes nombrado: minSdkVersion) debido a las actualizaciones de los componentes de Ionic, se debe trabajar con un API superior a 28.

3.5. Método utilizado para evaluar la usabilidad de la plataforma

Puntualizando, para la evaluación de la plataforma se lo ha pensado realizar aplicando una técnica de investigación como es la encuesta; siguiendo un número determinado de preguntas que serán contestadas por una muestra de 47 miembros activos de la corporación una vez hayan interactuado con el aplicativo móvil. Aplicando el respectivo cuestionario y mediante el modelo de calidad ISO/IEC 25010 se van a evaluar tres subcaracterísticas (estética, inteligibilidad, operabilidad) que corresponden a la usabilidad. Dentro del capítulo IV correspondiente al Marco de Resultados y discusión se desglosa toda la información referente a la validación de la plataforma.

CAPITULO IV

4. MARCO DE RESULTADOS Y DISCUSIÓN

Al mencionar el estándar ISO/IEC 25010, destinado para cumplir con el último objetivo propuesto en el trabajo presente; (Blas et al., 2016, p.138) puntualiza a la calidad del producto como la unión de varias características, subcaracterísticas y atributos, las cuales en conjunto aportan significativamente al producto final.

Debido a la amplia selección de características propuestas en el modelo de calidad anteriormente mencionado, se observa favorable aplicar la característica de Usabilidad en la plataforma; para ello es indispensable hacer uso de una técnica de investigación como la encuesta debido a que facilitará exponer resultados de una manera más útil y ordenada.

4.1. Usabilidad: subcaracterísticas y propósito para evaluar

Dentro del último objetivo planteado en el capítulo I, para evaluar la plataforma “LADRA” se utilizó el estándar ISO/IEC 25010 y a la Usabilidad como característica de calidad; misma que es descrita como “la forma en la cual un software puede ser fácil de entender, rápido de aprender y resultar amigable hacia el cliente” (ISO 25000, 2019).

La usabilidad según la ISO/IEC 25010 posee seis subcaracterísticas, de las cuales para la evaluación correspondiente se ha considerado tres subcaracterísticas como adecuadas para la medición de Usabilidad del producto software que se desarrolló, la Tabla 1-4 describe el propósito que tiene cada subcaracterísticas a ser utilizada.

Tabla 1-4: Características, subcaracterísticas y propósito a evaluar

Característica	Usabilidad	
		Propósito
Subcaracterísticas	Inteligibilidad	Valora si el producto software es apropiado a las necesidades del cliente final
	Operabilidad	Valora por medio del usuario si el software logra ser operable y si se lo puede controlar sin ninguna dificultad.

	Estética	Valora por medio del diseño de interfaz de usuario (IU), si el software logra una buena interacción y resulta ser amigable con el cliente final.
--	----------	--

Fuente: (ISO 25000, 2019)

Realizado por: Usca Danny, 2021

4.2. Escalas de evaluación

Parte fundamental es la forma de evaluación por medio de la encuesta preparada que será aplicada a usuarios de la corporación “LADRA”, de tal manera según cada interrogante de la encuesta se ha optado por usar una escala cualitativa específicamente en este caso se usó la escala de Likert de 7 puntos descrita en la Tabla 2-4.

Tabla 2-4: Escala cualitativa (escala de Likert) a usar

Puntuación	1	2	3	4	5	6	7
Escala cualitativa	No estoy en absoluto de acuerdo	No estoy de acuerdo	Algo en desacuerdo	No estoy ni en acuerdo ni en desacuerdo	Un poco de acuerdo	De acuerdo	Estoy muy de acuerdo

Realizado por: Usca Danny, 2021

4.3. Tamaño de la muestra

La corporación “LADRA”, en la actualidad y debido a la emergencia sanitaria declarada en nuestro país, se encuentra laborando con un total de 15 integrantes, a esto se suman usuarios activos y rescatistas voluntarios que son alrededor de 32 involucrados, mediante esta información se determinó el tamaño de la población en 47 personas; acto seguido se le aplicó la ecuación para obtener la muestra en una población finita detallada en la siguiente formula:

$$n = \frac{N * Z_a^2 * p * q}{d^2 * (N - 1) + Z_a^2 * p * q}$$

Donde:

n = valor de la muestra a encontrar

N = tamaño de la población

Z = nivel de confianza

d = nivel de precisión

p = probabilidad de éxito

q = probabilidad de fracaso

Para la evaluación de la plataforma móvil se fijó los siguientes valores:

N = 47 personas

Z = 1,96 correspondiente a un nivel de confianza de 95%

d = 0,1 que se convierte en un 10% de error admitido

p = 0,5 de probabilidad de éxito

q = 0,5 de probabilidad de fracaso

$$n = \frac{47 * 1,96^2 * 0,5 * 0,5}{0,1^2 * (47 - 1) + 1,96^2 * 0,5 * 0,5} = \mathbf{32 \text{ personas}}$$

4.3.1. Tipo de muestreo probabilístico

Empleando la fórmula para encontrar el tamaño de la muestra (n) se obtuvo un total de 32 personas, se aplicó un muestreo probabilístico estratificado; donde la población fue dividida en grupos dependiendo de su edad. Se optó por este tipo de muestreo debido a la facilidad para recabar información y realizar una comparación entre los diferentes estratos; con más detalle podemos observarlo en la Tabla 3-4.

Tabla 3-4: Muestreo probabilístico estratificado

Edad (años)	Población	Porcentaje (Población)	Cantidad de muestra
18 – 28	17	36,17%	14
29 – 39	21	44,68%	15
40 – 50	9	19,15%	3
Total	47	100%	32

Realizado por: Usca Danny, 2020

4.4. Adecuación del Cuestionario USE de Arnold Lund para la validación

En el Anexo E y Anexo F, se describe el cuestionario USE, una herramienta validada por Arnold Lund la cual contiene 30 preguntas todas con aspectos dedicados a la usabilidad del software, además de hacer uso de la escala de Likert en cada interrogante; es, sin lugar a duda una herramienta eficaz para cumplir con el propósito de evaluar la usabilidad de la plataforma.

Para aplicar la encuesta a los usuarios de la Corporación “LADRA” ello se realizó una comparativa y se adaptó el Cuestionario USE conforme a lo que describe Norma ISO/IEC 25010 en cuanto a Usabilidad, y así se seleccionó 13 preguntas con más relevancia y que podían brindar mayores resultados, como se muestra detalladas en la Tabla 4-4:

Tabla 4-4: Adaptación del cuestionario USE para evaluar la Usabilidad

Norma ISO/IEC 25010	USE Questionnaire	Preguntas
Inteligibilidad	Capacidad para reconocer su adecuación	<ul style="list-style-type: none"> • Me ayuda a ser más productivo. • Es útil. • Satisface mis necesidades. • Me ahorra tiempo cuando lo uso.
Operabilidad	Capacidad para ser usado	<ul style="list-style-type: none"> • Es fácil de usar • Es amigable con el usuario • Requiere la menor cantidad de pasos posibles para lograr lo que quiero hacer con la plataforma • Puedo usarlo sin instrucciones escritas • Puedo usarlo con éxito cada vez
Estética	Estética de la interfaz de usuario	<ul style="list-style-type: none"> • Estoy satisfecho con ella • Se lo recomendaría a un amigo • Es agradable de usar

Realizado por: Usca Danny, 2021

En el anexo G se puede visualizar el cuestionario final adecuado que fue utilizado para obtener los resultados; el cuestionario que fue aplicado el día lunes 21 de diciembre del 2020, por motivos

de la emergencia sanitaria que se vive en la actualidad se hizo uso de la plataforma digital QuestionPro para recolectar los presentes resultados.

4.5. Análisis de los resultados obtenidos al realizar la evaluación de la plataforma

Partiendo de las métricas obtenidos al realizar la encuesta a los usuarios de la Corporación “LADRA”, se procede a realizar el análisis de resultados aplicando un análisis descriptivo y análisis inferencial.

