



UNIVERSIDAD POLITÉCNICA SALESIANA

SEDE QUITO

CARRERA DE INGENIERÍA DE SISTEMAS

**DESARROLLO DE UNA APLICACIÓN WEB Y MÓVIL QUE PERMITA LA
ASIGNACIÓN AUTOMÁTICA, GESTIÓN Y MONITOREO DE LOS
TRÁMITES DE TRANSFERENCIA QUE ADMINISTRA EL CONSORCIO
JURÍDICO ORQUERA & ORQUERA.**

Trabajo de titulación previo a la obtención del

Título de Ingenieros de Sistemas

AUTORES: Cristina Daniela Rodríguez Peralta

David Antonio Cadena Egas

TUTOR: Walter Fernando Gaibor Naranjo

Quito - Ecuador

2022

**CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE
TITULACIÓN**

Nosotros, Cristina Daniela Rodríguez Peralta y David Antonio Cadena Egas con documentos de identificación N° 1725449571 y N° 1720842481, manifestamos que:

Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Quito, 10 de marzo de 2022

Atentamente,



.....

Cristina Daniela Rodríguez Peralta

1725449571



.....

David Antonio Cadena Egas

1720842481

**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO
DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA**

Nosotros, Cristina Daniela Rodríguez Peralta y David Antonio Cadena Egas con documentos de identificación N° 1725449571 y N° 1720842481, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo del Proyecto Técnico: “Desarrollo De Una Aplicación Web Y Móvil Que Permita La Asignación Automática, Gestión Y Monitoreo De Los Trámites De Transferencia Que Administra El Consorcio Jurídico Orquera & Orquera”, el cual ha sido desarrollado para optar por el título de: Ingenieros de Sistemas, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia, suscribo este documento en el momento que hago entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Quito, 10 de marzo de 2022



.....

Cristina Daniela Rodríguez Peralta

1725449571



.....

David Antonio Cadena Egas

1720842481

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Walter Fernando Gaibor Naranjo con documento de identificación N° 1713106647, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: DESARROLLO DE UNA APLICACIÓN WEB Y MÓVIL QUE PERMITA LA ASIGNACIÓN AUTOMÁTICA, GESTIÓN Y MONITOREO DE LOS TRÁMITES DE TRANSFERENCIA QUE ADMINISTRA EL CONSORCIO JURÍDICO ORQUERA & ORQUERA realizado por Cristina Daniela Rodríguez Peralta con documento de identificación N° 17225449571 y David Antonio Cadena Egas con documento de identificación N° 1720842481, obteniendo como resultado final el trabajo de titulación bajo la opción de Proyecto Técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Quito, 10 de marzo 2022



Ing. Walter Fernando Gaibor Naranjo, MSc

1713106647

DEDICATORIA

Me gustaría hacer un reconocimiento especial y dedicar este trabajo de Tesis a todas las personas que me brindaron su apoyo para que este objetivo culmine con éxito.

Para Dios por bendecir mi camino a lo largo de esta etapa.

Para mis padres Verónica Egas y Carlos Cadena, por brindarme todo su apoyo desde el primer día que decidí continuar con mis estudios.

Para mi abuelita Selma Colem, que es uno de los pilares fundamentales en mi vida y me ha brindado toda su sabiduría para que esto sea posible.

Para mi tía Alejandra Minda, por ser esa persona maravillosa y estar presente con entusiasmo y alegría en todo momento que lo necesite.

Para mi novia y hermanas, quienes con su presencia, consejos y apoyo hicieron que esto sea mucho más sencillo.

Por último, para el Ing. Walter Gaibor y para la Universidad Politécnica Salesiana, por ayudarme a potenciar cada uno de mis conocimientos en el campo de la Ingeniería de Sistemas.

David Antonio Cadena Egas

DEDICATORIA

A mi familia, en especial a Olga Jumbo, quien es soporte e inspiración de todos mis logros.

Al hombre más inteligente de este mundo material, Sebastián Rodríguez, por confiar tanto en mis capacidades.

A Nicolas Rodríguez, por ser tan diferente a mí y permitirme aprender tanto de ti.

A mis hermanas, Karen y Domenika, por estar siempre conmigo y ayudarme en todo.

A mi abue Nancy, por sus consejos y motivación.

A mis padres, Freddy Rodríguez y Laura Peralta, por darme las mejores experiencias de vida, enseñarme que todo es posible y ser los mejores en todo lo que hacen.

A mis tíos, Vivianna Peralta y Victor Rodríguez, por guiarme con los mejores consejos de vida.

A David, Anahi, Aitanna, Martina y Agustin.

A mi novio, por acompañarme en todo este camino.

Cristina Daniela Rodríguez Peralta

ÍNDICE GENERAL

INTRODUCCIÓN	1
Antecedentes.....	1
Problema.....	1
Justificación	2
Objetivos.....	3
Objetivo general	3
Objetivos específicos.....	3
Metodología.....	3
CAPÍTULO I.....	5
MARCO TEÓRICO.....	5
1.1. Consortio jurídico Orquera & Orquera.....	5
1.2. Sistemas de Información	6
1.2.1. Entrada.....	6
1.2.2. Procesamiento.....	6
1.2.3. Salida	6
1.3. Scrum.....	6
1.3.1. Product Owner	7
1.3.2. Scrum Máster.....	7
1.3.3. Development Team.....	7
1.4. Sprint	7
1.5. UWE.....	8

1.6. Trámite	8
1.7. Symfony	8
1.8. React Native	9
1.9. Bootstrap.....	9
CAPÍTULO II	10
REQUERIMIENTOS Y ANÁLISIS.....	10
2.1. Proceso de almacenamiento de trámites de transferencia de dominio	
10	
2.2. Levantamiento de requerimientos	12
2.2.1. Listado de peticiones en lenguaje común	12
2.3. Requerimientos.....	12
2.3.1. Requerimientos funcionales.....	12
2.3.2. Requerimiento de sistema y técnicos.....	13
2.4. Historias de usuario	13
2.5. Diagrama de contexto.....	15
2.6. Diagramas de caso de uso.....	15
2.6.1. Lista de actores	15
2.6.2. Diagrama de caso de uso de administrador	16
2.6.3. Diagrama de caso de uso de colaborador	17
2.7. Diagrama de secuencia	18
2.7.1. Diagrama de Secuencia del Administrador	19
2.7.2. Diagrama de Secuencia del Colaborador.....	21

2.8.	Diagrama de actividades.....	21
2.9.	Diagrama de clases	23
2.10.	Diagrama de datos	24
CAPÍTULO III.....		25
DISEÑO		25
3.1.	Arquitectura de software	25
3.2.	Diagrama de paquetes.....	26
3.3.	Diagrama de componentes	27
3.4.	Diseño de algoritmo de asignación de trámites.....	28
3.5.	Diseño de interfaces	30
3.5.1.	Prototipo de interfaz de inicio de sesión.....	30
3.5.2.	Prototipo de interfaz de ingreso de datos.....	31
3.5.3.	Prototipo de interfaz de listado de datos.....	32
3.5.4.	Prototipo de interfaz de menú lateral.....	33
CAPÍTULO IV CODIFICACIÓN		35
4.1.	Configuración de entorno de desarrollo	35
4.2.	Interfaz.....	36
4.3.	Funcionalidades del sistema clasificado por clases.....	36
4.4.	API's del sistema.....	38
4.5.	Codificación	38
4.5.1.	Controlador.....	38
4.5.2.	Entity	40

4.5.3. Vistas	41
4.6. Interfaces reales del sistema	42
4.6.1. Aplicativo web.....	42
4.6.2. Aplicativo móvil.....	46
CAPITULO V	48
PRUEBAS Y CORRECCIONES	48
5.1. Pruebas unitarias.....	48
5.2. Pruebas integrales.....	62
5.3. Número de pruebas realizadas.....	64
5.4. Pruebas de carga	65
5.5. Prueba de rendimiento.....	65
CONCLUSIONES	67
RECOMENDACIONES	69
LISTA DE REFERENCIAS	70

ÍNDICE DE TABLAS

Tabla 1. Tipos de trámites de transferencia de dominio	5
Tabla 2. Roles de scrum	8
Tabla 3. Historias de usuario.....	14
Tabla 4. Actores del sistema	16
Tabla 5. Descripción de clases	37
Tabla 6. Descripción de las API's	38
Tabla 7. P001	48
Tabla 8. P002	49
Tabla 9. P003	50
Tabla 10. P004	51
Tabla 11. P005	52
Tabla 12. P006	53
Tabla 13. P007	54
Tabla 14. P008	55
Tabla 15. P009	56
Tabla 16. P010	57
Tabla 17. P011	58
Tabla 18. P012	59
Tabla 19. P013	60
Tabla 20. P014	61
Tabla 21. P015	62
Tabla 22. P016	63
Tabla 23. Número de intentos por prueba.....	64

ÍNDICE DE FIGURAS

Figura 1.	11
Figura 2. Diagrama de contexto.....	15
Figura 3. Diagrama de caso de uso de administrador	17
Figura 4. Diagrama de caso de uso de colaborador	18
Figura 5. Diagrama de secuencia de administrador del sistema	20
Figura 6. Diagrama de secuencia del colaborador	21
Figura 7. Diagrama de actividades del sistema.....	22
Figura 8. Diagrama de clases	23
Figura 9. Diagrama de datos	24
Figura 10. Arquitectura de software	25
Figura 11. Diagrama de paquetes.....	26
Figura 12. Diagrama de componentes.....	27
Figura 13. Pesos de trámites	28
Figura 14. Pantalla de inicio de sesión.....	30
Figura 15. Pantalla de ingreso.....	31
Figura 16. Pantalla de lista.....	32
Figura 17. Pantalla de menú lateral.....	33
Figura 18. Pantalla móvil de lista.....	34
Figura 19. Estructura de carpetas en vs code	35
Figura 20. Controlador.....	39
Figura 21. Entity	40
Figura 22. Estructura de get y set de un entity.....	41
Figura 23. Estructura de las vistas	42
Figura 24. Pantalla real de inicio de sesión.....	43

Figura 25. Pantalla real de “trámites asignados a mi”	43
Figura 26. Pantalla real de ingreso de datos.....	44
Figura 27. Pantalla real de actualización de datos	44
Figura 28. Ejemplo de eliminado lógico	45
Figura 29. l.....	46
Figura 30. Pantalla real de aplicación móvil filtrar datos.....	47
Figura 31. Resumen de resultados de prueba de carga	65
Figura 32. Gráfico de prueba de rendimiento	66

RESUMEN

El sistema de información que gestiona y monitorea los trámites de transferencia da solución a la problemática que presenta el consorcio jurídico Orquera & Orquera, este permite la gestión ordenada de los mismos que la organización administra. Cabe mencionar que el sistema facilita el ingreso de usuarios y clientes, almacenando toda esta data para la generación de reportes, mejora la asignación de trámites al usuario dependiendo de su carga laboral y el tiempo de ejecución, no obstante, esto es posible gracias a la elaboración de un algoritmo.

La documentación presenta a continuación una división en capítulos para comprender mejor, tales como el análisis, diseño, desarrollo y pruebas del software en sus dos versiones: web y móvil, las cuales son desarrollados bajo dos lenguajes de programación que son PHP con su framework Symfony 5.2 y React Native respectivamente.

Además, para que toda la información sea legible se realiza diagramas de cada una de sus características empezando por su base de datos hasta sus interfaces, las mismas que se desarrollan con colores distintivos de la organización.

Después de realizar todo el proceso de análisis se continua con la etapa final de pruebas, la cual además de garantizar la calidad, también verifica la funcionalidad de la aplicación y cada uno de sus módulos.

ABSTRACT

The information system that manages and monitors the procedures for transferring the solution to the problem presented by the Orquera & Orquera legal consortium, this allows the orderly management of the same as the administrative organization. It is so worth to mention that the system facilitates the entry of users and clients, storing all this data for the generation of reports, improves the assignment of procedures to the user according to their workload and execution time, however, this is possible thanks to the elaboration of an algorithm.

The documentation presents below a division into chapters to better understand, such as the analysis, design, development and testing of the software in its two versions: web and mobile, which are developed under two programming languages that are PHP with its Symfony framework 5.2 and React Native respectively.

In addition, so that all the information is legible, diagrams of each of its characteristics are made, starting with its database to its interfaces, which are developed with distinctive colors of the organization.

After carrying out the entire analysis process, we continue with the final testing stage, which in addition to guaranteeing quality, also verifies the function of the application and each of its modules.

INTRODUCCIÓN

Antecedentes

El inicio de la digitalización es un paso muy importante en una organización, esto en complemento con el gran número de soluciones tecnológicas que existen, permite satisfacer las necesidades que presenta el consorcio jurídico Orquera & Orquera. El incremento de uso celular hace que esta gama de posibilidades sea aún más amplia y que el acceso a la información sea de forma inmediata, es por eso, que la implementación de sistemas que optimicen el uso de recursos desde cualquier parte sin necesidad de estar presente en un lugar físico, es relevante en cualquier organización de este tipo.

Problema

En la actualidad, el consorcio jurídico Orquera & Orquera administra toda la información de los trámites de transferencia de dominio de sus clientes en una sola hoja de cálculo. Este sistema actual ha traído consigo grandes dificultades y se ve reflejado en un servicio y gestión deficiente ocasionando pérdida de información, falta de organización e inadecuado registro de los datos.

Adicionalmente, con el continuo crecimiento de los datos, resulta compleja la búsqueda de información, ésta puede tardar hasta más de 20 minutos y esto sin duda minimiza la productividad por parte de los funcionarios y puede llegar a generar una pérdida significativa de clientes.

En base a esta problemática, la empresa requiere implementar una forma diferente de trabajo para sus funcionarios y colaboradores, de manera que un trámite pueda ser ingresado, actualizado y monitorizado por cualquier colaborador de la organización desde cualquier ubicación en la que éste se encuentre.

Justificación

El avance tecnológico que se ha alcanzado con el desarrollo de la red global (internet), permite que las empresas puedan acceder a la información desde cualquier lugar y dispositivo. De esta forma, este estudio es relevante porque pretende aprovechar las ventajas que provee el internet, desarrollando una aplicación móvil que permita acceder a la información de los trámites desde cualquier punto geográfico, así mismo la construcción de otra aplicación web que mejore el ingreso, actualización y administración ordenada de los mismos, lo cual representa una mejora sustancial de la disponibilidad de la información y su correspondiente socialización entre los colaboradores de la organización con el fin de mitigar los efectos de la problemática previamente planteada.

Así mismo, el aplicativo web cuenta con un algoritmo que permite la asignación equitativa de trámites dependiendo de la carga laboral de cada colaborador. La automatización de todo el proceso a través del aplicativo web facilita la gestión eficiente y segura de la información de la cartera de clientes para fines estratégicos de la empresa. El sistema presenta la información relevante de manera ordenada con el propósito de ayudar a la toma de decisiones gerenciales y la adecuada gestión de la organización.

Objetivos

Objetivo general

Desarrollar una aplicación web y móvil que permita la asignación automática, gestión y monitoreo de los trámites de transferencia que administra el consorcio jurídico Orquera & Orquera

Objetivos específicos

- Levantar el proceso de la organización para el correcto seguimiento de los trámites de transferencia.
- Diseñar el comportamiento del sistema web y aplicativo móvil de manera que permita la gestión de los trámites de transferencia.
- Desarrollar un algoritmo que asigne automáticamente un trámite permitiendo una carga de trabajo equitativa.
- Gestionar y monitorear los documentos asociados a los trámites de transferencia.
- Gestionar los respectivos informes que se requieren para los clientes y para verificar el adecuado funcionamiento de la organización.

Metodología

Se utiliza las siguientes metodologías para la implementación de la aplicación web:

Scrum es la metodología de la gestión del proyecto de desarrollo de la aplicación, debido a que es un marco de trabajo, que facilita la interacción con el cliente de forma frecuente y, además, posibilita la determinación de plazos de entrega de los diferentes avances del sistema (resultados de cada Sprint) (CertiProf, 2019).

Se utiliza UWE, para la aplicación de la Ingeniería de Software e Ingeniería Web en el desarrollo de la aplicación Web, pues proporciona una serie de elementos (diagramas) necesarios para plasmar el diseño, implementación y funcionamiento de la aplicación (UML-Based Web Engineering, 2016).

CAPÍTULO I

MARCO TEÓRICO

1.1. Consorcio jurídico Orquera & Orquera

El consorcio jurídico Orquera & Orquera, se encuentra ubicado en la ciudad de Quito, teniendo una presencia importante de seis años en el mercado. La empresa requiere mejorar la ejecución de sus actividades con el objetivo de utilizar las tecnologías de la información para su beneficio. Con la finalidad de eliminar las actuales deficiencias que se presentan en la gestión actual de la organización y de su información, las mismas que causan problemas de seguimiento de los trámites gestionados y por gestionar en su cartera de clientes.

La organización administra trámites de transferencia de dominio en sus diferentes tipos (Tabla 1).

Tabla 1.

