

UNIVERSIDAD ESTATAL DE BOLÍVAR

FACULTAD DE CIENCIAS ADMINISTRATIVAS, GESTIÓN EMPRESARIAL E INFORMÁTICA

CARRERA DE SOFTWARE

TRABAJO DE INTEGRACIÓN CURRICULAR PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SOFTWARE

FORMA: PROYECTO TECNOLÓGICO

TEMA:

APLICACIÓN WEB PARA LA GESTIÓN DE PAGOS EN LA JUNTA ADMINISTRADORA COMUNITARIA DE AGUA POTABLE Y SANEAMIENTO LA BUENA ESPERANZA, PARROQUIA CANGAHUA, CANTÓN CAYAMBE.

AUTOR:

MARCO JOSUE QUISHPE CUELLAR

DIRECTORA:

ING. GALUTH IRENE GARCIA CAMACHO

GUARANDA- ECUADOR

2026

TEMA DEL PROYECTO TECNOLÓGICO

APLICACIÓN WEB PARA LA GESTIÓN DE PAGOS EN LA JUNTA ADMINISTRADORA COMUNITARIA DE AGUA POTABLE Y SANEAMIENTO LA BUENA ESPERANZA, PARROQUIA CANGAHUA, CANTÓN CAYAMBE.

DEDICATORIA

A mis padres, por ser los pilares sobre los que he construido mis sueños. Su amor incondicional, su apoyo en cada paso que doy en mi vida y su esfuerzo constante para tratar de darme las oportunidades para seguir estudiando han sido mi mayor motor. Mamá, por tus palabras de aliento que siempre me llenan de esperanza en los momentos de dudas y miedos, y Papá, por tus enseñanzas y tu ejemplo de fortaleza y perseverancia que me han inspirado a seguir adelante sin importar los desafíos que se presenten en mi camino. Les agradezco por mostrarme que los límites solo existen en nuestra mente y que todo es posible con fe, trabajo y dedicación.

A Dios, por haberme dado la fortaleza, la sabiduría y la fe necesaria para enfrentar cada desafío. A través de sus bendiciones he encontrado el camino correcto incluso en los momentos más oscuros, he aprendido a confiar en que todo ocurre según un propósito divino. Y porque cada día me brinda la oportunidad de crecer, aprender y avanzar con la certeza de que nunca estoy solo.

A mis hermanas, por ser mi refugio, mi alegría y mi inspiración constante. Sus palabras de ánimo, sus sonrisas en los días difíciles y su constante confianza en mis capacidades me han dado la fuerza para superar los obstáculos. Cada una de ustedes me ha brindado su apoyo incondicional y siempre han sido ejemplo de unión sin importar los momentos malos.

AGRADECIMIENTO

Expreso un sincero agradecimiento a mi directora de tesis, Ing. Galuth García, a mis pares académicos, Ing. Mónica Bonilla, Ing. Raúl Chávez por su valiosa orientación en el desarrollo de este proyecto, ya que su experiencia y recomendaciones fueron esenciales en todo el proceso académico.

Agradezco también a todas las personas, amigos, familia que han sido parte de todo este proceso de la Universidad, donde tuve muchos tropiezos, pero gracias a sus palabras de aliento me han ayudado a sobresalir y enfocarme en terminar la carrera.

Y por último a todos los docentes con los que a lo largo de la carrera he tenido la oportunidad de recibir clases, gracias por compartir todos sus conocimientos, por sus palabras, y haber buscado la manera de que cada día seamos más exigentes, cada día tratemos de aprender más para llegar a ser unos buenos profesionales.

CERTIFICADO DE VALIDACIÓN

UEB
UNIVERSIDAD ESTATAL DE BOLIVAR

FACULTAD DE CIENCIAS ADMINISTRATIVAS, GESTIÓN EMPRESARIAL E INFORMÁTICA

CERTIFICADO DE VALIDACIÓN

Ing. Galuth García, Ing. Mónica Bonilla e Ing. Raúl Chávez, en su orden Directora y Pares Académicos del Trabajo de Integración Curricular "APLICACIÓN WEB PARA LA GESTIÓN DE PAGOS EN LA JUNTA ADMINISTRADORA COMUNITARIA DE AGUA POTABLE Y SANEAMIENTO LA BUENA ESPERANZA, PARROQUIA CANGAHUA, CANTÓN CAYAMBE" desarrollado por el señor Marco Josue Quishpe Cuellar.

CERTIFICAN

Que, luego de revisado el Trabajo de Integración Curricular en su totalidad, cumple con las exigencias académicas de la carrera SOFTWARE, por lo tanto, autorizamos su presentación y defensa.

Guaranda, 25 de febrero del 2026

Firmado electrónicamente por: GALUTH IRENE GARCIA CAMACHO
Validar únicamente con FirmaEC

Ing Galuth García
Director

Firmado electrónicamente por: MONICA ELIZABETH BONILLA MANOBANDA
Validar únicamente con FirmaEC

Ing. Mónica Bonilla
Par Académico

Firmado electrónicamente por: RAUL MARCELO CHAVEZ BENAVIDES
Validar únicamente con FirmaEC

Ing. Raúl Chávez
Par Académico

DERECHOS DE AUTOR

UEB | UNIVERSIDAD ESTATAL DE BOLÍVAR

BIBLIOTECA GENERAL

DERECHOS DE AUTOR

Yo **MARCO JOSUE QUISHPE CUELLAR** portador de la Cédula de Identidad No **1755812144** en calidad de autor y titular de los derechos morales y patrimoniales
del Trabajo de Titulación:
"APLICACIÓN WEB PARA LA GESTIÓN DE PAGOS EN LA JUNTA ADMINISTRADORA COMUNITARIA DE AGUA POTABLE Y SANEAMIENTO LA BUENA ESPERANZA, PARROQUIA CANGAHUA, CANTÓN CAYAMBE", modalidad **PROYECTO TECNOLÓGICO**, de conformidad con el Art. 114 del CÓDIGO ORGÁNICO DE LA ECONOMÍA SOCIAL DE LOS CONOCIMIENTOS, CREATIVIDAD E INNOVACIÓN, concedemos a favor de la Universidad Estatal de Bolívar, una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra, con fines estrictamente académicos. Conservamos a mi/nuestro favor todos los derechos de autor sobre la obra, establecidos en la normativa citada.

Así mismo, autorizo/autorizamos a la Universidad Estatal de Bolívar, para que realice la digitalización y publicación de este trabajo de titulación en el Repositorio Digital, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

El autor declara que la obra objeto de la presente autorización es original en su forma de expresión y no infringe el derecho de autor de terceros, asumiendo la responsabilidad por cualquier reclamación que pudiera presentarse por esta causa y liberando a la Universidad de toda responsabilidad.



Marco Quishpe

INDICE DE CONTENIDOS

TEMA DEL PROYECTO TECNOLÓGICO i

DEDICATORIA ii

AGRADECIMIENTO iii

CERTIFICADO DE VALIDACIÓN iv

DERECHOS DE AUTOR v

INDICE DE CONTENIDOS vi

INDICE DE TABLAS ix

INDICE DE FIGURAS xii

INTRODUCCIÓN 1

RESUMEN 2

CHUZHAK RIMAY 3

ABSTRACT 4

CAPÍTULO I 5

FORMULACIÓN GENERAL DEL PROYECTO 5

1.1. Tema 5

1.2. Descripción del Problema 5

1.3. Justificación 6

1.4. Objetivos: 7

1.4.1. General 7

1.4.2. Específicos 7

CAPÍTULO II 8

MARCO TEÓRICO 8

2.1. Antecedentes 8

2.2. Científico 9

2.2.1. Metodologías de Desarrollo de Software 9

2.2.2. Metodologías Ágiles 10

2.2.3. Tarifa Base 11

2.3. Conceptual 11

2.3.1. Sistema web 11

2.3.2. Base de datos 12

2.3.3. Gestor de Base de Datos 12

2.3.4. Lenguajes de Programación 13

2.3.5. Framework 13

2.3.6. Lectura de Medidor 15

2.3.7. Consumo de agua 15

2.3.8. Cobro de agua 15

2.3.9. PHP 15

2.3.10. HTML 16

2.3.11. CSS 16

2.3.12. Bootstrap 16

2.3.13. Servidor Web 17

2.3.14. Front-End 17

2.3.15. Backend 17

2.3.16. Pruebas de Aceptación 17

2.4. Legal 18

2.4.1. Ley Orgánica de Recursos Hídricos, Usos y Aprovechamiento del Agua 18

2.4.2. Ley Orgánica de protección de Datos 18

2.4.3. Ley de Propiedad Intelectual 18

2.4.4. Ley de Comercio Electrónico. Firmas y mensajes de datos. 19

2.5. Georeferencial 19

CAPÍTULO III 20

METODOLOGÍA 20

3.1. Metodología de Desarrollo de Software 20

3.1.1. Programación Extrema XP 20

3.3. Técnicas e Instrumentos de Recopilación de Datos 23

3.3.1. Entrevista 23

3.3.2. Observación 24

CAPÍTULO IV 24

INGENIERÍA DEL PROYECTO 24

4.1. Fase I: Planificación 24

4.2. Fase II: Diseño 50

4.3. Fase III: Codificación 86

4.4. Fase IV: Pruebas 105

a. Pruebas unitarias..... 105

b. Pruebas de Seguridad 113

c. Pruebas de integración 117

CONCLUSIONES..... 135

RECOMENDACIONES 136

BIBLIOGRAFÍA..... 137

INDICE DE TABLAS

Tabla 1. Comparativa de Metodologías de Desarrollo de Software..... 10

Tabla 2. Clasificación de los tipos de gestores de base de datos..... 12

Tabla 3: Comparativa de frameworks para desarrollo..... 13

Tabla 4. Roles del Proyecto..... 26

Tabla 5: Modelo de Historias de usuario..... 27

Tabla 6:Historia de Usuario: Autenticación al sistema. 28

Tabla 7:Historia de Usuario: Módulo de usuarios..... 29

Tabla 8:Historia de Usuario: Módulo de socios. 30

Tabla 9:Historia de Usuario: Módulo de medidores 32

Tabla 10:Historia de Usuario: Módulo de multas. 33

Tabla 11:Historia de Usuario: Módulo de lecturas..... 35

Tabla 12:Historia de Usuario: Módulo de pagos..... 36

Tabla 13:Historia de Usuario: Módulo de reportes. 37

Tabla 14:Historia de Usuario: Módulo de notificaciones..... 38

Tabla 15:Historia de Usuario: Módulo de Historial de Pagos..... 39

Tabla 16:Historia de Usuario: Módulo de multas. 40

Tabla 17:Historia de Usuario: Módulo de pagar servicio..... 41

Tabla 18:Historia de Usuario: Módulo de actualizar perfil. 42

Tabla 19. Estimación de Historias de Usuario. 43

Tabla 20. Plan Tentativo de Entregas 48

Tabla 21. Historias de usuario de la Iteración 1. 52

Tabla 22 Tarea 1: Diseño de la base de datos..... 52

Tabla 23 Tarea 2: Diseño de la interfaz para autenticación 53

Tabla 24 Tarea 3: Diseño de la interfaz para recuperar contraseña..... 53

Tabla 25 Historias de usuario de la iteración 2 54

Tabla 26 Tarea 1: Diseño de la base de datos de los usuarios..... 54

Tabla 27 Tarea 2: Desarrollo del backend para usuarios..... 54

Tabla 28 Tarea 3: Desarrollo del frontend para usuarios 55

Tabla 29 Tarea 1: Diseño de la base de datos para socios..... 55

Tabla 30 Tarea 2: Desarrollo del backend para socios..... 55

Tabla 31 Tarea 3: Desarrollo del frontend para socios..... 56

Tabla 32 Historias de usuario de la iteración 3 56

Tabla 33 Tarea 1: Diseño de la base de datos para medidores 57

Tabla 34 Tarea 2: Desarrollo del backend para el módulo medidores 57

Tabla 35 Tarea 3: Desarrollo del frontend para el módulo medidores..... 57

Tabla 36 Tarea 1: Diseño de la base de datos para el módulo de lecturas 58

Tabla 37 Tarea 2: Desarrollo del backend para el módulo de lecturas..... 58

Tabla 38 Tarea 3: Desarrollo del frontend para el módulo de lecturas 59

Tabla 39 Tarea 1: Diseño de la base de datos para el módulo de multas 59

Tabla 40 Tarea 2: Desarrollo del backend para el módulo de multas 59

Tabla 41 Tarea 3: Desarrollo del frontend para el módulo de multas 60

Tabla 42 Historias de usuario de la iteración 4 60

Tabla 43 Tarea 1: Diseño de la base de datos para el módulo de Pagos. 61

Tabla 44 Tarea 2: Desarrollo del backend para el módulo de pagos..... 61

Tabla 45 Tarea 3: Desarrollo del frontend para el módulo de pagos 61

Tabla 46 Tarea 1: Interfaz para reporte de recaudación 62

Tabla 47 Tarea 2: Interfaz para reporte de cobros pendientes..... 62

Tabla 48 Tarea 1: Diseño de la base de datos para el módulo de notificaciones 63

Tabla 49 Tarea 2: Desarrollo del backend para el módulo de notificaciones..... 63

Tabla 50 Tarea 3: Desarrollo del frontend para el módulo de notificaciones. 63

Tabla 51 Historias de usuario de la iteración 5 64

Tabla 52 Tarea 1: Diseño de la base de datos para el módulo Historial de Pagos. 64

Tabla 53 Tarea 2: Desarrollo del backend para el módulo de Historial de Pagos..... 65

Tabla 54 Tarea 3: Desarrollo del frontend para el módulo de Historial de Pagos. 65

Tabla 55 Tarea 1: Desarrollo del backend para el módulo de actualizar perfil..... 65

Tabla 56 Tarea 2: Desarrollo del frontend para el módulo de actualizar perfil..... 66

Tabla 57 Historias de usuario de la iteración 6 66

Tabla 58 Tarea 1: Diseño de la base de datos para el módulo de multas 67

Tabla 59 Tarea 2: Desarrollo del backend para el módulo de multas 67

Tabla 60 Tarea 3: Desarrollo del frontend para el módulo de multas 67

Tabla 61 Tarea 1: Diseño de la base de datos para el módulo de pagar servicio 68

Tabla 62 Tarea 2: Desarrollo del backend para el módulo de pagar servicio. 68

Tabla 63 Tarea 3: Desarrollo del frontend para el módulo de pagar servicio. 69

Tabla 64: Pruebas Unitarias módulo usuarios 105

Tabla 65:Pruebas unitarias módulo socios 106

Tabla 66:Pruebas unitarias módulo medidores..... 107

Tabla 67: Pruebas unitarias módulo multas 108

Tabla 68: Pruebas unitarias módulo lecturas 109

Tabla 69:Pruebas unitarias módulo pagos 110

Tabla 70: Pruebas unitarias módulo reportes 111

Tabla 71: Pruebas unitarias módulo historial de pagos 112

Tabla 72 Resultado de pruebas realizadas en OWASP ZAP 114

Tabla 73 Tabla de Casos de Prueba módulo de lecturas 120

Tabla 74 Casos de Prueba módulo multas 124

Tabla 75 Casos de Prueba módulo reportes 128

Tabla 76 Casos de prueba de integración 131

INDICE DE FIGURAS

Figura 1. Dirección de la Comunidad 20

Figura 2. Fases de la metodología XP 23

Figura 3. Arquitectura del Sistema Web 26

Figura 4 Diagrama de casos de uso: Administrador, Secretario/a, Tesorero/a 50

Figura 5 Diagrama de casos de uso: socio 51

Figura 6 Modelo Relacional de la Base de datos. 70

Figura 7:Modelo Entidad-Relación 71

Figura 8. Diseño de la interfaz de login de la aplicación web. 72

Figura 9. Diseño de la gestión de usuarios. 73

Figura 10. Diseño de la interfaz para agregar un nuevo usuario. 74

Figura 11. Diseño para editar información del usuario. 75

Figura 12. Diseño de la gestión de socios. 76

Figura 13. Diseño de agregar un nuevo socio. 76

Figura 14. Diseño de editar socio. 77

Figura 15. Diseño de los pagos..... 78

Figura 16. Diseño de Gestión de multas..... 79

Figura 17. Diseño de agregar multas. 80

Figura 18. Diseño de la gestión de lecturas. 81

Figura 19. Diseño de agregar lectura. 82

Figura 20. Diseño de interfaz de reportes. 83

Figura 21. Diseño de historial de pagos. 84

Figura 22. Diseño de interfaz de multas 85

Figura 23.Diseño de pagar servicio. 86

Figura 24 Controladores que manejan las solicitudes de la aplicación. 87

Figura 25 Modelos que interactúan con la base de datos. 87

Figura 26 Archivo de rutas para llamar a las diferentes interfaces. 88

Figura 27 Views de la aplicación 89

Figura 28 Ejemplo del módulo de lecturas 90

Figura 29 Ejemplo del módulo de medidores..... 91

Figura 30 Página de Inicio de Sesión 92

Figura 31 Tabla para la gestión de información de los usuarios. 93

Figura 32 Modal para agregar un nuevo usuario. 93

Figura 33 Modal para desactivar un usuario. 94

Figura 34 Tabla para la gestión de socios de la junta. 94

Figura 35 Modal para agregar un nuevo socio. 95

Figura 36 Modal para desactivar un socio. 95

Figura 37 Tabla para la gestión de los medidores. 96

Figura 38 Modal para agregar un nuevo medidor. 96

Figura 39 Modal para hacer traspaso de nombre de un medidor. 97

Figura 40 Tabla para visualizar los pagos pendientes. 97

Figura 41 Tabla para la gestión de las multas. 98

Figura 42 Modal para generar una nueva multa. 99

Figura 43 Modal para anular una multa. 99

Figura 44 Tabla para registrar lecturas. 100

Figura 45 Modal para agregar nueva lectura. 100

Figura 46 Formulario para enviar notificaciones a los socios. 101

Figura 47 Módulo para generar reportes. 102

Figura 48 Tabla para visualizar el historial de pagos. 102

Figura 49 Tabla para ver y pagar multas. 103

Figura 50 Ventana para pago de la multa 103

Figura 51 Tabla de valor a pagar del servicio. 104

Figura 52 Ventana para pago del servicio. 104

Figura 53 Pruebas realizadas en OWASP ZAP 113

Figura 54 Ejecución de OWASP ZAP 114

Figura 55 Ejecución de pruebas 120

Figura 56 Tabla lecturas con el registro creado 123

Figura 57 Tabla comprobantes con el total calculado 123

Figura 58 Ejecución de pruebas módulo multas 124

Figura 59 Resultado de Base de datos 127

Figura 60 Ejecución de pruebas 128

Figura 61 Ejecución de pruebas 131

Figura 62 Base de datos 134

INTRODUCCIÓN

Actualmente la transformación digital se ha vuelto algo fundamental para optimizar procesos en diferentes organizaciones ya sean públicas o privadas. Las juntas de agua potable como entidades encargadas de garantizar la administración correcta del agua muchas veces enfrentan problemas relacionados con la gestión de pagos, control de multas, registrar socios y usuarios, ya que al hacerlo de forma manual están propensos a errores, retrasos en algunos casos pérdida de información.

Ante este problema, este trabajo propone el desarrollo de una aplicación web para la gestión de pagos y multas, para que permita automatizar los procesos y mejorar la eficiencia, llevando un registro bien ordenado de socios, valores a pagar, generación de multas, registro de lecturas, registro de medidores y generación de reportes.

Para el desarrollo se utilizó la metodología ágil XP ya que nos permitirá adaptarse a los cambios constantes, al tener una entrega progresiva se puede obtener feedback de cada iteración e ir mejorando posibles fallas. Adicional a esto se integró herramientas y tecnologías modernas como lo es laravel, Bootstrap y MySQL, todas estas tecnologías juntas brindan una interfaz amigable, intuitiva, escalable y sobre todo robusta.

RESUMEN

La Junta Administradora Comunitaria de Agua Potable y Saneamiento Buena Esperanza, organización comunitaria dedicada a brindar y garantizar el acceso a servicios básicos de agua potable y saneamiento en la comunidad, sin embargo, uno de los desafíos que enfrentaba la junta de agua es la gestión del pago del servicio básico del agua potable, ya que estos procesos se efectúan manualmente y solo hay una persona encargada para los cobros.

Ante esta problemática este proyecto se desarrolló con el objetivo de sistematizar los procesos que son parte de la gestión de cobros del servicio de agua potable de la Junta Administradora Comunitaria de Agua Potable y Saneamiento La Buena Esperanza, mediante el desarrollo de una aplicación web que permitirá agilizar el proceso de pagos. Con esto se logra disminuir el tiempo de la carga administrativa, ya que el socio puede realizar el pago sin necesidad de acudir a las oficinas de la junta y eso mejora la experiencia totalmente.

Esta aplicación fue desarrollada usando laravel para el frontend, utilizando PHP considerado como un lenguaje de programación de alto nivel. Para el back-end se usó PHP ya que nos brinda una amplia cantidad de librerías que permiten desarrollar programas más robustos y seguros.

La metodología usada fue XP la cuál fue de suma importancia para que el proyecto se haga de manera organizada y poder entregar un software de calidad de acuerdo a los requerimientos que se establecieron.

Palabras clave: Junta de Agua, Tarifa, Metodología XP, socios.

CHUZHAK RIMAY

Buena Esperanza Ayllu Yakumanta Ch'uya Kaymantawan Kamachiq Huñuqa huk ayllu tantanakuymi, ayllupi qallariy yaku upyanapaq ch'uya kay qaranakunatapas qunapaq hinallataq takyachinapaq llamk'aq. Ichaqa, kay yaku kamachiq huñupaq huk sasachakuyninqa qallariy yaku upyanapaq qaranakunamanta qullqi huñuy kamachiyninmi, imaraykuchus kay ruwaykunaqa makillawan ruwakun, hinaspapas qullqi huñuymantaqa huk runallam kamachin.

Kay sasachakuyta qhawarispa, kay rurayqa paqarichisqa karqan Buena Esperanza Ayllu Yaku Ch'uya Kay Kamachiq Huñupaq yaku qaranamanta qullqi huñuy ruraykunata allichanapaq, huk web aplicación nisqata ruwaspa, chaymi qullqi quna rurayta aswan utqayman allichanqa. Kayqa kamachiy llank'ayta pisiyachin, huñu ukhukuna hunt'asqata internetpi pagayta atisqankurayku, mana huñup oficinankunaman riywan, chaytaq allin yachayta anchatapuni allinchan.

Kay aplicaciónqa Laravel nisqawanmi ruwasqa karqan ñawpaq kaqninpaq, PHP nisqatataqmi, chayqa hatun ñiqi programación simi hina qhawarisqam. PHP nisqaqa qhipa kaqninpaqmi servichikurqan, imaraykuchus achka librerías nisqakunata quwanchik, chaykunawanmi aswan sinchi, aswan waqaychasqa programakunata ruwayta atinchik.

Llamk'achisqa ruway yachayqa XP karqan, kayqa ancha chaniyuq karqan proyectota allin wakichisqata ruwanapaq, hinallataq churasqa mañakuykunaman hina allin software qunapaq.

K'askarichina simikuna: Junta de Agua, Arancel, Metodología XP, sociokuna.

ABSTRACT

JACAPS a community organization dedicated to providing and guaranteeing access to basic drinking water and sanitation services in the community, faced one of the challenges of managing payments for the basic drinking water service. These processes were carried out manually, and only one person was responsible for collections.

To address this problem, this project was developed with the objective of systematizing the processes involved in managing water service payments for the Buena Esperanza Community Water and Sanitation Board. This was achieved through the development of a web application that will streamline the payment process. This will reduce the administrative burden, as members can make payments entirely online without having to visit the board's offices, significantly improving the user experience.

This application was developed using Laravel for the frontend and PHP, considered a high-level programming language. PHP was used for the back-end because it provides a wide range of libraries that allow for the development of more robust and secure programs.

The XP methodology was used, which was crucial for the project's organization and the delivery of high-quality software that met the established requirements.

Keywords: Water Board, Tariff, XP Methodology, partners.

CAPÍTULO I

FORMULACIÓN GENERAL DEL PROYECTO

1.1.Tema

Aplicación Web para la gestión de pagos en la Junta Administradora de Agua Potable y Saneamiento La Buena Esperanza, Parroquia Cangahua, Cantón Cayambe.

1.2.Descripción del Problema

La Junta Administradora Comunitaria de Agua Potable y Saneamiento Buena Esperanza, es una organización comunitaria dedicada a brindar y garantizar el acceso a servicios básicos de agua potable y saneamiento en la comunidad, misma que está ubicada en el cantón Cayambe, parroquia Cangahua, la cual está conformada por 5 miembros cuyas denominaciones son: Presidenta, Secretario, Tesorero, Primer Vocal y Segundo Vocal, cuenta aproximadamente con 300 socios los cuales hacen uso de sus servicios, a través de su gestión buscan administrar eficientemente los recursos hídricos y atender las necesidades de sus asociados.

Sin embargo, uno de los desafíos que enfrenta la junta de agua es la gestión del pago del servicio básico del agua potable, ya que estos procesos se efectúan manualmente y solo hay una persona encargada para los cobros es la tesorera, quien hace efectivo el pago la primera semana del mes en un horario específico, durante ese día los socios deben acudir a la junta, hacer fila y al momento del cobro proporcionar el nombre, número de cédula y el sector al que pertenecen, posteriormente se registra el pago y se emite un recibo, en algunos casos los usuarios tienen problemas, al desconocer el valor a pagar y al no disponer de esa cantidad no pueden realizar el pago, lo que genera que al siguiente mes se le agregue una tarifa adicional por atraso.

En general, los problemas identificados son los siguientes:

- Largas filas y aglomeraciones el día de los cobros.
- Horarios limitados que pueden no ser convenientes para todos los socios.
- Dificultad para acceder a la información y conocer el estado de sus cuentas y los pagos realizados.
- Retrasos en los pagos debido a la falta de notificaciones o recordatorios.
- Entrega atrasa de informes a los directivos de la junta.

1.3.Justificación

Actualmente, las aplicaciones web han jugado un papel fundamental en esta revolución digital, ya que permiten a los usuarios acceder y utilizar diversos servicios y herramientas desde cualquier dispositivo con acceso a Internet (Ridge, 2023).

Una de las ventajas que nos brindan estas aplicaciones es automatizar y mejorar procesos. Según Mejía (2009), “La automatización consiste en tener a mano una información en tiempo real que sea accesible a todo el personal involucrado en la operación” (Mejía, 2009).

Por lo tanto, la automatización de diferentes procesos mejora el flujo de trabajo en una organización, ya que es posible reducir costos, tiempo, minimizar fallas y aumentar la productividad.

Las juntas de agua comunitarias al tener la responsabilidad de administrar los servicios que otorgan a sus socios o usuarios se ven obligados a buscar formas de realizar los cobros y al no contar con los suficientes recursos tecnológicos y económicos los realizan de forma manual, lo que genera tiempos de espera inadecuados e inconformidad para los socios.

En ese sentido, el desarrollo de un sistema para la gestión de pagos en la junta administradora comunitaria de agua potable y saneamiento “Buena Esperanza” contribuye a:

- Mejorar la eficiencia Operativa, ya que se reducirá significativamente la carga de trabajo administrativo.
- Facilitar la gestión y control de dichos pagos.
- Reducir tiempos de espera.
- Mejorar la calidad del servicio a los habitantes de la comunidad.

El proyecto propuesto beneficia directamente a la directiva y a los socios de la junta de agua, atendiendo un problema real mediante un sistema tecnológico que sea accesible y eficiente, además el proyecto puede servir como base para futuras iniciativas de modernización en otras comunidades con características similares.

1.4.Objetivos:

1.4.1. General

Implementar una aplicación web para la gestión de pagos de la Junta de Agua Potable y Saneamiento la Buena Esperanza para la optimización de sus procesos.

1.4.2. Específicos

- Establecer los requerimientos funcionales y no funcionales del sistema de gestión de pagos de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza.
- Diseñar la arquitectura de la aplicación web, y la estructura de la base de datos que se ajuste a las necesidades de los usuarios.
- Sistematizar los procesos de pagos de servicios a través de una interfaz amigable e intuitiva.

- Testear la aplicación web para detectar los posibles errores y comprobar si cumple con los requisitos del cliente.