4.5.1. Análisis descriptivo

Como se muestra en la Tabla 4-4 se tiene tres aspectos de evaluación a analizar, para facilitar la tabulación respectiva se ha realizado un análisis descriptivo por cada una de las subcaracterísticas de usabilidad, debido a que cada aparato cuenta con interrogantes diferentes.

Entre los parámetros a tener en cuenta para obtener resultados más relevantes se ha tomado en cuenta los siguientes aspectos descritos en la Tabla 5-4:

Tabla 5-4: Parámetros utilizados para obtener resultados de la tabulación

Parámetro	Descripción
N	Corresponde al tamaño de la muestra
Puntuación promedio	Corresponde a un aspecto de evaluación promedio, según la escala de Likert que va del 1 al 7, se tomó en cuenta el valor de 4, para calcular este parámetro se multiplica la cantidad de preguntas de cada sección por el valor que se propuso.
Puntuación promedio obtenido	Corresponde al promedio obtenido al aplicar el cuestionario
Desviación estándar	Corresponde a la dispersión del conjunto de los valores
Valor máximo	Valor más bajo obtenido en la encuesta
Valor mínimo	Valor más alto obtenido en la encuesta
Valor ideal	Valor perfecto que se espera al realizar la encuesta

Realizado por: Usca Danny, 2021

4.5.1.1. Inteligibilidad

La subcaracterística de inteligibilidad evalúa si la plataforma es apropiada a la necesidad que buscar el usuario.

En esta sección se tienen 4 interrogantes detalladas como:

Pregunta-1.- Me ayuda a ser más productivo

Pregunta-2.- Es útil

Pregunta-3.- Satisface mis necesidades

Pregunta-4.- Me ahorra tiempo cuando lo uso

La Tabla 6-4 corresponde a los valores obtenidos de la subcaracterística de Inteligibilidad para cada encuesta aplicada se realizó la sumatoria con el propósito de usar este valor al momento de obtener los resultados finales.

Tabla 6-4: Tabulación de las preguntas de la subcaracterística de Inteligibilidad

N° Encuesta	Inteligibilidad				Sumatoria
	Pregunta 1	Pregunta 2	Pregunta 3	Pregunta 4	
1	6	6	6	7	25
2	7	7	6	7	27
3	6	7	6	7	26
4	6	7	6	7	26
5	7	7	7	6	27
6	7	7	7	6	27
7	6	7	5	7	25
8	6	7	7	6	26
9	6	6	6	7	25
10	7	7	7	6	27
11	6	6	6	7	25
12	6	7	7	7	27
13	6	6	6	6	24
14	6	6	6	5	23
15	6	6	6	6	24
16	6	6	6	6	24
17	7	7	7	6	27
18	6	7	7	7	27
19	5	5	6	5	21
20	7	7	7	6	27
21	6	6	6	5	23
22	6	7	7	7	27
23	6	7	6	7	26
24	7	6	7	7	27
25	6	7	6	6	25

N° Encuesta	Inteligibilidad				Sumatoria
	Pregunta 1	Pregunta 2	Pregunta 3	Pregunta 4	
26	7	6	6	7	26
27	7	6	7	6	26
28	6	6	7	6	25
29	7	7	6	7	27
30	6	6	7	6	25
31	7	7	7	6	27
32	7	7	7	6	27

Realizado por: Usca Danny, 2021

Después de realizar la tabulación de los datos en el aspecto de Inteligibilidad, como resultado obtenemos los siguientes valores, en la Tabla 7-4, se aprecia que, de una muestra de 32 usuarios como puntuación promedio obtenido su valor es de 25,66 de igual forma se obtiene un valor mínimo de 21, un valor máximo de 27 y una desviación estándar de 1,52.

El último valor da a entender que el promedio individual de cada respuesta por parte de los usuarios en comparación con el valor cercano de su media no está demasiado disperso, de tal forma que están de acuerdo en su mayoría con las interrogantes que se planteó en la encuesta.

Tabla 7-4: Resultados de la subcaracterística de Inteligibilidad

Inteligibilidad						
N	Puntuación promedio	Puntuación promedio obtenida	Desviación estándar	Valor mínimo	Valor máximo	Valor ideal
32	16	25,66	1,52	21	27	28

Realizado por: Usca Danny, 2021

Al analizar los resultados se puede llegar a la conclusión de que, tener un valor ideal de 28 sería lo más adecuado, pero, a pesar de no cumplir con lo estimado, como resultado se tiene que la puntuación promedio obtenida es de 25,66 superando por 9,66 a la puntuación promedio de 16.

Como se muestra en el Gráfico 1-4 se puede decir que los usuarios de la Corporación “LADRA” que fueron encuestados están muy de acuerdo que la plataforma les resulta útil, son más productivos al momento de usarla, cumple también con sus necesidades principales y les ahorra tiempo significativo cuando realizan su trabajo.

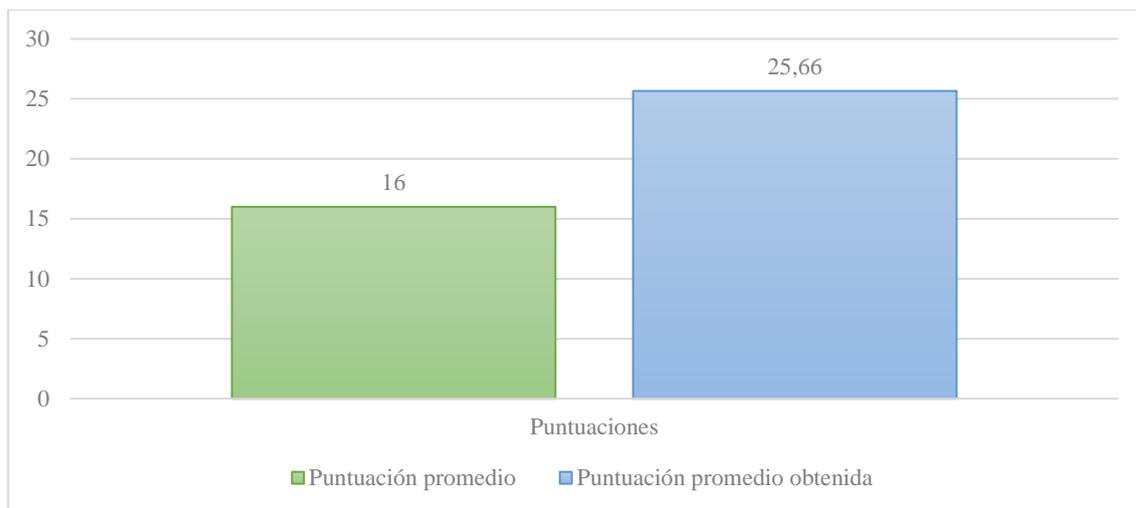


Gráfico 1-4: Comparación entre puntuaciones obtenidas de las respuestas - Inteligibilidad

Realizado por: Usca Danny, 2021

4.5.1.2. Operabilidad

La subcaracterística de operabilidad evalúa por medio del usuario si el software puede ser controlado sin dificultad alguna y a su vez si puede ser operable.

En esta sección se tienen 5 interrogantes detalladas como:

Pregunta-5.- Es fácil de usar

Pregunta-6.- Es amigable con el usuario

Pregunta-7.- Requiere la menor cantidad de pasos posibles para lograr lo que quiero hacer con la plataforma

Pregunta-8.- Puedo usarlo sin instrucciones escritas

Pregunta-9.- Puedo usarlo con éxito cada vez

La Tabla 8-4 corresponde a los valores obtenidos de la subcaracterística de Operabilidad para cada encuesta aplicada se realizó la sumatoria con el propósito de usar este valor al momento de obtener los resultados finales.