Tipos de trámites de transferencia de dominio

Trámites de transferencia de dominio		
Compra y venta	Donación	Adjudicación
Compra y venta con hipoteca	Donación con reserva de usufructo	
Cancelación de hipoteca con compra y venta e hipoteca abierta		

Nota: Se muestra la clasificación de trámites de transferencia de dominio

Elaborado por: Los autores

1.2. Sistemas de Información

A lo largo de los años se ha propuesto varias definiciones de lo que es un sistema de información, sin embargo, la que plantea Andreu et al., (1991), es la que más se asemeja hoy en día, en la cual indica que un sistema de información se define como: “conjunto de procesos que permite satisfacer las necesidades del usuario, a través de la recopilación, almacenamiento y distribución de la información presentada de forma precisa y ordenada para mejorar la estrategia de una empresa y ayudar a la toma de decisiones” (pág. 149).

1.2.1. Entrada

Se refiere a toda aquella actividad que se encarga de captar los datos requeridos para poder procesarlos, estos datos se obtienen mediante formularios, o por registros que se ingresaron con anterioridad (Stair & Reynolds, 2010).

1.2.2. Procesamiento

El procesamiento consiste en la realización de cálculos para obtener datos críticos que servirán para que la empresa se enfoque en el cumplimiento de sus objetivos. Estos datos son útiles y esenciales para el negocio, por que proporcionan una forma de identificar posibles mejoras (Stair & Reynolds, 2010).

1.2.3. Salida

Es toda la información resultante del procesamiento, este contiene data importante para la toma de decisiones (Stair & Reynolds, 2010).

1.3. Scrum

Es un marco de trabajo que es usado en cualquier ámbito en donde se plantean proyectos, se ejecuta de manera incremental y se centra en la entrega de avances en cada sprint (CertiProf, 2019).

1.3.1. Product Owner

El Product Owner (PO) es el encargado de asegurar el valor del producto y mantiene un amplio conocimiento del negocio. Él permite la comunicación efectiva entre los requerimientos de los Stakeholders y el funcionamiento del equipo de desarrollo (CertiProf, 2019).

1.3.2. Scrum Máster

Tiene por objetivo facilitar la comprensión de las interacciones dentro de scrum team, además que garantiza la ejecución de cada uno de los roles dentro del mismo (CertiProf, 2019).

1.3.3. Development Team

Son profesionales que se encargan de realizar mejoras incrementales del producto, un incremento se puede denominar “Sprint” y posteriormente puede ser lanzado a producción. El equipo requiere de un conjunto de tareas que serán completadas según lo establecido. Todas estas tareas son obligatorias y permite que los desarrolladores puedan adaptarse a las necesidades cambiantes del usuario (CertiProf, 2019).

1.4. Sprint

Se le llama así a cada iteración, en la cual se presenta un incremento, se le define un lapso el cual no debe ser mayor a un mes (CertiProf, 2019). En el caso de este proyecto se define la siguiente estructura con respecto a los integrantes:

Tabla 2.

Roles de scrum

Miembro	Roles de Scrum
David Cadena y Cristina Rodríguez	Development Team
Orquera & Orquera	StakeHolders
David Cadena y Cristina Rodríguez	Product Owner

Nota: Designación de roles por integrante equipo

Elaborado por: Los autores

1.5. UWE

Es un método que mediante diferentes diagramas permite el análisis y diseño de una aplicación web, este está basado bajo el lenguaje de modelado unificado, simplificando la implementación del código de software (TechTerms, 2009).

1.6. Trámite

Es un conjunto de diligencias, que son consideradas como requisitos para llevar a cabo un procedimiento judicial, estos son otorgados por los organismos gubernamentales pertinentes (Suarez, 1992).

1.7. Symfony

Es un framework de PHP, el cual no solo permite autogenerar código mediante consola, también permite generar la base de datos completa con el uso de migraciones, es bastante versátil y puede ser usado en el backend de un proyecto en la creación de APIs (Symfony, 2021).

1.8. React Native

Es un framework de javascript que permite desarrollar aplicaciones nativas, como Android y iOS, sin dejar de lado la importancia de la experiencia de usuario (Facebook, 2020).

1.9. Bootstrap

Es un conjunto de herramientas de código abierto para el diseño de interfaces, su uso es muy importante para la adaptación en diferentes medidas de pantalla y dispositivos desde los cuales se puede tener acceso el sistema (Bootstrap, s.f.).

CAPÍTULO II

REQUERIMIENTOS Y ANÁLISIS

La base de la ingeniería de software se centra en el levantamiento de los requerimientos y en el modelado de diagramas que se usan para mejorar la comprensión del sistema. Como consecuencia, para desarrollar una solución óptima es relevante el análisis del comportamiento actual que se requiere optimizar, a través de un diagrama de proceso que permite recopilar requerimientos con mayor detalle, los mismos que sirven de soporte para la realización de diagramas de contexto, casos de uso, datos, clases, actividades, secuencia y diseño de la aplicación.

2.1. Proceso de almacenamiento de trámites de transferencia de dominio

Para entender la manera en que se trabaja y de esta forma obtener requerimientos más profundos, se levanta el proceso que maneja el consorcio para el almacenamiento de los archivos de los trámites de transferencia de dominio encontrando así, cinco actores: funcionario, municipio, cliente, notaría y registro de la propiedad (Figura 1).

2.2. Levantamiento de requerimientos

Para el inicio del desarrollo se establece contacto con el cliente, el cual provee de información para ejecutar la reingeniería de estas peticiones, como resultado se obtiene requerimientos funcionales, de sistema y técnicos.

2.2.1. Listado de peticiones en lenguaje común

- El usuario registra, actualiza, elimina clientes y trámites en el sistema.
- Los usuarios bajo el rol de administrador y colaborador son capaces de administrar el proceso del trámite.
- El usuario realiza el registro de clientes, trámites y asigna el trámite al cliente
- Solo el usuario con el rol de administrador puede ingresar nuevos abogados al sistema
- El sistema presenta informes detallados en base a la data ingresada
- La aplicación móvil permite acceder al estado de los trámites.
- Los trámites ingresados se pueden reasignar dependiendo a las funciones de cada colaborador.

2.3. Requerimientos

2.3.1. Requerimientos funcionales

- El sistema asigna automáticamente un abogado a un trámite.
- El sistema guarda todos los datos acerca del flujo del trámite desde el inicio hasta su fin.
- Los usuarios son registrados en el sistema bajo un determinado rol que restringe sus funciones en él.

- El sistema muestra una tabla con información acerca del talento humano (usuarios del sistema) y todos los trámites, para los usuarios que estén bajo el rol de administrador.
- El sistema permite generar y descargar informes con datos relevantes para la toma de decisiones, los mismos que están disponibles para los administradores.
- El sistema permite que todos los usuarios puedan acceder al registro, actualización de los datos de clientes y trámites.
- Para la reasignación de un trámite se toma en cuenta las funciones que el usuario disponga.
- La información histórica acerca del manejo del trámite está disponible para su debido seguimiento.

2.3.2. *Requerimiento de sistema y técnicos*

- La aplicación móvil está disponible para el sistema operativo Android.
- El sistema proporciona errores que sean informativos y orientados al usuario final.
- El sistema es accesible desde cualquier navegador web.
- El sistema encripta la información de contraseñas, con el fin de mantener la integridad de los datos.
- Los accesos al sistema son gestionados únicamente por el administrador, quien designa al usuario su respectivo rol dependiendo de las funciones de este.

2.4. Historias de usuario

La tabla muestra las historias de usuario, las mismas que están enfocadas en definir el rol, la funcionalidad y el resultado esperado (objetivo) del sistema. Es muy común aplicar las historias de usuario en marcos ágiles, porque esta representación permite describir los requerimientos del usuario en un lenguaje no técnico (UML-Based Web Engineering, 2016). En la tabla 3 se visualiza el resultado.

Tabla 3.

Historias de usuario

Usuario	Funcionalidad	Objetivo
● Administrador	Crear usuario	Llevar un control de todos los miembros del consorcio.
● Administrador	Asignar roles	Controlar el acceso de los usuarios en el sistema.
● Administrador	Descargar reportes	Tener a disponibilidad los reportes para la toma de decisiones.
● Administrador	Asignar funcionalidades	Otorgar a cada usuario funcionalidades en los trámites.
● Administrador - Colaborador	Crear un tipo de trámite	Gestionar el trámite del cliente
● Administrador - Colaborador	Crear cliente	Administrar los clientes que han solicitado el servicio en el consorcio.
● Administrador – Colaborador	Crear trámite	Verificar el estado del trámite y dar un seguimiento al mismo.
● Abogado - Colaborador	Realizar búsquedas	Filtrar únicamente los registros de los trámites que el usuario necesita en el aplicativo móvil.

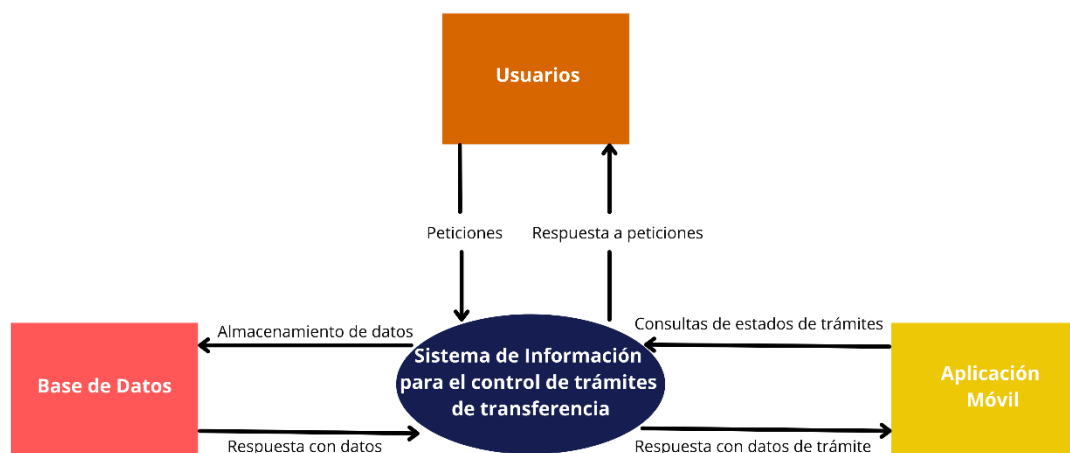
*Nota: Descripción de historias de usuario**Elaborado por: Los autores*

2.5. Diagrama de contexto

Este diagrama muestra las entidades y la manera en que se relacionan con el sistema como la base de datos, la cual almacena la información necesaria a cargar en las vistas, los usuarios que mediante las interfaces pueden ingresar y solicitar información. Por último, está la aplicación móvil, como se muestra en la figura 2.

Figura 2.

Diagrama de contexto



Nota: Análisis del contexto del sistema y los componentes con los que se relaciona

Elaborador por: Los autores

2.6. Diagramas de caso de uso

2.6.1. Lista de actores

El diagrama de casos de uso describe claramente la forma en que un usuario interactúa con el sistema mediante las acciones que puede ejecutar en él.

Permite identificar los roles, actores y funciones del software. Estos actores pueden ser entidades, sistemas e incluso usuarios que están involucrados en el proceso, tal como se muestra a continuación.

Tabla 4.

Actores del sistema

#	Actor	Descripción
1	Administrador (del sistema)	Está encargado de: <ul style="list-style-type: none">● Manejo total del sistema es decir tiene acceso a todas las funcionalidades del sistema● Descargar reportes● Crear nuevos usuarios(abogados), funciones, tipos de trámites
2	Colaborador	Está encargado de: <ul style="list-style-type: none">● Crear un nuevo cliente● Registrar y gestionar trámites de transferencia

Nota: Descripción de actores del sistema

Elaborado por: Los autores

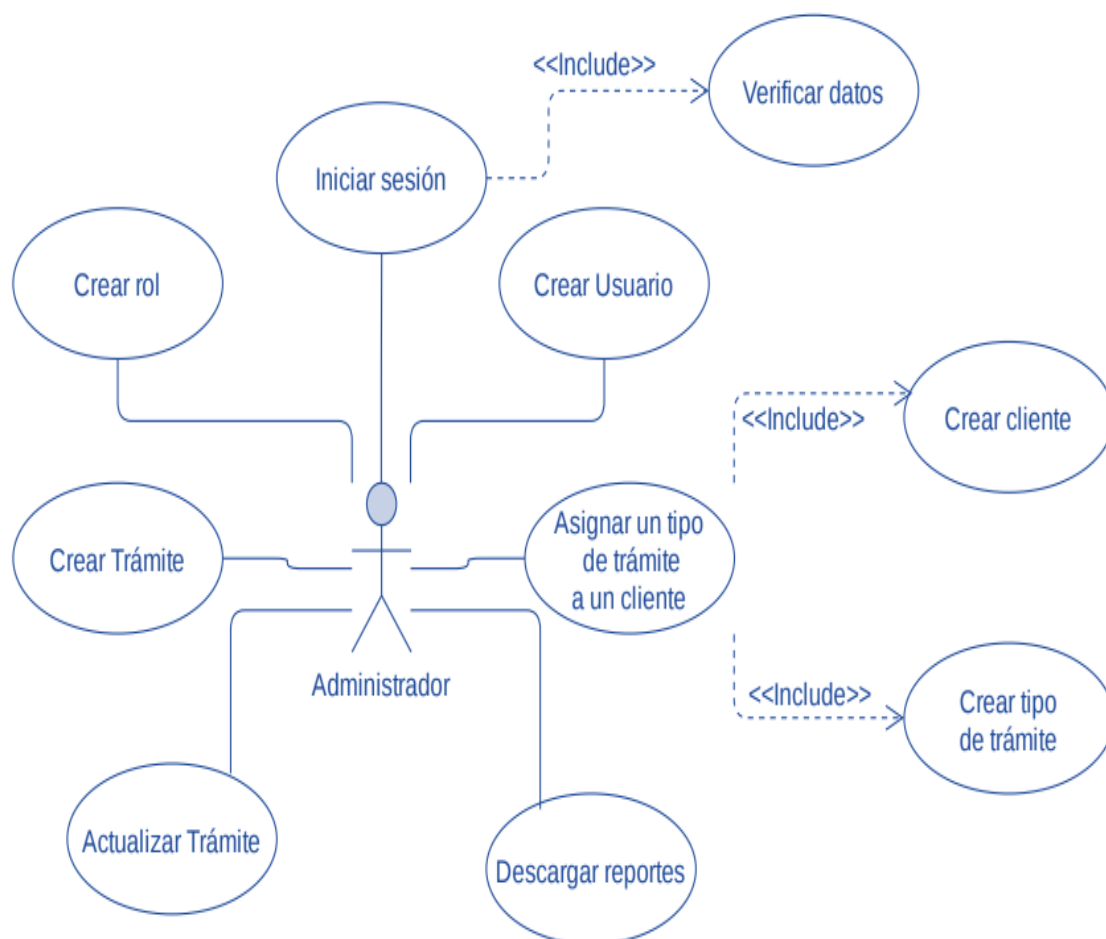
2.6.2. Diagrama de caso de uso de administrador

El sistema proporciona un rol administrador el cual es responsable de gestionar los reportes, las cuentas de usuarios y las funciones.

Todas las funciones que este actor es capaz de realizar en el sistema web se puede evidenciar mediante el diagrama de casos de uso que se muestra en la figura 3.

Figura 3.

Diagrama de caso de uso de administrador



Nota: Diagrama caso de uso del abogado administrador de sistema

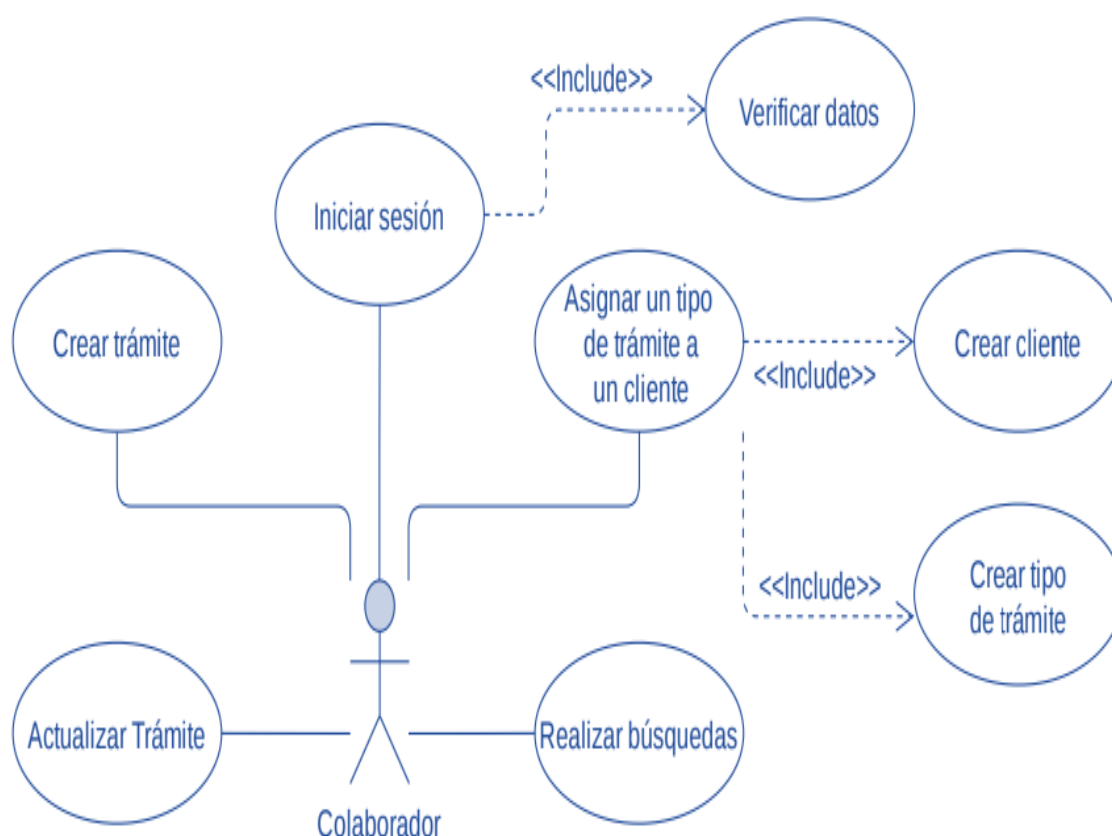
Elaborado por: Los autores

2.6.3. Diagrama de caso de uso de colaborador

Cada colaborador es responsable de gestionar los trámites asignados por el sistema. Todas las funciones que este actor es capaz de realizar en el sistema web se puede evidenciar mediante el diagrama de caso de uso (Figura 4).