CAPÍTULO II

MARCO TEÓRICO

2.1.Antecedentes

Guamán (2023), en su trabajo de titulación "Propuesta y desarrollo de un prototipo para el registro de pagos en la junta de agua Monseñor "Leónidas Proaño" de la comunidad de Guartanchún almacenamiento en la nube". Desarrolló un prototipo para la recaudación de los cobros de agua potable de a través de recursos en la nube para que de esta manera se puedan agilizar los procesos de la Junta de Agua. En la creación de este prototipo se utilizó la metodología Scrum, la cual dividió en 4 Sprint, cada uno de suma importancia debía cumplir con tareas específicas y con fechas límites. Todo esto con el fin de tener un prototipo que cumpla con todos los requerimientos que se establecieron al inicio. Algunos de los resultados que se obtuvieron son: cumplir con los parámetros de integridad y consistencia de la información, agilizar los procesos de búsqueda, alojamiento en la nube, realizar facturas de forma eficiente, consultar valores pendientes y generación de reportes.

Flores (2024), en su trabajo de titulación "Aplicación web progresiva para la gestión de cobranza del servicio del agua de regadío para la junta modular de riego y drenaje "Quillán Loma"". El trabajo se basa en la intención de automatizar los procesos que interviene al realizar el cobro del servicio de agua de riego implementando una aplicación web progresiva, con el fin de disminuir el tiempo empleado en realizar búsqueda de valores a pagar y en consecuencia realizar el cobro respectivo. Para el desarrollo de esa aplicación se usó React en la versión 18.2.0 para el frontend, para el Backend se utilizó Node.js versión 21.4.0 y la metodología de desarrollo aplicada fue XP (Xtreme

Programming), la misma que contó con cuatro fases: Planeación, Diseño, Desarrollo, Pruebas, las cuales cada una tenía su tiempo de ejecución y cumplía los requisitos establecidos. Como resultado se resalta que la implementación de esta aplicación ha mejorado la gestión de la información y los procesos que conlleva la gestión de cobros.

Semblantes (2021), en su trabajo de titulación "Desarrollo de un sistema web para el proceso de facturación, toma de lectura y consulta de planillas, en la Junta Administradora de Agua potable mollepamba de la parroquia picaihua". Desarrolló un sistema web para el proceso de facturación, tomar lecturas y consultar planillas, el eje principal del proyecto desarrollado debe ser de fácil accesio y comprensión para las personas que lo manejen, y al mismo tiempo debe ser intuitiva para los usuarios que ingresen para que puedan consultar de forma clara y sencilla sus valores a pagar. Para el desarrollo de esa aplicación se usó la metodología OpenUP considerando el tamaño del proyecto, tiempo de entrega, número de personas que intervienen. El framework elegido fue .NET ya que se destaca que tiene muchas ventajas importantes como: disminuir tiempos de desarrollo al contar con librerías implementadas.

Tras esta revisión bibliográfica de trabajos de grado se puede identificar la importancia que tienen los sistemas web en el control de gestión de pagos de las juntas de agua y nos ofrecen una visión de todos los desafíos y los diferentes enfoques que tienen estos sistemas asociados con la gestión de cobros, los mismos que nos brindan una base para continuar con este proyecto y concluirlo de manera óptima.

2.2.Científico

2.2.1. Metodologías de Desarrollo de Software

Una metodología de desarrollo de software es un proceso utilizado para desarrollar un producto en particular, es decir, es una forma de organizar el desarrollo por parte de un

equipo de desarrolladores (Yelenevych, 2023), estas metodologías son las que nos proporcionarán una guía al equipo de desarrollo, estableciendo las fases, actividades necesarias para llevar a cabo un proyecto de software de manera eficiente y efectiva.

2.2.2. Metodologías Ágiles

Las metodologías ágiles son aquellas que permiten adaptar la forma de trabajo a las condiciones del proyecto, consiguiendo flexibilidad e inmediatez en la respuesta para amoldar el proyecto y su desarrollo a las circunstancias específicas del entorno. Gracias a los sprints y la entrega continua, los equipos pueden hacer ajustes rápidamente, mejorando la calidad del producto y reduciendo errores (Sotomayor, 2024).

Las metodologías ágiles más comunes o más utilizadas son: XP, Scrum, Kanban y estas pueden adaptarse a diversos tipos de proyectos y definir el proceso de desarrollo e implementación.

Tabla 1. Comparativa de Metodologías de Desarrollo de Software

	XP	**Scrum**	**Kanban**
Tamaño del equipo	Equipos pequeños (hasta 12 miembros).	Equipos medianos (7-10 personas).	Sin límite definido, adaptable a cualquier tamaño.
Etapas	- Planificación - Diseño - Codificación -Pruebas	- Sprint Planning - Daily Scrum - Sprint Review - Retrospective	- Flujo continuo, sin etapas fijas (se visualiza el trabajo en columnas)
Roles	-Tracker - Programador - Cliente - Coach -Tester - Manager	- Product Owner - Scrum Master - Equipo de Desarrollo	- Sin roles definidos (equipos auto organizados)
Artefactos clave	- Historias de usuario - Tarjetas CRC - Prototipos (mockups)	- Product Backlog - Sprint Backlog - Gráfico Burndown	- Tablero Kanban

Requiere	Programación individual o de pareja	Grupos entre 3 a 9 miembros	No aplica
Entrega de producto	Releases frecuentes (incluso diarias).	Incrementos al final de cada sprint.	Entrega continua según capacidad del equipo.
Ventajas	- Alta calidad de código - Feedback constante	- Estructura clara - Entregas incrementales - mejora continua	- Sin plazos fijos - optimización de flujo - Ideal para soporte/mantenimiento
Desventajas	- Requiere alta disciplina técnica.	- Rigidez durante el Sprint, requiere compromiso del equipo	- Falta de estructura - Riesgo de desorganización - Sin límites claros
Adaptabilidad	- Ideal para equipos pequeños con alta incertidumbre técnica	- Bueno para proyectos con requisitos cambiantes, pero no caóticos	- Perfecto para entornos impredecibles o prioridades variables

Según las características mencionadas, la mejor opción para el proyecto es la metodología XP (programación extrema), debido a su adaptabilidad a los cambios requeridos por el cliente, su enfoque en equipos pequeños, la posibilidad de realizar entregas frecuentes y su estructura basada en historias de usuario lo que nos garantiza flexibilidad y calidad en cada iteración.

2.2.3. Tarifa Base

Precio unitario fijado por las autoridades para los servicios públicos realizados a su cargo (Salazar, 2022)

2.3.Conceptual

2.3.1. Sistema web

Los sistemas web son aplicaciones o plataformas informáticas que funcionan a través de una red, generalmente internet, y son accesibles desde navegadores web en dispositivos

como computadoras, tabletas o teléfonos móviles. Estos sistemas permiten a los usuarios interactuar, compartir información, realizar transacciones y de interactuar en línea (Borovskoy, 2023).

2.3.2. Base de datos

Una base de datos es un software informático que permite almacenar grandes cantidades de información, de una forma organizada y accesible para su futuro uso. Este sistema de almacenamiento de datos permite realizar búsquedas o ingresar nuevas informaciones de una manera sencilla y rápida desde un ordenador (López, 2021).

2.3.3. Gestor de Base de Datos

Un gestor de bases de datos es una aplicación de software que está diseñada para facilitar la organización, el almacenamiento, la recuperación y la gestión de datos dentro de una base de datos. Actúa como intermediario entre los usuarios y la base de datos, garantizando un manejo y acceso eficientes a los datos.

Tabla 2. Clasificación de los tipos de gestores de base de datos.

Tipo de gestores de base de Datos	**Características**	**Ejemplos**
Bases de Datos Relacionales	- Almacenan datos estructurados en tablas (filas y columnas). - Usan SQL para consultas. - Garantizan Atomicidad, Escalabilidad, Consistencia, Aislamiento, Durabilidad. - Ideal para transacciones complejas y relaciones definidas.	- MySQL - PostgreSQL - SQL Server - Oracle - SQLite
Bases de Datos no Relacionales	- Almacenan datos no estructurados o semiestructurados (JSON, documentos, grafos, etc.). - Sin esquema fijo, mayor flexibilidad. - Escalabilidad horizontal. - Optimizadas para grandes volúmenes de datos y velocidad.	- MongoDB - Firebase - Cassandra, - Redis - Couchbase

En este caso se eligió MySQL como gestor de base de datos gracias a su facilidad de integración, su rendimiento, su escalabilidad para almacenar grandes volúmenes de datos.

2.3.4. Lenguajes de Programación

Un lenguaje de programación es un conjunto de reglas y sintaxis que permite a los programadores dar instrucciones a una computadora para realizar tareas específicas. Estos lenguajes actúan como un puente entre los humanos y las máquinas, facilitando la creación de software, aplicaciones y sistemas que ejecutan las instrucciones definidas en el código (Cimas, 2020).

2.3.5. Framework

Es una estructura o conjunto de herramientas y componentes que nos proporciona una base para desarrollar aplicaciones o crear páginas web de una manera más organizada, robusta y escalable. Es como toda una serie de componentes y directrices sobre los cuales podemos montar cualquier aplicación, aportando una estructura prediseñada que permite a los equipos de trabajo centrarse en resolver problemas específicos, sin tener que pensar tanto en la arquitectura del proyecto online (García, 2025).

Tabla 3: Comparativa de frameworks para desarrollo

Característica	Laravel	React	Angular
Lenguaje	PHP	JavaScript	TypeScript
Arquitectura	Modelo-Vista-Controlador	Arquitectura basada en componentes combinada con un flujo de datos unidireccional	Modelo-Vista-Controlador

Curva de Aprendizaje	Alta: intuitiva gracias a su documentación y estructura clara.	Media	Alta
Seguridad	Integrada (CSRF, XSS, hashing, auth)	Reacter Router Dom	Guards
Entorno de Ejecución	Servidores Apache(Apache, Nginx)	Node.js	Node.js
API RESTful	Si	Si	Si
Base de datos	MySQL, PostgreSQL, SQLite, SQL Server	Mongo DB PostgreSQL MySQL	Firebase MySQL Mongo DB
Comunidad	Muy grande, activa y bien documentada	Comunidad grande y activa	Documentación extensa
Tipo de Proyecto	Sistemas web completos, APIs, apps empresariales	Flexible para aplicaciones más pequeñas	Para proyectos empresariales.

Tras un análisis de estas opciones, se elegio laravel principalmente por ofrecer una solución integral y orientada a aplicaciones web completas, por su compatibilidad con múltiples gestores de bases de datos brindando mayor flexibilidad y escalabilidad, además su arquitectura Modelo-Vista-Controlador (MVC) ofrece un equilibrio entre

productividad, organización y seguridad, y a su vez cuenta con una amplia documentación que facilita la continuidad de los proyectos.

2.3.6. Lectura de Medidor

La lectura de medidores de agua es un proceso crucial para gestionar el consumo y la facturación del agua en entornos residenciales, comerciales e industriales. Consiste en medir el volumen de agua consumido por una propiedad durante un período específico.

2.3.7. Consumo de agua

Es el volumen de agua utilizada por persona, este volumen es medido o expresado en litros (metros cúbicos). Para su calculo se utilizan datos del consumo anterior y el consumo actual.

2.3.8. Cobro de agua

Es la recaudación mensual de tasas por el servicio de agua potable y alcantarillado, calculado en base a los metros cúbicos consumidos en un periodo.

2.3.9. PHP

Es un lenguaje de programación orientado a objetos, de código abierto muy popular para el desarrollo web que puede ajustarse en HTML, es considerado como uno de los lenguajes más potentes y populares para el desarrollo web y de aplicaciones (Ortega, 2024).

Las características principales de este lenguaje son:

- Código abierto y gratuito
- Permite separación de códigos
- Tiene una comunidad amplia donde se puede compartir conocimientos y encontrar información.
- Permite desarrollo de páginas dinámicas.

2.3.10. HTML

HTML es HyperText Markup Language, y quiere decir “lenguaje de marcado de hipertexto”. Se usa para la elaboración de páginas web por medio de etiquetas que sirven para definir el texto y otros elementos que compondrán nuestras páginas web.HTML es un lenguaje de marcación que tiene sus normas, su estructura y una serie de convenciones, es muy sencillo y muy fácil de aprender y representa un standard que sirve como referencia del software que conecta con la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código HTML que además permite el contenido multimedia (imágenes, vídeos, sonido, juegos, animaciones) dentro de las mismas (Perandones, 2024).

2.3.11. CSS

CSS (Cascading Style Sheets) es un lenguaje de diseño web que permite dotar de estilo gráfico a las páginas escritas en HTML**.** Actualmente, su uso en desarrollo web es generalizado, pues el 97 % de las webs emplean este lenguaje. CSS permite declarar de forma sencilla cómo se verán los diferentes elementos de una web, dotándolos de posición, forma, color, fuente, etc (Ferreira, 2023).

2.3.12. Bootstrap

Bootstrap es un framework de desarrollo web gratuito y de código abierto. Está diseñado para facilitar el proceso de desarrollo de los sitios web responsivos y orientados a los dispositivos móviles, proporcionando una colección de sintaxis para diseños de plantillas (Deyimar, 2023).

2.3.13. Servidor Web

Un servidor web es un ordenador que almacena, procesa y entrega archivos de sitios web a los usuarios desde un navegador. Los servidores web están formados por hardware y software que utilizan el Protocolo de Transferencia de Hipertexto (HTTP) para responder a las peticiones de los usuarios de la web realizadas a través de la World Wide Web (Boada, 2025).

2.3.14. Front-End

Es la parte del desarrollo que se encarga de crear las interfaces con las que el usuario interactúa, son todas aquellas tecnologías de diseño y desarrollo web que corren en el navegador y que se encargan de la interactividad con los usuarios. Para esta tarea se requiere tecnología como HTML5 para manejo del contenido, CSS para crear el diseño, maquetación y la parte visual de la aplicación y el lenguaje de programación Javascript para usar algunos frameworks o librerías que expanden capacidades de interacción de las interfaces y consumir los servicios proporcionados por el backend (Zelaya, 2020).

2.3.15. Backend

Es la capa de acceso a datos de un software o cualquier dispositivo, que no es directamente accesible por los usuarios, además contiene la lógica de la aplicación que maneja dichos datos. El Backend también accede al servidor, que es una aplicación especializada que entiende la forma como el navegador realiza las peticiones (Zelaya, 2020).

2.3.16. Pruebas de Aceptación

Las pruebas de aceptación son un proceso de control de calidad que determina el grado de aprobación de una aplicación por parte de los usuarios finales. Un equipo de control de calidad realiza pruebas de aceptación para garantizar que el software o la aplicación se ajusten a los requisitos del negocio y a las necesidades del usuario final. Una prueba de aceptación arroja un resultado de aprobado o reprobado (Gillis, 2024).

2.4.Legal

2.4.1. Ley Orgánica de Recursos Hídricos, Usos y Aprovechamiento del Agua

Art. 40.- Las juntas administradoras de agua potable son organizaciones comunitarias, sin fines de lucro, que tienen la finalidad de prestar el servicio público de agua potable. Su accionar se fundamenta en criterios de eficiencia económica, sostenibilidad del recurso hídrico, calidad en la prestación de los servicios y equidad en el reparto del agua.

2.4.2. Ley Orgánica de protección de Datos

Ley Orgánica de Protección de Datos. Constitución de la República del Ecuador. (2008). Artículo 66, numeral 19. "Garantía y reconocimiento del derecho a la protección de datos de carácter personal, que incluye el acceso y la toma de decisiones sobre dicha información y datos y su adecuada protección. La recopilación, archivo, procesamiento, difusión o difusión de dichos datos personales requiere el permiso del titular o el permiso conforme a la ley."

2.4.3. Ley de Propiedad Intelectual

Sección V

Art. 28.- Los programas de ordenador se consideran obras literarias y se protegen como tales. Dicha protección se otorga independientemente de que hayan sido incorporados en un ordenador y cualquiera sea la forma en que estén expresados, ya sea en forma legible por el hombre (código fuente) o en forma legible por máquina (código objeto), ya sean programas operativos y programas aplicativos, incluyendo diagramas de flujo, planos, manuales de uso, y en general, aquellos elementos que conformen la estructura, secuencia y organización del programa.

Art. 29.- Es titular de un programa de ordenador, el productor, esto es la persona natural o jurídica que toma la iniciativa y responsabilidad de la realización de la obra.

Se considerará titular, salvo prueba en contrario, a la persona cuyo nombre conste en la obra o sus copias de la forma usual. Dicho titular está además legitimado para ejercer en nombre propio los derechos morales sobre la obra, incluyendo la facultad para decidir sobre su divulgación.

El productor tendrá el derecho exclusivo de realizar, autorizar o prohibir la realización de modificaciones o versiones sucesivas del programa, y de programas derivados del mismo.

2.4.4. Ley de Comercio Electrónico. Firmas y mensajes de datos.

Considerando:

Que el uso de sistemas de información y de redes electrónicas, incluida la internet, ha adquirido importancia para el desarrollo del comercio y la producción, permitiendo la realización y concreción de múltiples negocios de trascendental importancia, tanto para el sector público como para el sector privado;

Art. 1.- Esta ley regula los mensajes de datos, la firma electrónica, los servicios de certificación, la contratación electrónica y telemática, la prestación de servicios electrónicos, a través de redes de información, incluido el comercio electrónico y la protección a los usuarios de estos sistemas (*LEY DE COMERCIO ELECTRONICO, FIRMAS Y MENSAJES DE DATOS*, 2002).

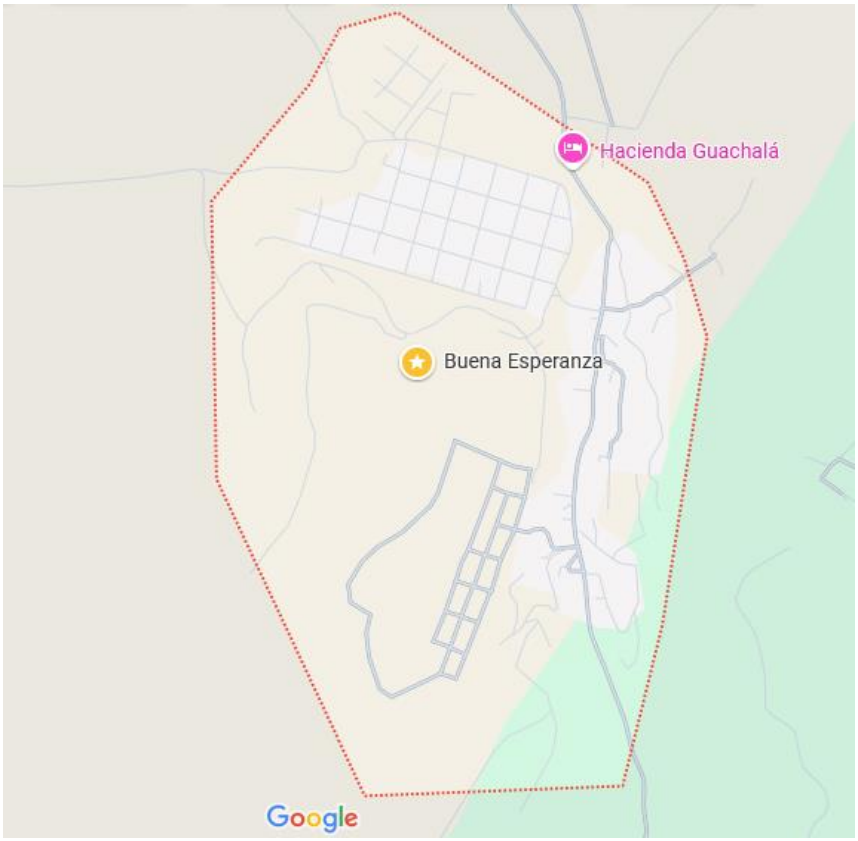
2.5.Georeferencial

La Junta Administradora de Agua Potable y Saneamiento está ubicada en la comunidad la Buena Esperanza, en el cantón Cayambe, parroquia Cangahua.

Latitud: 0°1°42.046”

Longitud: 78°10°35.309”

Figura 1. Dirección de la Comunidad



CAPÍTULO III

METODOLOGÍA

3.1. Metodología de Desarrollo de Software

Para el desarrollo del proyecto se utilizó la metodología ágil XP.

3.1.1. Programación Extrema XP

Es una metodología de desarrollo de software ágil, creada específicamente para promover la aplicación de prácticas de ingeniería apropiadas para la creación de software. Tiene como objetivo principal que un equipo de desarrollo pueda producir software de mejor calidad de forma constante y a su vez busca promover una buena calidad de vida para el equipo (Mendoza, 2020).

En este caso, al no contar con un equipo de desarrollo se trabajó con ciclos cortos, es decir dividir objetivos semanales teniendo en cuenta las características y funcionalidades más importantes, en cada ciclo se definió tareas concretas a realizar, ya sea para implementar un módulo o corregir algún error, de esta manera se evita sobrecarga de trabajo y se avanza de manera progresiva

a. Fases

Esta metodología consta de 4 fases que son:

- **Planificación**

Este es el primer paso, aquí se identificaron las historias de usuario para tener una idea o una base de los objetivos que va a tener el proyecto. Estas historias de usuario son tarjetas donde se detallan las funcionalidades específicas y se determinan basándonos en las necesidades del cliente, cada función o historia de usuario se divide según su prioridad y se descompone en versiones (Mancuzo, 2020).

Para la elaboración de estas historias de usuario se utilizó técnicas como la entrevista y la observación, de esta manera conocer como es el flujo de trabajo actualmente.

- **Diseño**

En esta fase se tomaron en cuenta las historias de usuario y se diseña la arquitectura del código, se decide el lenguaje de programación, el entorno, las bibliotecas y los marcos de trabajo que sea adapten mejor al proyecto.

El diseño incluye una planificación de programación colaborativa, flexible e integral, para luego pasar a la siguiente fase, donde se evaluará una versión de prueba (Mancuzo, 2020).

Para esta fase se realizó diagramas de casos de uso, identificando los actores y los objetivos de cada uno, además se realizó un diagrama entidad relación identificando las

entidades claves que va a tener el sistema, se definió sus atributos y las relaciones que van a tener. Para esto se utilizó Draw.io, una herramienta gratuita que nos permite crear diferentes diagramas, gracias a su facilidad de uso y su interfaz intuitiva.

- **Codificación**

En esta fase se trabajó en la codificación real del software. El código se escribió de manera iterativa y fue probado regularmente. Esto asegura que el código sea de alta calidad y en caso de haber errores, estos se puedan solucionar rápidamente.

- **Pruebas**

Una de las características de la metodología XP es el cambio constante, por eso cuando el código de una función está listo se somete a una serie de pruebas unitarias continuas, con el objetivo de corregir fallas periódicamente (Mancuzo, 2020).

Esta fase se complementó con revisiones constantes realizadas por el Coach y los pares académicos. Estas revisiones ayudaron a detectar errores tempranos y garantizar que se vaya cumpliendo los requisitos que se plantearon en la planificación.

Además, se realizaron pruebas de aceptación con el objetivo de validar que cada función o módulo individual funcione correctamente antes de su puesta en marcha.

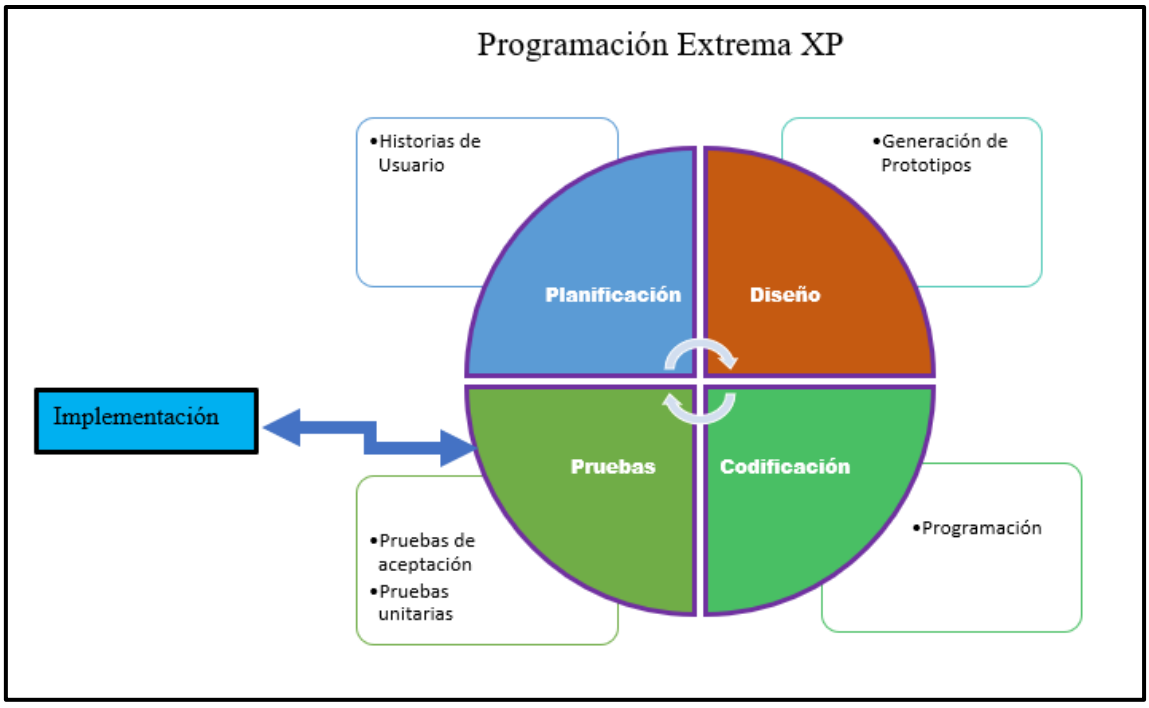


Figura 2. Fases de la metodología XP

3.2. Gestión de Versiones

Para el correcto desarrollo de este proyecto se usó un sistema de control de versiones utilizando GitHub. Este mecanismo ha permitido que pueda llevar un seguimiento detallado de los cambios que se vayan realizando en el código, gracias a esto se tiene la posibilidad de revertir o deshacer cambios no deseados en caso de que sea necesario, con esto se puede mantener la calidad del proyecto en cada iteración o fase.

3.3.Técnicas e Instrumentos de Recopilación de Datos

3.3.1. Entrevista

Es una técnica de recogida de información que además de ser una de las estrategias utilizadas en procesos de investigación, tiene ya un valor en sí misma. El objetivo de una entrevista es obtener información de forma oral y personalizada sobre acontecimientos, experiencias, opiniones de personas (Folgueiras, 2016).

Para el proyecto propuesto, se aplicó a la presidenta de la junta de agua, con la finalidad de analizar la situación actual de los procesos de pagos.

3.3.2. Observación

La observación es una técnica que consiste en analizar en tiempo real el flujo de trabajo Esta técnica fue útil para:

- Observar el proceso de pago, toma y registro de lectura.

Como se puede evidenciar la matriz de observación aplicada fue la siguiente. Anexo 2 Matriz de Observación

CAPÍTULO IV

INGENIERÍA DEL PROYECTO

La metodología XP permitió el desarrollo eficiente del sistema web para la Junta de Agua gracias a su enfoque en la adaptación y mejora continua durante el proceso de desarrollo, la misma que consta de las siguientes fases: Planificación, Diseño, Codificación, Pruebas.

4.1. Fase I: Planificación

La primera fase en la metodología XP es la planificación, en la que se determinarán los requisitos del sistema y su funcionamiento.

a. Requerimientos del sistema

En base a la recolección de información es importante que el proceso de cobros a los socios sea automatizado mediante una aplicación web y con esto reducir la carga del personal administrativo y mejorar los procesos.

Los módulos a implementar son:

Módulo de Autenticación: Este módulo permitirá autenticar al administrador, tesorero, secretario y a los socios de la Junta de Agua.

Módulo de usuarios: Este módulo mostrará el listado de los usuarios de la junta y su respectivo rol, además se podrá agregar nuevos usuarios, editar su información o desactivarlos.

Módulo de socios: Este módulo mostrará el listado de los socios de la junta, además se podrá agregar nuevos socios, editar su información o desactivarlos.

Módulo de Medidores: Este módulo mostrará el listado de medidores activos pertenecientes a los socios de la junta, además se podrá registrar nuevos medidores, ver su información, hacer traspasos de nombre o desactivarlos.

Módulo de Lecturas: Aquí se mostrará las lecturas registradas de los usuarios, además se irá agregando las lecturas de cada mes para calcular el valor a pagar.