Tabla 8-4: Tabulación de las preguntas de la subcaracterística de Operabilidad

N° Encuesta	Operabilidad					Sumatoria
	Pregunta 5	Pregunta 6	Pregunta 7	Pregunta 8	Pregunta 9	
1	6	7	6	6	6	31
2	6	7	6	7	6	32

N° Encuesta	Operabilidad					Sumatoria
	Pregunta 5	Pregunta 6	Pregunta 7	Pregunta 8	Pregunta 9	
3	7	7	7	6	7	34
4	7	6	6	7	6	32
5	7	6	6	7	6	32
6	7	7	6	6	7	33
7	6	7	6	7	7	33
8	7	6	7	7	6	33
9	6	6	6	6	6	30
10	7	7	6	7	7	34
11	6	6	5	6	6	29
12	7	7	6	7	7	34
13	6	6	5	7	6	30
14	7	7	6	7	7	34
15	6	6	5	6	7	30
16	7	6	6	6	4	29
17	6	6	6	7	6	31
18	6	6	7	7	7	33
19	6	5	6	6	7	30
20	6	7	7	6	6	32
21	6	5	6	6	7	30
22	6	6	7	7	6	32
23	7	7	7	7	6	34
24	7	7	7	6	7	34
25	6	7	6	6	7	32
26	6	7	7	6	6	32
27	7	7	7	6	7	34
28	6	6	7	6	7	32
29	7	7	6	7	6	33
30	7	6	6	6	7	32
31	6	6	7	6	7	32
32	7	7	6	7	6	33

Realizado por: Usca Danny, 2021

Después de realizar la tabulación de los datos en el aspecto de Operabilidad, como resultado obtenemos los siguientes valores, en la Tabla 9-4, se aprecia que, de una muestra de 32 usuarios como puntuación promedio obtenido su valor es de 32,06 de igual forma se obtiene un valor mínimo de 29, un valor máximo de 34 y una desviación estándar de 1,54.

El último valor da a entender que el promedio individual de cada respuesta por parte de los usuarios en comparación con el valor cercano de su media no está demasiado disperso, de tal forma que están de acuerdo en su mayoría con las interrogantes que se planteó en la encuesta.

Tabla 9-4: Resultados de la subcaracterística de Operabilidad

Operabilidad						
N	Puntuación promedio	Puntuación promedio obtenida	Desviación estándar	Valor mínimo	Valor máximo	Valor ideal
32	20	32,06	1,54	29	34	35

Realizado por: Usca Danny, 2021

Al analizar los resultados se puede llegar a la conclusión de que, tener un valor ideal de 35 sería lo más adecuado, pero, a pesar de no cumplir con lo estimado, como resultado se tiene que la puntuación promedio obtenida es de 32,06 superando por 12,06 a la puntuación promedio de 20.

Como se muestra en el Gráfico 2-4 se puede decir que los usuarios de la Corporación “LADRA” que fueron encuestados están muy de acuerdo que la plataforma es fácil de usar, amigable con el usuario, cada vez que los usuarios hacen uso de la plataforma realizan su trabajo con éxito ya que lo pueden usar sin instrucciones escritas y finalmente requieren la menor cantidad de pasos para lograr sus objetivos dentro de la aplicación.

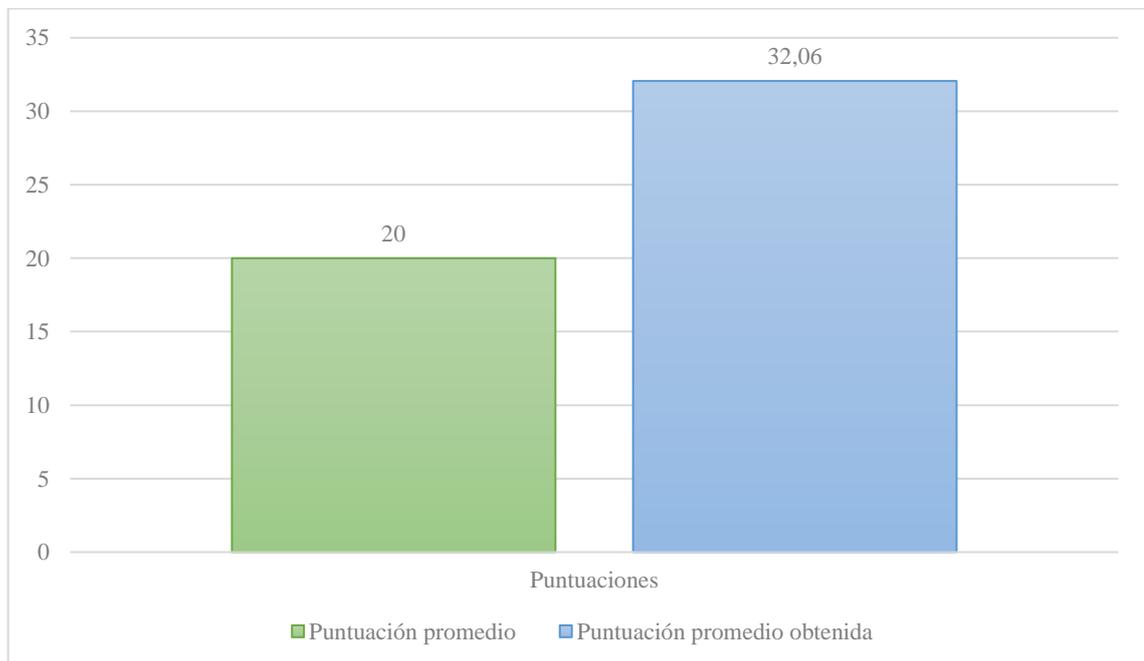


Gráfico 2-4: Comparación entre puntuaciones obtenidas de las respuestas - Operabilidad

Realizado por: Usca Danny, 2020

4.5.1.3. Estética

La subcaracterística de estética evalúa por medio del diseño de la interfaz de usuario, si la plataforma “LADRA” resulta ser amigable y tiene una buena interacción con el cliente.

En esta sección se tienen 3 interrogantes detalladas como:

Pregunta-10.- Estoy satisfecho con ella

Pregunta-11.- Se lo recomendaría a un amigo

Pregunta-12.- Es agradable de usar

La Tabla 10-4 corresponde a los valores obtenidos de la subcaracterística de Estética para cada encuesta aplicada se realizó la sumatoria con el propósito de usar este valor al momento de obtener los resultados finales.

Tabla 10-4: Tabulación de las preguntas de la subcaracterística de Estética

N° Encuesta	Estética			Sumatoria
	Pregunta 10	Pregunta 11	Pregunta 12	
1	6	6	7	19
2	7	6	7	20
3	6	7	7	20
4	7	7	7	21
5	6	7	7	20
6	7	6	6	19
7	6	7	7	20
8	7	6	6	19
9	7	6	7	20
10	7	6	7	20
11	7	7	7	21
12	6	7	7	20
13	7	6	6	19
14	6	6	7	19
15	4	6	6	16
16	6	7	7	20
17	7	7	6	20
18	6	7	7	20
19	6	7	7	20
20	7	7	6	20
21	5	6	6	17
22	6	7	7	20

N° Encuesta	Estética			Sumatoria
	Pregunta 10	Pregunta 11	Pregunta 12	
23	6	6	7	19
24	6	7	7	20
25	6	7	6	19
26	7	7	6	20
27	7	6	7	20
28	6	7	6	19
29	7	6	7	20
30	6	7	6	19
31	6	7	7	20
32	7	7	6	20

Realizado por: Usca Danny, 2021

Después de realizar la tabulación de los datos en el aspecto de Estética, como resultado obtenemos los siguientes valores, en la Tabla 11-4, se aprecia que, de una muestra de 32 usuarios como puntuación promedio obtenido su valor es de 19,56 de igual forma se obtiene un valor mínimo de 16, un valor máximo de 21 y una desviación estándar de 0,98.

El último valor da a entender que el promedio individual de cada respuesta por parte de los usuarios en comparación con el valor cercano de su media no está demasiado disperso, de tal forma que están de acuerdo en su mayoría con las interrogantes que se planteó en la encuesta.