Figura 4.

Diagrama de caso de uso de colaborador



Nota: Diagrama de caso de uso del abogado colaborador del sistema

Elaborado por: Los autores

2.7. Diagrama de secuencia

El diagrama de secuencia es usado para describir la interacción entre los objetos del sistema y la forma en como estos trabajan juntos. Es muy conocido en el mundo de la tecnología porque permite entender los requerimientos de un nuevo sistema y ayuda a documentar el proceso existente de la organización.

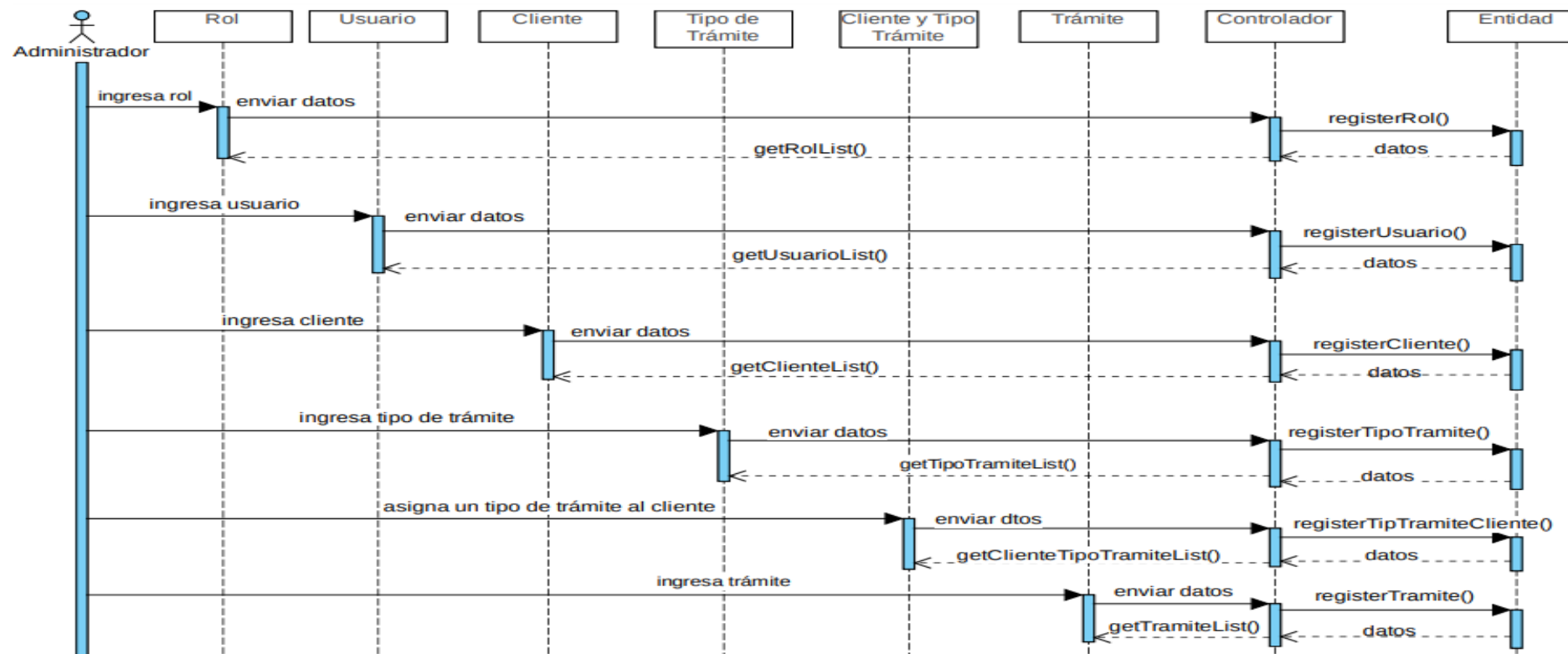
Se presenta un diagrama de secuencia para cada actor del sistema, Administrador y Colaborador.

2.7.1. Diagrama de Secuencia del Administrador

Esta representación es la interacción entre el administrador y los diferentes componentes del sistema web, vista, controlador y entidad. Como se muestra en la figura 5, el administrador crea un usuario asignándole un rol, posteriormente crea un cliente, un tipo de trámite y asigna un tipo de trámite al cliente. Finalmente, el administrador ingresa toda la información referente al proceso del trámite.

Figura 5.

Diagrama de secuencia de administrador del sistema



Nota: Diagrama de secuencia del abogado administrador del sistema

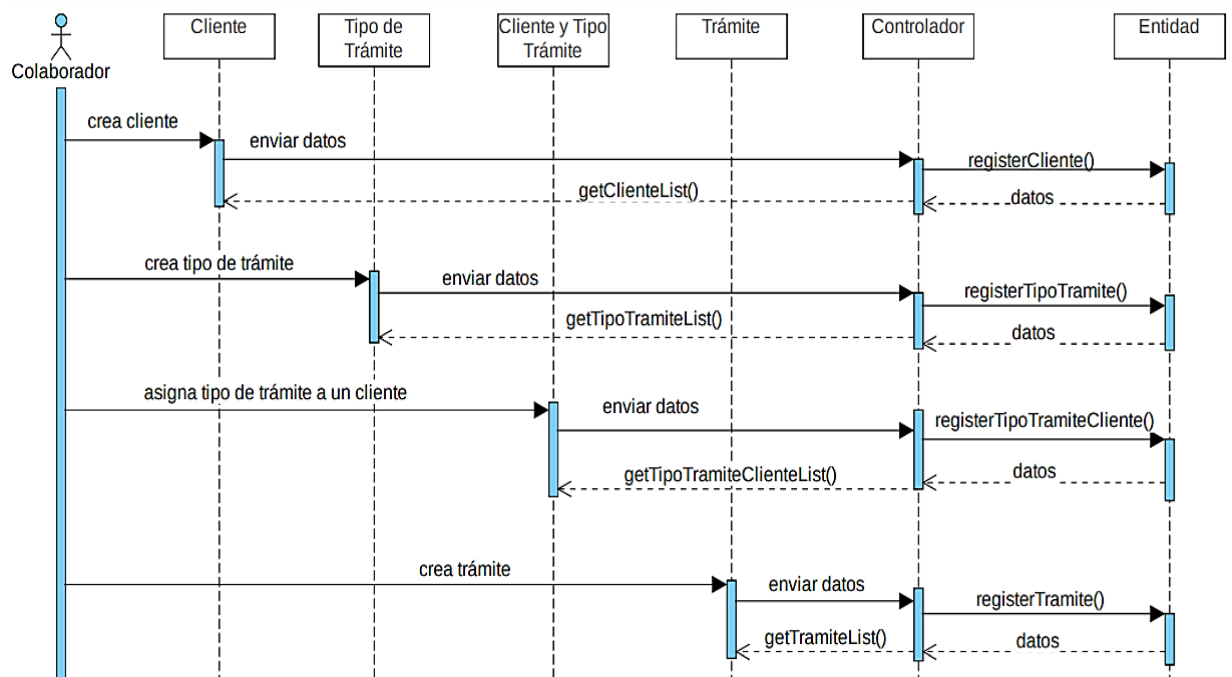
Elaborado por: Los autores

2.7.2. Diagrama de Secuencia del Colaborador

Esta representación es la interacción entre el colaborador y los diferentes componentes del sistema web, vista, controlador y entidad. Como se muestra en la figura 6, el colaborador crea un cliente, luego genera un tipo de trámite para que posteriormente asigne un cliente a un tipo de trámite. Finalmente, el colaborador ingresa toda la información referente al proceso del trámite.

Figura 6.

Diagrama de secuencia del colaborador



Nota: Diagrama de secuencia del usuario bajo el rol de abogado colaborador

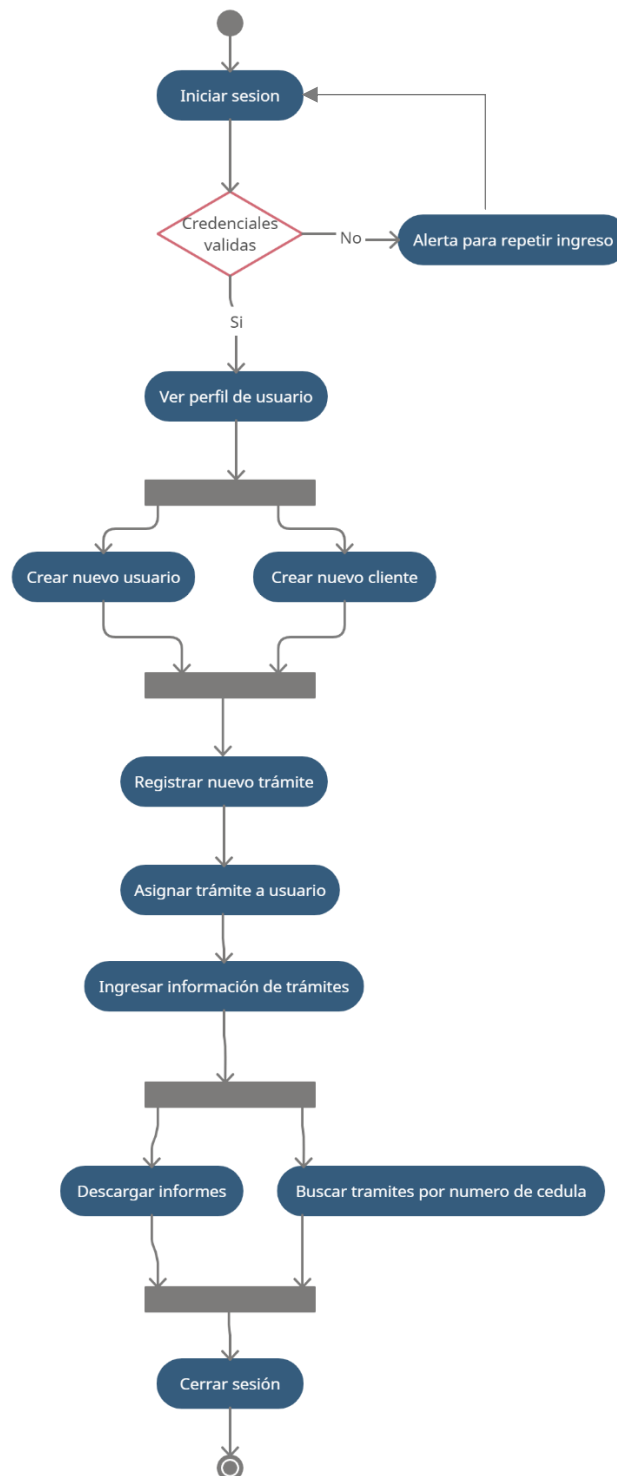
Elaborado por: Los autores

2.8. Diagrama de actividades

Este diagrama replica el comportamiento del sistema, el cual se describe bajo el flujo de las actividades, a la par se muestra un proceso que sigue el usuario al interactuar con el sistema (Figura 7).

Figura 7.

Diagrama de actividades del sistema



Nota: Diagrama de actividades del sistema

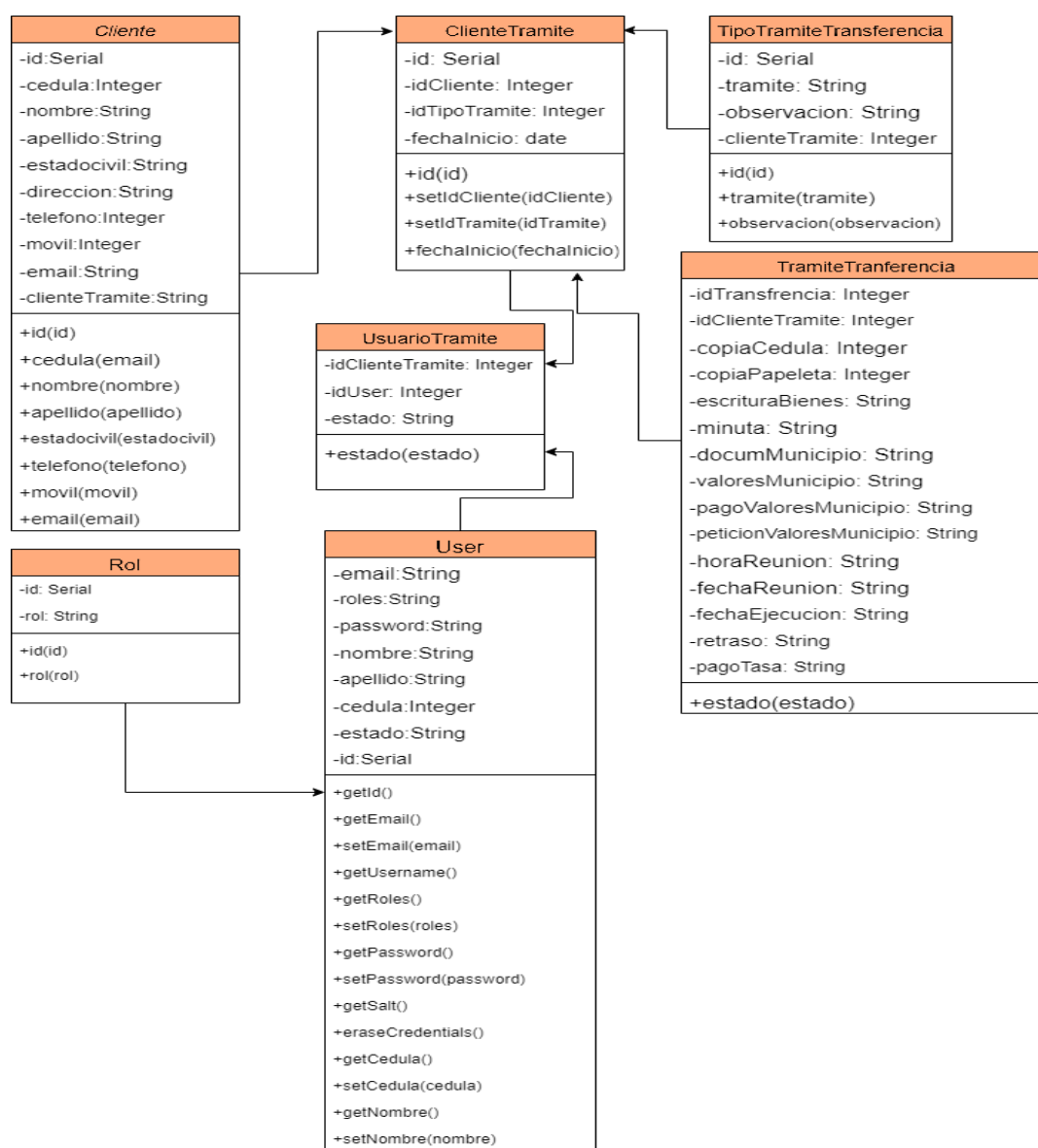
Elaborador por: Los autores

2.9. Diagrama de clases

A continuación, se observa la estructura estática del software en donde se muestra los objetos (clases) a nivel de entidades y la interacción entre ellos representados en un diagrama (Figura 8).

Figura 8.