Módulo de Pagos: Se mostrará los detalles del pago que debe realizarse de acuerdo al servicio.

Módulo de multas o infracciones: Se podrá registrar la multa pertinente, seleccionando la falta cometida.

Módulo de Reportes: Se podrá generar reportes de todo tipo.

Módulo de Historial de Pagos: En este módulo el socio puede ver su historial de pagos y podrá visualizar sus comprobantes correspondientes

Módulo de pago de servicios: En este módulo el socio consulta el valor a pagar por el consumo de agua y tiene la opción de pagarlo.

b. Arquitectura del Sistema

Para esta aplicación web se aplicó la arquitectura MVC (modelo-vista-controlador), además se usó el gestor de base de datos MYSQL para gestionar y administrar la información.

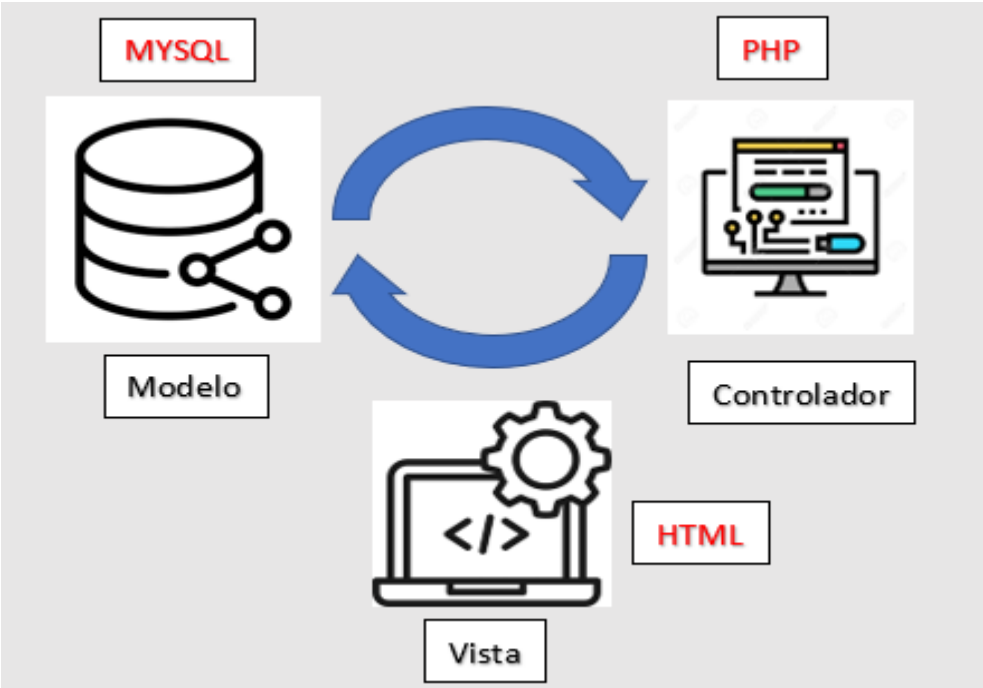


Figura 3. Arquitectura del Sistema Web

c. Roles

Basado en XP se consideró los siguientes roles, definiendo las actividades a realizar y las responsabilidades del equipo.

Tabla 4. Roles del Proyecto

Nombre	Rol	Función

Marco Quishpe	Programador y Tester	Realizar la planificación, análisis, diseño, codificación y hacer las respectivas pruebas funcionales de la aplicación.
Ing. Galuth García	Coach y seguimiento	Supervisar las actividades del proyecto y realizar revisiones constantes de los avances en el desarrollo.
Sra. Mery Coyago	Cliente	Revisar las funcionalidades de la aplicación.

d. Historias de Usuario

En las historias de usuario se describe rápidamente lo que se quiere implementar en la aplicación web de acuerdo con los requerimientos establecidos.

Tabla 5: Modelo de Historias de usuario

Historia de usuario	Número: 1
Nombre de Historia:	
Usuario:	
Prioridad de negocio:	Puntos estimados:
	Puntos Reales:
Programador Responsable:	Iteración asignada:
Descripción:	
Criterios de aceptación	

Tabla 6:Historia de Usuario: Autenticación al sistema.

Historia de usuario	Número: 1
Nombre de Historia: Autenticación al sistema	
Usuario: Administrador de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza.	
Prioridad de negocio: Alta	Puntos estimados: 9
	Puntos Reales: 10
Programador Responsable: Marco Quishpe	Iteración asignada: 1
Descripción: Como administrador de la Junta debe ingresar al sistema verificando el usuario y contraseña, para crear a los usuarios que tienen acceso al sistema y gestionar los procesos de cobros del servicio de agua potable.	
Criterios de aceptación	
Criterio de aceptación 1 Dado que el usuario se encuentre en la página de inicio de sesión Cuando ingrese el usuario y contraseña Entonces el usuario accederá al sistema con un rol específico (Administrador, Tesorero, secretario, Usuario) Criterio de aceptación 2	

Dado que el usuario se encuentre en la página de inicio de sesión Cuando ingrese de manera errónea el usuario y contraseña Entonces la aplicación desplegará un aviso de credenciales incorrectas Criterio de aceptación 3 Dado que el usuario olvide su contraseña Cuando seleccione la opción de olvide mi contraseña Entonces la aplicación desplegará una pantalla para la recuperación en el que podrá proporcionar información para generar una nueva clave.

Tabla 7:Historia de Usuario: Módulo de usuarios.

Historia de usuario	Número: 2
Nombre de Historia: Módulo de usuarios	
Usuario: Administrador de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza.	
Prioridad de negocio: Alta	Puntos estimados: 17
	Puntos Reales: 15
Programador Responsable: Marco Quishpe	Iteración asignada:2
Descripción: Como administrador de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza, debe ingresar al módulo de usuarios	

Para gestionar la información de los usuarios, agregar, editar, desactivar usuarios
Criterios de aceptación
Criterio de aceptación 1: Realizar búsquedas de usuarios por nombre, la aplicación filtrará en una tabla los datos registrados en relación al nombre que haya ingresado en el buscador. Criterio de aceptación 2: Modificar la información del usuario, para lo cual debe seleccionar el botón de editar, y en la pantalla se desplegará un modal para modificar la información y guardarla. Criterio de aceptación 3: Desactivar un usuario Debe dar click en el botón de desactivar Entonces se desplegará una alerta de confirmación para desactivar. Criterio de aceptación 4: Agregar un nuevo usuario Debe dar click en el botón de agregar usuario y se desplegará un modal para ingresar la información del nuevo usuario y guardarla.

Tabla 8:Historia de Usuario: Módulo de socios.

Historia de usuario	Número: 3
Nombre de Historia: Módulo de Socios	
Usuario: secretario de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza.	

Prioridad de negocio: Alta	Puntos estimados: 13
	Puntos Reales: 15
Programador Responsable: Marco Quishpe	Iteración asignada:2
Descripción: Como secretario de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza Debe ingresar al módulo de socios Para gestionar la información de los socios de la junta, agregar, o desactivar.	
Criterios de aceptación	
Criterio de aceptación 1: Dado que el secretario necesite buscar al socio por su nombre Debe ingresar la información en el buscador Entonces la aplicación filtrará en una tabla los datos en relación al nombre que haya ingresado en el buscador. Criterio de aceptación 2: Dado que el secretario necesite desactivar un socio Debe dar click en el botón de desactivar Entonces se desplegará una alerta de confirmación. Criterio de aceptación 3: Dado que el secretario quiera agregar un nuevo socio Debe dar click en el botón de agregar socio Entonces se desplegará un modal para ingresar la información del nuevo socio y guardarla.	

Tabla 9:Historia de Usuario: Módulo de medidores

<table>
<tr><td>Historia de usuario</td><td>Número: 4</td></tr>
<tr><td colspan="2">Nombre de Historia: Módulo de medidores</td></tr>
<tr><td colspan="2">Usuario: secretario de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza.</td></tr>
<tr><td rowspan="2">Prioridad de negocio: Alta</td><td>Puntos estimados: 16</td></tr>
<tr><td>Puntos Reales: 15</td></tr>
<tr><td>Programador Responsable: Marco Quishpe</td><td>Iteración asignada:3</td></tr>
<tr><td colspan="2">Descripción:
Como secretario de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza
Debe ingresar al módulo de medidores
Para gestionar la información de los medidores registrados en la junta para agregar, hacer traspasos, desactivar</td></tr>
<tr><td colspan="2">Criterios de aceptación</td></tr>
<tr><td colspan="2">Criterio de aceptación 1:
Dado que el secretario quiera agregar un nuevo medidor
Debe dar click en el botón de añadir medidor
Entonces se desplegará un modal para ingresar la información del nuevo medidor y guardarla.
Criterio de aceptación 2:</td></tr>
</table>

Dado que el administrador requiera hacer un traspaso de un medidor Debe dar click en el botón de traspasos. Entonces en la pantalla se desplegará un modal para ingresar la información nueva y guardarla.

Tabla 10:Historia de Usuario: Módulo de multas.

Historia de usuario	Número: 5
Nombre de Historia: Módulo de multas	
Usuario: Tesorero de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza.	
Prioridad de negocio: Alta	Puntos estimados: 14
	Puntos Reales: 12
Programador Responsable: Marco Quishpe	Iteración asignada:3
Descripción: Como administrador de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza Debe ingresar al módulo de multas Para gestionar las multas	
Criterios de aceptación	
Criterio de aceptación 1: Dado que el tesorero quiera ingresar al módulo de multas En el menú debe seleccionar la opción de multas	

Entonces le mostrará las multas detallando el valor y la razón

Criterio de aceptación 2:

Dado que el tesorero requiera generar una multa al socio

Debe dar click en el botón de generar multa

Entonces se le desplegará un modal para añadir una multa, indicando el usuario, la fecha, el valor y la razón.

Criterio de aceptación 3:

Dado que el tesorero quiera buscar una multa por el nombre del socio

Debe ingresar la información en el buscador

Entonces la aplicación filtrará en una tabla las multas relacionadas con ese nombre.

Criterio de aceptación 4:

Dado que el tesorero necesite filtrar las multas por estado: multas pagadas o multas pendientes

Debe dar click en la opción filtrar por estado

Entonces la aplicación filtrará la información según el estado seleccionado.

Tabla 11:Historia de Usuario: Módulo de lecturas.

<table>
<tr><td>Historia de usuario</td><td>Número: 6</td></tr>
<tr><td colspan="2">Nombre de Historia: Módulo de lecturas</td></tr>
<tr><td colspan="2">Usuario: Tesorero de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza.</td></tr>
<tr><td rowspan="2">Prioridad de negocio: Alta</td><td>Puntos estimados: 10</td></tr>
<tr><td>Puntos Reales: 12</td></tr>
<tr><td>Programador Responsable: Marco Quishpe</td><td>Iteración asignada:3</td></tr>
<tr><td colspan="2">Descripción:
Como tesorero de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza
Debe ingresar al módulo de lecturas
Para ingresar nuevas lecturas</td></tr>
<tr><td colspan="2">Criterios de aceptación</td></tr>
<tr><td colspan="2">Criterio de aceptación 1:
Dado que el tesorero quiera ingresar las lecturas
Debe dar click en agregar lecturas
Entonces debe buscar el medidor a registrar e ingresar la lectura actual, y guardar</td></tr>
</table>

Tabla 12:Historia de Usuario: Módulo de pagos.

<table>
<tr><td>Historia de usuario</td><td>Número: 7</td></tr>
<tr><td colspan="2">Nombre de Historia: Módulo de pagos</td></tr>
<tr><td colspan="2">Usuario: Tesorero de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza.</td></tr>
<tr><td rowspan="2">Prioridad de negocio: Alta</td><td>Puntos estimados: 13</td></tr>
<tr><td>Puntos Reales: 12</td></tr>
<tr><td>Programador Responsable: Marco Quishpe</td><td>Iteración asignada:3</td></tr>
<tr><td colspan="2">Descripción:
Como tesorero de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza
Debe ingresar al módulo de pagos
Para buscar planillas pendientes y pagarlas.</td></tr>
<tr><td colspan="2">Criterios de aceptación</td></tr>
<tr><td colspan="2">Criterio de aceptación 1:
Dado que el tesorero ingrese al módulo de pagos.
Debe seleccionar en el menú la opción de pagos.
Entonces ahí se mostrará una tabla con los datos de las planillas faltantes a pagar
Criterio de aceptación 2:
Dado que el tesorero requiera marcar una planilla como pagada
Debe buscar por nombre o cédula del socio
Entonces le mostrará los datos del medidor a pagar y ahí poder marcarlo como pagado.</td></tr>
</table>

Tabla 13:Historia de Usuario: Módulo de reportes.

<table>
<tr><td>Historia de usuario</td><td>Número: 8</td></tr>
<tr><td colspan="2">Nombre de Historia: Módulo de reportes</td></tr>
<tr><td colspan="2">Usuario: Administrador de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza.</td></tr>
<tr><td rowspan="2">Prioridad de negocio: Alta</td><td>Puntos estimados: 10</td></tr>
<tr><td>Puntos Reales: 12</td></tr>
<tr><td>Programador Responsable: Marco Quishpe</td><td>Iteración asignada:4</td></tr>
<tr><td colspan="2">Descripción:
Como administrador de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza
Debe ingresar al módulo de reportes
Para generar todo tipo de reportes como: cantidad recaudada por mes, o cuentas pendientes de cada mes, socios activos e inactivos, medidores activos e inactivos, historial de traspasos.</td></tr>
<tr><td colspan="2">Criterios de aceptación</td></tr>
<tr><td colspan="2">Criterio de aceptación 1:
Dado que el administrador ingrese en el módulo de reportes
Debe seleccionar el tipo de reportes que requiere y de click en generar
Entonces la aplicación mostrará un preview del reporte solicitado.
Criterio de aceptación 2:
Dado que administrador haya generado un reporte
Debe seleccionar la opción de descargar PDF
Entonces la aplicación iniciará la descarga del reporte en formato PDF.</td></tr>
</table>

Tabla 14:Historia de Usuario: Módulo de notificaciones.

Historia de usuario	Número: 9
Nombre de Historia: Módulo de notificaciones	
Usuario: Administrador de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza.	
Prioridad de negocio: Alta	Puntos estimados: 13
	Puntos Reales: 12
Programador Responsable: Marco Quishpe	Iteración asignada:3
Descripción: Como administrador de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza Debe ingresar al módulo de notificaciones Para enviar mensajes a los socios que requiera.	
Criterios de aceptación	
Criterio de aceptación 1: Dado que el administrador requiera ingresar en el módulo de notificaciones Debe seleccionar la opción notificaciones en el menú. Entonces se mostrará la interfaz para notificaciones. Criterio de aceptación 2: Dado que administrador requiera enviar una notificación a los socios Debe seleccionar el destinatario siempre y cuando esté activo e ingresar el mensaje. Entonces la aplicación enviará el mensaje a los socios seleccionados.	

Por el lado del socio:

Tabla 15:Historia de Usuario: Módulo de Historial de Pagos

Historia de usuario	Número: 10
Nombre de Historia: Módulo de historial de pagos	
Usuario: Socio de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza.	
Prioridad de negocio: Alta	Puntos estimados: 14
	Puntos Reales: 12
Programador Responsable: Marco Quishpe	Iteración asignada:5
Descripción: Como socio de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza Debe ingresar al módulo de historial de pagos Para ver el historial de pagos y ver las comprobantes correspondientes.	
Criterios de aceptación	
Criterio de aceptación 1: Dado que el socio ingrese el módulo de historial de pagos Debe seleccionar en el menú la opción de historial de pagos	

Entonces se le desplegará una tabla con su historial. Criterio de aceptación 2: Dado que el socio quiera buscar su pago o comprobante por fecha Debe ingresar la fecha en el buscador Entonces la aplicación filtrará esa fecha para mostrar el comprobante correspondiente Criterio de aceptación 3: Dado que el socio desee descargar algún comprobante Debe dar click en el botón Ver Comprobante, y se le descargará su comprobante en formato PDF.

Tabla 16:Historia de Usuario: Módulo de multas.

Historia de usuario	Número: 11
Nombre de Historia: Módulo de multas	
Usuario: Socio de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza.	
Prioridad de negocio: Alta	Puntos estimados: 18
	Puntos Reales: 20
Programador Responsable: Marco Quishpe	Iteración asignada:6
Descripción: Como socio de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza	

Podrá realizar consultas al módulo de multas, para revisar valores pendientes y poder pagarlas mediante tarjeta.
Criterios de aceptación
Criterio de aceptación 1: Dado que el socio quiera ingresar al módulo de multas Debe seleccionar en el menú la opción de multas Entonces se mostrará una tabla con las multas que tiene generadas indicando la fecha, el valor y la razón Criterio de aceptación 2: Dado que el socio quiera pagar una multa Debe dar click en el botón pagar en su respectiva multa. Entonces se le desplegará una interfaz para proceder a hacer el pago.

Tabla 17:Historia de Usuario: Módulo de pagar servicio.

Historia de usuario	Número: 12
Nombre de Historia: Módulo de pagar servicio	
Usuario: Socio de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza.	
Prioridad de negocio: Alta	Puntos estimados: 22
	Puntos Reales: 24
Programador Responsable: Marco Quishpe	Iteración asignada:6
Descripción:	

Como socio de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza Necesito ingresar al módulo de pagar servicio Para cancelar el valor a pagar por el consumo de agua
Criterios de aceptación
Criterio de aceptación 1: Dado que el socio requiera ingresar al módulo de pagar servicio Debe seleccionar en el menú la opción de pagar servicio Entonces la aplicación le mostrará el valor a pagar por el servicio de dicho mes o periodo. Criterio de aceptación 2: Dado que el socio quiera pagar el servicio Debe dar clic en el botón de pagar Entonces se le desplegará una interfaz para proceder al pago.

Tabla 18:Historia de Usuario: Módulo de actualizar perfil.

Historia de usuario	Número: 13
Nombre de Historia: Módulo de actualizar perfil	
Usuario: Socio de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza.	
Prioridad de negocio: Alta	Puntos estimados: 13
	Puntos Reales: 11
Programador Responsable: Marco Quishpe	Iteración asignada:5

Descripción: Como socio de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza Debe ingresar al módulo de actualizar perfil Para modificar datos personales.
Criterios de aceptación
Criterio de aceptación 1: Dado que el socio ingresa al módulo de actualizar perfil Debe seleccionar la opción de editar Entonces la aplicación le permitirá modificar datos como: número de teléfono, correo electrónico, contraseña.

e. Estimación de historias de usuario.

Tabla 19. Estimación de Historias de Usuario.

N°	Historia de Usuario	Tareas	Puntos Reales	Días	Horas
1	Autenticación del sistema	Diseño de la base de datos	3	0.38	3
		Diseño de la interfaz para autenticación	5	0.62	5

		Interfaz de recuperar contraseña.	2	0.25	2
2	Módulo de usuarios	Diseño de la base de datos	3	0.38	6
		Desarrollo del backend de usuarios	6	0.75	6
		Desarrollo del frontend de usuarios.	6	0.75	6
3	Módulo de socios	Diseño de la base de datos	3	0.38	3
		Desarrollo del backend de socios	6	0.75	6
		Desarrollo del frontend de socios	6	0.75	6
4	Módulo de medidores	Diseño de la base de datos	3	0.38	3
		Diseño del backend de medidores	6	0.75	6

		Diseño del frontend de medidores	6	0.75	6
5	Módulo pagos	Diseño de la base de datos	2	0.25	2
		Desarrollo del backend para pagos	4	0.50	4
		Desarrollo del frontend para pagos.	4	0.50	4
6	Módulo de multas	Diseño de la base de datos	3	0.38	3
		Desarrollo del backend para multas	5	0.62	5
		Desarrollo del frontend para multas.	4	0.50	4
7	Módulo de lecturas	Diseño de la base de datos	3	0.38	3
		Desarrollo del backend para lecturas	5	0.62	5

		Desarrollo del frontend para lecturas	4	0.50	4
8	Módulo de notificaciones	Diseño de la base de datos	2	0.25	2
		Desarrollo del backend para notificaciones	5	0.62	5
		Desarrollo del frontend para notifiaciones	5	0.62	5
9	Módulo de reportes	Creación de reporte de cantidad recaudada	6	0.75	6
		Creación de reporte de pagos pendientes	6	0.75	6
10	Módulo de Historial de Pagos	Diseño de la base de datos	3	0.38	3
		Desarrollo del backend	5	0.62	5

		para Historial de Pagos			
		Desarrollo del frontend para Historial de Pagos	4	0.50	4
11	Módulo de multas	Diseño de la base de datos	5	0.62	5
		Desarrollo del backend para pagar multas	8	1	8
		Desarrollo del frontend para pagar multas	7	0.87	7
12	Módulo de pagar servicio	Diseño de la base de datos	5	0.62	5
		Desarrollo del backend para pagar servicio	12	1.5	12
		Desarrollo del frontend	7	0.87	7

		para pagar servicio			
13	Módulo de actualizar perfil	Desarrollo del backend para actualizar perfil	6	0.75	6
		Desarrollo del frontend para actualizar perfil.	5	0.62	5
TOTAL:			180	22.48	180

f. Plan de entregas

Para realizar las tareas de cada historia de usuario, se realizó un cronograma especificando las iteraciones con su tiempo estimado.

Tabla 20. Plan Tentativo de Entregas

N°	Historia de usuario	Tiempo estimado		Iteración asignada					
		Días	Horas	1	2	3	4	5	6

1	Autenticación del sistema	1.25	10	X					
2	Módulo usuarios	1.88	15		X				
3	Módulo de socios	1.88	15		X				
4	Módulo de medidores	1.88	15			X			
5	Módulo de pagos	1.25	10				X		
6	Módulo de multas	1.5	12			X			
7	Módulo de lecturas	1.5	12			X			
8	Módulo de notificaciones	1.49	12				X		
9	Módulo de reportes	1.5	12				X		
10	Módulo de Historial de Pagos	1.5	12					X	
11	Módulo de pagar multas	2.49	20						X
12	Módulo de pagar servicio	2.99	24						X

13	Módulo de actualizar perfil.	1.37	11					X	

4.2.Fase II: Diseño

En esta fase se crearon los diagramas de casos de uso, y el modelo entidad relación para comprender mejor como será el flujo de la aplicación:

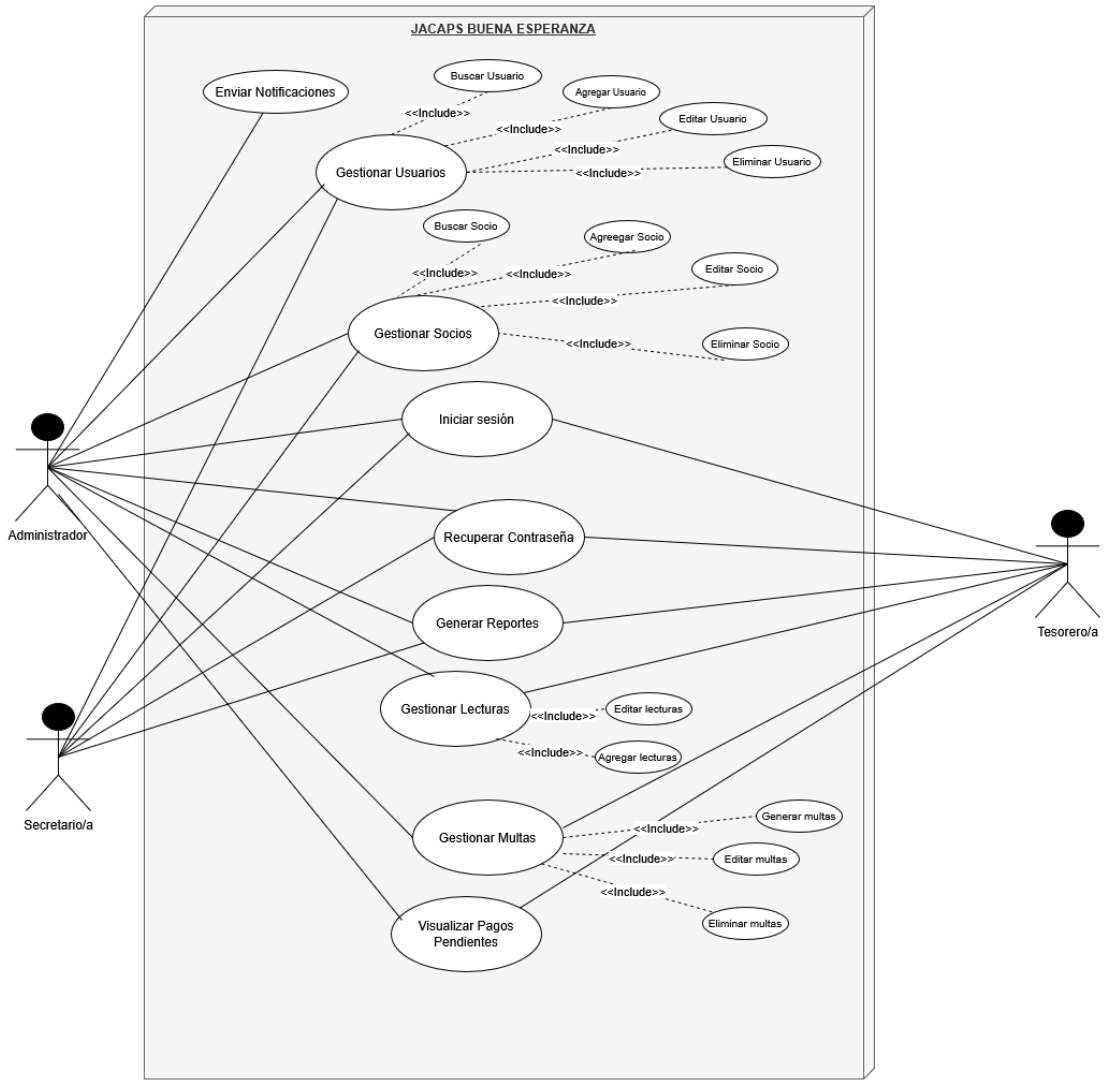


Figura 4 Diagrama de casos de uso: Administrador, Secretario/a, Tesorero/a

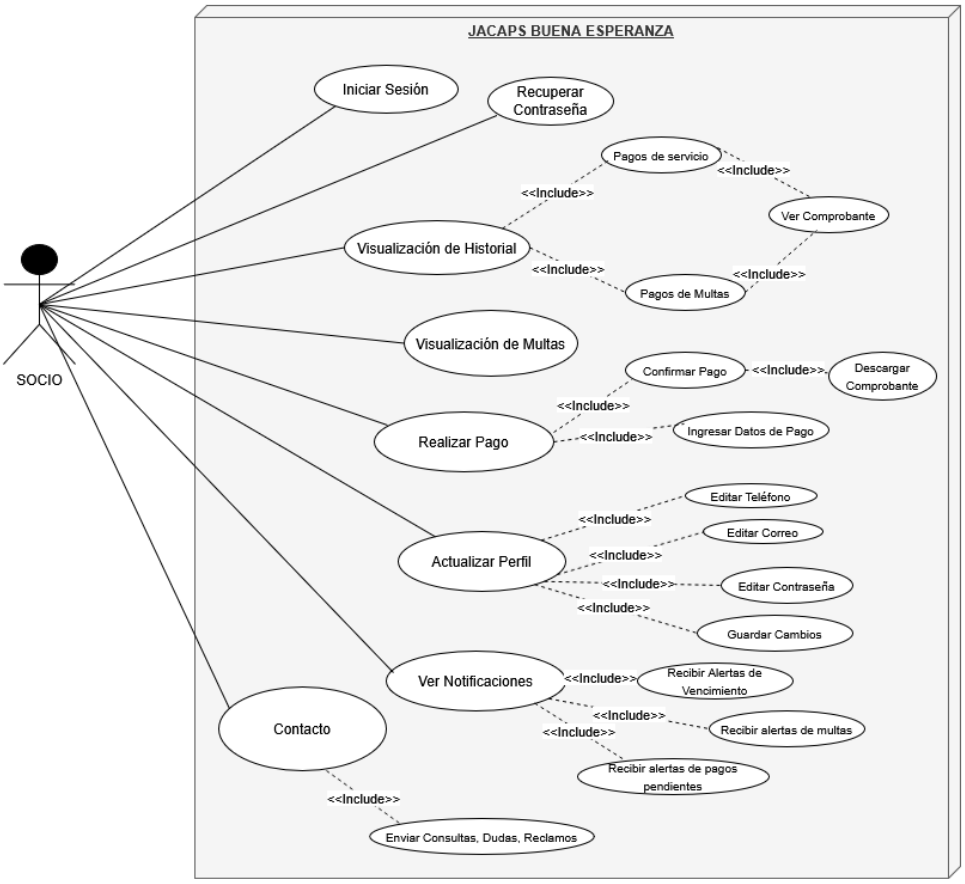


Figura 5 Diagrama de casos de uso: socio

a. Iteraciones

- **Iteración 1**

De acuerdo a la Tabla 19, el plan de entrega el primer período consta de 1 iteración:

Tabla 21. Historias de usuario de la Iteración 1.