Tabla 11-4: Resultados de la subcaracterística de Estética

Estética						
N	Puntuación promedio	Puntuación promedio obtenida	Desviación estándar	Valor mínimo	Valor máximo	Valor ideal
32	12	19,56	0,98	16	21	21

Realizado por: Usca Danny, 2021

Al analizar los resultados se puede llegar a la conclusión de que, tener un valor ideal de 21 sería lo más adecuado, pero, a pesar de no cumplir con lo estimado, como resultado se tiene que la puntuación promedio obtenida es de 19,56 superando por 7,56 a la puntuación promedio de 12.

Como se muestra en el Gráfico 3-4 se puede decir que los usuarios de la Corporación “LADRA” que fueron encuestados están muy de acuerdo que la plataforma es agradable de usar, se sienten muy satisfechos con ella y en futuro la recomendarían a sus amigos.

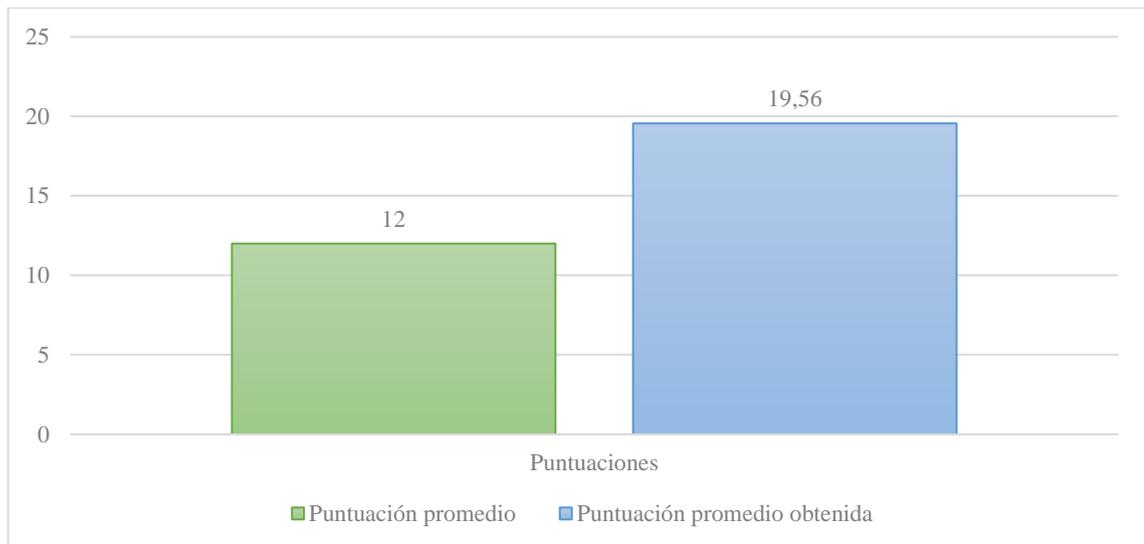


Gráfico 3-4: Comparación entre las puntuaciones obtenidas de las respuestas - Estética

Realizado por: Usca Danny, 2021

4.5.2. *Análisis Inferencial*

Por medio de los datos obtenidos en el análisis descriptivo se procede a realizar un análisis inferencial, mismo que va a servir para poder llegar a una conclusión acerca de la usabilidad de la plataforma “LADRA”. Para un correcto análisis es necesario seguir varios pasos con el fin de obtener una buena interpretación de los resultados.

En cada subcaracterística de la usabilidad se aplicó la prueba de Shapiro-Wilk a fin de verificar el tipo de distribución a la que pertenece (si es o no una distribución normal). Para ello se plantea dos hipótesis de la siguiente manera:

- H_0 : si p-valor \geq a alfa (α), tiene a una distribución normal
- H_1 : si p-valor $<$ a alfa (α), no tiene una distribución normal

La variable p-valor es la probabilidad de un conjunto de datos (inteligibilidad, operabilidad, estética) al aplicar la prueba de Shapiro-Wilks mientras que alfa corresponde al nivel de significancia, para el análisis se seleccionó un nivel de significancia del 5% (0,05).

Independientemente del resultado obtenido se debe calcular el valor de μ , el cual es un valor predefinido promedio base; calculado como producto entre el total de preguntas de cada una de las subcaracterísticas de usabilidad por el valor medio de la escala de Likert (valor medio = 4), en la Tabla 12-4 se detalla el valor de μ a ser usado en cada análisis.

Tabla 12-4: Valor de mu para cada subcaracterística de la usabilidad

Usabilidad	
Subcaracterística	Valor de mu
Inteligibilidad	16
Operabilidad	20
Estética	12

Realizado por: Usca Danny, 2021

Si se acepta la hipótesis nula (H_0) se procede a aplicar un test paramétrico como la Prueba “t” de Student, con ayuda del software R la forma correcta de aplicar el test es con la función `t.test(muestra, mu)`.

Caso contrario si se rechaza la hipótesis nula, es decir se acepta la hipótesis alternativa (H_1) se aplica un test no paramétrico como la Prueba de los rangos con signo de Wilcoxon, para ello se hace uso de la función `wilcox.test(muestra, mu, exact=FALSE)`, en el software R.

Finalmente, para contrastar los resultados obtenidos al aplicar el test correspondiente, se realiza la interpretación de los datos, planteando una hipótesis nula y una hipótesis alternativa. En donde:

- H_0 : si $\mu =$ puntuación promedio obtenida de la subcaracterística a analizar
- H_1 : si $\mu \neq$ puntuación promedio obtenida de la subcaracterística a analizar

4.5.2.1. Inteligibilidad

Para verificar la normalidad de la inteligibilidad se hizo uso del software RStudio, donde se aplica la función `shapiro.test` al conjunto de promedios individuales de Inteligibilidad tal como se muestra en la Figura 1-4.

```
> data_inteligibilidad <-c(25,27,26,26,27,27,25,26,25,27,25,27,24,23,24,24,27,27,21,
27,23,27,26,27,25,26,26,25,27,25,27,27)
> shapiro.test(data_inteligibilidad)

      shapiro-wilk normality test

data:  data_inteligibilidad
W = 0.82989, p-value = 0.0001567
```

Figura 1-4: Cálculo de la normalidad de la inteligibilidad con la función `shapiro.test`

Realizado por: Usca Danny, 2021

El valor de p resultante es de 0,0001567, lo cual determina que los datos no tienen una distribución normal. Por lo tanto, se hará uso del test no paramétrico de Wilcoxon para contrastar sus medias.

Prueba de Wilcoxon

Como parámetros previos a aplicar el test de Wilcoxon se tiene que, para la subcaracterísticas de inteligibilidad se va a utilizar un valor de $\mu=16$, como se fijó en la Tabla 12-4 y un valor de 0,05 que corresponde al 5% de nivel de significancia, como aclaración p-valor debe ser mayor a 0,05.

Para realizar la interpretación final de los resultados de inteligibilidad se planteó la siguiente hipótesis:

- H_0 : si $\mu(16)$ es igual a la puntuación promedio obtenida de inteligibilidad
- H_1 : si $\mu(16)$ es diferente a la puntuación promedio obtenida de inteligibilidad

```
> wilcox.test(data_inteligibilidad, mu = 16, exact = FALSE)
      wilcoxon signed rank test with continuity correction

data:  data_inteligibilidad
V = 528, p-value = 6.442e-07
alternative hypothesis: true location is not equal to 16
```

Figura 2-4: Test de Wilcoxon para validar la subcaracterística de inteligibilidad

Realizado por: Usca Danny, 2021

La Figura 2-4 muestra, que se rechaza la hipótesis nula (H_0) y se acepta la hipótesis alternativa (H_1), pues p-valor es de 6,442e-07, siendo este resultado mucho menor a nivel de significancia de 0,05. Por lo tanto, se concluye que los usuarios de la Corporación “LADRA” establecen que la plataforma móvil es útil, satisface los requerimientos funcionales planteados y les ahorra tiempo de proceso, siendo más productivos.