Diagrama de clases



Nota: Diagrama de clases del sistema

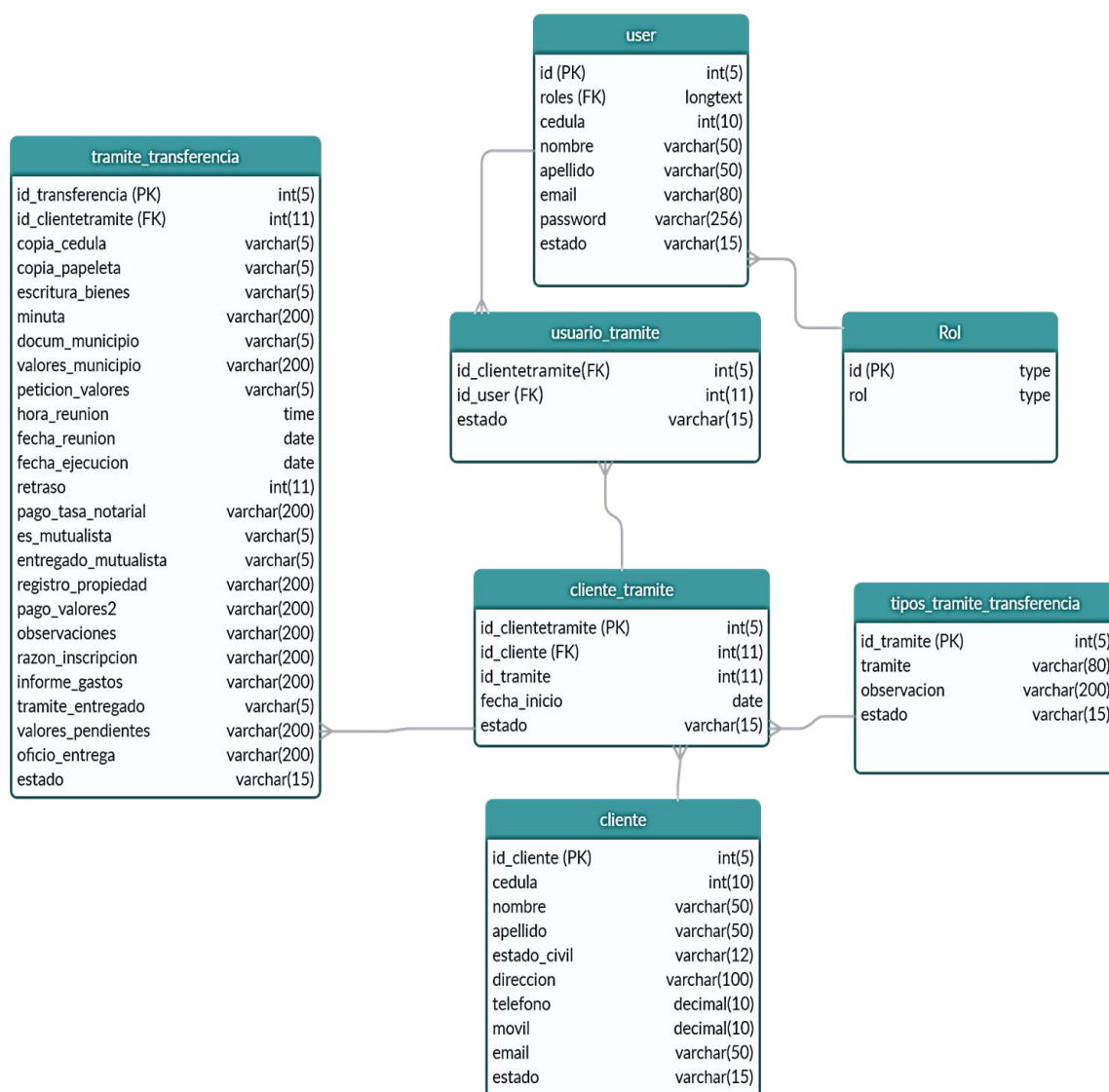
Elaborado por: Los autores

2.10. Diagrama de datos

La composición de los datos representados en un diagrama entidad-relación, el mismo que agrupa un conjunto de tablas, en las cuales se visualiza sus características tales como tipo y longitud (Figura 9).

Figura 9.

Diagrama de datos



Nota: Diagrama de datos

Elaborado por: Los autores

CAPÍTULO III

DISEÑO

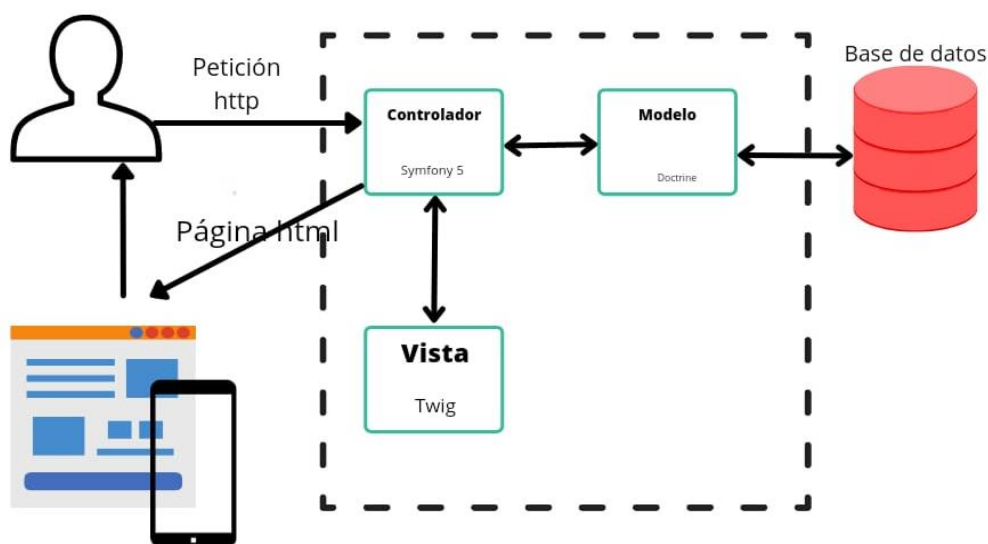
En el capítulo anterior se obtiene información importante que permite visualizar el funcionamiento del sistema, en este se muestra su diseño no solo la arquitectura del software sino también un pequeño avance de cómo se ve una vez que se termina su desarrollo.

3.1. Arquitectura de software

La arquitectura permite mantener un orden estandarizado de los archivos del software de forma que la comunicación entre los datos y la aplicación sea más eficiente, por esta razón se usa MVC como modelo arquitectónico.

Figura 10.

Arquitectura de software



Nota: Arquitectura de software

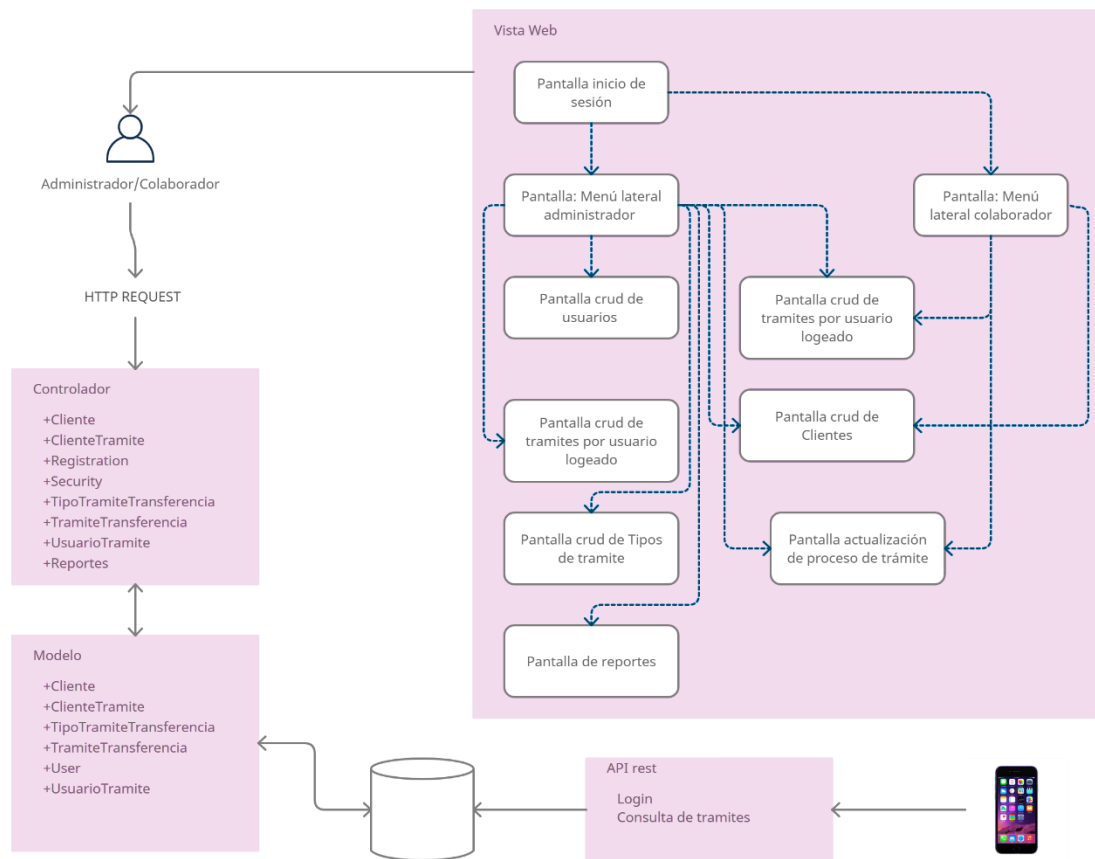
Elaborado por: Los autores

3.2. Diagrama de paquetes

Aunque se define el modelo arquitectónico, se ve como es el flujo de los datos para tener más claro el panorama, el diagrama tiene la misma división que la arquitectura.

Figura 11.

Diagrama de paquetes



Nota: Diagrama de paquetes

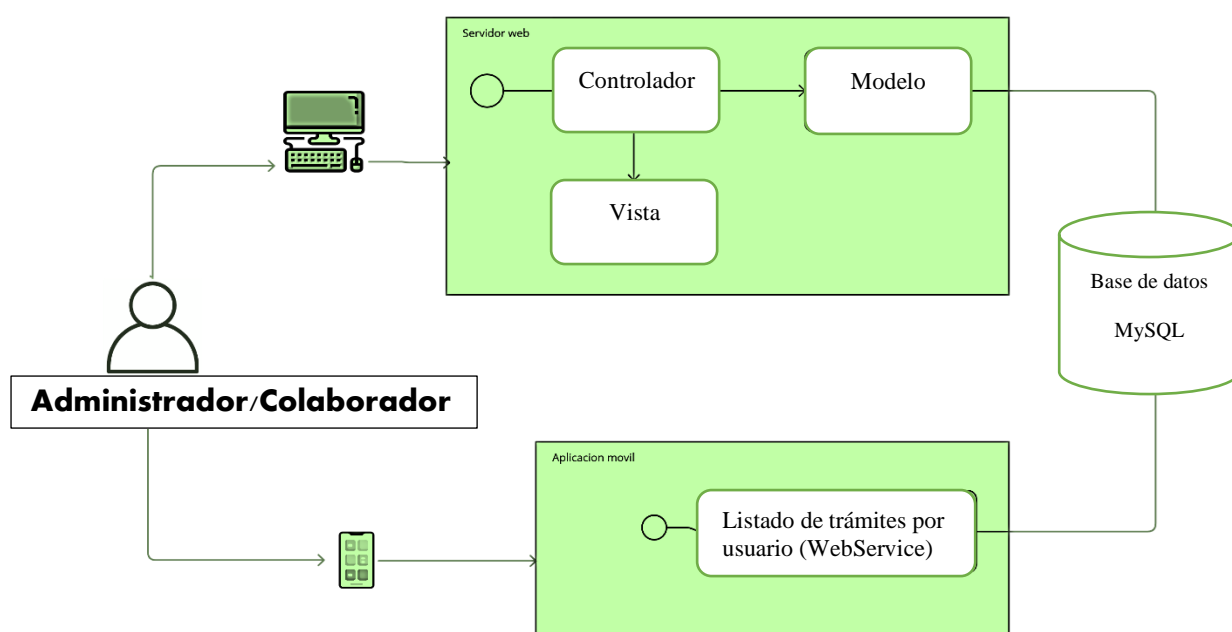
Elaborado por: Los autores

3.3. Diagrama de componentes

En los componentes del sistema se muestra el entorno y como se relaciona y comunica entre todas sus partes, en este caso en particular se tiene una aplicación móvil, la cual consume un api que proviene del aplicativo web, el mismo que, extrae los datos de la base, tal como diagrama la figura 12.

Figura 12.

Diagrama de componentes



Nota: Diagrama de componentes

Elaborado por: Los autores

3.4. Diseño de algoritmo de asignación de trámites

Una parte distintiva de este software se centra en la asignación de trámites aleatoriamente. Para hacer esto posible, se coloca a cada tipo de trámite diferentes pesos con referencia a su carga de trabajo y al tiempo, como se muestra a continuación

Figura 13.

Pesos de trámites

Tipo de tramite	Carga	Peso
Compra y venta	1	1
Compra y venta con hipoteca	2	2
Cancelación de hipoteca con compra y venta e hipoteca	3	2
Donación	5	2
Donación con reserva de usufructo	5	2
Adjudicación	4	3

Nota: Pesos de trámites

Elaborador por. Los autores

Conociendo cada uno de sus pesos, se procede al cálculo el cual tiene los siguientes pasos:

- Obtener el número de trámites que tienen asignados cada usuario.
- Distinguir el número de trámites asignados a un usuario por tipo.
- Luego se multiplica el número de cada peso por el número por tipo de trámite y se suma.

De esta forma, se puede desarrollar el algoritmo permitiendo que este se ejecute cada vez que se crea un trámite, este método elimina la excesiva carga de trabajo hacia un solo colaborador.

3.5. Diseño de interfaces

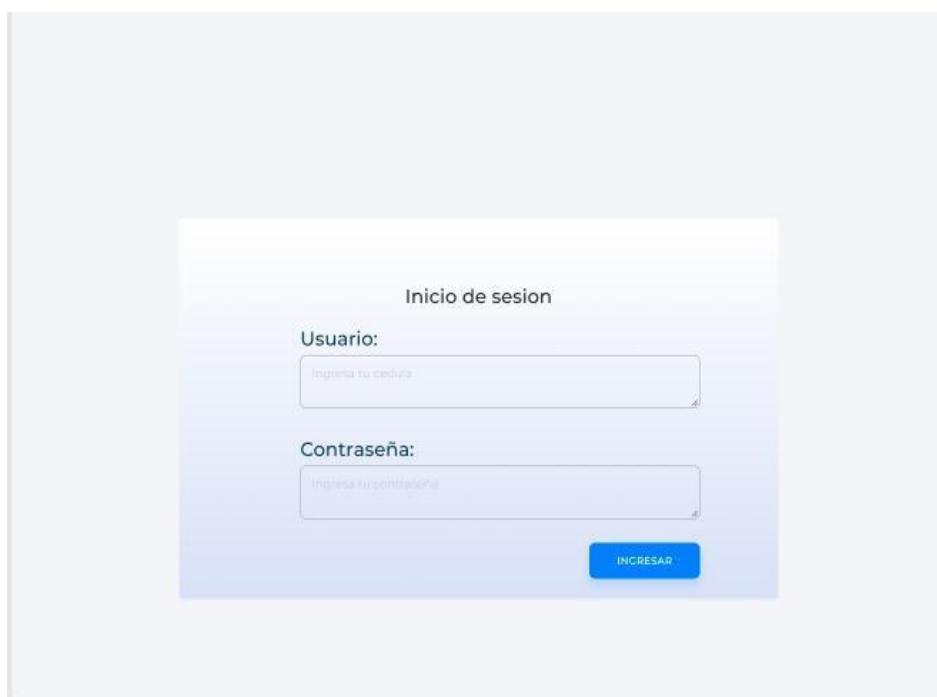
Ya se tiene la diagramación interna de la aplicación. En este segmento se detalla cómo se ven las interfaces, las cuales sirven para que el usuario interactúe con el software, y a continuación, se presenta el prototipo general en donde se muestra la vista de inicio de sesión, el formulario de ingreso, la lista y el menú lateral.

3.5.1. Prototipo de interfaz de inicio de sesión

Para el inicio de sesión el usuario ingresa sus credenciales, el correo electrónico y la contraseña respectivamente, esta interfaz también muestra mensajes en caso de que un email no este ingresado en la base y cuando la contraseña no sea correcta.

Figura 14.

Pantalla de inicio de sesión

El prototipo muestra una interfaz de inicio de sesión centrada en la pantalla. El título "Inicio de sesión" está en la parte superior. Debajo, hay dos campos de entrada: "Usuario:" con el placeholder "Ingresa tu cédula" y "Contraseña:" con el placeholder "Ingresa tu contraseña". A la derecha de cada campo hay un ícono de lupa. En la parte inferior derecha, hay un botón azul con el texto "INGRESAR".

Nota: Pantalla de inicio de sesión

Elaborado por: Los autores

3.5.2. Prototipo de interfaz de ingreso de datos

Cada entidad que se está registrando cuenta con sus propios campos, pero este prototipo embebe las características necesarias en cuanto a colores, tipo de letra y ubicación de cajas de texto.

Figura 15.