N°	Historia de Usuario	Prioridad	Riesgo
1	Autenticación del sistema	Alta	Alto

1. Autenticación del sistema

Tabla 22 Tarea 1: Diseño de la base de datos

Tarea	
Número: 1	Número de historia: 1
Nombre de historia: autenticación al sistema	
Puntos estimados: 3	
Fecha de inicio: 24/05/2025	Fecha de finalización:24/05/2025
Descripción: Modelar el esquema de la base de datos que intervienes en la autenticación del sistema.	

Tabla 23 Tarea 2: Diseño de la interfaz para autenticación

Tarea	
Número: 1	Número de historia: 1
Nombre de historia: autenticación al sistema	
Puntos estimados: 5	
Fecha de inicio:24/05/2025	Fecha de finalización:24/05/2025
Descripción: Desarrollo de la interfaz en la que los usuarios de la junta según sus roles puedan ingresar al aplicativo.	

Tabla 24 Tarea 3: Diseño de la interfaz para recuperar contraseña

Tarea	
Número: 1	Número de historia: 1
Nombre de historia: autenticación al sistema	
Puntos estimados: 2	
Fecha de inicio:25/05/2025	Fecha de finalización:25/05/2025
Descripción: Desarrollo de la interfaz en la que los usuarios de la junta puedan recuperar la contraseña indicando su cédula y correo.	

- **Iteración 2**

  De acuerdo a la Tabla 19 el plan de entrega para el segundo período consta de 2 iteraciones:

Tabla 25 Historias de usuario de la iteración 2

N°	Historia de Usuario	Prioridad	Riesgo
1	Módulo de usuarios	Alta	Alto
2	Módulo de socios	Alta	Alto

1. Módulo de usuarios.

Tabla 26 Tarea 1: Diseño de la base de datos de los usuarios

Tarea	
Número: 1	Número de historia: 2
Nombre de historia: Módulo de usuarios	
Puntos estimados: 3	
Fecha de inicio:25/05/2025	Fecha de finalización:25/05/2025
Descripción: Diseñar el esquema de la base de datos con las entidades que intervienen en este módulo	

Tabla 27 Tarea 2: Desarrollo del backend para usuarios

Tarea	
Número: 2	Número de historia: 2
Nombre de historia: Módulo de usuarios	
Puntos estimados: 6	
Fecha de inicio:27/05/2025	Fecha de finalización:27/05/2025

Descripción: Desarrollo de las consultas relacionadas con gestionar la información que va a manejar el frontend.

Tabla 28 Tarea 3: Desarrollo del frontend para usuarios

Tarea	
Número: 3	Número de historia: 2
Nombre de historia: Módulo de usuarios	
Puntos estimados: 6	
Fecha de inicio:28/05/2025	Fecha de finalización:28/05/2025
Descripción: Desarrollo de la interfaz gráfica del módulo de usuarios.	

2. Módulo de socios

Tabla 29 Tarea 1: Diseño de la base de datos para socios

Tarea	
Número: 1	Número de historia: 3
Nombre de historia: Modulo de socios	
Puntos estimados: 3	
Fecha de inicio:29/05/2025	Fecha de finalización:29/05/2025
Descripción: Diseñar el esquema con las entidades que intervienen en el módulo de socios.	

Tabla 30 Tarea 2: Desarrollo del backend para socios

Tarea	
Número: 2	Número de historia: 3

Nombre de historia: Modulo de socios	
Puntos estimados: 6	
Fecha de inicio:29/05/2025	Fecha de finalización:30/05/2025
Descripción: Desarrollo de las consultas que van a gestionar la información de los socios a través del frontend.	

Tabla 31 Tarea 3: Desarrollo del frontend para socios

Tarea	
Número: 3	Número de historia: 3
Nombre de historia: Modulo de socios	
Puntos estimados: 6	
Fecha de inicio:30/05/2025	Fecha de finalización:30/05/2025
Descripción: Desarrollo de la interfaz del módulo de socios.	

- **Iteración 3**

  De acuerdo a la Tabla 19, el plan de entrega en el tercer periodo consta de 3 iteraciones:

Tabla 32 Historias de usuario de la iteración 3

N°	Historia de Usuario	Prioridad	Riesgo
1	Módulo de medidores	Alta	Alto
2	Módulo de lecturas	Alta	Alto
3	Módulo de multas	Alta	Alto

1. Módulo de medidores

Tabla 33 Tarea 1: Diseño de la base de datos para medidores

Tarea	
Número: 1	Número de historia: 4
Nombre de historia: Modulo de medidores	
Puntos estimados: 3	
Fecha de inicio:31/05/2025	Fecha de finalización:1/06/2025
Descripción: Diseño del esquema de la base de datos con las entidades que intervienes en este módulo.	

Tabla 34 Tarea 2: Desarrollo del backend para el módulo medidores

Tarea	
Número: 2	Número de historia: 4
Nombre de historia: Modulo de medidores	
Puntos estimados: 6	
Fecha de inicio:1/06/2025	Fecha de finalización:1/06/2025
Descripción: Codificación de las consultas necesarias para gestión de los medidores en el frontend.	

Tabla 35 Tarea 3: Desarrollo del frontend para el módulo medidores

Tarea	
Número: 3	Número de historia: 4
Nombre de historia: Modulo de medidores	

Puntos estimados: 6	
Fecha de inicio:02/06/2025	Fecha de finalización:02/06/2025
Descripción: Desarrollo de la interfaz gráfica para gestionar la información de los medidores.	

2. Módulo de lecturas

Tabla 36 Tarea 1: Diseño de la base de datos para el módulo de lecturas

Tarea	
Número: 1	Número de historia: 6
Nombre de historia: Modulo de lecturas	
Puntos estimados: 3	
Fecha de inicio:	Fecha de finalización:
Descripción: Diseño de la base de datos para la gestión de lecturas	

Tabla 37 Tarea 2: Desarrollo del backend para el módulo de lecturas

Tarea	
Número: 2	Número de historia: 6
Nombre de historia: Modulo de lecturas	
Puntos estimados: 5	
Fecha de inicio:03/06/2025	Fecha de finalización:03/06/2025
Descripción: Codificación con las consultas necesarias para gestionar la información desde el backend.	

Tabla 38 Tarea 3: Desarrollo del frontend para el módulo de lecturas

Tarea	
Número: 3	Número de historia: 6
Nombre de historia: Modulo de lecturas	
Puntos estimados: 4	
Fecha de inicio:03/06/2025	Fecha de finalización:04/06/2025
Descripción: Desarrollo de la interfaz gráfica del módulo de lecturas.	

3. Módulo de multas

Tabla 39 Tarea 1: Diseño de la base de datos para el módulo de multas

Tarea	
Número: 1	Número de historia: 5
Nombre de historia: Modulo de multas	
Puntos estimados: 3	
Fecha de inicio:05/06/2025	Fecha de finalización:05/06/2025
Descripción: Diseño de la base de datos con todas las entidades necesarias para la gestión de multas.	

Tabla 40 Tarea 2: Desarrollo del backend para el módulo de multas

Tarea	
Número: 2	Número de historia: 5
Nombre de historia: Modulo de multas	
Puntos estimados: 5	
Fecha de inicio:05/06/2025	Fecha de finalización:06/06/2025

Descripción: Codificación de las consultas desde el frontend que intervienen en la gestión de multas de los socios

Tabla 41 Tarea 3: Desarrollo del frontend para el módulo de multas

Tarea	
Número: 3	Número de historia: 5
Nombre de historia: Modulo de multas	
Puntos estimados: 4	
Fecha de inicio:06/06/2025	Fecha de finalización:06/06/2025
Descripción: Desarrollo de la interfaz gráfica para el módulo de multas.	

- **Iteración 4**

De acuerdo a la Tabla 19, el plan de entregas para el cuarto período consta de 3 iteraciones:

Tabla 42 Historias de usuario de la iteración 4

N°	Historia de Usuario	Prioridad	Riesgo
1	Módulo de Historial de Pagos	Alta	Alto
2	Módulo de reportes	Alta	Alto
3	Módulo de notificaciones	Alta	Alto

1. Módulo de Historial de Pagos

Tabla 43 Tarea 1: Diseño de la base de datos para el módulo de Pagos.

Tarea	
Número: 1	Número de historia: 7
Nombre de historia: Modulo de Pagos	
Puntos estimados: 2	
Fecha de inicio:07/06/2025	Fecha de finalización:07/06/2025
Descripción: Diseño de la base de datos con las entidades necesarias para mostrar los pagos pendientes	

Tabla 44 Tarea 2: Desarrollo del backend para el módulo de pagos

Tarea	
Número: 2	Número de historia: 7
Nombre de historia: Modulo de pagos	
Puntos estimados: 4	
Fecha de inicio:07/06/2025	Fecha de finalización:08/06/2025
Descripción: Codificación de las consultas respectivas que se solicitará desde el backend.	

Tabla 45 Tarea 3: Desarrollo del frontend para el módulo de pagos

Tarea	
Número: 3	Número de historia: 7
Nombre de historia: Modulo de pagos	
Puntos estimados: 4	
Fecha de inicio:08/06/2025	Fecha de finalización:08/06/2025

Descripción: Desarrollo de la interfaz gráfica para mostrar los pagos.

2. Módulo de reportes

Tabla 46 Tarea 1: Interfaz para reporte de recaudación

Tarea	
Número: 1	Número de historia: 8
Nombre de historia: Modulo de reportes	
Puntos estimados: 6	
Fecha de inicio:10/06/2025	Fecha de finalización:10/06/2025
Descripción: Diseño de la interfaz que permita generar un reporte de la cantidad recaudada por mes.	

Tabla 47 Tarea 2: Interfaz para reporte de cobros pendientes

Tarea	
Número: 2	Número de historia: 8
Nombre de historia: Modulo de reportes	
Puntos estimados: 6	
Fecha de inicio:11/06/2025	Fecha de finalización:11/06/2025
Descripción: Diseño de la interfaz que permita generar un reporte de la cantidad por cobrar de cada mes.	

3. Módulo de notificaciones

Tabla 48 Tarea 1: Diseño de la base de datos para el módulo de notificaciones

Tarea	
Número: 1	Número de historia: 9
Nombre de historia: Modulo de notificaciones	
Puntos estimados: 2	
Fecha de inicio:12/06/2025	Fecha de finalización:12/06/2025
Descripción: Diseño de la base de datos con entidades necesarias para el envío de notificaciones.	

Tabla 49 Tarea 2: Desarrollo del backend para el módulo de notificaciones.

Tarea	
Número: 2	Número de historia: 9
Nombre de historia: Modulo de notificaciones	
Puntos estimados: 5	
Fecha de inicio:	Fecha de finalización:
Descripción: Codificación de las consultas que van a ejecutarse desde el backend para el envío.	

Tabla 50 Tarea 3: Desarrollo del frontend para el módulo de notificaciones.

Tarea	
Número: 3	Número de historia: 9
Nombre de historia: Modulo de notificaciones	
Puntos estimados: 5	

Fecha de inicio:13/06/2025	Fecha de finalización:13/06/2025
Descripción: Desarrollo de la interfaz para el envío correcto de notificaciones a los socios.	

- **Iteración 5**

De acuerdo a la Tabla 19, en el plan de entrega en el quinto período consta de 2 iteraciones:

Tabla 51 Historias de usuario de la iteración 5

N°	Historia de Usuario	Prioridad	Riesgo
1	Módulo de Historial de Pagos	Alta	Alto
2	Módulo de actualizar Perfil	Alta	Alto

1. Módulo de Historial de Pagos

Tabla 52 Tarea 1: Diseño de la base de datos para el módulo Historial de Pagos.

Tarea	
Número: 1	Número de historia: 10
Nombre de historia: Modulo de Historial de Pagos	
Puntos estimados: 3	
Fecha de inicio:18/06/2025	Fecha de finalización:18/06/2025

Descripción: Diseño del esquema de la base de datos que llevará el control de los pagos ya cancelados.

Tabla 53 Tarea 2: Desarrollo del backend para el módulo de Historial de Pagos.

Tarea	
Número: 2	Número de historia: 10
Nombre de historia: Modulo de Historial de Pagos	
Puntos estimados: 5	
Fecha de inicio:19/06/2025	Fecha de finalización:19/06/2025
Descripción: Codificación con las consultas que serán solicitadas desde el frontend para mostrar la información.	

Tabla 54 Tarea 3: Desarrollo del frontend para el módulo de Historial de Pagos.

Tarea	
Número: 3	Número de historia: 10
Nombre de historia: Modulo de Historial de Pagos	
Puntos estimados: 4	
Fecha de inicio:20/06/2025	Fecha de finalización:20/06/2025
Descripción: Desarrollo de la interfaz para visualizar los comprobantes	

2. Módulo de actualizar perfil

Tabla 55 Tarea 1: Desarrollo del backend para el módulo de actualizar perfil

Tarea

Número: 1	Número de historia: 13
Nombre de historia: Modulo de actualizar perfil	
Puntos estimados: 6	
Fecha de inicio:21/06/2025	Fecha de finalización:21/06/2025
Descripción: Codificación con las consultas necesarias para poder modificar la información desde el frontend.	

Tabla 56 Tarea 2: Desarrollo del frontend para el módulo de actualizar perfil.

Tarea	
Número: 2	Número de historia: 13
Nombre de historia: Modulo de actualizar perfil	
Puntos estimados: 5	
Fecha de inicio:21/06/2025	Fecha de finalización:22/06/2025
Descripción: desarrollo de la interfaz para la gestión de los datos de cada socio.	

- **Iteración 6**

De acuerdo a la Tabla 19, el plan de entrega para el sexto período consta de 2 iteraciones:

Tabla 57 Historias de usuario de la iteración 6

N°	Historia de Usuario	Prioridad	Riesgo
1	Módulo de pagar multas	Alta	Alto

2	Módulo de pagar servicio	Alta	Alto

1. Módulo de pagar multas

Tabla 58 Tarea 1: Diseño de la base de datos para el módulo de multas

Tarea	
Número: 1	Número de historia: 11
Nombre de historia: Modulo de multas	
Puntos estimados: 5	
Fecha de inicio:22/06/2025	Fecha de finalización:23/06/2025
Descripción: Diseño de la base de datos con las entidades necesarias para registrar un pago de multas.	

Tabla 59 Tarea 2: Desarrollo del backend para el módulo de multas

Tarea	
Número: 2	Número de historia: 11
Nombre de historia: Modulo de multas	
Puntos estimados: 8	
Fecha de inicio:23/06/2025	Fecha de finalización:25/06/2025
Descripción: Codificación necesaria para poder consultar las multas y realizar un pago correctamente desde el frontend	

Tabla 60 Tarea 3: Desarrollo del frontend para el módulo de multas

Tarea	
Número: 3	Número de historia: 11

Nombre de historia: Modulo de multas	
Puntos estimados: 7	
Fecha de inicio:25/06/2025	Fecha de finalización:26/06/2025
Descripción: Desarrollo de la interfaz para mostrar las multas generadas y poder pagarlas.	

2. Módulo de pagar servicio

Tabla 61 Tarea 1: Diseño de la base de datos para el módulo de pagar servicio

Tarea	
Número: 1	Número de historia: 12
Nombre de historia: Modulo de pagar servicio	
Puntos estimados: 5	
Fecha de inicio:27/06/2025	Fecha de finalización:27/06/2025
Descripción: Diseño del esquema para generar un valor a pagar a través de las lecturas de cada medidor.	

Tabla 62 Tarea 2: Desarrollo del backend para el módulo de pagar servicio.

Tarea	
Número: 2	Número de historia: 12
Nombre de historia: Modulo de pagar servicio	
Puntos estimados: 12	
Fecha de inicio:28/06/2025	Fecha de finalización:30/06/2025
Descripción: Codificación con las consultas necesarias para obtener el valor a pagar y para registrar el pago.	

Tabla 63 Tarea 3: Desarrollo del frontend para el módulo de pagar servicio.

Tarea	
Número: 3	Número de historia: 12
Nombre de historia: Modulo de pagar servicio	
Puntos estimados: 7	
Fecha de inicio:2/07/2025	Fecha de finalización:2/07/2025
Descripción: Desarrollo de la interfaz para conocer el valor a pagar y poder pagarlo.	

b. Diseño del modelo de datos

El diseño de la base de datos se desarrolló en MySQL.

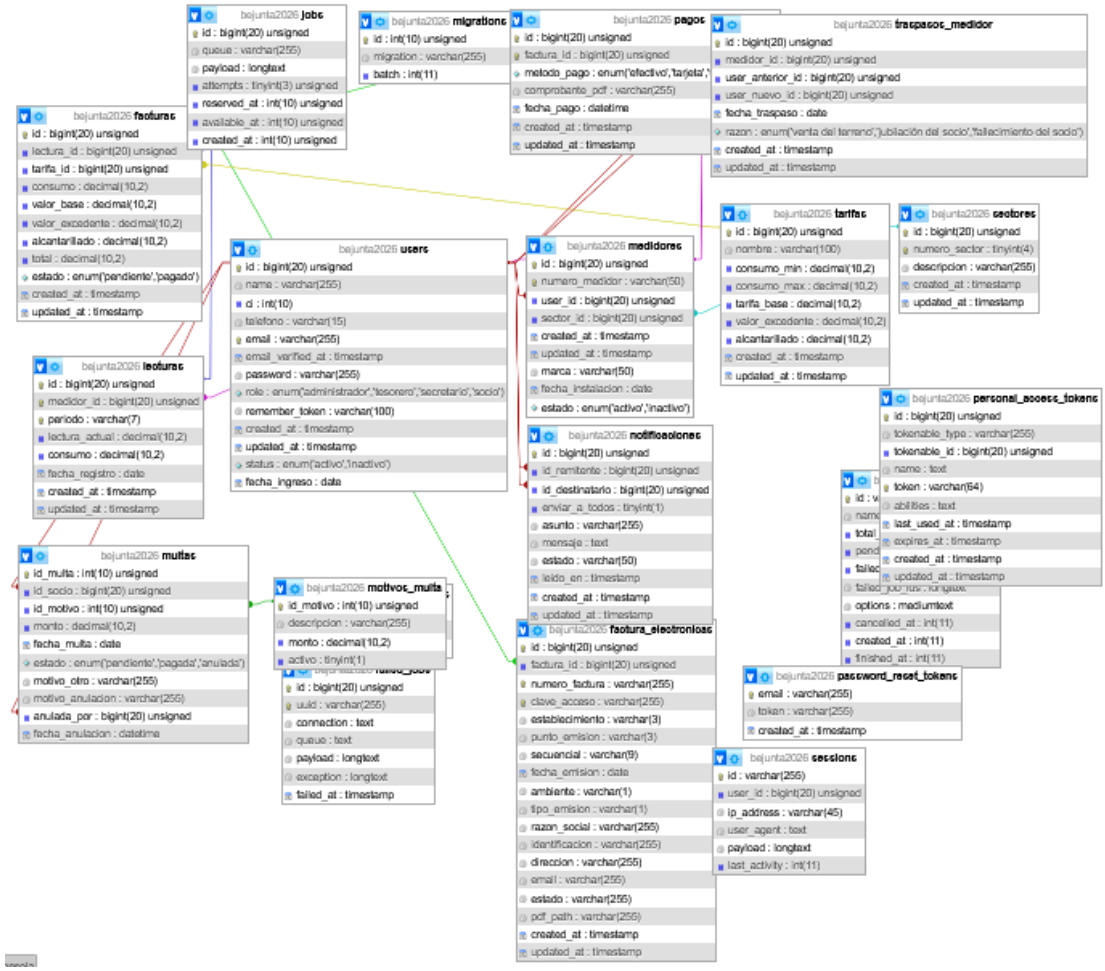


Figura 6 Modelo Relacional de la Base de datos.

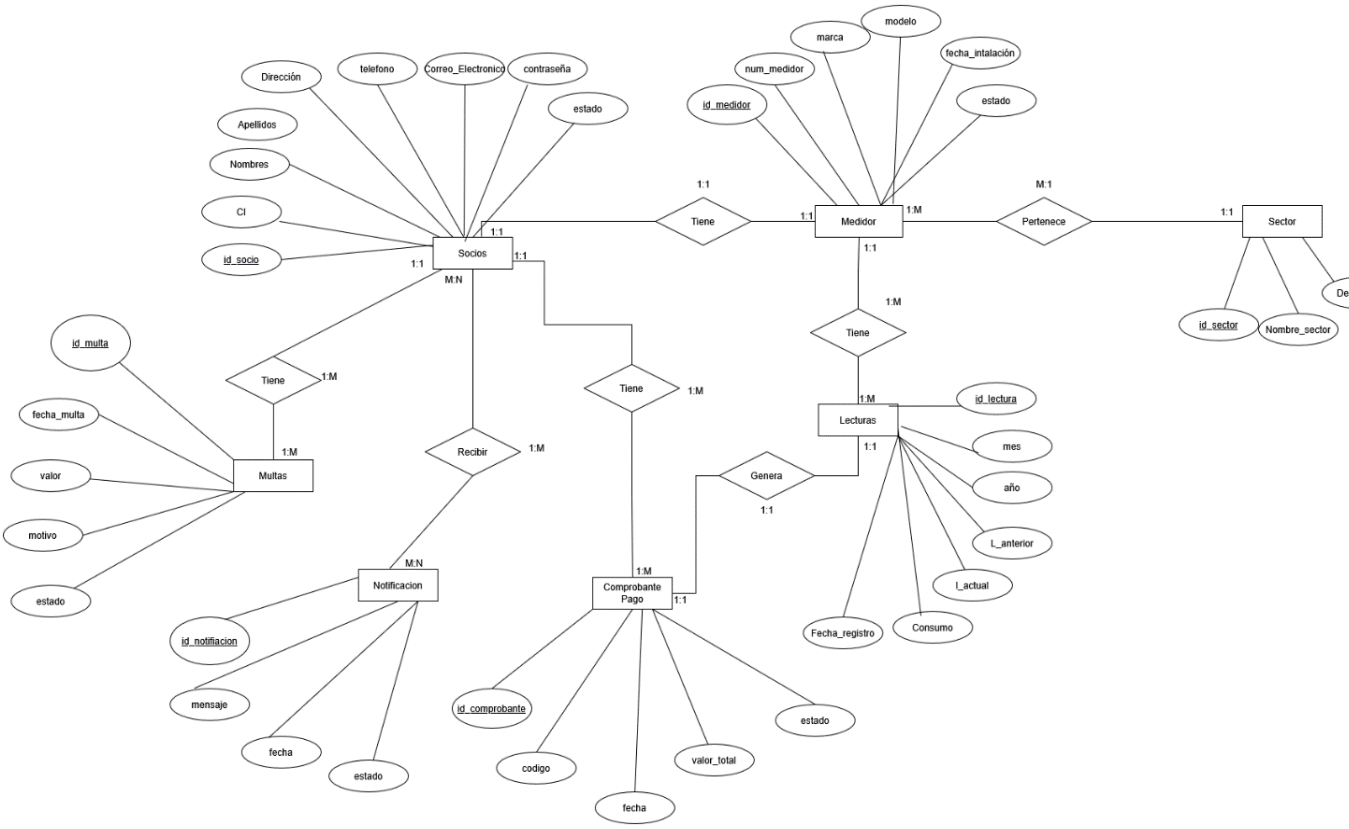


Figura 7:Modelo Entidad-Relación

c. Diseño de iteraciones.

A continuación, se presenta las interfaces de las historias de usuario de la aplicación web, estás fueron diseñadas en la herramienta Balsamiq ya que cuenta con varios elementos que facilitan la creación de estas.

- **Ingreso al sistema.**

  Para acceder al sistema es necesario ingresar las credenciales del usuario y contraseña.

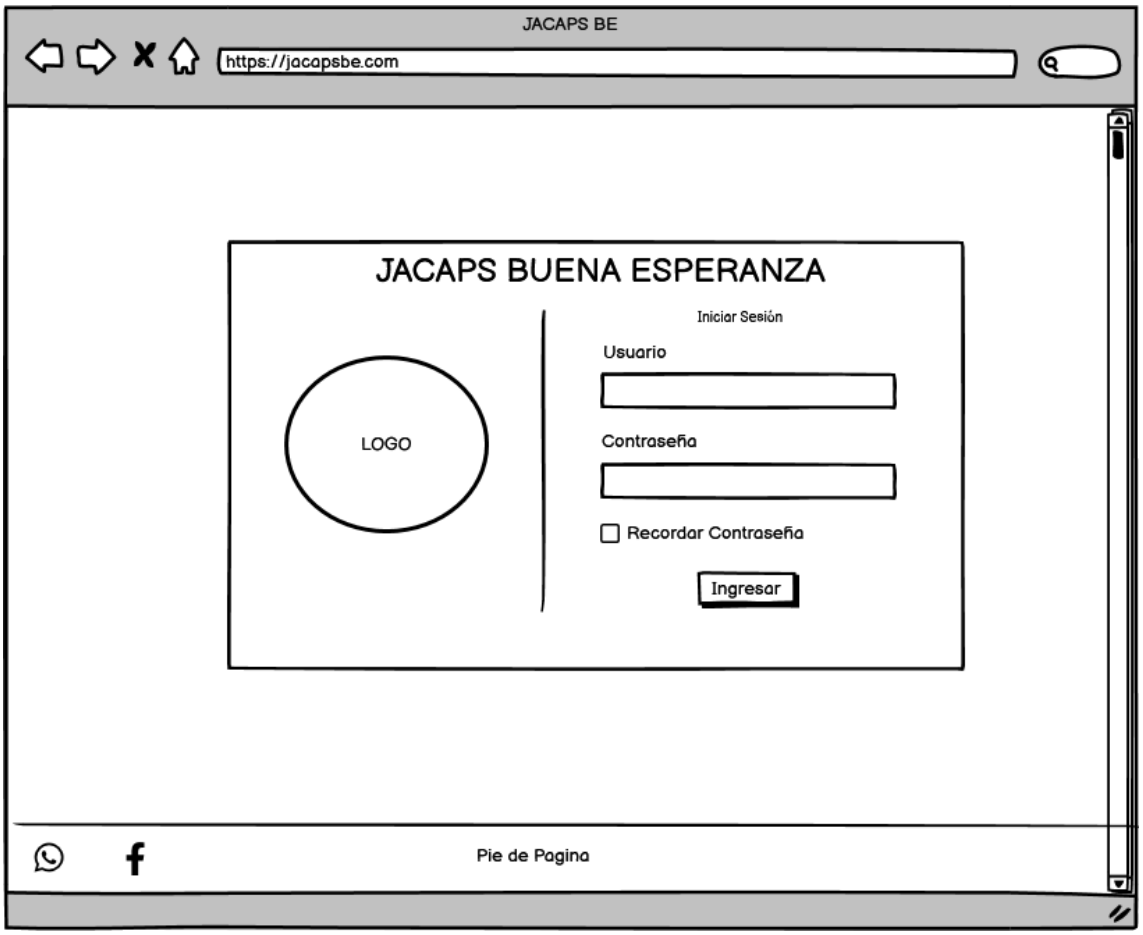


Figura 8. Diseño de la interfaz de login de la aplicación web.

- **Gestión de Usuarios**

El administrador puede visualizar los usuarios del sistema, además puedes realizar la búsqueda por nombre o cédula.