4.5.2.2. Operabilidad

Para verificar la normalidad de la operabilidad se hizo uso del software RStudio, donde se aplica la función `shapiro.test` al conjunto de promedios individuales de Operabilidad tal como se muestra en la Figura 3-4.

```

> data_operabilidad <-c(31,32,34,32,32,33,33,33,30,34,29,34,30,34,30,29,31,33,30,32,
30,32,34,34,32,32,34,32,33,32,32,33)
> shapiro.test(data_operabilidad)

      shapiro-wilk normality test

data:  data_operabilidad
w = 0.90058, p-value = 0.006407

```

Figura 3-4: Cálculo de la normalidad de la operabilidad con la función shapiro.test

Realizado por: Usca Danny, 2020

El valor de p resultante es de 0,006407, lo cual determina que los datos no tienen una distribución normal. Por lo tanto, se hará uso del test no paramétrico de Wilcoxon para contrastar sus medias.

Prueba de Wilcoxon

Como parámetros previos a aplicar el test de Wilcoxon se tiene que, para la subcaracterísticas de operabilidad se va a utilizar un valor de $\mu=20$, como se fijó en la Tabla 12-4 y un valor de 0,05 que corresponde al 5% de nivel de significancia, como aclaración p-valor debe ser mayor a 0,05.

Para realizar la interpretación final de los resultados de operabilidad se planteó la siguiente hipótesis:

- H_0 : si $\mu(20)$ es igual a la puntuación promedio obtenida de operabilidad
- H_1 : si $\mu(20)$ es diferente a la puntuación promedio obtenida de operabilidad

```

> wilcox.test(data_operabilidad, mu = 20, exact = FALSE)

      wilcoxon signed rank test with continuity correction

data:  data_operabilidad
V = 528, p-value = 7.145e-07
alternative hypothesis: true location is not equal to 20

```

Figura 4-4: Test de Wilcoxon para validar la subcaracterística de operabilidad

Realizado por: Usca Danny, 2020

La Figura 4-4 muestra, que se rechaza la hipótesis nula (H_0) y se acepta la hipótesis alternativa (H_1), pues p-valor es de 7,14e-07, siendo este resultado mucho menor a nivel de significancia de 0,05. Por lo tanto, se concluye que los usuarios de la Corporación “LADRA” establecen que la plataforma móvil es fácil de usar, lo pueden manipular sin necesidad de instrucciones, lo que

conlleva a tener éxito cada vez que es usada y más importante realizan el trabajo con la menor cantidad de pasos.

4.5.2.3. Estética

Para verificar la normalidad de la estética se hizo uso del software RStudio, donde se aplica la función `shapiro.test` al conjunto de promedios individuales de Estética tal como se muestra en la Figura 5-4.

```
> data_estetica <- c(19,20,20,21,20,19,20,19,20,20,21,20,19,19,16,20,20,20,20,20,17,20,19,20,19,20,20,19,20,19,20,20)
> shapiro.test(data_estetica)

      shapiro-wilk normality test

data:  data_estetica
W = 0.70967, p-value = 1.217e-06
```

Figura 5-4: Calculo de la normalidad de la estética con la función `shapiro.test`

Realizado por: Usca Danny, 2020

El valor de p resultante es de 1,217e-06, lo cual determina que los datos no tienen una distribución normal. Por lo tanto, se hará uso del test no paramétrico de Wilcoxon.

Prueba de Wilcoxon

Como parámetros previos a aplicar el test de Wilcoxon se tiene que, para la subcaracterísticas de estética se va a utilizar un valor de $\mu=12$, como se fijó en la Tabla 12-4 y un valor de 0,05 que corresponde al 5% de nivel de significancia, como aclaración p-valor debe ser mayor a 0,05.

Para realizar la interpretación final de los resultados de operabilidad se planteó la siguiente hipótesis:

- H_0 : si $\mu(12)$ es igual a la puntuación promedio obtenida de estética
- H_1 : si $\mu(12)$ es diferente a la puntuación promedio obtenida de estética

```
> wilcox.test(data_estetica, mu = 12, exact = FALSE)

      wilcoxon signed rank test with continuity correction

data:  data_estetica
V = 528, p-value = 4.003e-07
alternative hypothesis: true location is not equal to 12
```

Figura 6-4: Test de Wilcoxon para validar la subcaracterística de estética

Realizado por: Usca Danny, 2020

La Figura 6-4 muestra, que se rechaza la hipótesis nula (H_0) y se acepta la hipótesis alternativa (H_1), pues p-valor es de 4,003e-07, siendo este resultado mucho menor a nivel de significancia de 0,05. Por lo tanto, se concluye que los usuarios de la Corporación “LADRA” establecen que la plataforma móvil es agradable de usar, se encuentran satisfechos con sus características y se la recomendarían a sus amigos.

CONCLUSIONES

- A manera de disminuir el tiempo de codificación de la plataforma “LADRA”, se ha realizado un análisis de las diferentes herramientas necesarias para el rápido desarrollo. Debido a la amplia gama existente de lenguajes de programación y sus diferentes frameworks se realizó un análisis para buscar la tecnología que mayor se adecue a los requerimientos del sistema automatizado. Escogiendo dos frameworks Angular-Ionic y CodeIgniter, basados en los lenguajes de programación TypeScript y PHP respectivamente. Además, la arquitectura seleccionada fue Modelo-Vista-Controlador (MVC), que facilitó la mantenibilidad del código y su curva de aprendizaje relativamente baja.
- Posterior al análisis de la tecnología, en el desarrollo de la plataforma web y móvil, se utilizó la metodología ágil Scrum que brindó a lo largo del proyecto resultados tempranos mediante las entregas continuas de funcionalidades, se logró una mayor productividad al controlar riesgos y corregir errores en fases tempranas de desarrollo; obteniendo un total de 37 historias de usuario y 8 historias técnicas divididas en 15 sprints, que corresponden a una duración del proyecto de 30 semanas de trabajo, de tal forma que se construyó un producto software acorde a los requerimientos funcionales establecidos con el usuario final.
- Con la finalidad de obtener un producto software de calidad; se creó un plan de pruebas mediante la herramienta TestLink, la cual asignó por cada sprint una Build, evidenciando que cada Historia Técnica o de usuario cumplió con los tiempos planificados. Además, permitió validar las funcionalidades del sistema web y móvil, y corregir las fallas encontradas, dando una solución rápida para obtener un software eficiente.
- Teniendo en cuenta el modelo de Calidad del Producto Software (ISO/IEC 25010) y su característica de Usabilidad, se contempló la validación de la plataforma utilizando parámetros como la inteligibilidad, operabilidad y estética. Mediante el análisis descriptivo e inferencial aplicando el USE Questionary se obtuvo una puntuación promedio por cada ítem de usabilidad, para inteligibilidad un valor de 25,66 sobre 28, operabilidad un valor de 32,06 sobre 35 y estética un valor de 19,56 sobre 21. Los resultados sobrepasan la puntuación promedio establecida, concluyendo que la plataforma web y móvil tiene una interfaz agradable, es útil, fácil de usar, satisface los requerimientos funcionales planteados y les ahorra tiempo de proceso, siendo más productivos.

RECOMENDACIONES

- Se recomienda seguir utilizando el framework Angular-Ionic y CodeIgniter para la implementación de nuevas funcionalidades debido a que las bases de su desarrollo se encuentran codificadas en ese entorno de trabajo, lo que va a facilitar dar un buen mantenimiento.
- Se recomienda para futuras funcionalidades de la plataforma “LADRA”, el uso de la herramienta de administración de proyectos Jira, debido a su facilidad de uso con la metodología Scrum, su rápida adaptación con un plan de pruebas y su potente sistema de control de versiones.
- Es recomendable dar a conocer la plataforma desarrollada, al público general de la ciudad de Riobamba, pues los beneficios que ofrece el aplicativo son significativamente altos, beneficiando a los usuarios que busquen información sobre pérdida, adopción o rescate de mascotas.
- Se recomienda trabajar con despliegues continuos en un servidor remoto a fin de evitar complicaciones con los servicios de alojamiento en la nube y reducir posibles retrasos en el lanzamiento del software.