Pantalla de ingreso

Registro de nuevo funcionario

Cedula:

Nombres:

Apellidos:

Correo:

Cargo:

Contraseña:

Nota: Pantalla general de ingreso

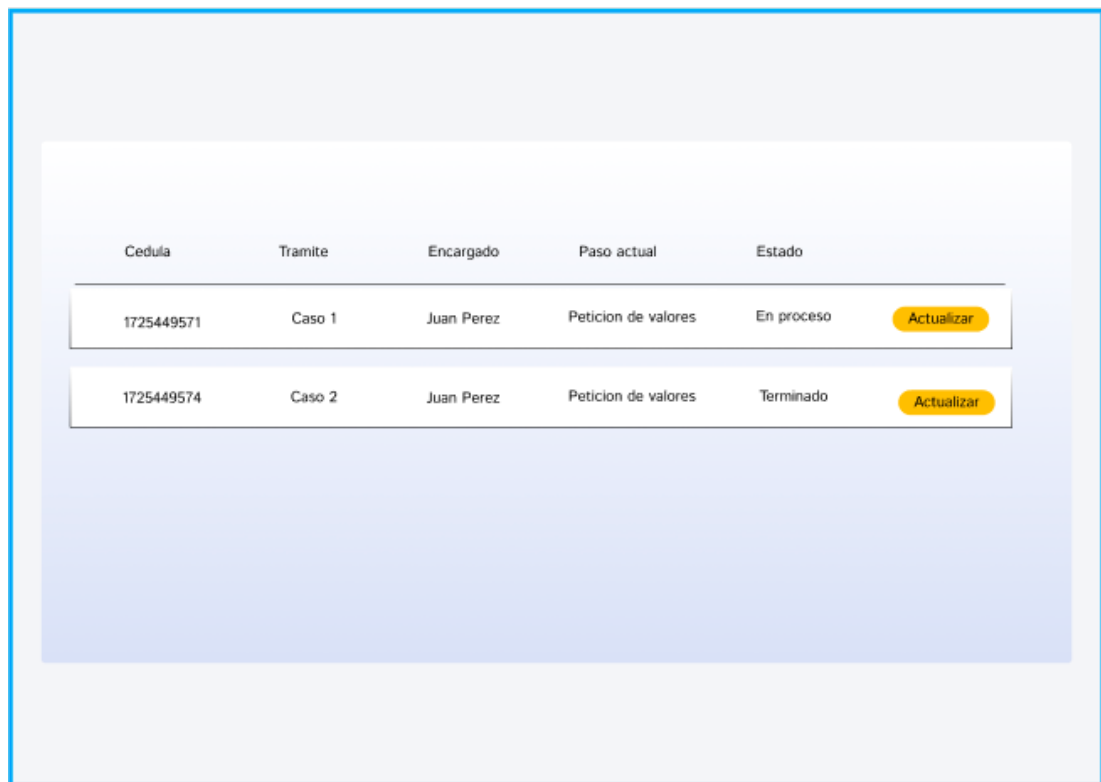
Elaborado por: Los autores

3.5.3. Prototipo de interfaz de listado de datos

Después de iniciar sesión esta pantalla es la primera vista con la que se puede interactuar, las listas tienen este diseño, en donde se muestra data importante acerca del elemento listado y un botón que permite la acción señalada en el mismo.

Figura 16.

Pantalla de lista



Cedula	Tramite	Encargado	Paso actual	Estado	
1725449571	Caso 1	Juan Perez	Peticion de valores	En proceso	Actualizar
1725449574	Caso 2	Juan Perez	Peticion de valores	Terminado	Actualizar

Nota: Pantalla de lista

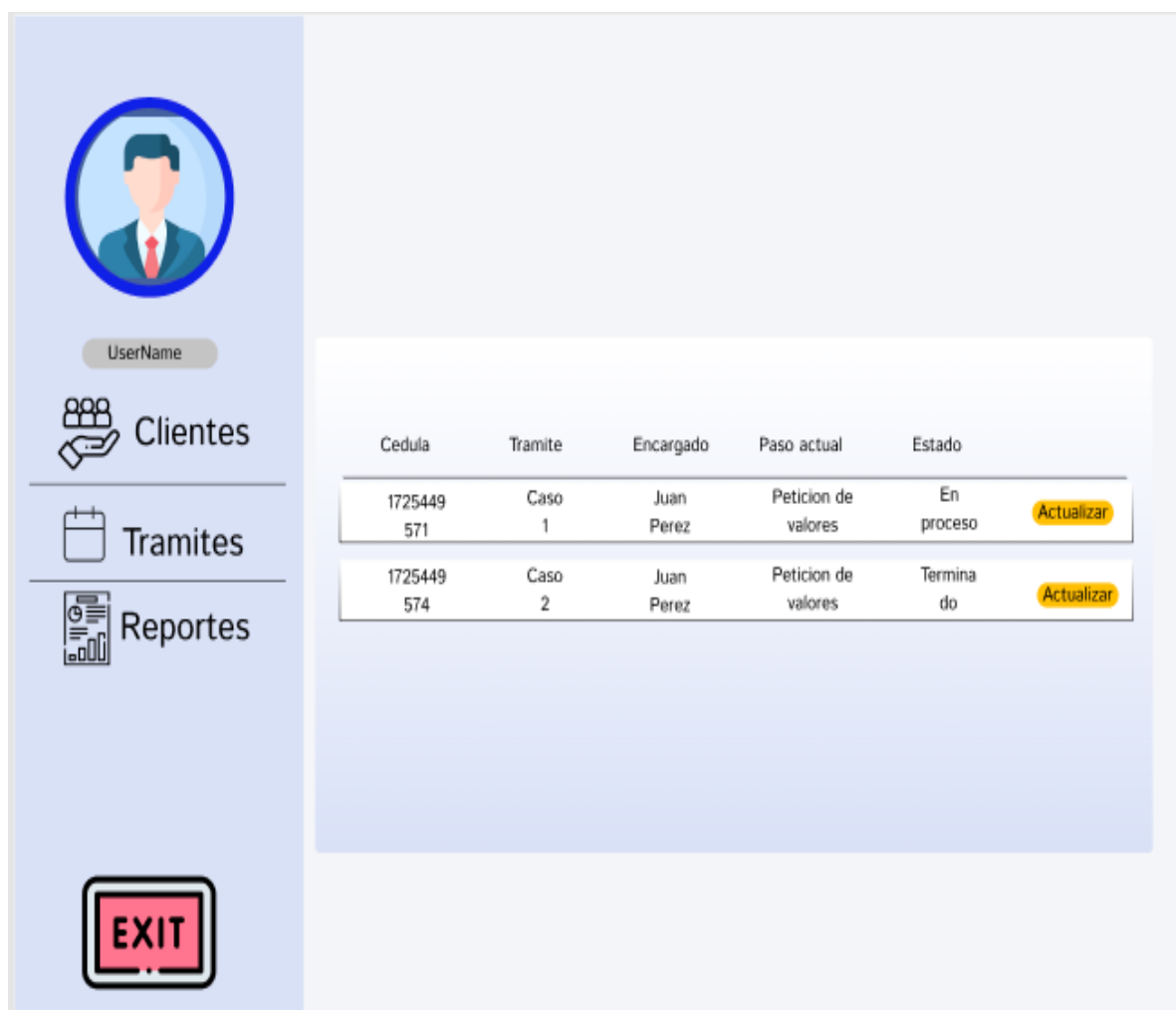
Elaborado por: Los autores

3.5.4. Prototipo de interfaz de menú lateral

El menú lateral está disponible en todas las vistas, así como, los enlaces a las interfaces dependiendo el rol.

Figura 17.

Pantalla de menú lateral



Nota: Pantalla de menú lateral

Elaborado por: Los autores

En el aplicativo móvil, la consulta de trámites se presenta de manera simple con el fin de que para los administradores y colaboradores se puede visualizar los trámites que se les ha asignado desde cualquier parte sin tener que ingresar al aplicativo web.

Figura 18.

Pantalla móvil de lista



Nota: Pantalla móvil de lista

Elaborado por: Los autores

El consorcio jurídico ha otorgado total libertad en cuanto a vistas, sin embargo, permite conocer los colores organizacionales y su logo, los cuales son usados de forma pertinente en las vistas internas del sistema, por otro lado, al ser las imágenes que se muestra solo prototipos, estas sirven de guía.

CAPÍTULO IV

CODIFICACIÓN

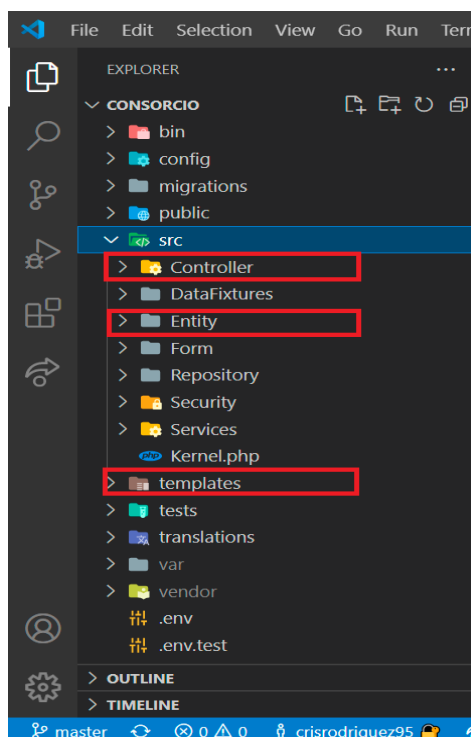
En este capítulo se implementa el análisis y diseño que son preparados en los capítulos anteriores.

4.1. Configuración de entorno de desarrollo

Para el desarrollo del aplicativo web se usa Symfony en su versión 5.2, el cual facilita la estructura de modelo, vista, controlador, para la creación del proyecto se necesita “Composer” y en este caso se codifica en el editor de texto “Visual Studio Code”, en donde al abrir el proyecto muestra el orden de las carpetas denotando el MVC.

Figura 19.

Estructura de carpetas en vs code



Nota: Estructura de carpetas en vs code

Elaborado por: Los autores

Con respecto al aplicativo móvil se usa React native para desarrollarlo, las API's también son construidas en Symfony para centralizar el consumo de datos debido a que el número de API's es pequeño como para separarlas del código central.

4.2. Interfaz

El uso de una plantilla de bootstrap que permite desarrollar las interfaces rápidamente, usar componentes de ésta es la propuesta aceptada por los sponsors, la misma que está presente en todo aplicativo web con los diferentes cambios necesarios para la debida personalización del sistema.

4.3. Funcionalidades del sistema clasificado por clases

En esta sección se describe las clases que son parte de la funcionalidad de todo el software excluyendo todas las carpetas autogeneradas por el framework.

Las funcionalidades son realizadas de acuerdo con cada una de las entidades que se encuentran en la base de datos.

Tabla 5.

Descripción de clases

Nombre de la clase	Descripción
Cliente	Esta clase concentra todo el crud de cliente, es decir, su ingreso, eliminación, actualización y visualización
ClienteTramite	Esta clase tiene por funcionalidad la asignación de tipo de trámite a un cliente en específico y con el uso del algoritmo se asigna automáticamente a un usuario dependiendo de la carga y del tiempo.
Registration	Esta clase está encargada del registro, visualización, actualización de un usuario nuevo.
Reportes	Se realiza la generación de reportes, que son descargados por los administradores
Security	Esta clase está dedicada a establecer la seguridad de la aplicación en el login y logout en donde se hace la comprobación de contraseña, también se hace las API's para el consumo en el aplicativo móvil
TipoTramiteTransferencia	Está encargada al crud de los tipos de trámites de transferencia que se disponen
Tramite	Se refiere a toda la actualización del proceso del trámite por parte de cada usuario asignado
UsuarioTramite	Se obtiene los trámites por usuario

*Nota: Descripción de funcionalidades por clases del sistema**Elaborado por: Los autores*

4.4. API's del sistema

Debido a que el producto del proyecto no solo es un aplicativo web, para la versión móvil se usa dos webs services los cuales, son vitales para el consumo de datos, el desarrollo de estos es centralizado en la versión web.

Tabla 6.

Descripción de las API's

Nombre de servicio	Descripción
login_api	Aprovechando los métodos que se encargan de realizar el login en la clase Security se desarrolla el primer servicio para que cada usuario pueda iniciar sesión
consulta_api	Este servicio provee la búsqueda filtrada por diferentes parámetros

Nota: Descripción de las API's

Elaborado por: Los autores

4.5. Codificación

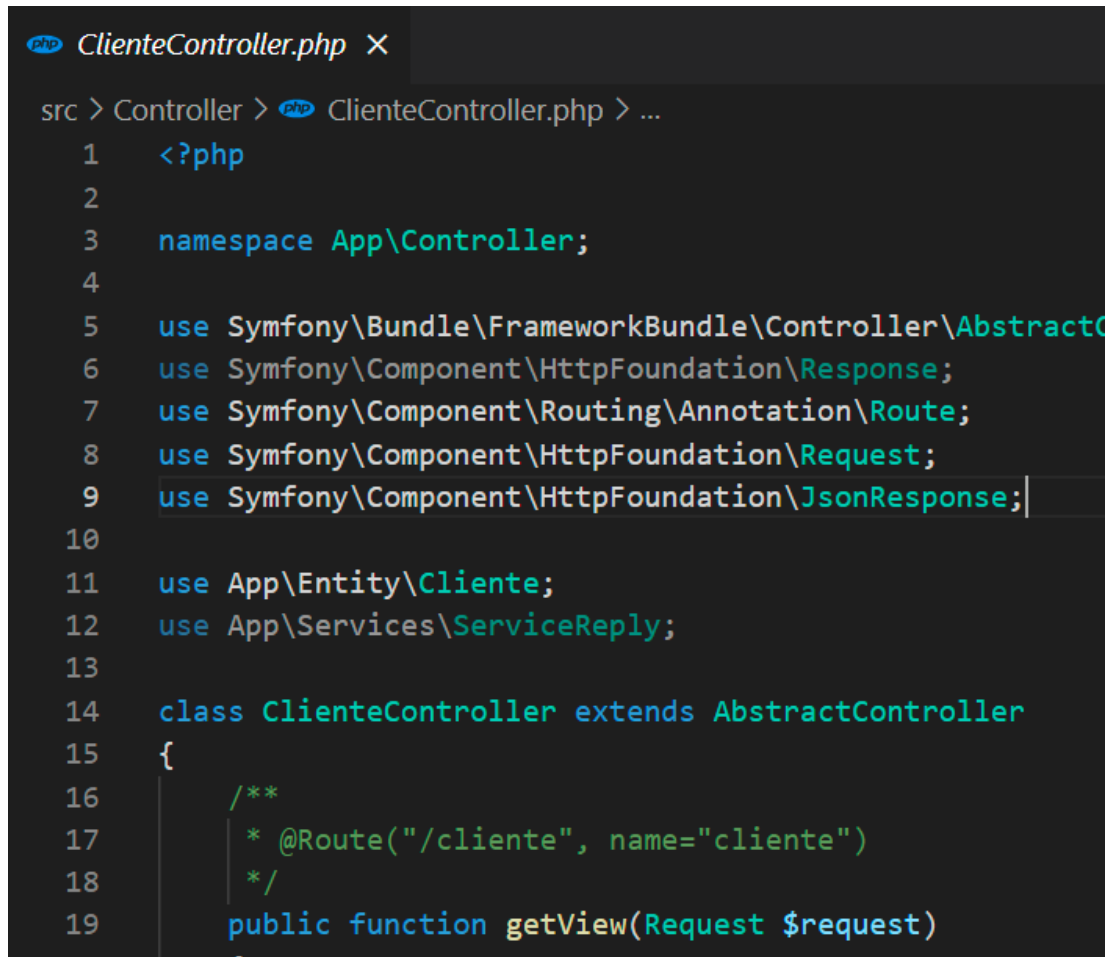
4.5.1. Controlador

El uso de un framework para el desarrollo facilita la estandarización de la estructura de los archivos, por ejemplo, para realizar una funcionalidad en el controlador, esté ya viene debidamente estructurado una vez que es creado mediante la terminal, se puede observar en la primera línea la etiqueta de PHP, siguiente de ésta se encuentra “namespace App\Controller;”, en los controladores las siguientes líneas corresponden a la importación de librerías y entidades necesarias para trabajar. Existe más de un tipo de controlador en este caso son de tipo “AbstractController”, para usarlos se escribe “extends” una palabra reservada. Las funciones corresponden a una

ruta en específico el cual se especifica en la parte superior antes de ser declarada como se muestra en la línea 16 a la 18.

Figura 20.