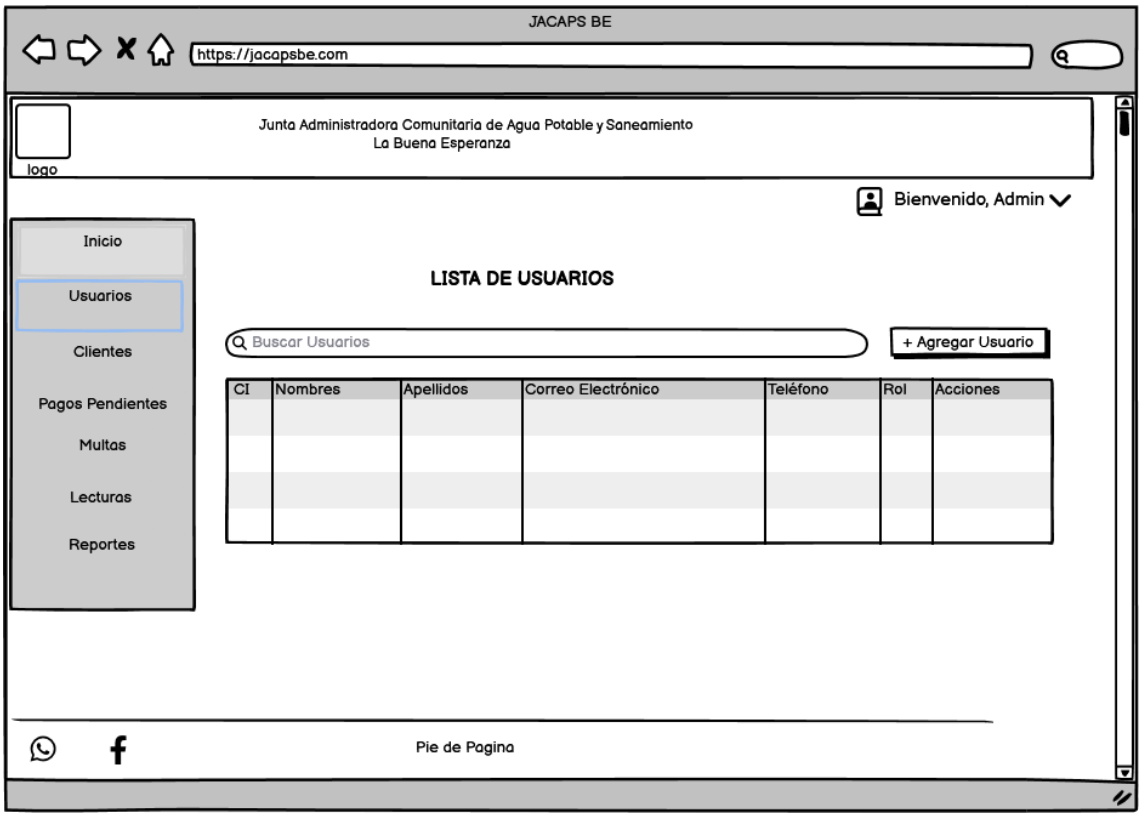


Figura 9. Diseño de la gestión de usuarios.

- **Agregar usuario**

Para agregar un usuario se debe dar click en el botón de agregar usuario y se desplegará un modal para ingresar la información del nuevo usuario del sistema.

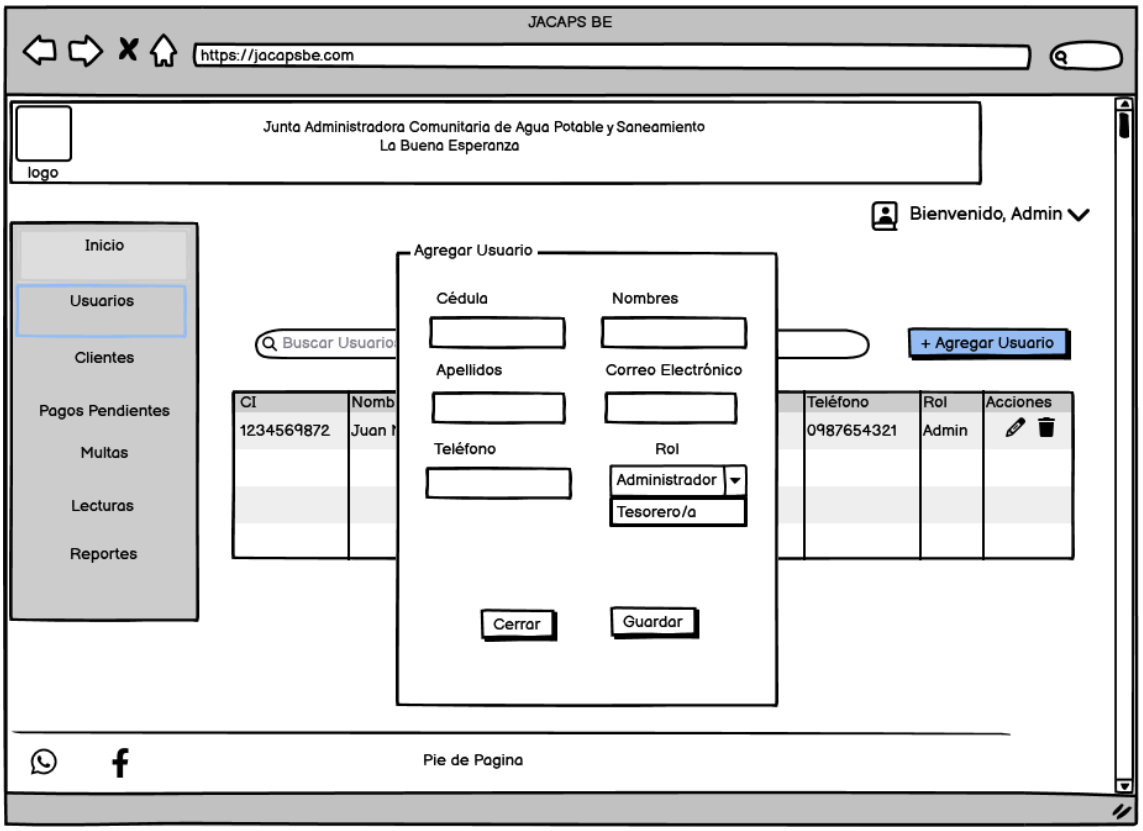


Figura 10. Diseño de la interfaz para agregar un nuevo usuario.

- **Editar información de usuario**

Al dar click en el botón de editar usuario se desplegará un modal de la información del usuario que se seleccionó para modificar lo que se requiera.

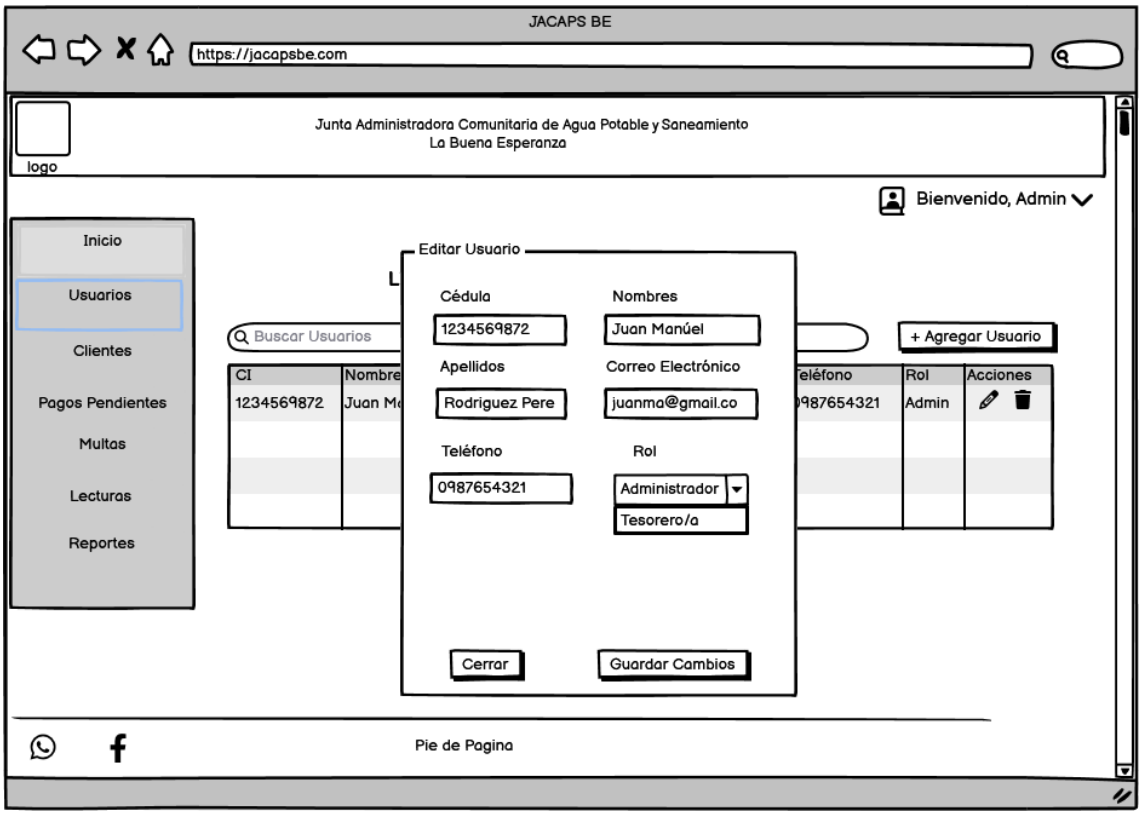


Figura 11. Diseño para editar información del usuario.

- **Gestión de Socios**

Aquí se muestra el listado de los socios de la junta, además puede realizar la búsqueda por su nombre.

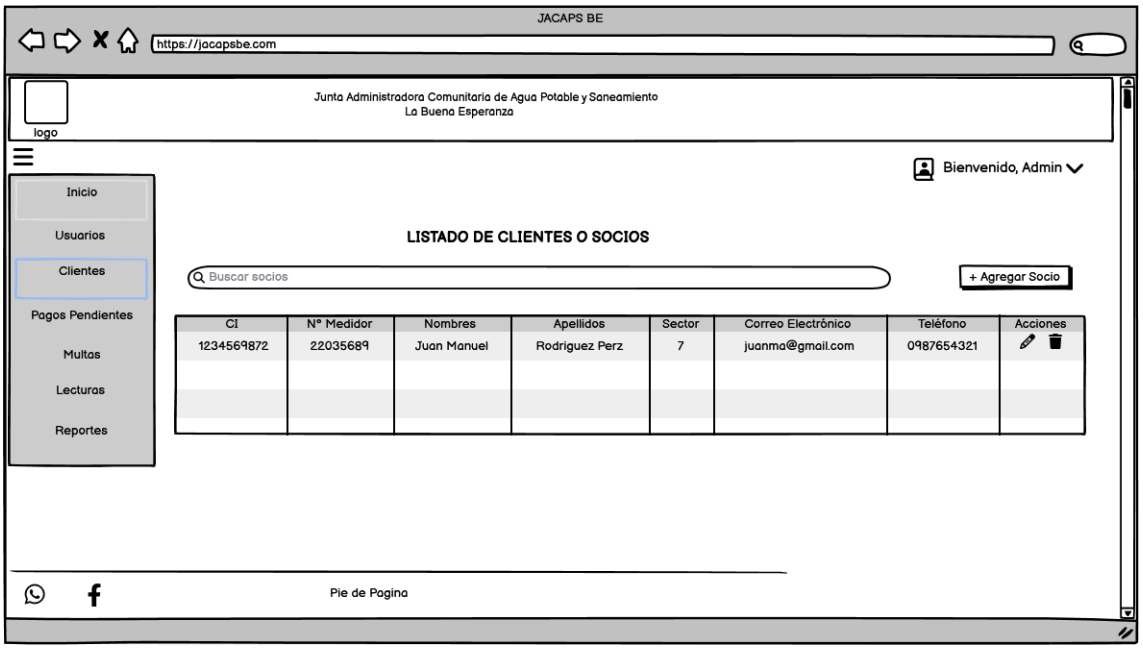


Figura 12. Diseño de la gestión de socios.

- **Agregar Socio**

Al dar click en el botón de agregar socio de mostrará un modal para añadir datos del nuevo socio a registrarse.

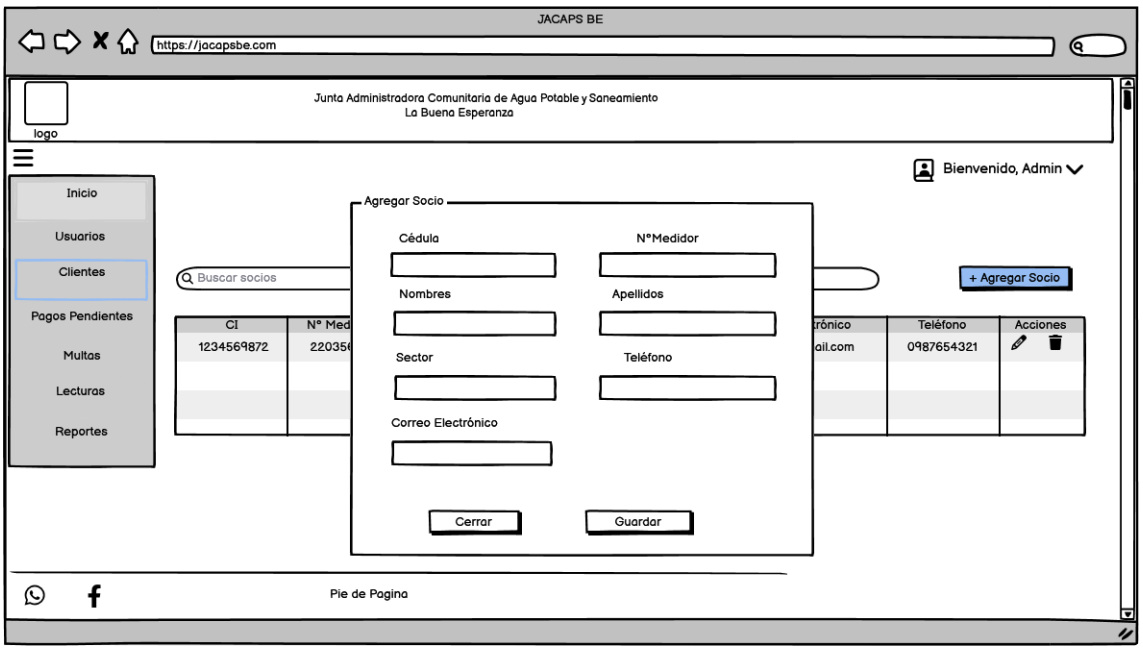


Figura 13. Diseño de agregar un nuevo socio.

- **Editar información de Socio**

Al dar click en el botón de editar, se mostrará un modal con los datos del socio seleccionado para poder editarlo si se requiere.

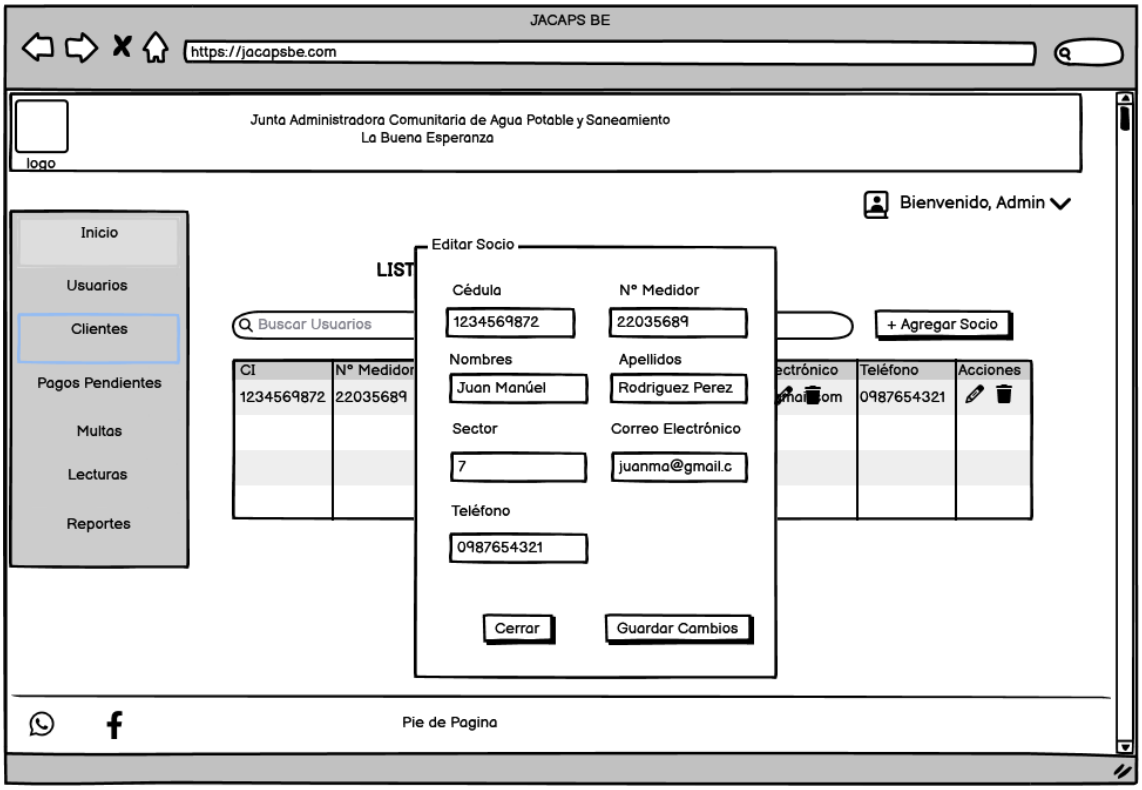


Figura 14. Diseño de editar socio.

- **Pagos Pendientes**

Aquí se puede visualizar un listado de los socios que tengan pagos pendientes.

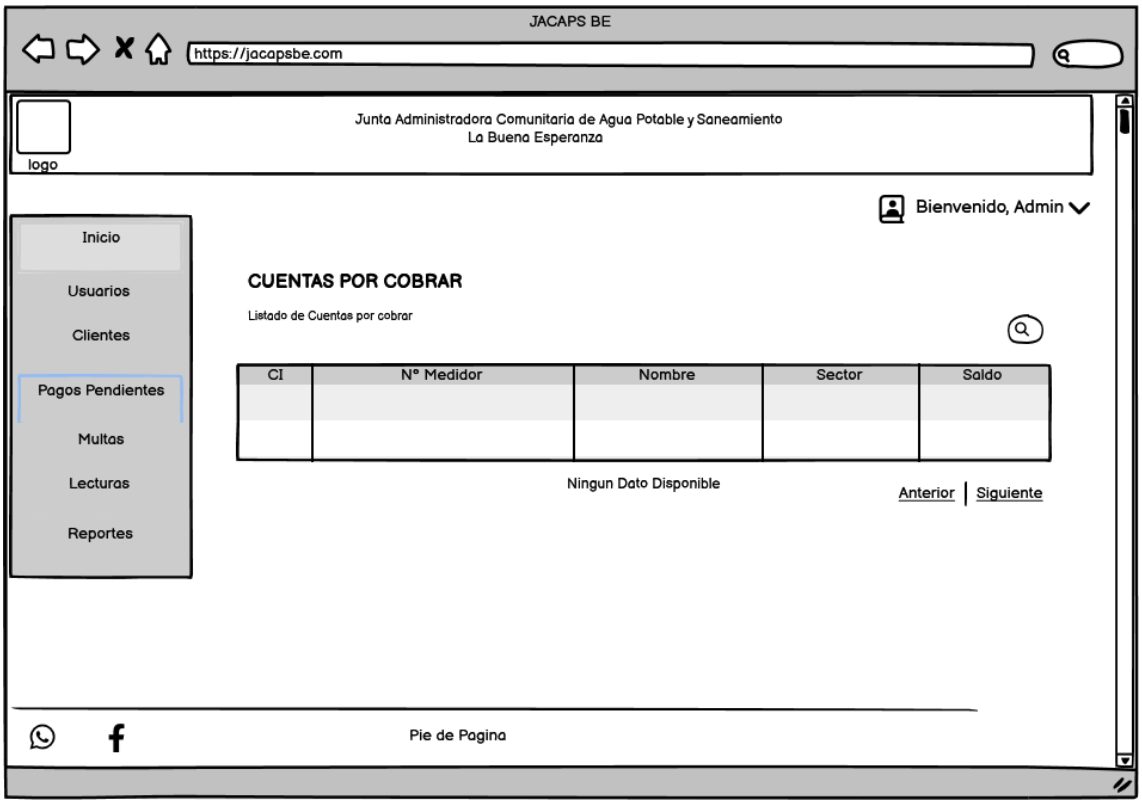


Figura 15. Diseño de los pagos

- **Gestión de Multas**

Aquí se podrá visualizar las multas que han sido generadas a los socios.

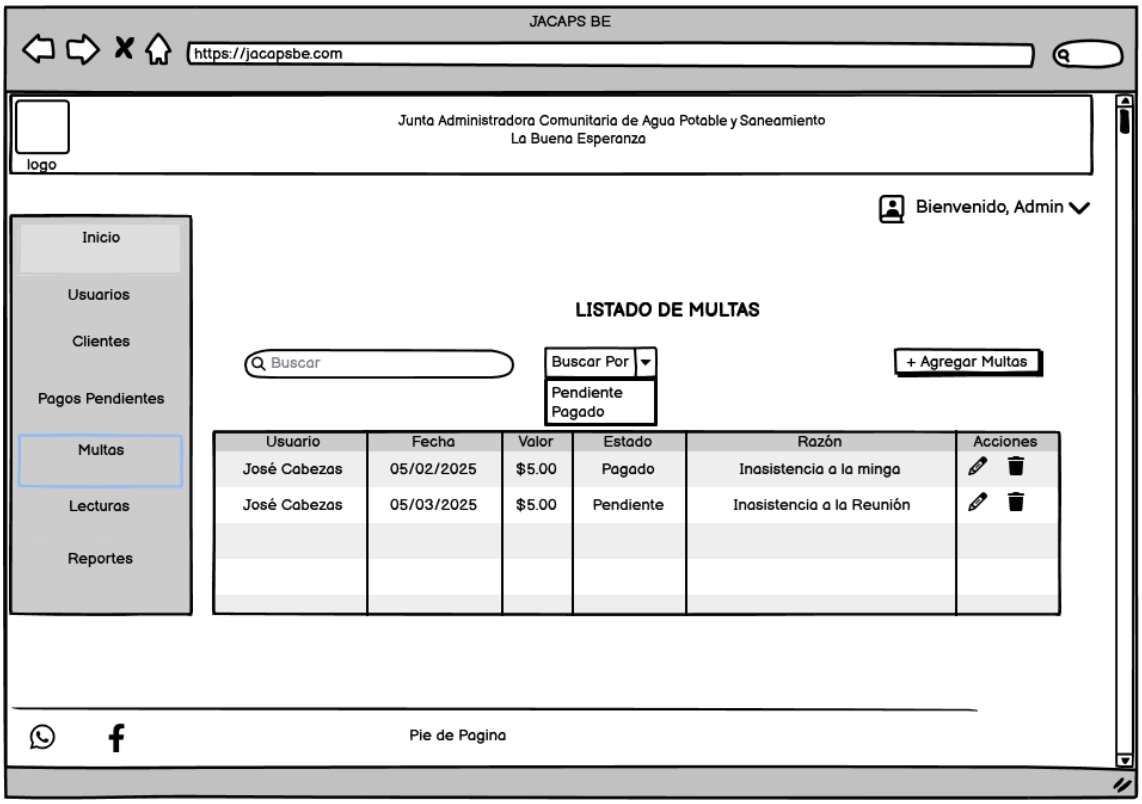


Figura 16. Diseño de Gestión de multas.

- **Agregar multas**

Al dar click en el botón de agregar multa se desplegará un modal para ingresar los datos del socio multado, y su razón.

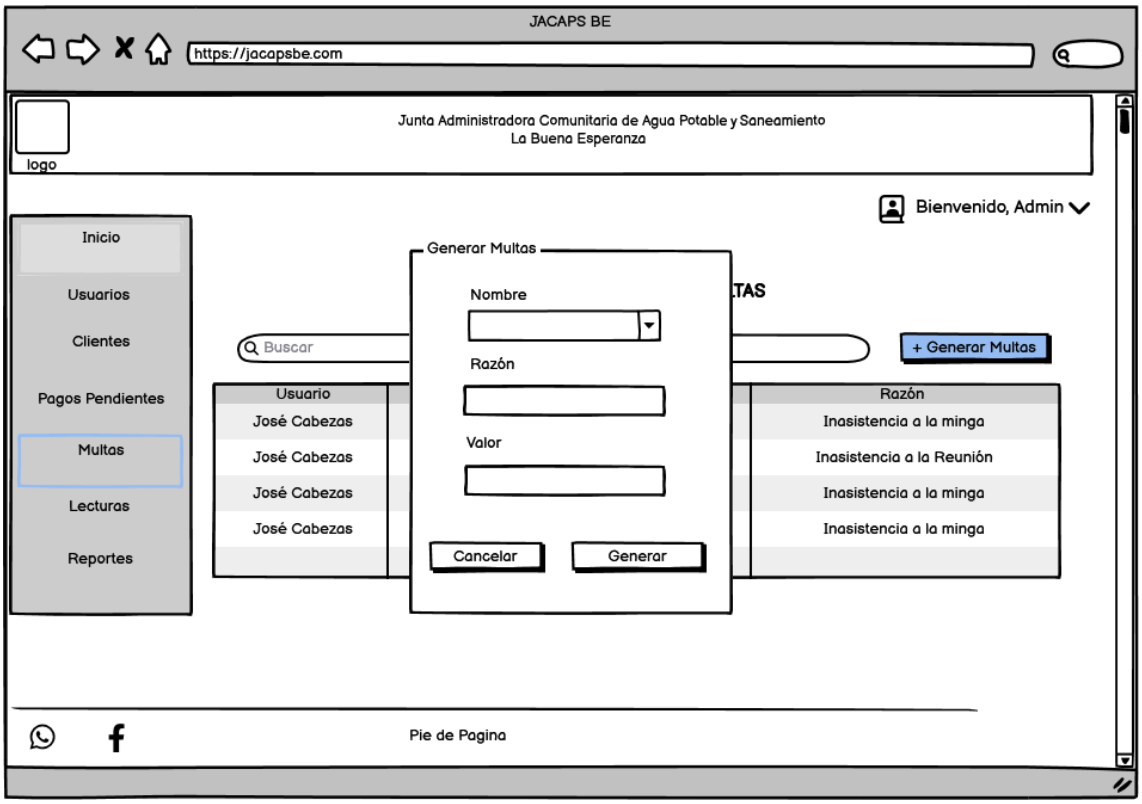


Figura 17. Diseño de agregar multas.

- **Gestión de Lecturas**

Aquí se podrá visualizar as lecturas que han sido generadas.

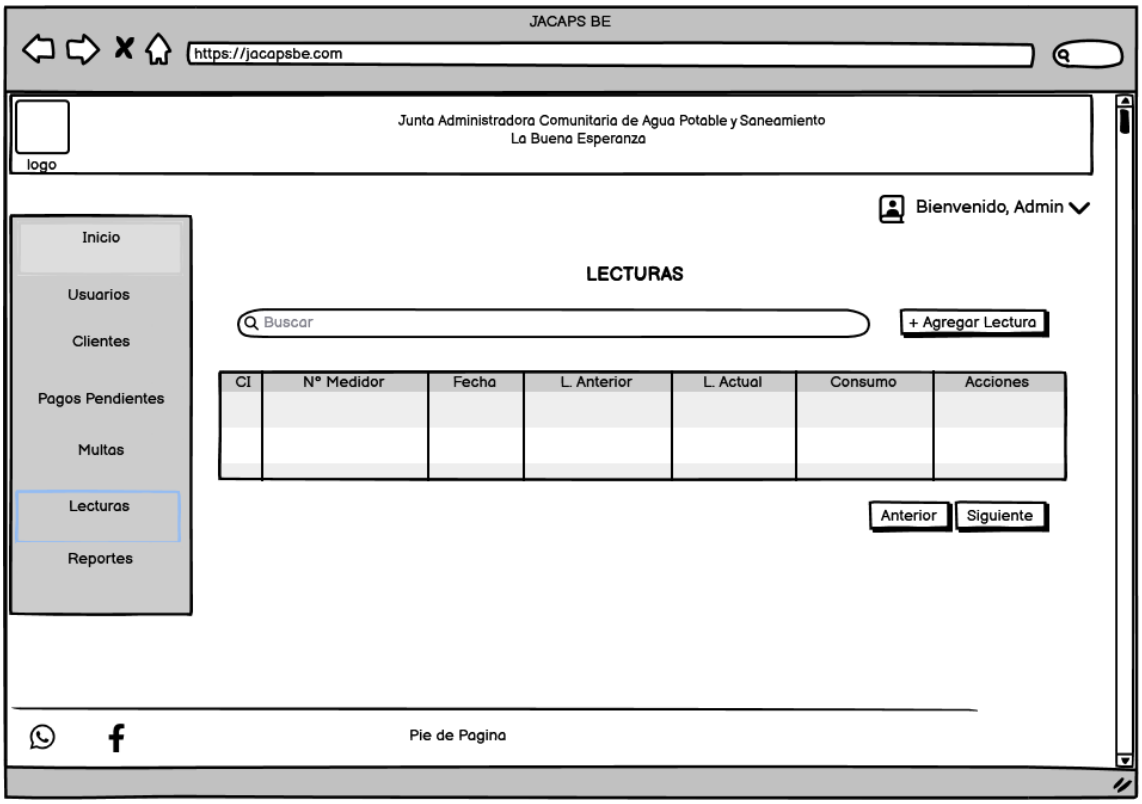


Figura 18. Diseño de la gestión de lecturas.