GLOSARIO

API: Comprende a una interfaz de programación de aplicaciones (Application Programming Interface), usado principalmente para la comunicación de nuestros servicios con distintos tipos de plataformas (Cerecedo,2020).

Backend: Aplicación o sistema mediante la cual se manipula información que ingresa en una interfaz o página web (Mercaderes Digitales, 2020).

Framework: Corresponde a un ambiente de trabajo y ejecución, que permite facilitar diferentes tareas y agilizar el desarrollo por medio de herramientas preestablecidas que contienen soluciones completas, (Romero, 2017).

FrontEnd: Interfaz gráfica mediante la cual el usuario interactúa directamente, en una aplicación web los componentes más comunes para su desarrollo son HTML y CSS (Chapaval, 2018).

REST: Es un modelo arquitectónico para programación web, apoyado completamente en HTTP, por medio de la cual se desarrollan interfaz de programación de aplicaciones (JosmanTek, 2014).

CSS: Su acrónimo proviene de Cascading Style Sheets que significa hojas de estilo en cascada, usado principalmente para agregar aspectos a cada elemento de una página en HTML, generando una visualización más agradable para el usuario (Plaza, 2017).

BIBLIOGRAFIA

AGUILAR, J., ¿Qué es el patrón MVC en programación y por qué es útil? | campusMVP.es. [en línea].2019. [Consulta: 22 septiembre 2020]. Disponible en: <https://www.campusmvp.es/recursos/post/que-es-el-patron-mvc-en-programacion-y-por-que-es-util.aspx>.

ALVARADO, P., Qué Es Xcode, Para Qué Sirve y Cómo Descargar. [en línea].2014. [Consulta: 19 septiembre 2020]. Disponible en: <https://www.ipadizate.es/2014/07/20/xcode-93212/>.

ANDROID STUDIO, Introducción a Android Studio | Desarrolladores de Android. [en línea].2020. [Consulta: 19 septiembre 2020]. Disponible en: <https://developer.android.com/studio/intro?hl=es-419>.

ANTÓN DORANTES, C., Laravel, el mejor framework en PHP. *Platzi* [en línea].2016. [Consulta: 25 febrero 2021]. Disponible en: <https://platzi.com/blog/laravel-framework-php/>.

BAHIT, E., POO y MVC en PHP. [en línea].2011. S.l.: [Consulta: 22 septiembre 2020]. Disponible en: <http://eugeniabahit.blogspot.com>.

BLANES, J.A., ¿Qué es React Native? *Deloitte* [en línea].2019. [Consulta: 25 febrero 2021]. Disponible en: <https://www2.deloitte.com/es/es/pages/technology/articles/que-es-react-native.html>.

BLAS, M.J., GONNET, S. y LEONE, H., Especificación de la Calidad en Software-as-a-Service : Definición de un Esquema de Calidad basado en el Estándar ISO / IEC 25010. ASSE 2016, 17o Simposio Argentino de Ingeniería en Software [en línea].2016, pp. 135-146. Disponible en: http://sedici.unlp.edu.ar/bitstream/handle/10915/57158/Documento_completo.pdf?sequence=1.

BOSNIC, S., PAPP, I. y NOVAK, S., The development of hybrid mobile applications with Apache Cordova. *24th Telecommunications Forum, TELFOR 2016* [en línea].2016. S.l.: Institute of Electrical and Electronics Engineers Inc., pp. 1–4. [Consulta: 17 septiembre 2020]. ISBN 9788674666494. DOI 10.1109/TELFOR.2016.7818919. Disponible en: <http://ieeexplore.ieee.org/document/7818919/>.

CASTEJÓN, J., Arquitectura y diseño de sistemas web modernos. , ISSN 1698-8841.

CERECEDO, D., ¿Qué es un API? - Byteflair. [en línea].2016. [Consulta: 31 diciembre 2020]. Disponible en: <https://byteflair.com/es/2013/01/que-es-un-api/>.

CHAPAVAL, N., Qué es Frontend y Backend. [en línea]. 2018, [Consulta: 15 abril 2021]. Disponible en: <https://platzi.com/blog/que-es-frontend-y-backend/>.

CHERREZ, G., Diseño e implementación de un sistema de localización, medición de velocidad y aceleración de un vehículo para determinar rutas alternas, utilizando tecnologías GPS Y GPRS. . S.l.: Escuela Superior Politécnica de Chimborazo.

CIMAN, M. y GAGGI, O., An empirical analysis of energy consumption of cross-platform frameworks for mobile development. *Pervasive and Mobile Computing* [en línea].2017, vol. 39, pp. 214–230. [Consulta: 18 septiembre 2020]. ISSN 15741192. DOI 10.1016/j.pmcj.2016.10.004. Disponible en: <https://linkinghub.elsevier.com/retrieve/pii/S1574119216303170>.

CODEIGNITER, CodeIgniter Web Framework. [en línea].2006. [Consulta: 19 septiembre 2020]. Disponible en: <https://codeigniter.com/>.

DAGNE, L., Flutter for Cross-Platform App and SDK Development. *Metropolia University of Applied Sciences* [en línea].2019, no. May. Disponible en: <https://www.theseus.fi/bitstream/handle/10024/172866/LukasDagneThesis.pdf?sequence=2&isAllowed=y>.

DE LEÓN, Á., Servidor Apache. [en línea].2019. [Consulta: 22 septiembre 2020]. Disponible en: <https://blog.infranetworking.com/que-es-apache-servidor/>.

DI GIACOMO, M., MySQL: Lessons learned on a digital library. *IEEE Software* [en línea].2005, vol. 22, no. 3, pp. 10–13. [Consulta: 22 septiembre 2020]. ISSN 07407459. DOI 10.1109/MS.2005.71. Disponible en: <https://ieeexplore.ieee.org/document/1438321/>.

DÍAZ, A.F., Creación de los Módulos de Localización de Personas, Emisión de Alertas y Definición de Rutas en el Sistema Skillbox, con Signalr de la Tecnología Web Asp. Net. ,

DÍAZ, J., SAMPEDRO, L. y VARGAS, F., (PDF) Instalación y configuración de Apache, un servidor Web gratis. [en línea].2002. [Consulta: 22 septiembre 2020]. Disponible en: https://www.researchgate.net/publication/28201149_Instalacion_y_configuracion_de_Apache_un_servidor_Web_gratis.

EGUILUZ, J., *Desarrollo web ágil con Symfony2* [en línea].2013. S.l.: s.n. [Consulta: 18 septiembre 2020]. Disponible en: https://www.academia.edu/15037504/Desarrollo_web_agil_con_Symfon_Javier_Eguiluz_1092.

ELROM, E., *Pro MEAN Stack Development* [en línea].2016. Berkeley, CA: Apress. [Consulta: 18 septiembre 2020]. Disponible en: <http://link.springer.com/10.1007/978-1-4842-2044-3>.

ESTDALE, J. y GEORGIADOU, E., Applying the ISO/IEC 25010 Quality Models to Software Product. *Communications in Computer and Information Science* [en línea].2018. S.l.: Springer Verlag, pp. 492–503. [Consulta: 23 septiembre 2020]. ISBN 9783319979243. DOI 10.1007/978-3-319-97925-0_42. Disponible en: http://link.springer.com/10.1007/978-3-319-97925-0_42.

FERNANDEZ, Y. y DIAZ, Y., Patrón Modelo-Vista-Controlador. | Díaz González | Telemática. [en línea].2012. [Consulta: 22 septiembre 2020]. Disponible en: <http://revistatelematica.cujae.edu.cu/index.php/tele/article/view/15/10>.