Controlador



```
src > Controller > ClienteController.php > ...
1  <?php
2
3  namespace App\Controller;
4
5  use Symfony\Bundle\FrameworkBundle\Controller\AbstractController;
6  use Symfony\Component\HttpFoundation\Response;
7  use Symfony\Component\Routing\Annotation\Route;
8  use Symfony\Component\HttpFoundation\Request;
9  use Symfony\Component\HttpFoundation\JsonResponse;
10
11 use App\Entity\Cliente;
12 use App\Services\ServiceReply;
13
14 class ClienteController extends AbstractController
15 {
16     /**
17      * @Route("/cliente", name="cliente")
18      */
19     public function getView(Request $request)
20     {
```

Nota: Estructura de un controlador

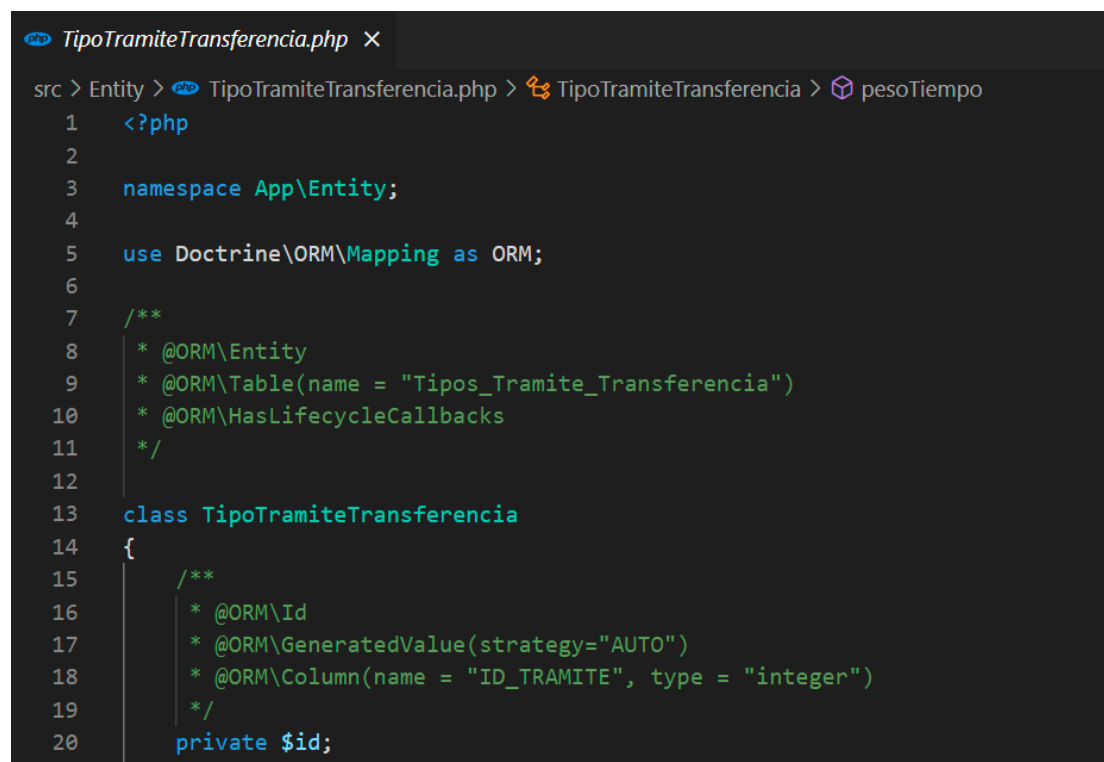
Elaborado por: Los autores

4.5.2. Entity

En symfony, las entidades usan doctrine para obtener la data necesaria, en cada entity se declara características como el tipo del campo en la base, con el fin de que mediante la terminal se puedan realizar las debidas migraciones, aunque esta sentencia puede ser excluida si se tiene una base de datos antes creada, cada entidad corresponde a una tabla en la base.

Figura 21.

Entity



```
src > Entity > TipoTramiteTransferencia.php > TipoTramiteTransferencia > pesoTiempo
1  <?php
2
3  namespace App\Entity;
4
5  use Doctrine\ORM\Mapping as ORM;
6
7  /**
8   * @ORM\Entity
9   * @ORM\Table(name = "Tipos_Tramite_Transferencia")
10  * @ORM\HasLifecycleCallbacks
11  */
12
13  class TipoTramiteTransferencia
14  {
15      /**
16       * @ORM\Id
17       * @ORM\GeneratedValue(strategy="AUTO")
18       * @ORM\Column(name = "ID_TRAMITE", type = "integer")
19       */
20      private $id;
```

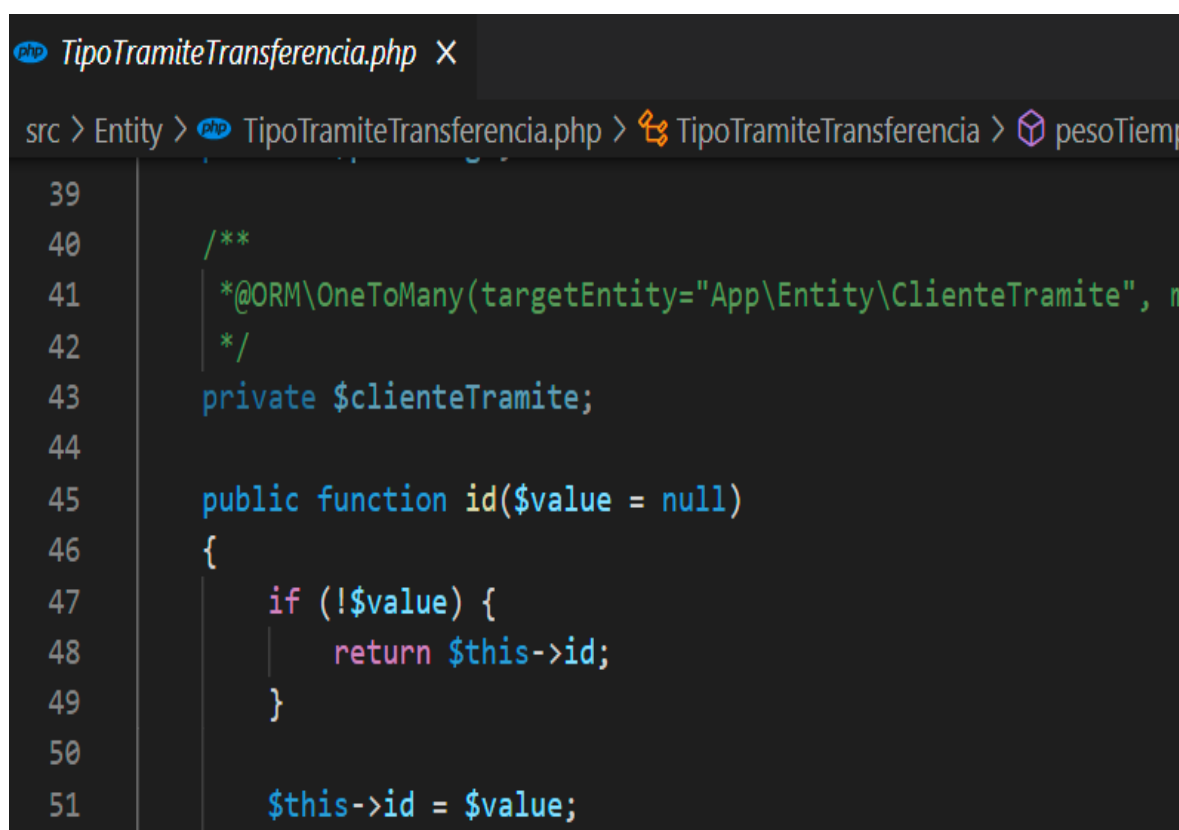
Nota: Estructura inicial de una entity

Elaborado por: Los autores

Acompañado de las variables por cada campo se incluye un método que permite su obtención y colocación dentro de la funcionalidad que se desee, que representa a un get y un set.

Figura 22.

Estructura de get y set de un entity



```
39
40     /**
41      * @ORM\OneToMany(targetEntity="App\Entity\ClienteTramite", m
42      */
43     private $clienteTramite;
44
45     public function id($value = null)
46     {
47         if (!$value) {
48             return $this->id;
49         }
50
51         $this->id = $value;
```

Nota: Estructura de get y set de un entity

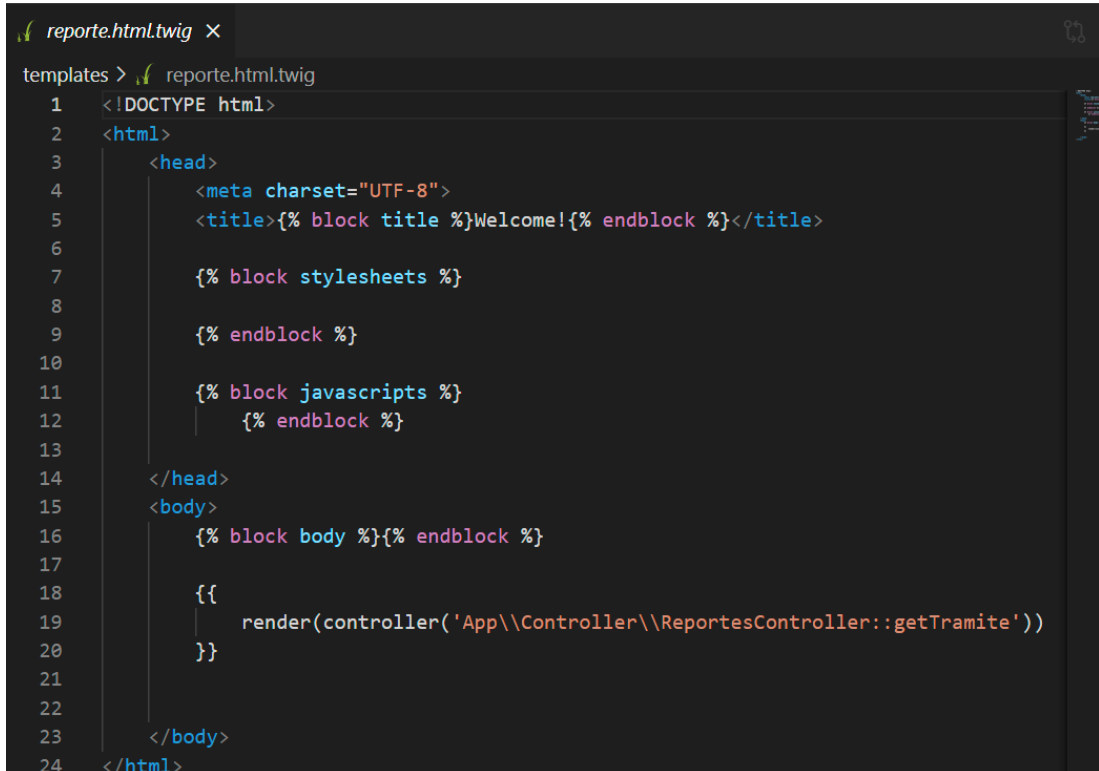
Elaborado por: Autores

4.5.3. Vistas

Para el desarrollo de las vistas se usa comandos que en esta ocasión no se usan, sin embargo, se usa html con twig que es el lenguaje con el que las vistas son aceptadas en symfony, aunque tiene una nomenclatura diferente las etiquetas HTML se siguen usando.

Figura 23.