- **Agregar Lecturas**

Al dar click en el botón de agregar lecturas se mostrará un modal para seleccionar el medidor e ingresar su respectiva lectura del mes.

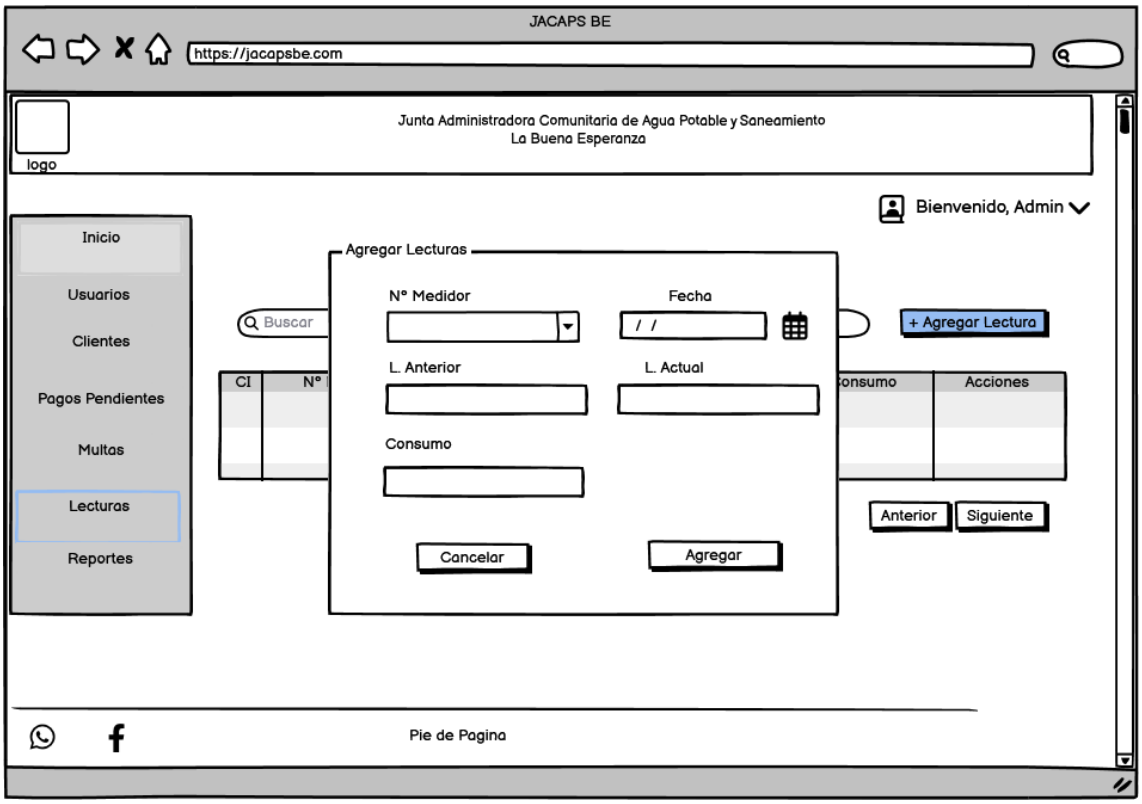


Figura 19. Diseño de agregar lectura.

- **Reportes**

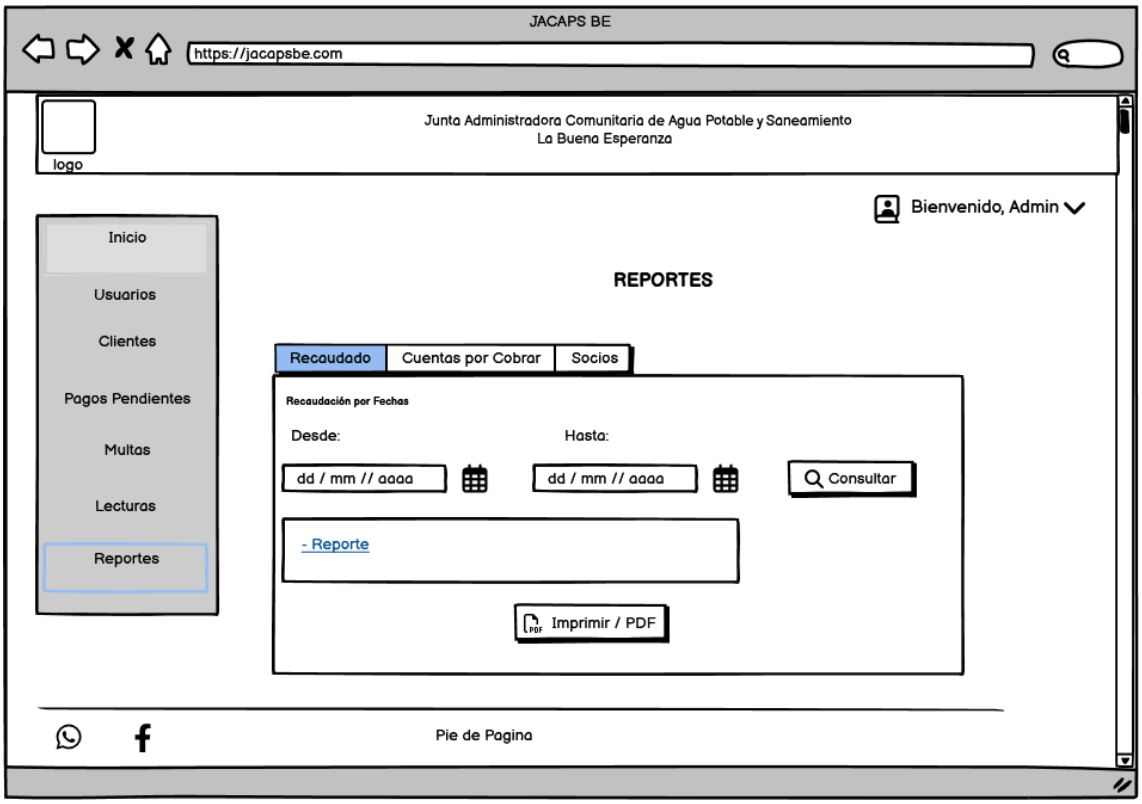


Figura 20. Diseño de interfaz de reportes.

Socio

- **Historial de Pagos**

Aquí el socio podrá visualizar el historial de sus pagos y podrá ver sus comprobantes.



Figura 21. Diseño de historial de pagos.

- **Multas**

Aquí el usuario puede visualizar sus multas que tenga pendientes.

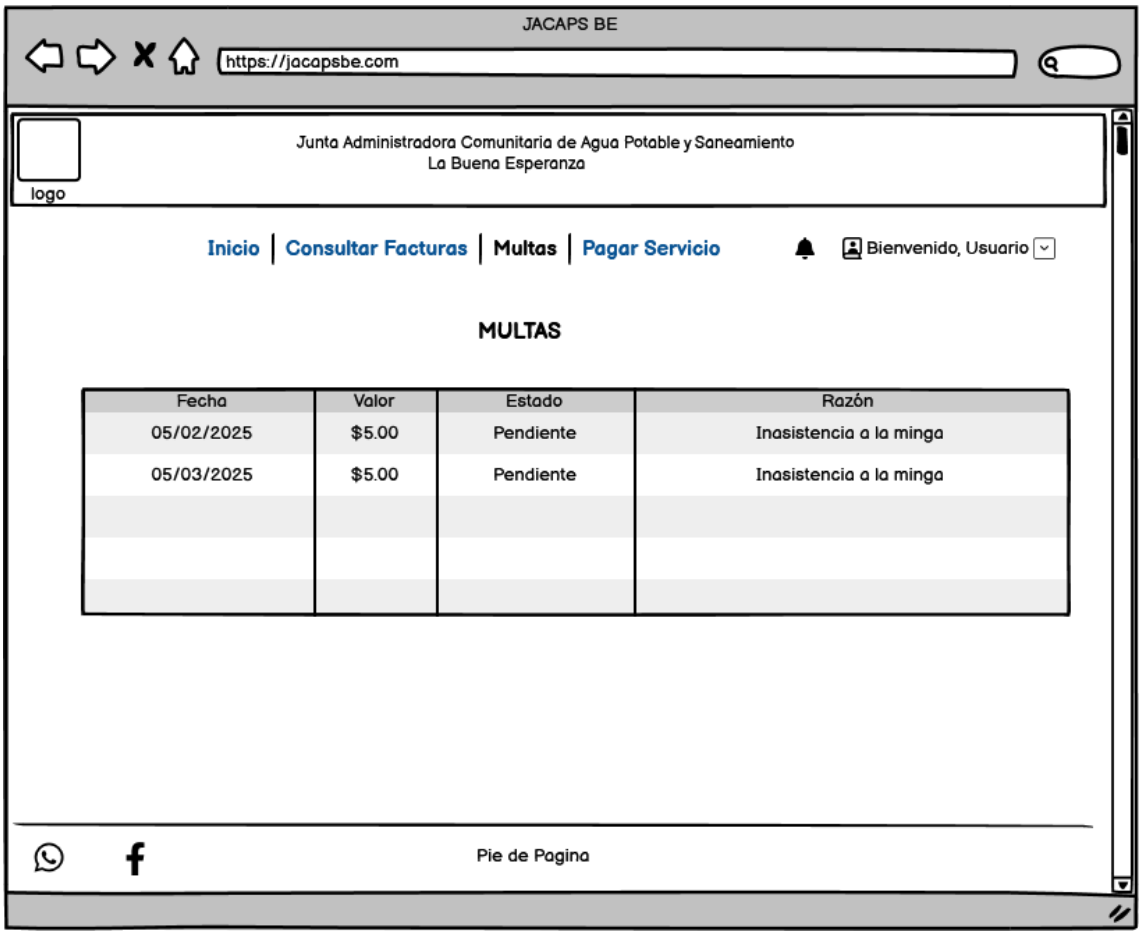


Figura 22. Diseño de interfaz de multas

- **Pagar servicio**

Aquí el socio podrá realizar el pago del servicio de agua potable que se le generé cada mes.

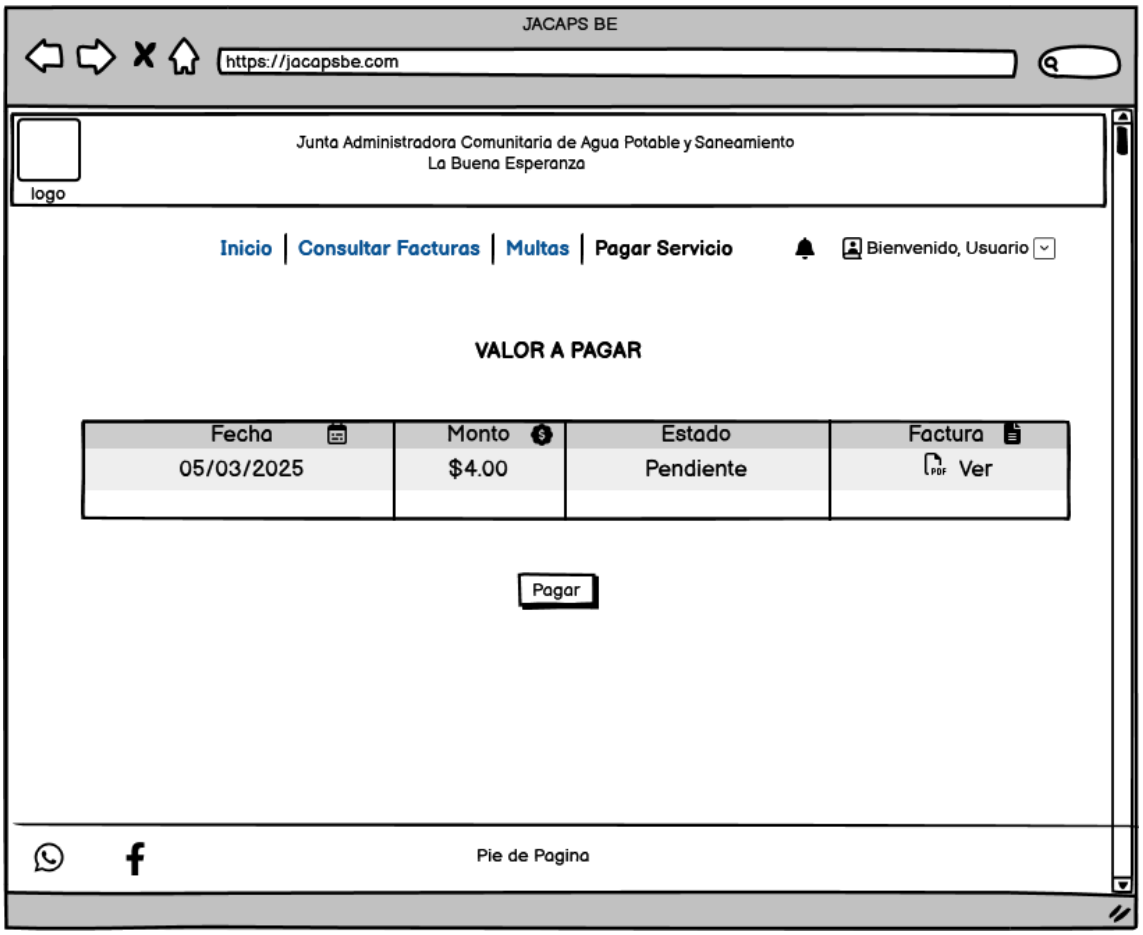


Figura 23.Diseño de pagar servicio.

4.3.Fase III: Codificación

Para la codificación de la aplicación web se utiliza php y html. Además, como gestor de base de datos se utilizó MySQL.

a. Backend

Ya que estamos trabajando con MVC(modelo-vista controlador), para el backend tenemos los modelos y los controladores, los mismos que manejarán las solicitudes de la lógica de la aplicación



Figura 24 Controladores que manejan las solicitudes de la aplicación.

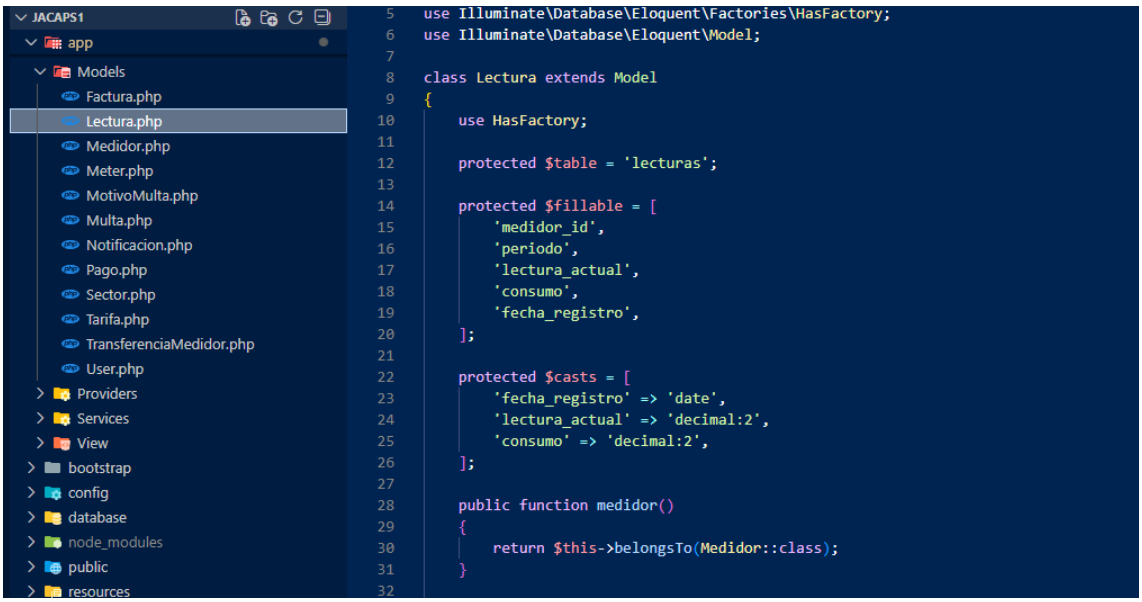


Figura 25 Modelos que interactúan con la base de datos.



Figura 26 Archivo de rutas para llamar a las diferentes interfaces.

b. Frontend

En la parte del frontend tenemos las vistas de la aplicación, las mismas que se comunicarán con el modelo y con los controladores para su correcto funcionamiento.

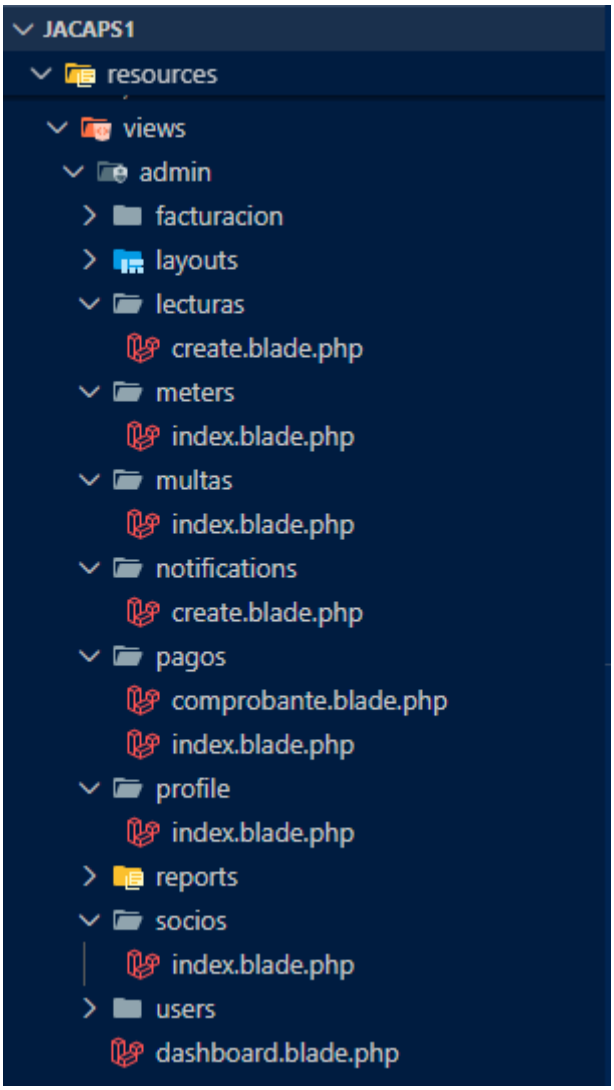


Figura 27 Views de la aplicación

create.blade.php

resources > views > admin > lecturas > create.blade.php

```
<nav aria-label="breadcrumb">
    <form action="{{ route('admin.lecturas.store') }}" method="POST">
        @csrf

        {{-- BUSCADOR DE MEDIDOR --}}
        <div class="mb-3 position-relative">
            <label class="form-label">Socio / Medidor:</label>

            <input type="text"
                   id="medidor_search"
                   class="form-control"
                   placeholder="Escriba nombre del socio o número de medidor"
                   autocomplete="off"
                   required>

            <input type="hidden" name="medidor_id" id="medidor_id">

            <div id="medidor_results"
                 class="list-group position-absolute w-100"
                 style="z-index:1050; display:none;"></div>

            @error('medidor_id')
                <span class="text-danger">{{ $message }}</span>
            @enderror
        </div>
        {{-- LECTURA ANTERIOR --}}
<div class="mb-3" id="lectura_anterior_box" style="display:none;">
    <label class="form-label text-muted">
        Lectura anterior:
        <strong id="lectura_anterior_valor">0.00</strong> m³
    </label>
</div>
```

Figura 28 Ejemplo del módulo de lecturas

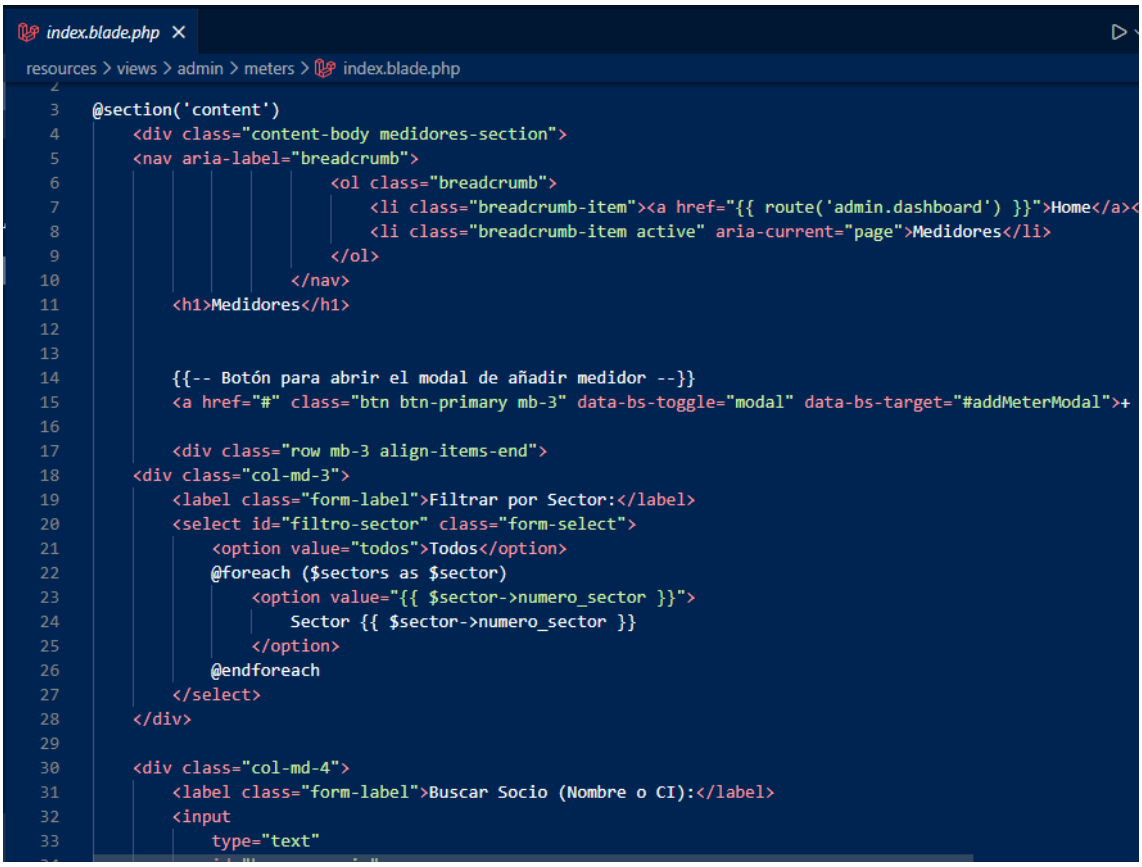


Figura 29 Ejemplo del módulo de medidores

A continuación, se presentan las pantallas de los módulos que se desarrollaron para la aplicación web según las historias de usuario mencionadas anteriormente.

a. Autenticación al sistema.

Al ingresar a la aplicación se mostrará la página de autenticación, en la cual el usuario debe disponer de las credenciales, y según su rol tiene acceso a los módulos.

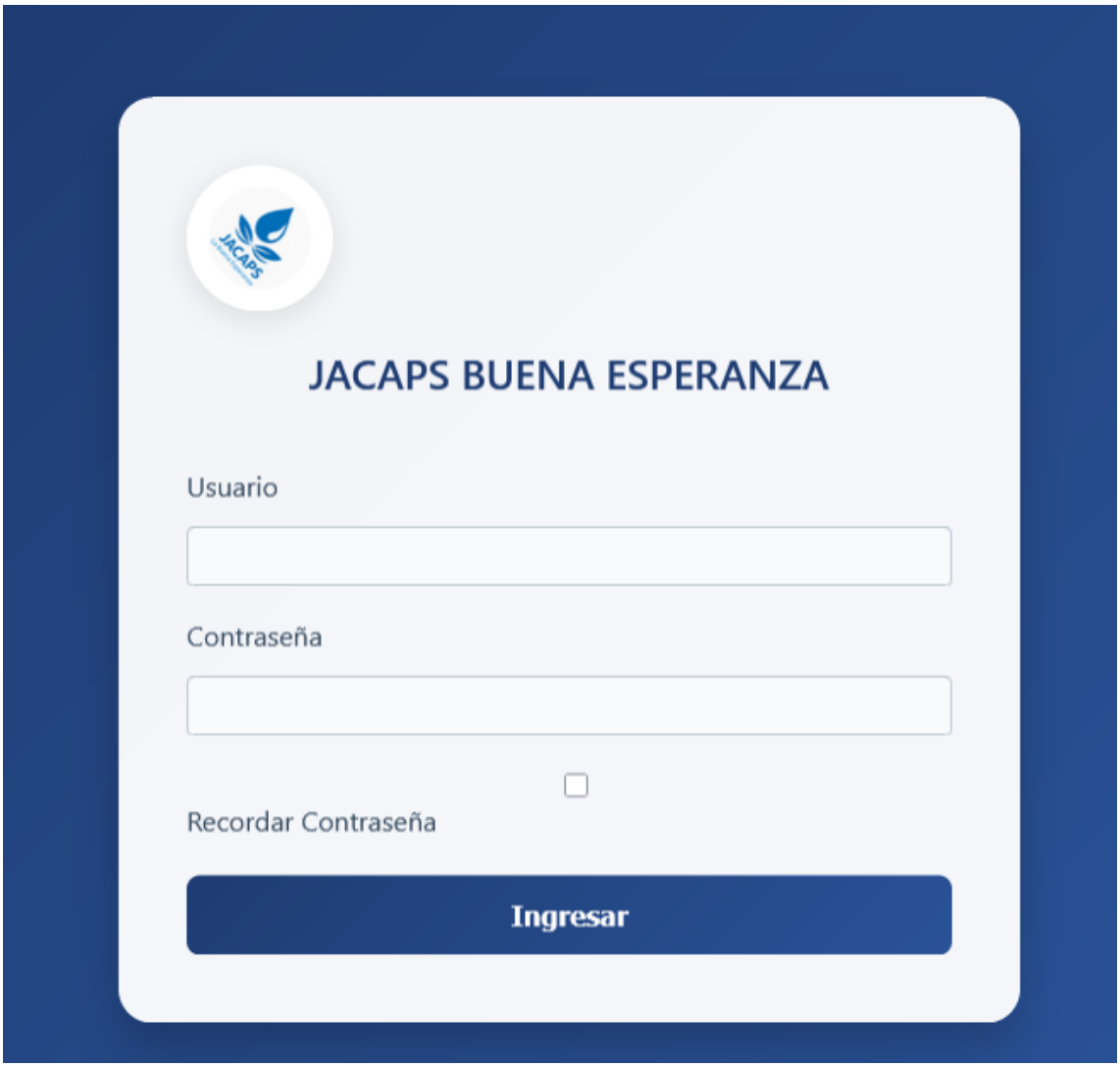


Figura 30 Página de Inicio de Sesión

b. Módulo de usuarios.

Este módulo permitirá la gestión de los usuarios del sistema, aquí podrá hacer visualizar, insertar. Estos datos se mostrarán en una tabla.