FERREÑO, E., Android Studio: esto es lo que puedes hacer con este programa. [en línea].2020. [Consulta: 19 septiembre 2020]. Disponible en: <https://elandroidelibre.elespanol.com/2020/01/android-studio-que-es-y-para-que-se-utiliza.html>.

GARCÍA, M., ¿QUE ES XAMPP Y COMO PUEDO USARLO? – Nettix Perú. [en línea].2020. [Consulta: 19 septiembre 2020]. Disponible en: <https://www.nettix.com.pe/blog/web-blog/que-es-xampp-y-como-puedo-usarlo>.

GARZAS, J., Análisis y comparativa de herramientas de gestión de pruebas de uso gratuito. [en línea].2013. [Consulta: 23 septiembre 2020]. Disponible en: <https://www.javiergarzas.com/2013/10/herramientas-de-gestion-de-pruebas.html>.

HADLEY, M.J., Web Application Description Language (WADL). [en línea].2006. S.l.: [Consulta: 10 septiembre 2020]. Disponible en: <http://research.sun.com/techrep/>.

HALILI, F. y RAMADANI, E., Web Services: A Comparison of Soap and Rest Services. *Modern Applied Science*, vol. 12, no. 3. ISSN 1913-1852. DOI 10.5539/mas.v12n3p175.

HEARD, I.R. y ARDILA, N.R., Native vs. Hybrid Mobile Applications as Society Enters the Internet of Things. *International Journal of Hyperconnectivity and the Internet of Things*, vol. 2, no. 2, pp. 30–42. ISSN 2473-4365. DOI 10.4018/ijhiot.2018070103.

HEREDIA, V., 56 de cada 100 personas tiene celular en el Ecuador. *El Comercio* [en línea].2017, [Consulta: 8 noviembre 2020]. Disponible en: <https://www.elcomercio.com/guaifai/celulares-ecuador-inec-estadistica-crecimiento.html>.

HU, Y., NANDA, A. y YANG, Q., Measurement, analysis and performance improvement of the Apache Web server. *1999 IEEE International Performance, Computing and Communications Conference, IPCCC 1999* [en línea].1999. S.l.: Institute of Electrical and Electronics Engineers Inc., pp. 261–267. [Consulta: 21 septiembre 2020]. ISBN 0780352580. DOI 10.1109/PCCC.1999.749447. Disponible en: <http://ieeexplore.ieee.org/document/749447/>.

ISITAN, M. y KOKLU, M., Comparison and Evaluation of Cross Platform Mobile Application Development Tools. *International Journal of Applied Mathematics, Electronics and Computers* [en línea].2020, vol. 8, no. 4, pp. 273-281. [Consulta: 22 febrero 2021]. ISSN 2147-8228. DOI 10.18100/ijamec.832673. Disponible en: www.dergipark.org.tr/ijamec.

ISO 25000., ISO 25000 - Calidad de Software y datos [en línea]. 2019. [Consulta: 10 enero 2021]. Disponible en: <https://iso25000.com/index.php/normas-iso-25000/iso-25010/19-iso-iec-25010>.

ISO2500, ISO 25010. [en línea].2019. [Consulta: 23 septiembre 2020]. Disponible en: <https://iso25000.com/index.php/normas-iso-25000/iso-25010>.

JOBE, W., Native Apps Vs. Mobile Web Apps. *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 7, no. 4, pp. 27. ISSN 1865-7923. DOI 10.3991/ijim.v7i4.3226.

JOSMANTEK., Restful APIs - ¿Que es REST? - JosmanTek Blog. [blog]. 2014. [Consulta: 15 abril 2021]. Disponible en: <http://blog.josmantek.com/php/restful-apis-que-es-rest/>.

JUSTIN, J. y JUDE, J., Welcome to the World of Ionic. *Learn Ionic 2* [en línea].2017. Berkeley, CA: Apress, pp. 1–3. [Consulta: 19 septiembre 2020]. Disponible en: http://link.springer.com/10.1007/978-1-4842-2617-9_1.

KARANJIT, A., MEAN vs. LAMP Stack. [en línea].2016. S.l.: [Consulta: 18 septiembre 2020]. Disponible en: https://repository.stcloudstate.edu/csit_etds/11.

KHAN, A.I., AL-BADI, A. y AL-KINDI, M., Progressive web application assessment using AHP. *Procedia Computer Science* [en línea].2019. S.l.: Elsevier B.V., pp. 289–294. [Consulta:

11 septiembre 2020]. DOI 10.1016/j.procs.2019.08.041. Disponible en: www.sciencedirect.comwww.elsevier.com/locate/procedia1877-0509www.sciencedirect.comwww.elsevier.com/locate/procedia1877-0509.

KHANDEPARKAR, A., GUPTA, R. y SINDHYA, B., An Introduction to Hybrid Platform Mobile Application Development. *International Journal of Computer Applications* [en línea].2015. S.l.: [Consulta: 17 septiembre 2020]. Disponible en: www.ijcaonline.org.

LAAZIRI, Majida; et al. A Comparative study of PHP frameworks performance. *Procedia Manufacturing* [en línea].2019. S.l.: Elsevier B.V., pp. 864-871. [Consulta: 22 febrero 2021]. DOI 10.1016/j.promfg.2019.02.295. Disponible en: www.sciencedirect.com Available online at www.sciencedirect.comwww.elsevier.com/locate/procedia.

LÓPEZ, A., Qué es Postman y para qué sirve | OpenWebinars. [en línea].2019. [Consulta: 19 septiembre 2020]. Disponible en: <https://openwebinars.net/blog/que-es-postman/>.

LOZADA, J., *Revista CienciAmérica* [en línea].2014. S.l.: Universidad Tecnológica Indoamérica (UTI). [Consulta: 20 septiembre 2019]. Disponible en: <https://dialnet.unirioja.es/servlet/articulo?codigo=6163749>.

LUJÁN, S., Programación de aplicaciones web: Historia, principios básicos y clientes web. *Editorial Club Universitario* [en línea].2002, [Consulta: 8 noviembre 2020]. Disponible en: https://rua.ua.es/dspace/bitstream/10045/16995/1/sergio_lujan-programacion_de_aplicaciones_web.pdf.

MACÍAS, R., Generación de informes de proyectos ágiles en TestLink - Adictos al trabajo. *Autentia* [en línea].2013. [Consulta: 8 noviembre 2020]. Disponible en: <https://www.adictosaltrabajo.com/2013/06/05/trabajando-con-test-link/>.

MERCADERES DIGITALES., ¿Qué es un backend? [en línea]. 2020. [Consulta: 31 diciembre 2020]. Disponible en: <https://www.mercaderesdigitales.com/Diccionario%20de%20Marketing%20que-es-backend/>

MIN TUN, P., Choosing a Mobile Application Development Approach Mobile Financial Services in Myanmar View project Choosing a Mobile Application Development Approach. [en línea].2014, [Consulta: 16 septiembre 2020]. DOI 10.14456/ajmi.2014.4. Disponible en: <https://www.researchgate.net/publication/342437581>.

OLMEDO, M. y NACIMBA, P., DSpace ESPOCH.: Desarrollo de un portal web y una aplicación móvil para la ubicación de los sitios turísticos del cantón Guano. [en línea].2017. [Consulta: 24 septiembre 2020]. Disponible en: <http://dspace.espoch.edu.ec/handle/123456789/7356>.

PLAZA, J., CSS - Code - ¿Qué es CSS? [blog].2017. [Consulta: 15 abril 2021]. Disponible en: <https://3con14.biz/css/conceptos/4-¿qué-es-css.html>.

POULTER, A.J., JOHNSTON, S.J. y COX, S.J., Using the MEAN stack to implement a RESTful service for an Internet of Things application. *IEEE World Forum on Internet of Things, WF-IoT 2015 - Proceedings* [en línea].2015. S.I.: Institute of Electrical and Electronics Engineers Inc., pp. 280–285. [Consulta: 18 septiembre 2020]. ISBN 9781509003655. DOI 10.1109/WF-IoT.2015.7389066. Disponible en: <http://ieeexplore.ieee.org/document/7389066/>.