Estructura de las vistas

A screenshot of a code editor with a dark theme. The editor shows a file named 'reporte.html.twig' with 24 lines of code. The code is a Twig template for an HTML page. It starts with a DOCTYPE declaration, followed by an HTML tag. Inside the HTML tag, there is a head section containing a meta charset declaration, a title block, and blocks for stylesheets and javascripts. The body section contains a block for the main content and a block for rendering a controller action. The code is as follows:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <title>{% block title %}Welcome!{% endblock %}</title>
6
7     {% block stylesheets %}
8
9     {% endblock %}
10
11    {% block javascripts %}
12      {% endblock %}
13  </head>
14  <body>
15    {% block body %}{% endblock %}
16
17    {{
18      render(controller('App\\Controller\\ReportesController::getTramite'))
19    }}
20  </body>
21 </html>
```

Nota: Estructura de las vistas

Elaborado por: Los autores

4.6. Interfaces reales del sistema

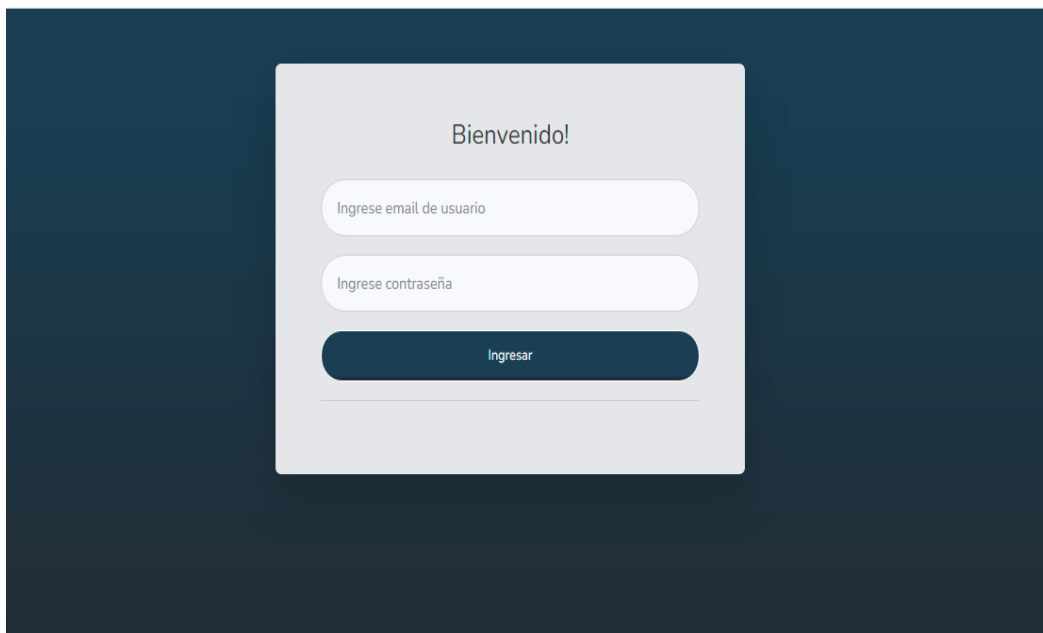
Se muestra las interfaces reales de las pantallas de forma general para evitar redundancia.

4.6.1. *Aplicativo web*

Primero se dispone de la interfaz de inicio de sesión la cual es vista por todos al iniciar la aplicación, está dispone de dos campos email y contraseña para autenticar al usuario.

Figura 24.

Pantalla real de inicio de sesión



The login screen features a dark blue background. In the center is a light gray rectangular box. Inside this box, the text "Bienvenido!" is displayed at the top. Below it are two white input fields with rounded corners. The first field is labeled "Ingrese email de usuario" and the second is labeled "Ingrese contraseña". Below these fields is a dark blue button with the text "Ingresar" in white. A thin horizontal line is positioned below the button.

Nota: Pantalla real de inicio de sesión

Elaborado por: Los autores

A continuación, se muestra la forma en que los trámites que están asignados al usuario que inicio sesión se presentan, y también se puede visualizar el menú lateral en este caso es un administrador.

Figura 25.

Pantalla real de “trámites asignados a mi”

Tramites asignados a mi											
Código	Usuario	Nombre	Apellido	Cliente	Nombre	Apellido	Tipo de Trámite	Fecha	Estado	Progreso	Acciones
57	1725449571	Daniela	Peralta	1719605469	Martha	Robayo	COMPRA Y VENTA	2022-01-03	FINALIZADO	100.0%	

Nota: Pantalla real de “trámites asignados a mi”

Elaborado por: Los autores

El ingreso de datos se hace mediante un modal el cual está validado una vez ingresado la tabla que está en el fondo se actualiza.

Figura 26.

Pantalla real de ingreso de datos

The screenshot shows a web application interface for 'ORQUERA Estudios Jurídicos'. A sidebar on the left contains navigation links: 'Tramites asignados a mi', 'PANEL', 'Usuarios', 'Clientes', 'Tramites', and 'Reportes'. The main area displays a table of users with columns: 'Cédula', 'Nombre', 'Estado', 'Rol', and 'Acciones'. A modal titled 'Crear Usuario' is open in the center, containing the following fields and options:

- Cédula:** Input field with the value '1725449571'.
- Nombre:** Input field with the value 'Daniela'.
- Apellido:** Input field with the value 'Peralta'.
- Correo Electronico:** Input field with the value 'danylove9569@hotmail.com'.
- Rol:** Dropdown menu with 'Administrador' selected.
- Contraseña:** Input field with masked characters '*****'.
- Estado:** Dropdown menu with 'Activo' selected.
- Buttons:** 'Crear cliente' and 'Cerrar'.

The background table shows the following data:

Cédula	Nombre	Estado	Rol	Acciones
1725449571	Daniela	Activo	ROLE_ADMIN	[Icon]
1701743997	nuevo	Inactivo	ROLE_ADMIN	[Icon]
1725449581	Abogado 1	Activo	ROLE_ADMIN	[Icon]
1725449555	Mariela	Activo	ROLE_USER	[Icon]
1720842481	usuario	Activo	ROLE_USER	[Icon]

Nota: Pantalla real de ingreso de datos Elaborado por: Los autores

Al actualizar el modal se carga con los datos actuales para poder realizar la edición de los campos que se requiera.

Figura 27.

Pantalla real de actualización de datos

The screenshot shows the same web application interface as Figure 26. The 'Crear Usuario' modal is open, but the data is updated. The modal contains the following fields and options:

- Cédula:** Input field with the value '1725449571'.
- Nombre:** Input field with the value 'Daniela'.
- Apellido:** Input field with the value 'Peralta'.
- Correo Electronico:** Input field with the value 'danylove9569@hotmail.com'.
- Rol:** Dropdown menu with 'Administrador' selected.
- Contraseña:** Input field with masked characters '*****'.
- Estado:** Dropdown menu with 'Activo' selected.
- Buttons:** 'Crear cliente' and 'Cerrar'.

The background table shows the following data:

Cédula	Nombre	Estado	Rol	Acciones
1725449571	Daniela	Activo	ROLE_ADMIN	[Icon]
1701743997	nuevo	Inactivo	ROLE_ADMIN	[Icon]
1725449581	Abogado 1	Activo	ROLE_ADMIN	[Icon]
1725449555	Mariela	Activo	ROLE_USER	[Icon]
1720842481	usuario	Activo	ROLE_USER	[Icon]
151541289	Pedro	Activo	ROLE_USER	[Icon]

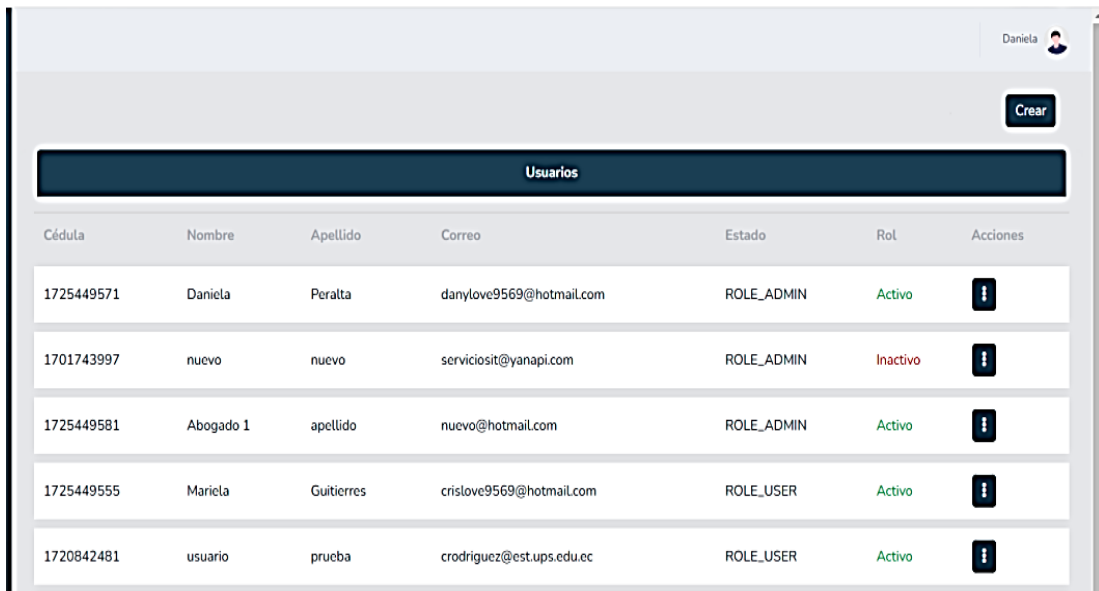
Nota: Pantalla real de actualización de datos

Elaborado por: Los autores

En caso de que un usuario salga de la organización, este no es eliminado permanentemente en su lugar se ejecuta un eliminado lógico, el cual es una actualización de estado de usuario de Activo a Inactivo.

Figura 28.

Ejemplo de eliminado lógico



The screenshot shows a web application interface for managing users. At the top right, there is a user profile for 'Daniela' and a 'Crear' button. Below this is a table titled 'Usuarios'. The table has the following columns: Cédula, Nombre, Apellido, Correo, Estado, Rol, and Acciones. There are five rows of user data. The 'Estado' column uses color coding: green for 'Activo' and red for 'Inactivo'. Each row has a three-dot menu icon in the 'Acciones' column.

Cédula	Nombre	Apellido	Correo	Estado	Rol	Acciones
1725449571	Daniela	Peralta	danylove9569@hotmail.com	Activo	ROLE_ADMIN	
1701743997	nuevo	nuevo	serviciosit@yanapi.com	Inactivo	ROLE_ADMIN	
1725449581	Abogado 1	apellido	nuevo@hotmail.com	Activo	ROLE_ADMIN	
1725449555	Mariela	Gutierrez	crislove9569@hotmail.com	Activo	ROLE_USER	
1720842481	usuario	prueba	crodriguez@est.ups.edu.ec	Activo	ROLE_USER	

Nota: Pantalla real de eliminado lógico

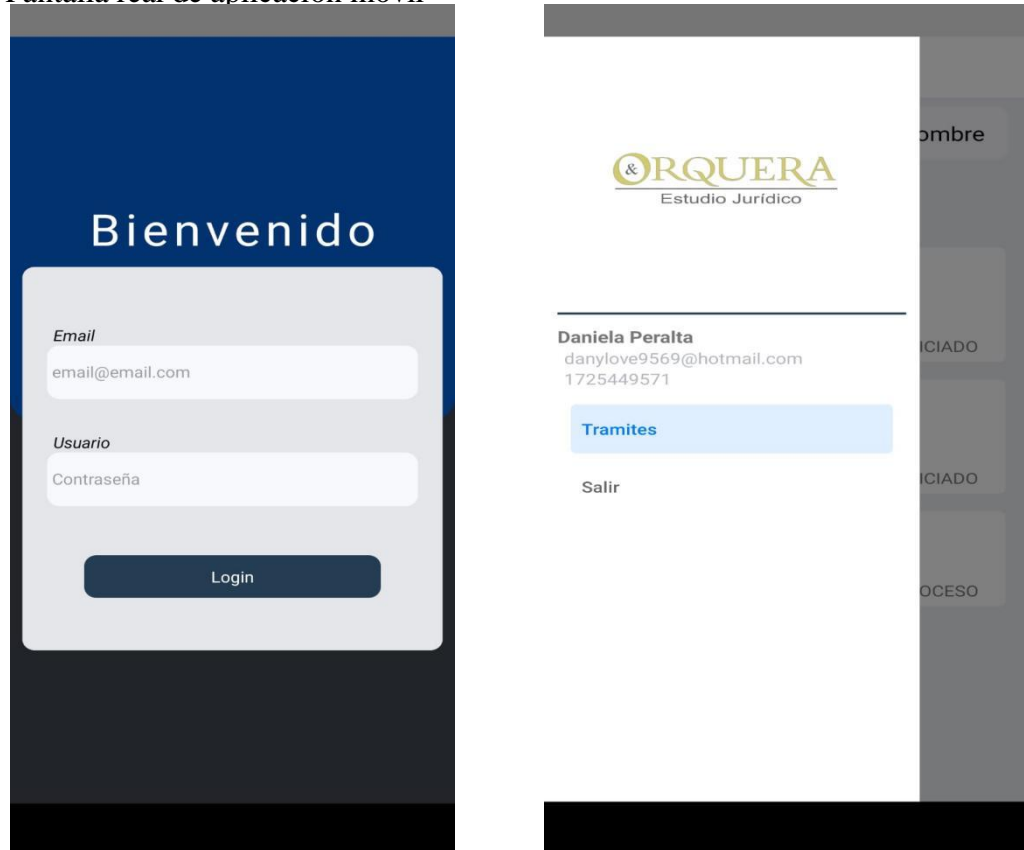
Elaborado por: Los autores

4.6.2. *Aplicativo móvil*

En la interfaz del móvil, tiene menor carga de vistas, se dispone de dos: la pantalla de inicio de sesión que funciona igual que en la web y la de búsqueda de trámites para verificar su estado.

Figura 29.

Pantalla real de aplicación móvil

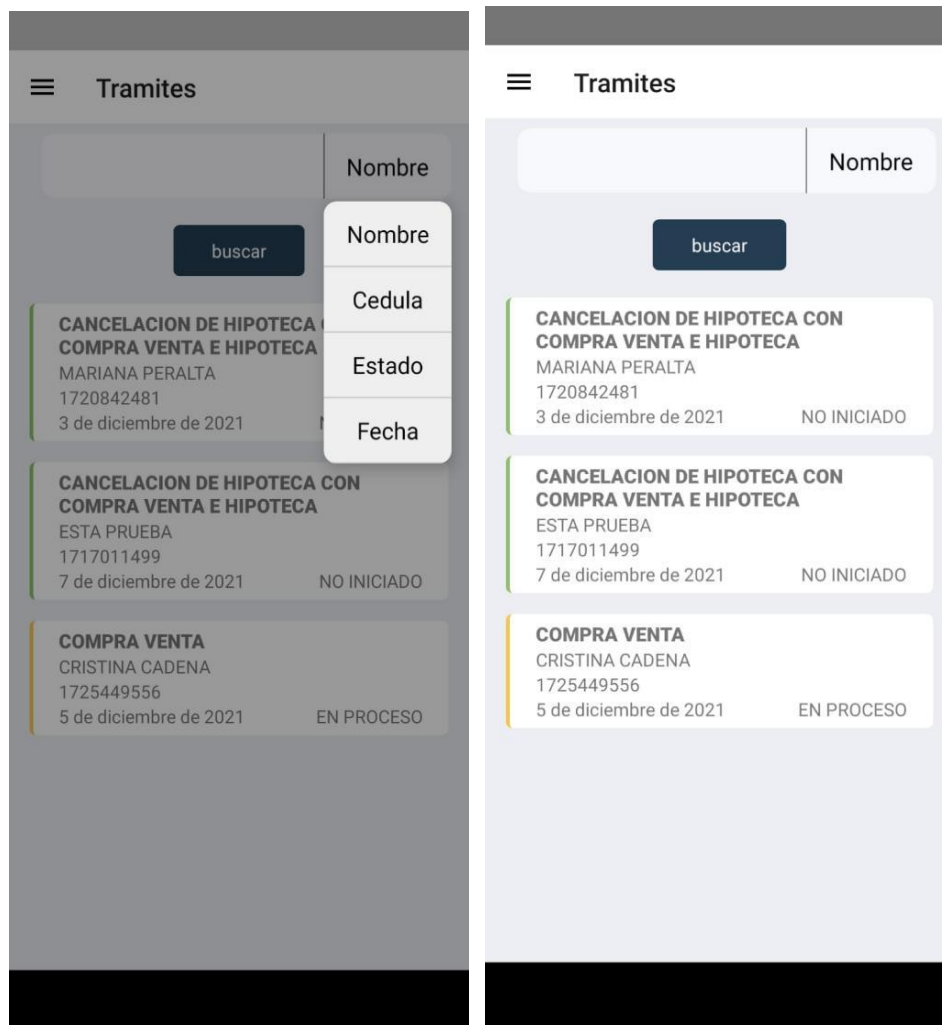


Nota: Pantalla real de aplicación móvil inicio de sesión y menú lateral

Elaborado por: Los autores

Figura 30.

Pantalla real de aplicación móvil filtrar datos



Nota: Pantalla real de aplicación móvil filtrar datos

Elaborado por: Los autores

CAPITULO V

PRUEBAS Y CORRECCIONES

Una vez terminado el desarrollo siguiendo el análisis realizado en los anteriores capítulos, el siguiente paso es asegurar que todo el sistema funcione correctamente, es por lo que se ha decidido aplicar pruebas funcionales, unitarias y de integración, las pruebas a continuación todas están en estado aprobado, pero se incluye una tabla con el número de intentos hasta llegar a ese estado de ser el caso.

5.1. Pruebas unitarias

Tabla 7.

P001

P001: Inicio de sesión con datos correctos	
Fecha:	2021-12-10
Descripción:	El usuario ingresa su correo y su contraseña, ambos son correctos por lo que pueden ingresar sin problema alguno.
Precondiciones:	El usuario a iniciar sesión esta guardado en la base de datos, se prueba con diferentes usuarios.
Datos Ingresados 1:	Usuario: danylove9569@hotmail.com Clave: clave de usuario
Resultado Esperado 1: El usuario ingresa normalmente	Resultado Obtenido 1: El usuario inicio sesión normalmente
Mensaje: ninguno	Mensaje: ninguno
Datos Ingresados 2:	Usuario: serviciosit@yanapi.com Clave: clave de usuario
Resultado Esperado 2: El usuario ingresa normalmente	Resultado Obtenido 2: El usuario inicio sesión normalmente
Mensaje: ninguno	Mensaje: ninguno

Evaluación de la Prueba:	APROBADA
---------------------------------	----------

Nota: Prueba de inicio de sesión con datos correcto

Elaborado por: Los autores

Tabla 8.

P002

P002: Inicio de sesión con datos incorrectos	
Fecha:	2021-12-10
Descripción:	El usuario ingresa su correo y su contraseña, ambos son incorrectos por lo que no pueden ingresar al sistema
Precondiciones:	El usuario a iniciar sesión esta guardado en la base de datos, además ingresa una contraseña incorrecta.
Datos Ingresados 1:	Usuario: danylove9569@hotmail.com Clave: abcd
Resultado Esperado 1: Se deniega el ingreso al sistema	Resultado Obtenido 1: Se denegó el ingreso al sistema
Mensaje: ¡La contraseña ingresada es incorrecta!	Mensaje: ¡La contraseña ingresada es incorrecta!
Datos Ingresados 2:	Usuario: danylove9569@hotmail.com Clave: abcd
Resultado Esperado 2: Se deniega el ingreso al sistema	Resultado Obtenido 2: Se denegó el ingreso al sistema
Mensaje: ¡La contraseña ingresada es incorrecta!	Mensaje: ¡La contraseña ingresada es incorrecta!
Evaluación de la Prueba:	APROBADA

Nota: Prueba de inicio de sesión con datos correcto

Elaborado por: Los autores

Tabla 9.

P003

P003: Inicio de sesión con usuario no creado	
Fecha:	2021-12-10
Descripción:	El usuario ingresa un correo incorrecto
Precondiciones:	Ingresar un correo que no exista en la base de datos
Datos Ingresados 1:	Usuario: dome_rodriguez@outlook.es Clave: clave de usuario
Resultado Esperado 1: El usuario no puede ingresar al sistema	Resultado Obtenido 1: El usuario no puede ingresar al sistema
Mensaje: Usuario no encontrado	Mensaje: Usuario no encontrado
Evaluación de la Prueba:	APROBADA

Nota: Prueba de inicio de sesión con datos incorrectos

Elaborado por: Los autores

Tabla 10.

P004

P004: Ingresar, actualizar, eliminar usuario	
Fecha:	2021-12-10
Descripción:	El usuario administrador ingresa un nuevo usuario, el cual se actualiza, elimina.
Precondiciones:	No haber creado ese usuario antes.
Datos Ingresados 1:	Email: dome_rodriguez@outlook.es Cédula: 1712268430 Nombre: Dominica Sofia Apellido: Rodríguez Rol: Administrador Clave: clave de usuario
Resultado Esperado 1: El usuario se registra	Resultado Obtenido 1: El usuario se registra
Mensaje: Registrado con éxito	Mensaje: Registrado con éxito
Datos Ingresados 2:	Clave: nueva clave
Resultado Esperado 2: Usuario actualizado	Resultado Obtenido 2: Usuario actualizado
Mensaje: ninguno	Mensaje: ninguno
Datos Ingresados 2:	Ninguno porque se ejecuta una eliminación
Resultado Esperado 2: Usuario cambia de estado	Resultado Obtenido 2: Usuario cambia de estado
Mensaje: ninguno	Mensaje: ninguno
Evaluación de la Prueba:	APROBADA

Nota: Prueba de ingresar, actualizar y eliminar usuario

Elaborado por: Los autores

Tabla 11.

P005

P005: Ingresar, actualizar, eliminar cliente	
Fecha:	2021-12-10
Descripción:	El usuario ingresa un nuevo cliente, el cual se actualiza, elimina.
Precondiciones:	No haber creado ese cliente antes.
Datos Ingresados 1:	Email: david.cadenya@hotmail.com Cédula: 1720842681 Nombre: David Antonio Apellido: Cadena Estado civil: Soltero Dirección: Casa Teléfono: 02222222
Resultado Esperado 1: El cliente es registrado	Resultado Obtenido 1: El cliente es registrado
Mensaje: Registrado con éxito	Mensaje: Registrado con éxito
Datos Ingresados 2:	Apellido: Daniels
Resultado Esperado 2: Cliente actualizado	Resultado Obtenido 2: Cliente actualizado
Mensaje: ninguno	Mensaje: ninguno
Datos Ingresados 2:	Ninguno porque se ejecuta una eliminación
Resultado Esperado 2: Cliente cambia de estado	Resultado Obtenido 2: Cliente cambia de estado
Mensaje: ninguno	Mensaje: ninguno
Evaluación de la Prueba:	APROBADA

Nota: Prueba de ingresar, actualizar y eliminar cliente

Elaborado por: Los autores

Tabla 12.

P006

P006: Ingresar, actualizar tipos de trámite			
Fecha:	2021-12-10	H.U:	HU-001
Descripción:	El usuario administrador ingresa un nuevo tipo de trámite, el cual se actualiza.		
Precondiciones:	No haber creado ese tipo de trámite antes.		
Datos Ingresados 1:	Nombre: HIPOTECA Carga: 1 Tiempo: 1		
Resultado Esperado 1: El tipo de trámite es registrado	Resultado Obtenido 1: El tipo de trámite es registrado		
Mensaje: Registrado con éxito	Mensaje: Registrado con éxito		
Datos Ingresados 2:	Nombre: HIPOTECA Editada		
Resultado Esperado 2: Tipo de trámite actualizado	Resultado Obtenido 2: Tipo de trámite actualizado		
Mensaje: ninguno	Mensaje: ninguno		
Evaluación de la Prueba:	APROBADA		

Nota: Prueba de ingresar, actualizar tipo de trámite

Elaborado por: Los autores

Tabla 13.

P007

P007: Descargar reportes	
Fecha:	2021-12-10
Descripción:	El usuario administrador selecciona del menú lateral cualquiera de los reportes
Precondiciones:	Haber iniciado sesión
Datos Ingresados 1:	El usuario da clic en reporte trámites
Resultado Esperado 1: El reporte se descarga	Resultado Obtenido 1: El reporte se descarga
Mensaje: ninguno	Mensaje: ninguno
Datos Ingresados 2:	El usuario da clic en reporte trámites por tipo
Resultado Esperado 2: El reporte se descarga	Resultado Obtenido 2: El reporte se descarga
Mensaje: ninguno	Mensaje: ninguno
Evaluación de la Prueba:	APROBADA