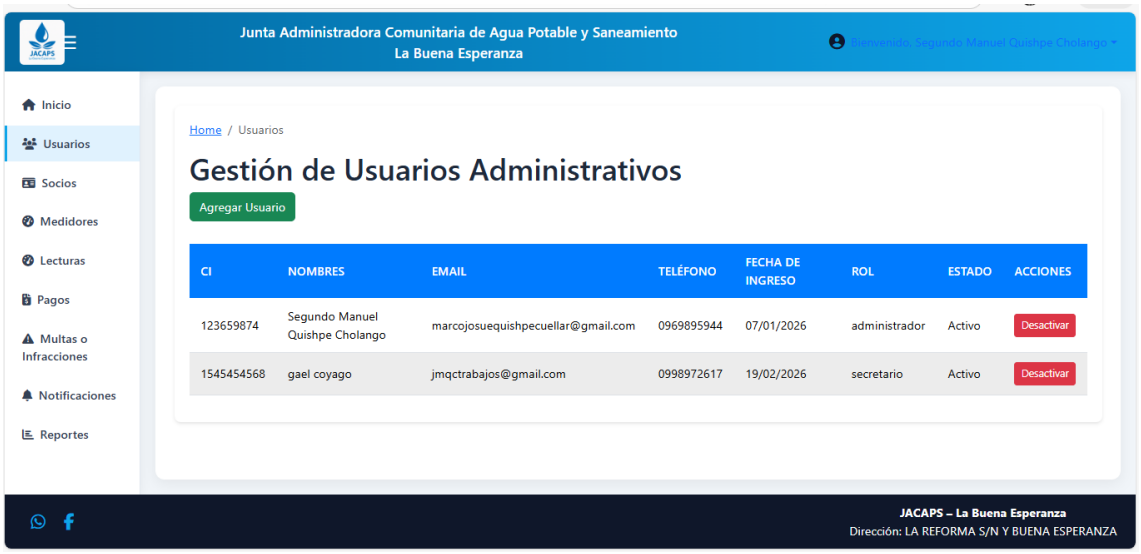


Figura 31 Tabla para la gestión de información de los usuarios.

En la siguiente figura se presenta el modal para ingresar los datos de un nuevo usuario que se requiera ser agregado al sistema.

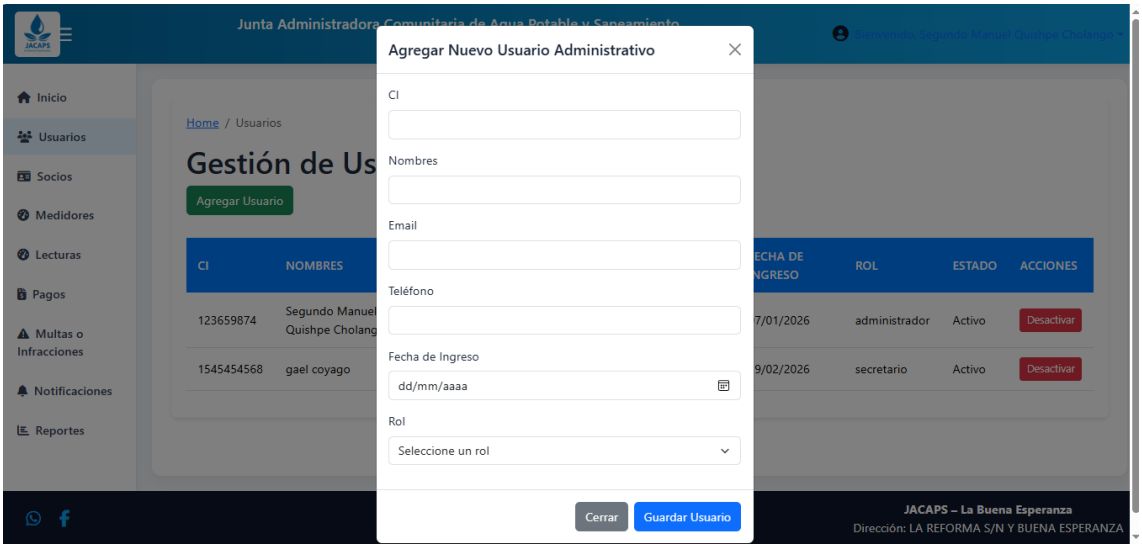


Figura 32 Modal para agregar un nuevo usuario.

De igual manera se puede mostrar el modal para desactivar un usuario existente.

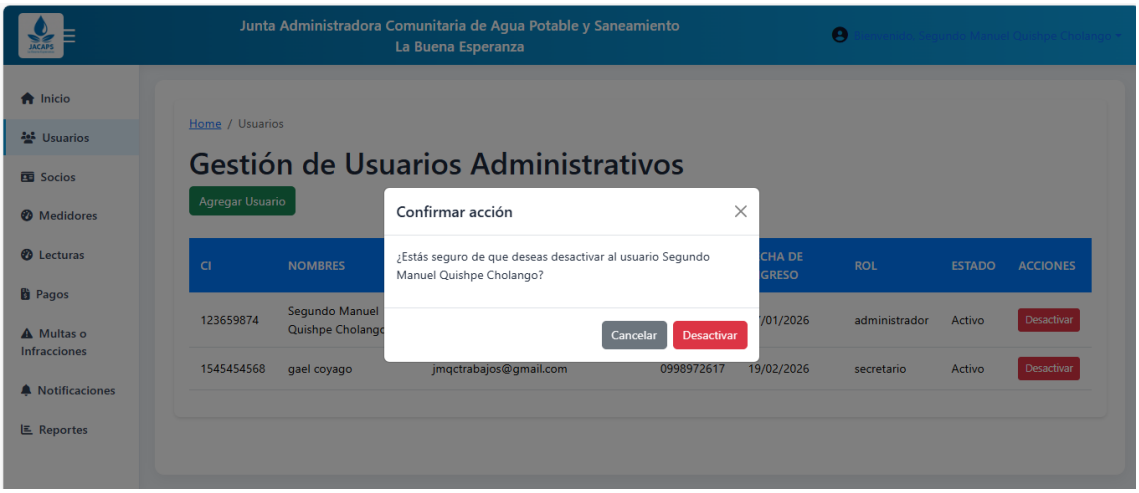


Figura 33 Modal para desactivar un usuario.

c. Módulo de socios de la junta

Aquí se podrá visualizar en una tabla los socios registrados en la junta de agua, se podrá realizar una búsqueda por el nombre o número de cédula. Además, se podrá agregar un nuevo socio o a su vez cambiar su estado a inactivo.



Figura 34 Tabla para la gestión de socios de la junta.

En esta figura se muestra el modal para agregar un nuevo socio a la junta.

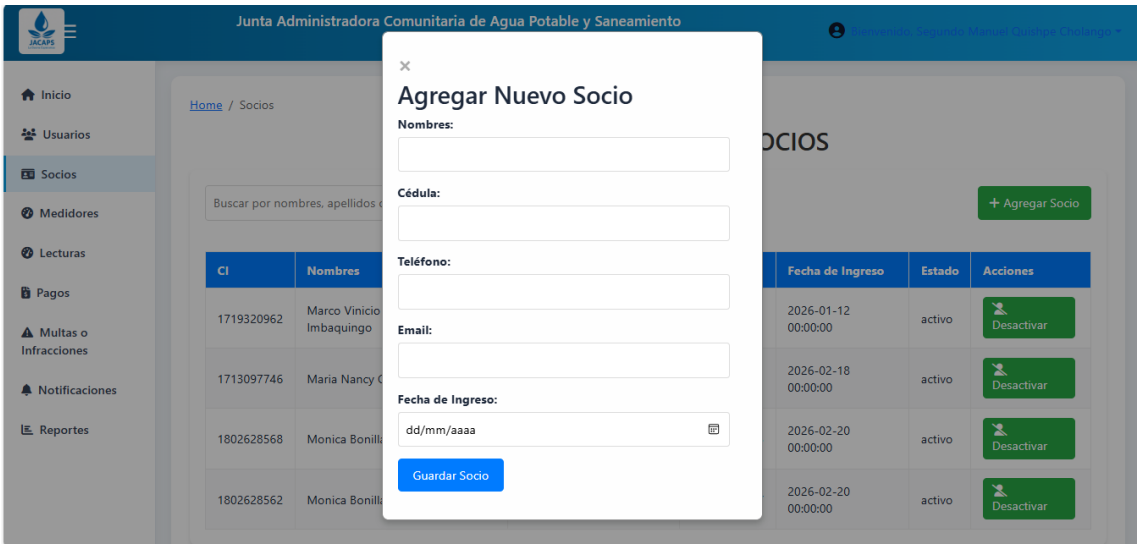


Figura 35 Modal para agregar un nuevo socio.

De igual forma en esta figura muestra el modal para desactivar un socio seleccionado.

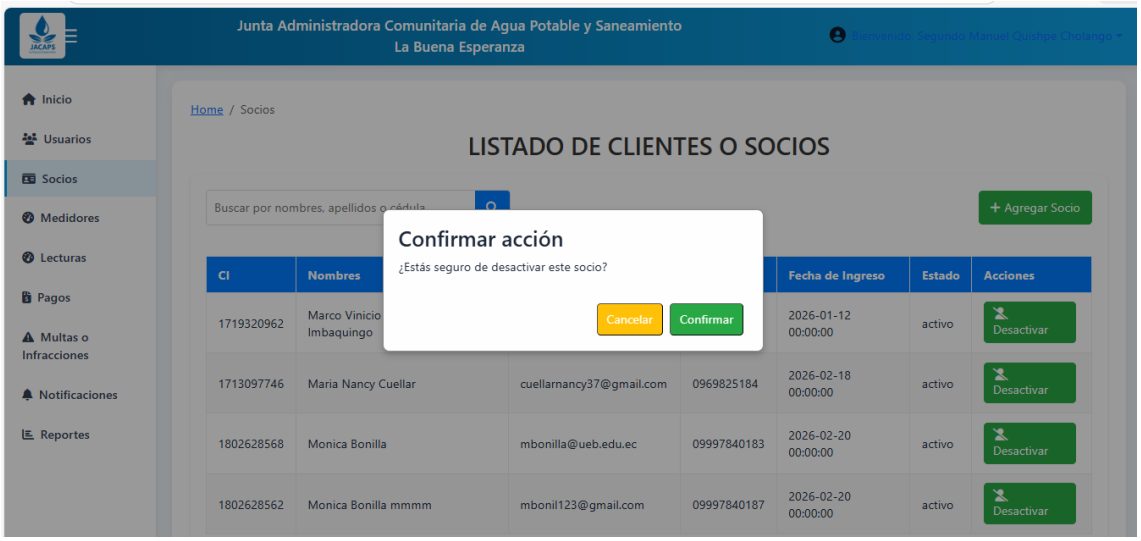


Figura 36 Modal para desactivar un socio.

d. Módulo de medidores

Aquí se muestra en una tabla los medidores que están activos en la junta de agua, además se podrá añadir nuevos medidores y a su vez hacer traspasos de nombre, y también cambiar su estado a inactivo.

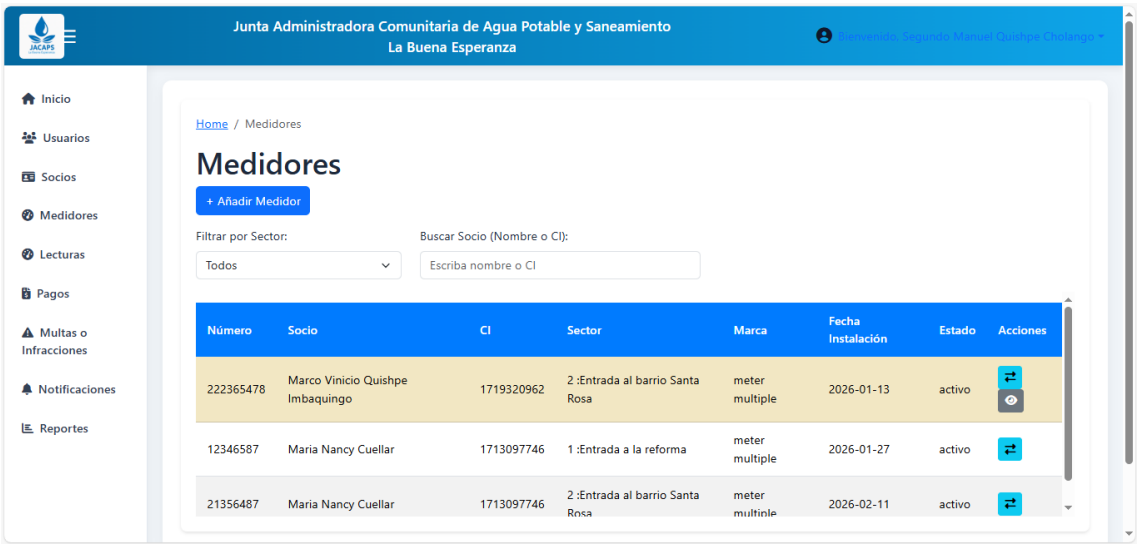


Figura 37 Tabla para la gestión de los medidores.

En la siguiente figura se muestra el modal para añadir un nuevo medidor, en el cuál ingresará los datos del mismo.

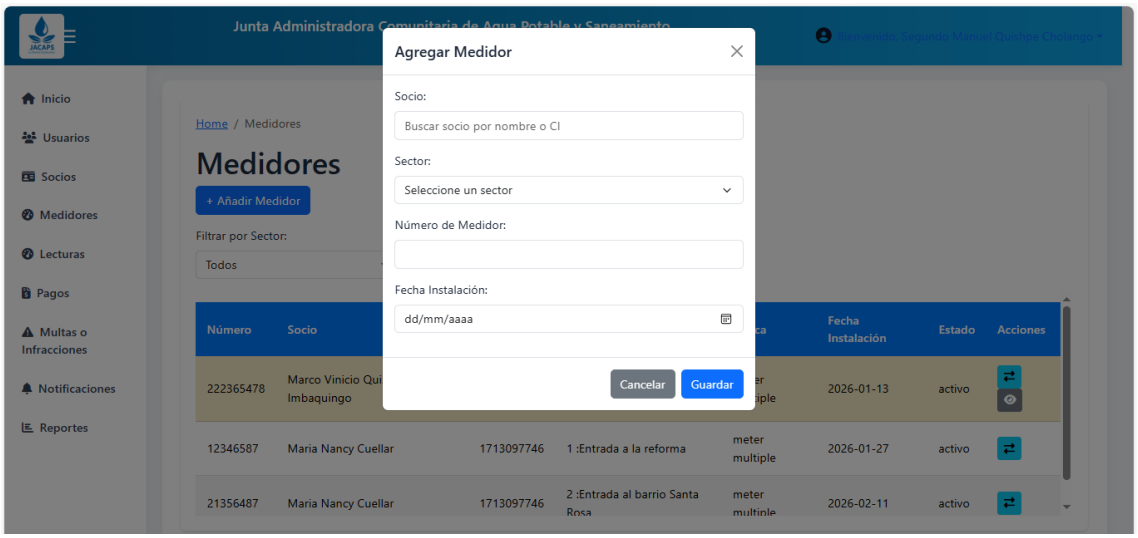


Figura 38 Modal para agregar un nuevo medidor.

Y como se muestra en la siguiente figura, el modal para hacer el traspaso de nombre de un medidor seleccionado.

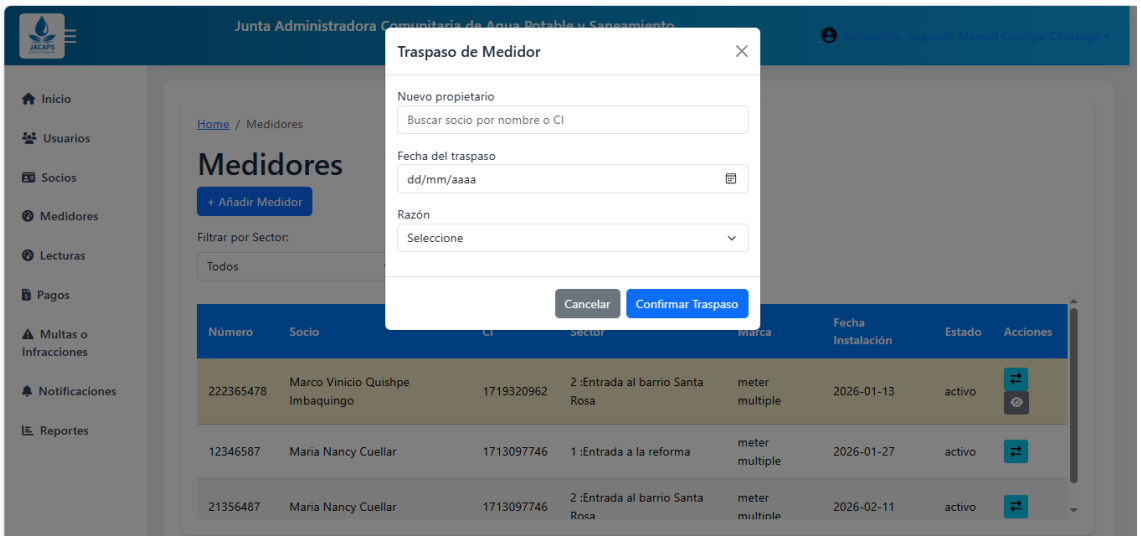


Figura 39 Modal para hacer traspaso de nombre de un medidor.

e. Módulo de pagos.

En este módulo se muestra los pagos pendientes del servicio que tenga algún socio de la junta. Además, nos muestra el total de los pagos pendientes.

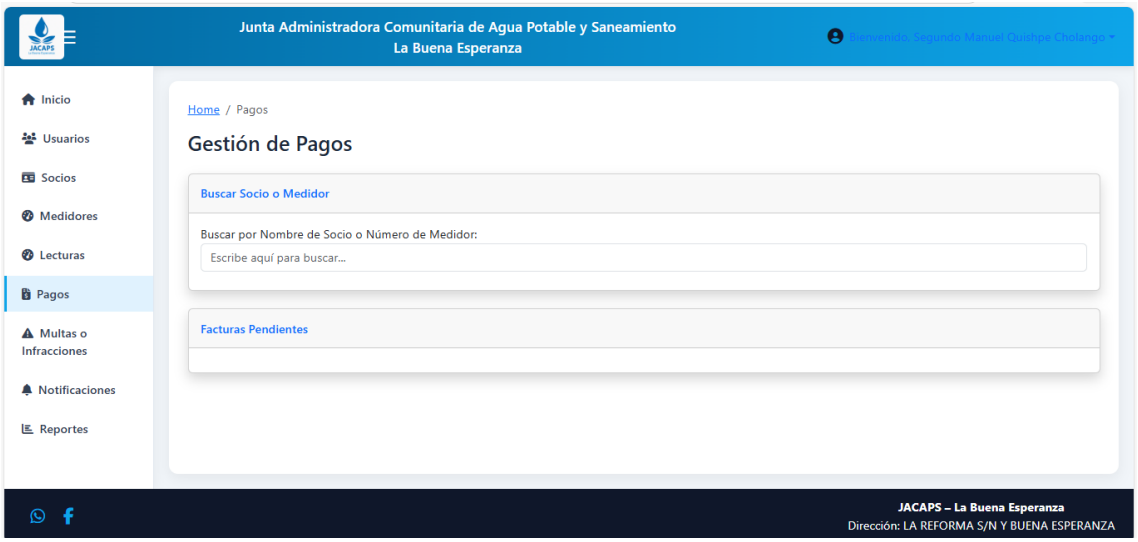


Figura 40 Tabla para visualizar los pagos pendientes.

f. Módulo de multas.

En este módulo se muestra el listado de las multas que han sido generadas a los socios, se pueden generar nuevas multas, anular multas, marcar como pagadas y se puede buscar por el nombre y también se puede filtrar según su estado, ya sea pagada o pendiente.

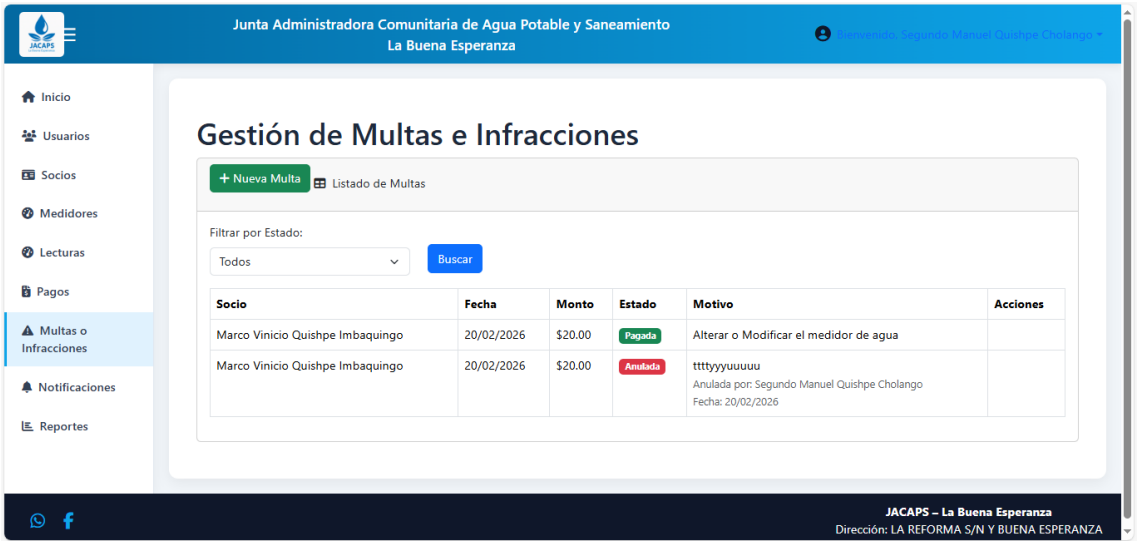


Figura 41 Tabla para la gestión de las multas.

En la siguiente figura se muestra el modal para generar una nueva multa, donde se deberá seleccionar el socio, y el motivo correspondiente.

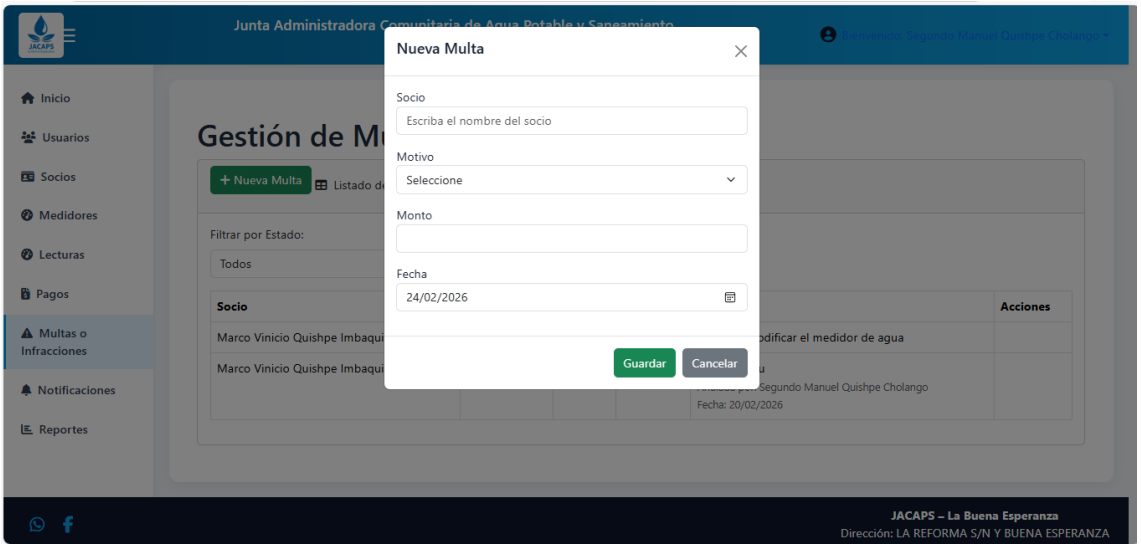


Figura 42 Modal para generar una nueva multa.

Así mismo como se muestra en la siguiente figura, se podrá abrir un modal para anular una multa.

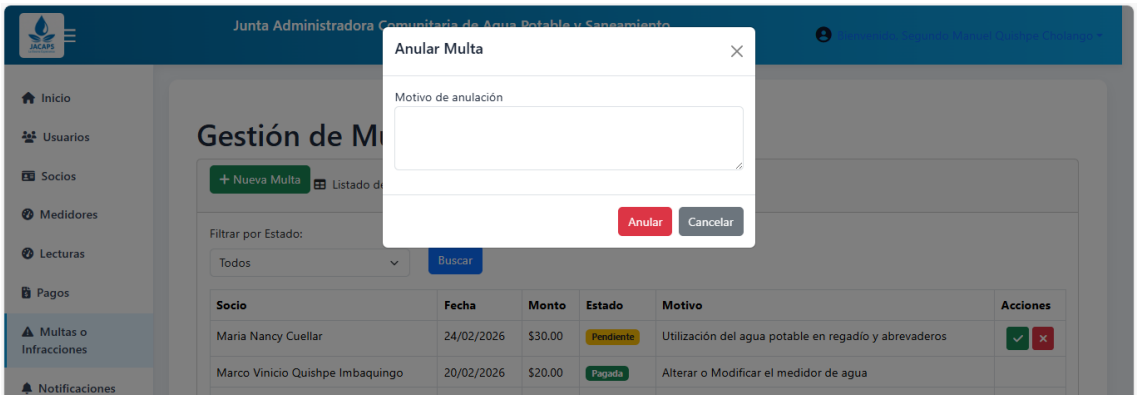


Figura 43 Modal para anular una multa.

g. Módulo de lecturas.

En este módulo se puede agregar nuevas lecturas para que con el consumo calculado se genere el valor a pagar para el medidor correspondiente.

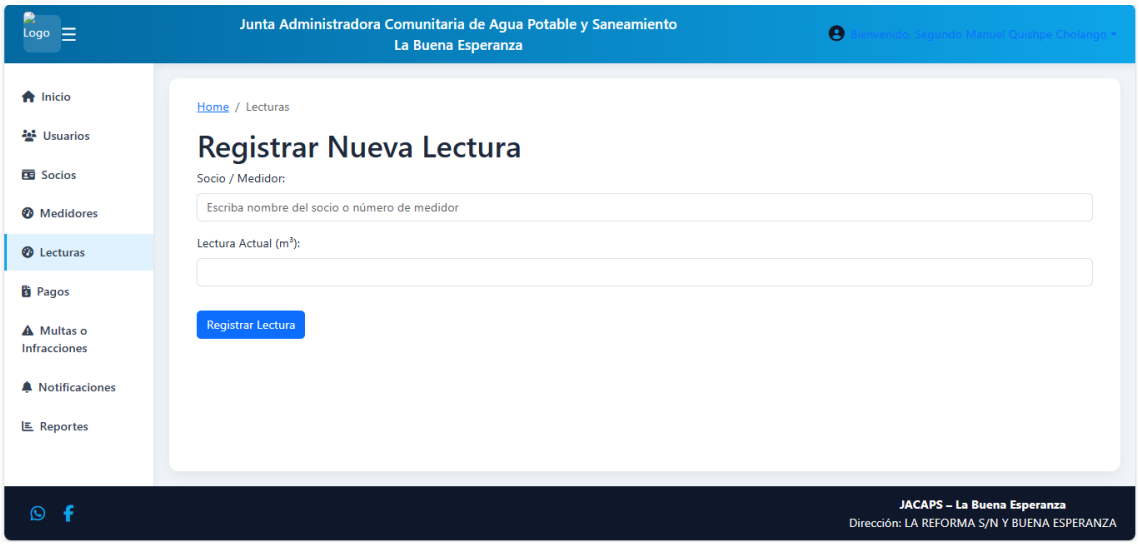


Figura 44 Tabla para registrar lecturas.

Y como se aprecia en la siguiente figura, también se puede agregar una nueva lectura para los medidores, para lo cuál se nos mostrará un modal en el que seleccionaremos el medidor y automáticamente nos va a mostrar el registro de la lectura anterior, para que así no haya ningún error con el cálculo del valor.