PUSKARIC, Hrvoje; et al. DEVELOPMENT OF WEB BASED APPLICATION USING SPA ARCHITECTURE. [en línea].2019, [Consulta: 14 septiembre 2020]. ISSN 2620-2832. DOI 10.24874/PES01.02.044. Disponible en: <https://www.researchgate.net/publication/339355883>.

RADI, A.A., Evaluation of Xamarin Forms for MultiPlatform Mobile Application Development. *School of Computing and Information Systems* [en línea].2016, pp. 23. [Consulta: 8 noviembre 2020]. Disponible en: <http://scholarworks.gvsu.edu/cgi/viewcontent.cgi?article=1238&context=cistechlib>.

RAMÍREZ, P., Servidor Web Apache: ¿Qué es, cuáles son sus características y para qué se utiliza? - ITSoftware. [en línea].2017. [Consulta: 22 septiembre 2020]. Disponible en: <https://itsoftware.com.co/content/servidor-web-apache/>.

ROBLEDANO, Á., Qué es MySQL: Características y ventajas. [en línea].2019. [Consulta: 22 septiembre 2020]. Disponible en: <https://openwebinars.net/blog/que-es-mysql/>.

RODRÍGUEZ, Rafael, Manejo de test con TestLink - Adictos al trabajo. [en línea].2013. [Consulta: 23 septiembre 2020]. Disponible en: <https://www.adictosaltrabajo.com/2013/05/07/test-link/>.

RODRÍGUEZ, Ximena, Tipos de servidores web | OpenWebinars. [en línea].2019. [Consulta: 22 septiembre 2020]. Disponible en: <https://openwebinars.net/blog/tipos-servidores-web/>.

ROMERO, A., ¿QUE ES FRAMEWORK? [blog].2017. [Consulta: 15 abril 2021]. Disponible en: <https://programacion2.foroactivo.com/t46-que-es-framework>.

RUIZ, A., ¿Qué es Bootstrap? - Everriculum. [en línea].2014. [Consulta: 19 septiembre 2020]. Disponible en: <https://www.mastermarketingdigital.com/everriculum/2014/09/20/que-es-bootstrap/>.

SCHWABER, K., SCRUM Development Process. *Business Object Design and Implementation* [en línea].1997. London: Springer London, pp. 117–134. [Consulta: 23 septiembre 2020]. Disponible en: http://link.springer.com/10.1007/978-1-4471-0947-1_11.

SIVAJI, Ashok; et al. Cultural UX/Usability View project Digital government citizen focus View project Measuring Public Value UX based on ISO/IEC 25010 Quality Attributes Case Study on e-Government Website. [en línea].2014, [Consulta: 23 septiembre 2020]. DOI 10.13140/2.1.4303.8720. Disponible en: <https://www.researchgate.net/publication/267335773>.

SPEICHER, M., What is Usability? [en línea].2015. S.l.: Disponible en: <https://github.com/globis-ethz/jqmetrics>.

SRIVASTAVA, A., BHARDWAJ, S. y SARASWAT, S., SCRUM model for agile methodology. *Proceeding - IEEE International Conference on Computing, Communication and Automation, ICCCA 2017* [en línea].2017. S.l.: Institute of Electrical and Electronics Engineers Inc., pp. 864–869. [Consulta: 23 septiembre 2020]. ISBN 9781509064717. DOI 10.1109/CCAA.2017.8229928. Disponible en: <http://ieeexplore.ieee.org/document/8229928/>.

STEPNIAK, W. y NOWAK, Z., Performance analysis of SPA web systems. *Advances in Intelligent Systems and Computing* [en línea].2017. S.l.: Springer Verlag, pp. 235–247. [Consulta: 15 septiembre 2020]. ISBN 9783319465821. DOI 10.1007/978-3-319-46583-8_19. Disponible en: http://link.springer.com/10.1007/978-3-319-46583-8_19.

TANDEL, S., TANDEL, S.S. y JAMADAR, A., Engineering and Technology (A High Impact Factor. *International Journal of Innovative Research in Science* [en línea].2018, vol. 7, no. 9. [Consulta: 11 septiembre 2020]. ISSN 2347-6710. DOI 10.15680/IJRSET.2018.0709021. Disponible en: www.ijirset.com.

TERRERA, G., TestingBaires » ¿Querés conocer al detalle la herramienta Testlink? [en línea].2019. [Consulta: 23 septiembre 2020]. Disponible en:

<https://testingbaire.com/2019/06/22/queres-conocer-al-detalle-la-herramienta-testlink/>.

TESTLINK, Administración y Ejecución de Pruebas. . S.l.:

TESTLINK COMMUNITY, User Manual. . S.l.:

VALVERDE, F. y PASTOR, O., Dealing with REST Services in Model-driven Web Engineering Methods. [en línea].2009. S.l.: [Consulta: 17 septiembre 2020]. Disponible en: <https://www.researchgate.net/publication/228975781>.

VELASCO, R., Visual Studio Code: editor de texto de código abierto para programar. [en línea].2020a. [Consulta: 19 septiembre 2020]. Disponible en: <https://www.softzone.es/programas/utilidades/visual-studio-code/>.

VELASCO, R., VMware: instalar sistemas operativos con máquinas virtuales. [en línea].2020b. [Consulta: 19 septiembre 2020]. Disponible en: <https://www.softzone.es/programas/sistema/vmware/>.

XINYANG, F., JIANJING, S. y YING, F., REST: An alternative to RPC for web services architecture. *2009 1st International Conference on Future Information Networks, ICFIN 2009* [en línea].2009. S.l.: IEEE, pp. 7–10. [Consulta: 17 septiembre 2020]. ISBN 9781424451593. DOI 10.1109/ICFIN.2009.5339611. Disponible en: <http://ieeexplore.ieee.org/document/5339611/>.

YANG, Yahui; et al. Mobile Terminal Development Plan of Cross-Platform Mobile Application Service Platform Based on Ionic and Cordova. *Proceedings - 2017 International Conference on Industrial Informatics - Computing Technology, Intelligent Technology, Industrial Information Integration, ICIICII 2017* [en línea].2018. S.l.: Institute of Electrical and Electronics Engineers Inc., pp. 100–103. [Consulta: 18 septiembre 2020]. ISBN 9781538624340. DOI 10.1109/ICIICII.2017.28. Disponible en: <http://ieeexplore.ieee.org/document/8328596/>.



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

**DIRECCIÓN DE BIBLIOTECAS Y RECURSOS DEL APRENDIZAJE
UNIDAD DE PROCESOS TÉCNICOS Y ANÁLISIS BIBLIOGRÁFICO Y DOCUMENTAL**

REVISIÓN DE NORMAS TÉCNICAS, RESUMEN Y BIBLIOGRAFÍA

Fecha de entrega: 10 / 06 / 2021

INFORMACIÓN DEL AUTOR/A (S)
Nombres – Apellidos: <i>Danny Alexander Usca Farinango</i>
INFORMACIÓN INSTITUCIONAL
Facultad: <i>INFORMÁTICA Y ELECTRÓNICA</i>
Carrera: <i>INGENIERÍA EN SISTEMAS</i>
Título a optar: <i>INGENIERO EN SISTEMAS INFORMÁTICOS</i>
f. Analista de Biblioteca responsable: Lic. Luis Caminos Vargas Mgs.

**LUIS
ALBERTO
CAMINOS
VARGAS**

Firmado digitalmente por LUIS
ALBERTO CAMINOS VARGAS
Nombre de reconocimiento (DN):
c=EC, I=ROBAMBA,
serialNumber=0602766974,
cn=LUIS ALBERTO CAMINOS
VARGAS
Fecha: 2021.06.10 17:52:41
-05'00'



1055-DBRA-UTP-2021