*Nota: Prueba de descarga de reportes**Elaborado por: Los autores*

A continuación, se empieza a testear el funcionamiento del algoritmo de asignación de trámites se necesita tener en cuenta la tabla de tipos de trámites y sus pesos.

Tabla 14.

P008

P008: Asignar nuevo trámite a un usuario aleatorio	
Fecha:	2021-12-11
Descripción:	El usuario crea un nuevo trámite y automáticamente este es asignado a un usuario
Precondiciones:	Que ningún usuario tenga asignado un trámite
Datos Ingresados 1:	Cliente: David Cadena Tipo de trámite: HIPOTECA
Resultado Esperado 1: El trámite es asignado a cualquier usuario	Resultado Obtenido 1: El trámite es asignado a cualquier usuario
Mensaje: Registrado con éxito	Mensaje: Registrado con éxito
Evaluación de la Prueba:	APROBADA

Nota: Prueba de asignar nuevo trámite a usuario aleatorio

Elaborado por: Los autores

Tabla 15.

P009

P009: Asignar nuevo trámite al usuario que no tenga trámites	
Fecha:	2021-12-11
Descripción:	El usuario crea un nuevo trámite y automáticamente este es asignado a un usuario que no tenga trámites asignados
Precondiciones:	Que ningún usuario tenga asignado un trámite
Datos Ingresados 1:	Cliente: David Cadena Tipo de trámite: HIPOTECA
Resultado Esperado 1: El trámite es asignado a un usuario que no tiene trámites anteriores	Resultado Obtenido 1: El trámite es asignado a un usuario que no tiene trámites anteriores
Mensaje: Registrado con éxito	Mensaje: Registrado con éxito
Evaluación de la Prueba:	APROBADA

Nota: Prueba de asignar nuevo trámite a usuario que no tenga trámites

Elaborado por: Los autores

Tabla 16.

P010

P010: Asignar nuevo trámite a usuario aleatorio cuando dos usuarios dispongan el mismo valor de pesos	
Fecha:	2021-12-11
Descripción:	El usuario crea un nuevo trámite y automáticamente este es asignado a un usuario aleatorio entre el grupo que menos valor de pesos.
Precondiciones:	Que existan dos o varios usuarios con el mismo valor de pesos
Datos Ingresados 1:	Cliente: David Cadena Tipo de trámite: HIPOTECA
Resultado Esperado 1: El trámite es asignado a un usuario que está en el grupo de usuarios con el mismo valor de pesos	Resultado Obtenido 1: El trámite es asignado a un usuario que está en el grupo de usuarios con el mismo valor de pesos
Mensaje: Registrado con éxito	Mensaje: Registrado con éxito
Evaluación de la Prueba:	APROBADA

Nota: Prueba de asignar nuevo trámite a usuario aleatorio cuando dos usuarios dispongan el mismo valor de pesos

Elaborado por: Los autores

El funcionamiento del software también está regido por las validaciones en interfaz que se disponen por esta razón se testea algunas de ellas.

Tabla 17.

P011

P011: El ingreso de la cédula en cliente y usuario es de 10 caracteres	
Fecha:	2021-12-11
Descripción:	Se intenta crear un usuario y un cliente con un número de cédula con un número menor de 10 dígitos el sistema muestra un mensaje
Precondiciones:	Ingresar una cédula con 9 dígitos
Datos Ingresados 1:	Cédula Cliente: 170175666
Resultado Esperado 1: El sistema muestra un mensaje	Resultado Obtenido 1: El sistema muestra un mensaje
Mensaje: Registrado con éxito	Mensaje: Registrado con éxito
Datos Ingresados 1:	Cédula Usuario: 170175666
Resultado Esperado 1: El sistema muestra un mensaje	Resultado Obtenido 1: El sistema muestra un mensaje
Mensaje: Registrado con éxito	Mensaje: Registrado con éxito
Evaluación de la Prueba:	APROBADA

Nota: Prueba de ingreso de la cédula en cliente y usuario es de 10 caracteres

Elaborado por: Los autores

Tabla 18.

P012

P012: Ingreso de un cliente con estado civil casado	
Fecha:	2021-12-11
Descripción:	Se registra un cliente nuevo con estado civil casado, el sistema despliega un campo extra para agregar la cédula y nombre de su cónyuge
Precondiciones:	ninguna
Datos Ingresados 1:	Email: david.cadenya1@hotmail.com Cédula: 1720842682 Nombre: David Antonio Apellido: Cadena Estado civil: Casado Cédula cónyuge: 1725448723 Cónyuge: Cristina Rodríguez Dirección: Casa Teléfono: 02222222
Resultado Esperado 1: El sistema muestra dos campos extras para los datos de cónyuge	Resultado Obtenido 1: El sistema muestra dos campos extras para los datos de cónyuge
Mensaje: ninguno	Mensaje: ninguno
Evaluación de la Prueba:	APROBADA

Nota: Prueba de ingreso de un cliente con estado civil casado

Elaborado por: Los autores

Tabla 19.

P013

P013: Visualizar menú lateral de usuario administrador	
Fecha:	2021-12-11
Descripción:	Un usuario bajo el rol de administrador tiene acceso a todas las funcionalidades del sistema
Precondiciones:	Disponer de un usuario administrador
Datos Ingresados 1:	Usuario: danylove9569@hotmail.com Clave: clave de usuario
Resultado Esperado 1: El sistema muestra el menú lateral de administrador	Resultado Obtenido 1: El sistema muestra el menú lateral de administrador
Mensaje: ninguno	Mensaje: ninguno
Evaluación de la Prueba:	APROBADA

Nota: Prueba de visualizar menú lateral de usuario administrador

Elaborado por: Los autores

Tabla 20.

P014

P014: Visualizar menú lateral de usuario colaborador	
Fecha:	2021-12-11
Descripción:	Un usuario bajo el rol de colaborador solo puede acceder a clientes y trámites
Precondiciones:	Disponer de un usuario colaborador
Datos Ingresados 1:	Usuario: serviciosit@yanapi.com Clave: clave de usuario
Resultado Esperado 1: El sistema muestra el menú lateral de colaborador	Resultado Obtenido 1: El sistema muestra el menú lateral de colaborador
Mensaje: ninguno	Mensaje: ninguno
Evaluación de la Prueba:	APROBADA

Nota: Prueba de visualizar menú lateral de usuario colaborador

Elaborado por: Los autores

5.2. Pruebas integrales

Para las pruebas integrales se toma en cuenta los usuarios y clientes que se crean para las pruebas anteriores, terminando con la actualización de un trámite que se asigna a un usuario de forma aleatoria.

Tabla 21.

P015

P015: Actualizar el proceso del trámite	
Fecha:	2021-12-11
Descripción:	Una vez que se dispone de un trámite asignado se puede editar todos sus datos de proceso
Precondiciones:	Disponer un trámite asignado
Datos Ingresados 1:	Datos de proceso de trámite
Resultado Esperado 1: El usuario puede actualizar el proceso	Resultado Obtenido 1: El usuario puede actualizar el proceso
Mensaje: ninguno	Mensaje: ninguno
Evaluación de la Prueba:	APROBADA

Nota: Prueba de actualizar el proceso del trámite

Elaborado por: Los autores

Tabla 22.

P016

P016: Reasignar un trámite a otro colaborador	
Fecha:	2021-12-11
Descripción:	El usuario asignado puede reasignar el trámite
Precondiciones:	Disponer de un trámite
Datos Ingresados 1:	Usuario: Antonio Egas Comentario: realice proceso uno
Resultado Esperado 1: El usuario al que se le reasigna tiene el trámite	Resultado Obtenido 1: El usuario al que se le reasigna tiene el trámite
Mensaje: ninguno	Mensaje: ninguno
Evaluación de la Prueba:	APROBADA

Nota: Prueba de reasignar un trámite a otro colaborador

Elaborado por: Los autores

5.3. Número de pruebas realizadas

Para llegar a aprobar una prueba se realizan un número n de intentos como se muestra a continuación en la tabla 23.

Tabla 23.

Número de intentos por prueba

Código de prueba	Número de intentos fallidos	Número de intentos aprobados	Número de pruebas total
P001	0	5	5
P002	5	4	9
P003	0	3	3
P004	6	4	10
P005	7	5	11
P006	5	7	11
P007	10	6	16
P008	20	5	25
P009	11	2	13
P010	5	7	12
P011	6	2	8
P012	3	3	6
P013	1	2	3
P014	15	6	21
P015	12	10	22

Nota: Número de intentos por prueba

Elaborado por: Los autores

5.4. Pruebas de carga

En las pruebas de carga se usa JMeter para ejecutarlas, se selecciona solo tres url las cuales son: login, trámites y descarga de resultados. Cada una es parte de los procesos recurrentes del sistema.

Se configura la prueba con una carga de 50 usuarios cada 5 segundos por 1 minuto, y se obtiene los siguientes resultados siendo el porcentaje de error del 0%

Figura 31.

Resumen de resultados de prueba de carga

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput ↑	Received KB...	Sent KB/sec	Avg. Bytes
descarga de reporte	204	7364	3208	19133	2904.34	0.00%	37.7/min	34.39	0.17	56076.0
tramites	240	7290	1921	16051	2999.50	0.00%	43.9/min	40.12	0.21	56125.0
login	266	8724	516	62046	13811.71	0.00%	49.1/min	43.72	0.11	54691.3
TOTAL	710	7849	516	62046	8797.39	0.00%	2.2/sec	116.84	0.48	55573.8

Nota: Resultados de prueba de carga en JMeter

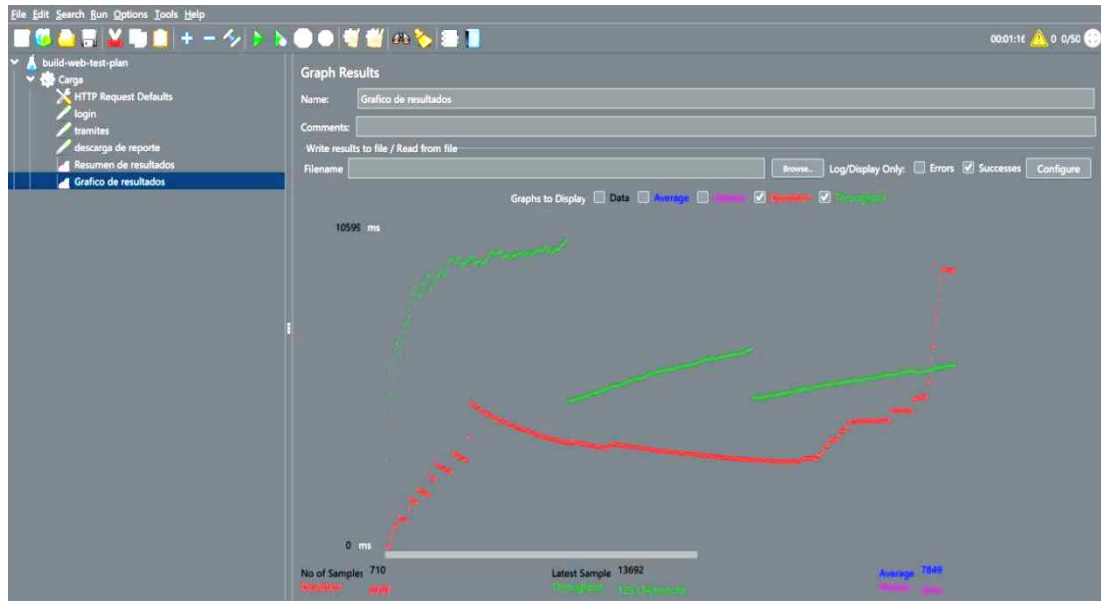
Elaborado por: Los autores

5.5. Prueba de rendimiento

Además, también se observa el rendimiento del sistema, para su mejor comprensión los datos se representan en la figura 32, en donde de color verde se tiene el rendimiento y de rojo la desviación. Frente a la carga de 50 usuarios cada 5 segundos el resultado es elevado, lo que implica una gran capacidad del sistema y su infraestructura.

Figura 32.

Gráfico de prueba de rendimiento



Nota: Resultados de prueba de rendimiento en JMeter

Elaborado por: Los autores

CONCLUSIONES

- La obtención de requisitos no solo se hace por medio de reuniones personales con el usuario si no también se puede conseguir levantando el proceso actual, el cual permite identificar de manera precisa brechas e incluso fallas que se manejan y que se solucionan con el desarrollo del software.
- El uso de herramientas que faciliten el desarrollo sin duda alguna reduce el tiempo que se invierte en programación, en el caso de esta aplicación se usa el framework Symfony, el cual provee de una terminal para autogenerar el código de la mayoría de los componentes, empezando por las entidades de base de datos hasta las vistas con twig.
- El 100% de los trámites se asignan equitativamente gracias al desarrollo e implementación del algoritmo, el cual toma pesos de los diferentes tipos, eliminando la sobre carga de trabajo de los colaboradores.
- Con el ingreso de información acerca de los trámites y sus asignaciones los usuarios administradores pueden de manera fácil acceder a reportes con cifras del número de trámites por colaborador y por tipo de tal manera que ellos tienen noción de ésta, y se tome decisiones.
- Es importante mencionar que además de que el usuario se asigna de forma equitativa, también es posible la reasignación, proceso que se almacena y para saber que usuario ha estado gestionando el trámite, se dispone de un registro histórico.
- El uso del sistema completo de forma web permite que el software esté disponible en todo lugar, pero la inclusión de un lado móvil mejora su accesibilidad con información resumida de los trámites que se tienen asignados actualmente desde cualquier lugar.

- El desarrollo de sistemas con características específicas para un software hace que este cumpla con los requisitos planteados, pero también dispone de características comunes en todos los programas como lo es la usabilidad, la cual mediante la ayuda de íconos y otros elementos permite que el aplicativo sea intuitivo.

RECOMENDACIONES

- Para el correcto uso del sistema y de todas sus funcionalidades se debe realizar una capacitación y en caso de que existan nuevos ingresos de personal, se debe organizar con el fin de tener un mejor uso.
- Se debe utilizar la aplicación web en dispositivos con el sistema operativo Android desde su versión 6.0 en adelante.
- Se debe elegir un usuario administrador para que se aquel que distribuya el APK en la organización mientras el aplicativo no esté en la tienda de Android.
- Es de suma importancia realizar un respaldo de la base de datos cada semana con el fin de que no exista la pérdida de datos en caso de algún accidente.
- Se debe evitar compartir las contraseñas debido a que esto podría afectar el correcto proceso de ingreso de datos imposibilitando obtener información verídica para la generación de reportes

LISTA DE REFERENCIAS

Andreu, R., Ricart, J., & Valor, J. (1991). *Estrategia y Sistemas de Información*.

Madrid: McGrawhill.

Bootstrap. (s.f.). *Bootstrap*. Obtenido de Bootstrap: <https://getbootstrap.com/>

CertiProf. (2019). *SCRUM MASTER PROFESSIONAL CERTIFICATE(SMPC)*.

Recuperado el 23 de febrero de 2021, de <https://certiprof.com/>

Facebook. (2020). *React Native · Learn once, write anywhere*. Obtenido de

<https://reactnative.dev/>

Hernandez, A. (2003). LOS SISTEMAS DE INFORMACIÓN: EVOLUCIÓN Y

DESARROLLO. *Proyecto social: Revista de relaciones laborales*, 149-165.

MDN Web Docs. (2021). *HTML: Lenguaje de etiquetas de hipertexto / MDN*.

Obtenido de <https://developer.mozilla.org/es/docs/Web/HTML>

PHP. (s.f.). *PHP: ¿Qué es PHP? - Manual*. Obtenido de

<https://www.php.net/manual/es/intro-what-is.php>

Stair, R., & Reynolds, G. (2010). *Principios de sistemas de información: un enfoque*

administrativo. Recuperado el 23 de febrero de 2021, de

http://fcaenlinea1.unam.mx/anexos/1144/1144_u7_act1

Suarez, A. (1992). *Diccionario de economía y administración*. Madrid: McGraw-

Hill.

Symfony. (2021). *Symfony*. Obtenido de What is: [https://symfony.com/what-is-](https://symfony.com/what-is-symfony)

[symfony](https://symfony.com/what-is-symfony)

TechTerms. (2009). *User Interface*. Recuperado el 23 de febrero de 2021, de https://techterms.com/definition/user_interface

UML-Based Web Engineering. (2016). *Que es UWE?* Recuperado el 23 de febrero de 2021, de Tutorial sobre UWE: <https://uwe.pst.ifi.lmu.de/>

XAMPP. (2021). *XAMPP Installers and Downloads for Apache Friends*. Obtenido de <https://www.apachefriends.org/es/index.html>