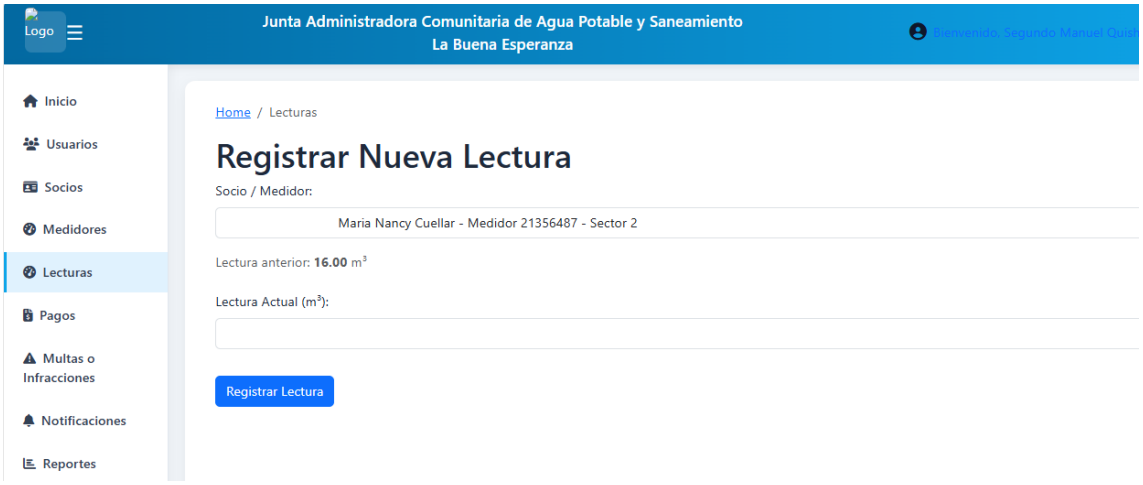


Figura 45 Modal para agregar nueva lectura.

h. Módulo de notificaciones.

En este módulo se mostrará un formulario para enviar notificaciones a los socios, en este caso se debe seleccionar el destinatario y se escribe el mensaje deseado, también tiene la opción de enviarle un mismo mensaje a todos los socios.

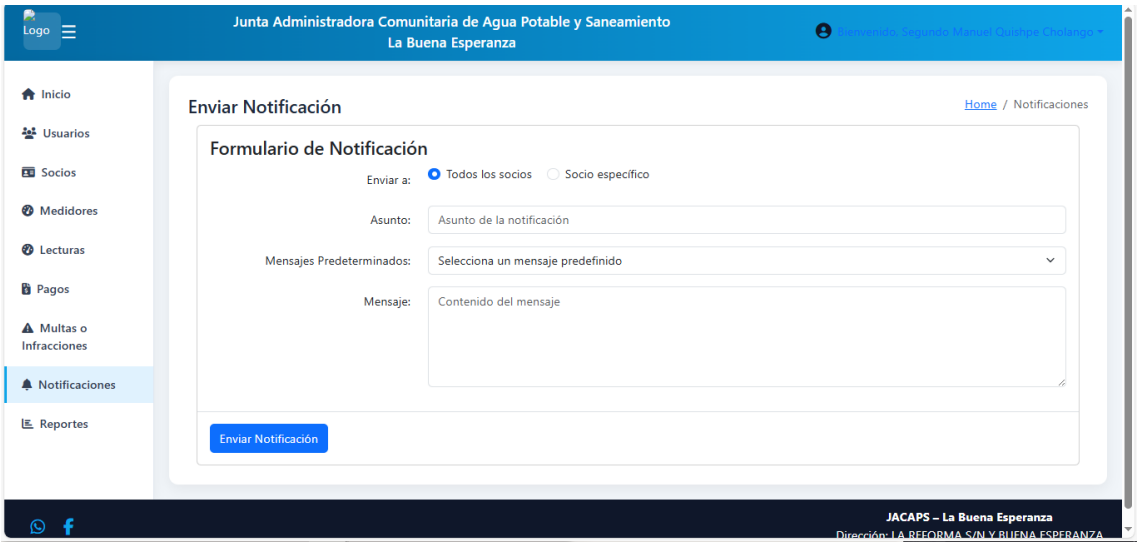


Figura 46 Formulario para enviar notificaciones a los socios.

i. Módulo de reportes.

En este módulo se puede generar 2 tipos de reportes, uno sobre el dinero recaudado y otro para las cuentas pendientes en cierto mes y año, con la opción de descargar en formato PDF.

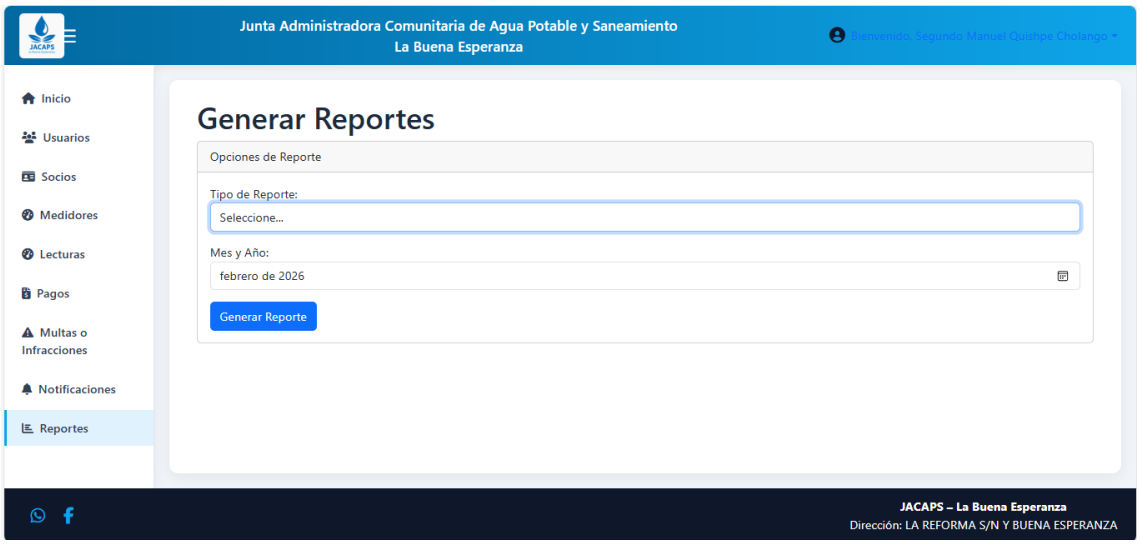


Figura 47 Módulo para generar reportes.

j. Módulo de historial de pagos.

En este módulo el socio visualizará una tabla de historial con los datos de las facturas que ya han sido pagadas, las mismas que tendrá la opción de descargar su respectivo comprobante.



Figura 48 Tabla para visualizar el historial de pagos.

k. Módulo de pagar multas

En este módulo el socio visualizará en una tabla las multas que tenga generadas mostrando la fecha, el valor y motivo de la multa con la opción de pagarla.



Figura 49 Tabla para ver y pagar multas.

Al presionar el botón para pagar se abrirá la ventana para el respectivo pago.

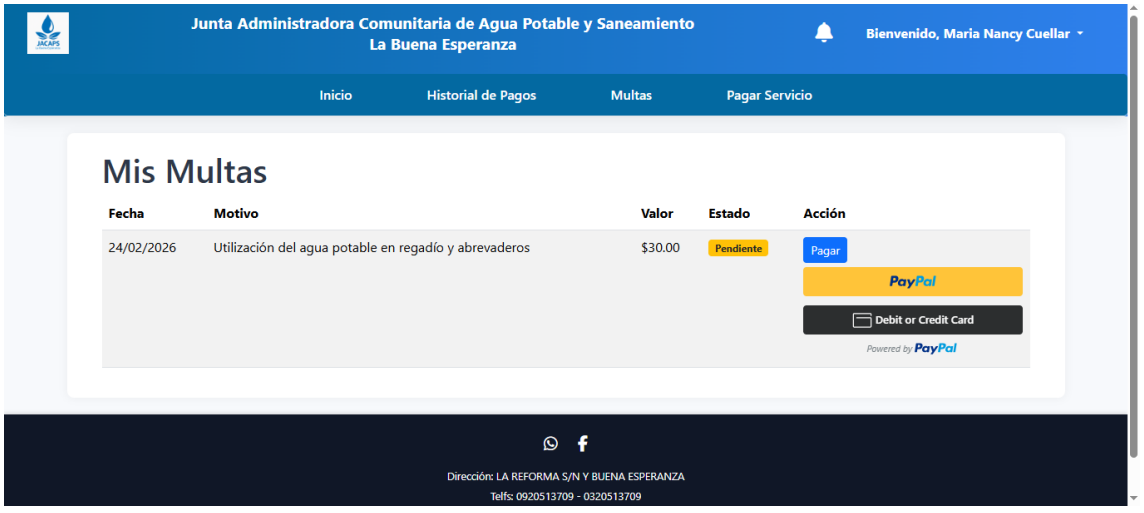


Figura 50 Ventana para pago de la multa

l. Módulo de pagar servicio.

En este módulo se muestra los valores a pagar que han sido generados a partir de las lecturas registradas en los medidores, así mismo con la opción de pagar cada valor.



Figura 51 Tabla de valor a pagar del servicio.

Así mismo al dar click en pagar se mostrará la ventana para proceder con el pago, como podemos ver en la siguiente figura.

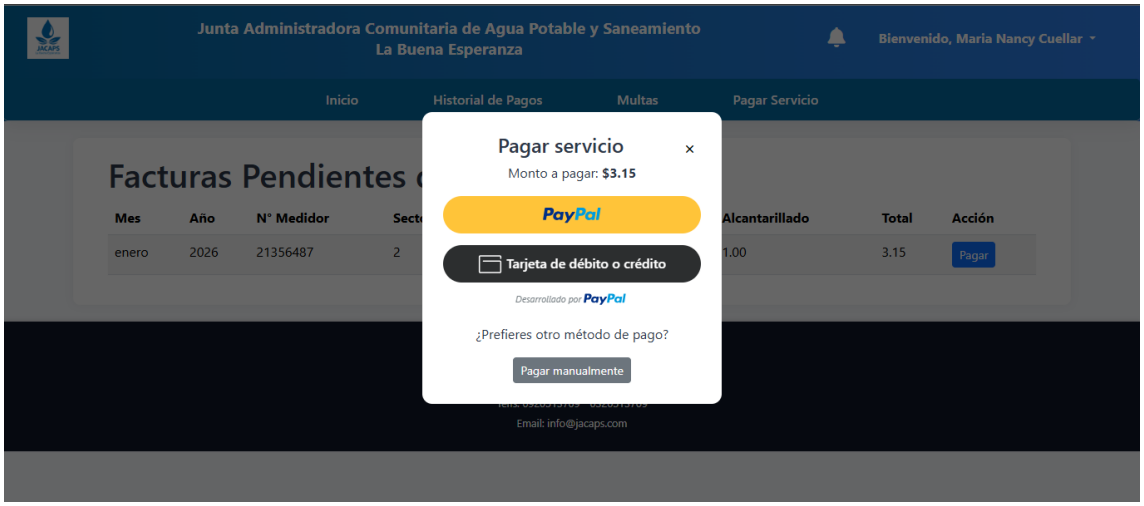


Figura 52 Ventana para pago del servicio.

4.4.Fase IV: Pruebas

a. Pruebas unitarias

Estas pruebas se ejecutaron con el objetivo de validar la aplicación web guiándose en los requerimientos de las historias de usuario.

Módulo de usuarios

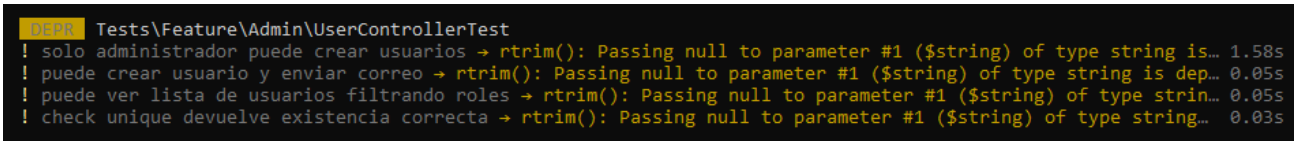


Tabla 64: Pruebas Unitarias módulo usuarios

Criterio	¿Completado?
Preparación del equipo XP	
El programador responsable completó el desarrollo del módulo de usuarios	SÍ
La historia fue integrada correctamente en la iteración asignada	SÍ
No existen errores bloqueantes asociados al módulo	SÍ
Preparación de requisitos	
La historia de usuario está claramente definida y comprendida	SÍ
Los criterios de aceptación están documentados y validados	SÍ
Las acciones permitidas (agregar, editar, desactivar, buscar) están definidas	SÍ
Preparación del entorno de pruebas	
El entorno de pruebas está disponible y operativo	SÍ
El módulo de usuarios es accesible solo para el rol Administrador	SÍ
Existen usuarios de prueba registrados en el sistema	SÍ
Preparación de datos	
Existen datos de prueba para búsqueda de usuarios	SÍ
Existen datos de prueba para edición de usuarios	SÍ
Existen datos de prueba para desactivación de usuarios	SÍ
Existen datos de prueba para creación de nuevos usuarios	SÍ
Criterios de aceptación – Búsqueda	
El administrador puede buscar usuarios por nombre	SÍ
La tabla filtra correctamente los usuarios según el nombre ingresado	SÍ
La búsqueda muestra resultados parciales y completos	SÍ
Criterios de aceptación – Desactivación	
El botón “Desactivar” está disponible para cada usuario	SÍ
Al seleccionar “Desactivar” se muestra una alerta de confirmación	SÍ
El usuario se desactiva solo después de confirmar la acción	SÍ
Criterios de aceptación – Creación	

El botón “Agregar usuario” está disponible	SÍ
Al seleccionar “Agregar usuario” se despliega un modal de registro	SÍ
Es posible ingresar la información del nuevo usuario	SÍ
El nuevo usuario se guarda correctamente	SÍ
Criterios generales de aceptación	
No existen defectos críticos asociados a la historia	SÍ
El módulo cumple con los criterios de aceptación definidos	SÍ

Módulo de socios

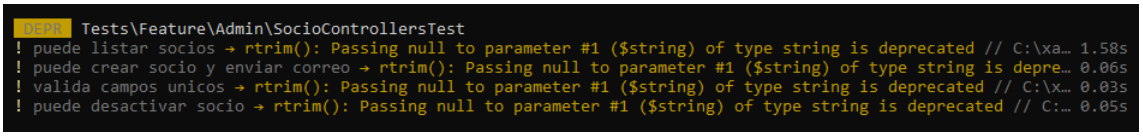


Tabla 65:Pruebas unitarias módulo socios

Criterio	¿Completado?
Preparación de requisitos	
La historia de usuario está claramente definida y comprendida	SÍ
Los criterios de aceptación están documentados y validados	SÍ
Las acciones permitidas (agregar, desactivar, buscar) están definidas	SÍ
Preparación del entorno de pruebas	
El entorno de pruebas está disponible y operativo	SÍ
El módulo de socios es accesible solo para el rol Secretario	SÍ
Existen socios de prueba registrados en el sistema	SÍ
Preparación de datos	
Existen datos de prueba para búsqueda de socios	SÍ
Existen datos de prueba para desactivación de socios	SÍ
Existen datos de prueba para creación de nuevos socios	SÍ
Criterios de aceptación – Desactivación	
El botón “Desactivar” está disponible para cada socio	SÍ
Al seleccionar “Desactivar” se muestra una alerta de confirmación	SÍ
El socio se desactiva solo después de confirmar la acción	SÍ
Criterios de aceptación – Creación	
El botón “Agregar socio” está disponible	SÍ
Al seleccionar “Agregar socio” se despliega un modal de registro	SÍ
Es posible ingresar la información del nuevo socio	SÍ
El nuevo socio se guarda correctamente	SÍ
Criterios generales de aceptación	
No existen defectos críticos asociados a la historia	SÍ
El módulo cumple con los criterios de aceptación definidos	SÍ

Módulo de medidores

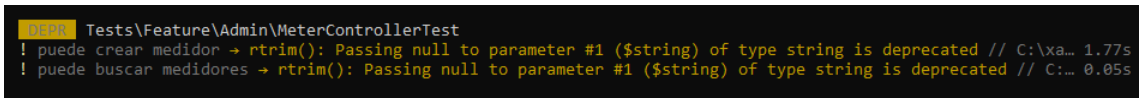


Tabla 66:Pruebas unitarias módulo medidores

Criterio	¿Completado?
Preparación del equipo XP	
El programador responsable completó el desarrollo del módulo de medidores	SÍ
La historia fue integrada correctamente en la iteración asignada	SÍ
No existen errores bloqueantes asociados al módulo	SÍ
Preparación de requisitos	
La historia de usuario está claramente definida y comprendida	SÍ
Los criterios de aceptación están documentados y validados	SÍ
Las acciones permitidas (agregar, traspasar, desactivar) están definidas	SÍ
Preparación del entorno de pruebas	
El entorno de pruebas está disponible y operativo	SÍ
El módulo de medidores es accesible para el rol Secretario	SÍ
Existen medidores de prueba registrados en el sistema	SÍ
Preparación de datos	
Existen datos de prueba para registro de medidores	SÍ
Existen datos de prueba para traspaso de medidores	SÍ
Existen datos de prueba para desactivación de medidores	SÍ
Criterios de aceptación – Registro de medidores	
El secretario puede agregar un nuevo medidor	SÍ
El botón “Añadir medidor” está disponible	SÍ
Al seleccionar “Añadir medidor” se despliega un modal de registro	SÍ
Es posible ingresar la información del nuevo medidor	SÍ
El medidor se guarda correctamente	SÍ
Criterios de aceptación – Traspaso de medidores	
El botón “Traspasos” está disponible	SÍ
Al seleccionar “Traspasos” se despliega un modal de traspaso	SÍ
Es posible ingresar la nueva información del medidor	SÍ
El traspaso se guarda correctamente	SÍ
Criterios generales de aceptación	
No existen defectos críticos asociados a la historia	SÍ
El módulo cumple con los criterios de aceptación definidos	SÍ

Módulo de multas

Tabla 67: Pruebas unitarias módulo multas

Criterio	¿Completado?
Preparación del equipo XP	
El programador responsable completó el desarrollo del módulo de multas	SÍ
La historia fue integrada correctamente en la iteración asignada	SÍ
No existen errores bloqueantes asociados al módulo	SÍ
Preparación de requisitos	
La historia de usuario está claramente definida y comprendida	SÍ
Los criterios de aceptación están documentados y validados	SÍ
Las acciones permitidas (visualizar, generar, buscar, filtrar multas) están definidas	SÍ
Preparación del entorno de pruebas	
El entorno de pruebas está disponible y operativo	SÍ
El módulo de multas es accesible solo para el rol Tesorero	SÍ
Existen multas de prueba registradas en el sistema	SÍ
Preparación de datos	
Existen datos de prueba para generación de multas	SÍ
Existen datos de prueba para búsqueda de multas	SÍ
Existen datos de prueba para filtrado por estado	SÍ
Criterios de aceptación – Registro de medidores	
El secretario puede agregar un nuevo medidor	SÍ
El botón “Añadir medidor” está disponible	SÍ
Al seleccionar “Añadir medidor” se despliega un modal de registro	SÍ
Es posible ingresar la información del nuevo medidor	SÍ
El medidor se guarda correctamente	SÍ
Criterios de aceptación – Traspaso de medidores	
El botón “Traspasos” está disponible	SÍ
Al seleccionar “Traspasos” se despliega un modal de traspaso	SÍ
Es posible ingresar la nueva información del medidor	SÍ
El traspaso se guarda correctamente	SÍ
Criterios generales de aceptación	
No existen defectos críticos asociados a la historia	SÍ
El módulo cumple con los criterios de aceptación definidos	SÍ

Módulo de lecturas

Tabla 68: Pruebas unitarias módulo lecturas

Criterio	¿Completado?
Preparación del equipo XP	
El programador responsable completó el desarrollo del módulo de lecturas	SÍ
La historia fue integrada correctamente en la iteración asignada	SÍ
No existen errores bloqueantes asociados al módulo	SÍ
Preparación de requisitos	
La historia de usuario está claramente definida y comprendida	SÍ
El criterio de aceptación está documentado y validado	SÍ
La acción de ingresar nuevas lecturas está claramente definida	SÍ
Preparación del entorno de pruebas	
El entorno de pruebas está disponible y operativo	SÍ
El módulo de lecturas es accesible solo para el rol Tesorero	SÍ
Existen medidores registrados en el sistema	SÍ
Preparación de datos	
Existen datos de prueba de medidores	SÍ
Existen datos de prueba para ingreso de lecturas	SÍ
Existen lecturas previas para validación del registro	SÍ
Criterios de aceptación – Ingreso de lecturas	
El tesorero puede ingresar al módulo de lecturas desde el menú	SÍ
El botón “Agregar lecturas” está disponible	SÍ
Al seleccionar “Agregar lecturas” se permite buscar el medidor	SÍ
El sistema permite ingresar la lectura actual del medidor	SÍ
La lectura se guarda correctamente	SÍ
Validaciones	
El sistema valida que el medidor exista	SÍ
El sistema valida que la lectura ingresada sea numérica	SÍ
Criterios generales de aceptación	
No existen defectos críticos asociados a la historia	SÍ
El módulo cumple con el criterio de aceptación definido	SÍ

Módulo de pagos

Tabla 69:Pruebas unitarias módulo pagos

Criterio	¿Completado?
Preparación del equipo XP	
El programador responsable completó el desarrollo del módulo de pagos	SÍ
La historia fue integrada correctamente en la iteración asignada	SÍ
No existen errores bloqueantes asociados al módulo	SÍ
Preparación de requisitos	
La historia de usuario está claramente definida y comprendida	SÍ
Los criterios de aceptación están documentados y validados	SÍ
Las acciones permitidas (buscar planillas, marcar como pagadas) están definidas	SÍ
Preparación del entorno de pruebas	
El entorno de pruebas está disponible y operativo	SÍ
El módulo de pagos es accesible solo para el rol Tesorero	SÍ
Existen planillas generadas en el sistema	SÍ
Preparación de datos	
Existen planillas pendientes de pago	SÍ
Existen socios registrados con medidores asociados	SÍ
Existen datos de prueba para búsqueda por nombre y cédula	SÍ
Criterios de aceptación – Visualización de planillas	
El tesorero puede ingresar al módulo de pagos desde el menú	SÍ
Se muestra una tabla con las planillas pendientes de pago	SÍ
La información de las planillas es clara y completa	SÍ
Criterios de aceptación – Búsqueda de planillas	
El tesorero puede buscar planillas por nombre del socio	SÍ
El tesorero puede buscar planillas por cédula del socio	SÍ
La tabla filtra correctamente las planillas buscadas	SÍ
Criterios de aceptación – Pago de planillas	
El sistema muestra los datos del medidor asociado a la planilla	SÍ
El tesorero puede marcar una planilla como pagada	SÍ
El estado de la planilla cambia correctamente a “Pagada”	SÍ
Criterios generales de aceptación	
No existen defectos críticos asociados a la historia	SÍ
El módulo cumple con los criterios de aceptación definidos	SÍ

Módulo de reportes

Tabla 70: Pruebas unitarias módulo reportes

Criterio	¿Completado?
Preparación del equipo XP	
El programador responsable completó el desarrollo del módulo de reportes	SÍ
La historia fue integrada correctamente en la iteración asignada	SÍ
No existen errores bloqueantes asociados al módulo	SÍ
Preparación de requisitos	
La historia de usuario está claramente definida y comprendida	SÍ
Los criterios de aceptación están documentados y validados	SÍ
Los tipos de reportes a generar están claramente definidos	SÍ
Preparación del entorno de pruebas	
El entorno de pruebas está disponible y operativo	SÍ
El módulo de reportes es accesible solo para el rol Administrador	SÍ
Existen datos registrados para la generación de reportes	SÍ
Preparación de datos	
Existen datos de recaudaciones por mes	SÍ
Existen cuentas pendientes registradas	SÍ
Existen socios activos e inactivos	SÍ
Existen medidores activos e inactivos	SÍ
Existen registros de traspasos de medidores	SÍ
Criterios de aceptación – Generación de reportes	
El administrador puede ingresar al módulo de reportes desde el menú	SÍ
El sistema permite seleccionar el tipo de reporte	SÍ
El botón “Generar” está disponible	SÍ
El sistema muestra un preview del reporte solicitado	SÍ
El reporte presenta información correcta y completa	SÍ
Criterios de aceptación – Descarga de reportes	
El sistema permite descargar el reporte en formato PDF	SÍ
El archivo PDF se genera correctamente	SÍ
El contenido del PDF coincide con el preview mostrado	SÍ
Criterios generales de aceptación	
No existen defectos críticos asociados a la historia	SÍ
El módulo cumple con los criterios de aceptación definidos	SÍ

Módulo de historial de pagos

Tabla 71: Pruebas unitarias módulo historial de pagos

Criterio	¿Completado?
Preparación del equipo XP	
El programador responsable completó el desarrollo del módulo de historial de pagos	SÍ
La historia fue integrada correctamente en la iteración asignada	SÍ
No existen errores bloqueantes asociados al módulo	SÍ
Preparación de requisitos	
La historia de usuario está claramente definida y comprendida	SÍ
Los criterios de aceptación están documentados y validados	SÍ
Las funcionalidades (ver historial, buscar por fecha, descargar comprobante) están definidas	SÍ
Preparación del entorno de pruebas	
El entorno de pruebas está disponible y operativo	SÍ
El módulo de historial de pagos es accesible solo para el rol Socio	SÍ
Existen pagos registrados para el socio	SÍ
Preparación de datos	
Existen registros de pagos históricos	SÍ
Existen comprobantes asociados a los pagos	SÍ
Existen datos de prueba para búsqueda por fecha	SÍ
Criterios de aceptación – Visualización del historial	
El socio puede ingresar al módulo de historial de pagos desde el menú	SÍ
Se muestra una tabla con el historial de pagos del socio	SÍ
La información del historial es clara y correcta	SÍ
Criterios de aceptación – Búsqueda de pagos	
El socio puede buscar pagos por fecha	SÍ
El sistema filtra correctamente los pagos según la fecha ingresada	SÍ
Criterios de aceptación – Descarga de comprobantes	
El botón “Ver Comprobante” está disponible	SÍ
El sistema permite descargar el comprobante en formato PDF	SÍ
El comprobante descargado corresponde al pago seleccionado	SÍ
Criterios generales de aceptación	
No existen defectos críticos asociados a la historia	SÍ
El módulo cumple con los criterios de aceptación definidos	SÍ

b. Pruebas de Seguridad

Este tipo de pruebas se realiza para identificar vulnerabilidades en el programa, con esto garantizamos la protección de datos y no corremos riesgos de futuros ataques.

Para esto se utilizó la herramienta OWASP ZAP, esta herramienta realiza ataques simultáneos al sistema en base al top 10 de las vulnerabilidades de los sitios web.

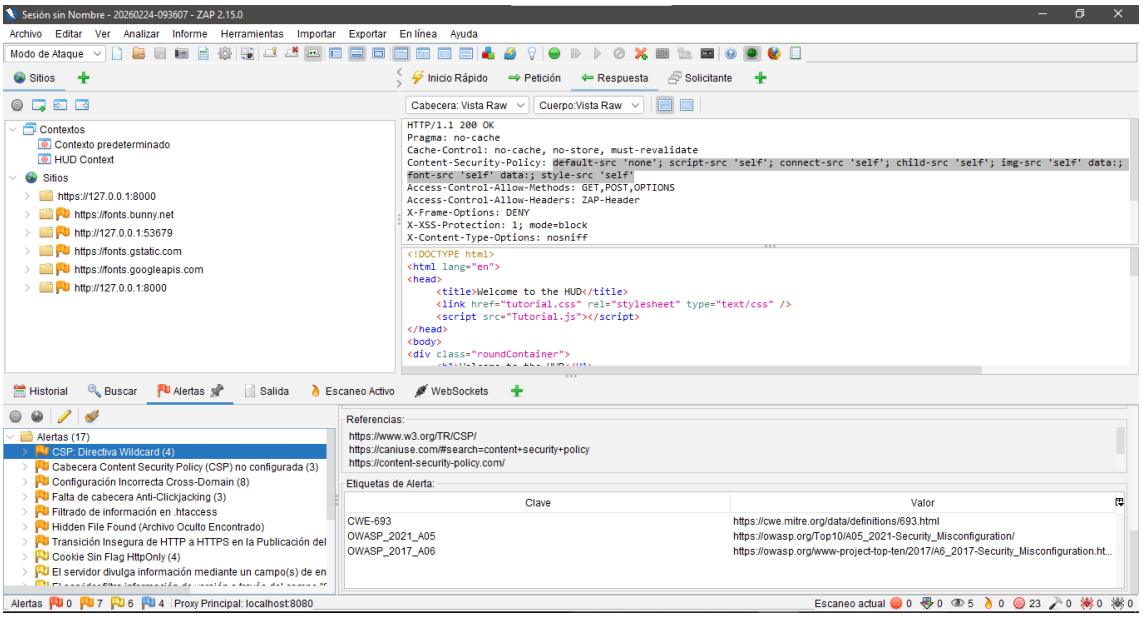


Figura 53 Pruebas realizadas en OWASP ZAP

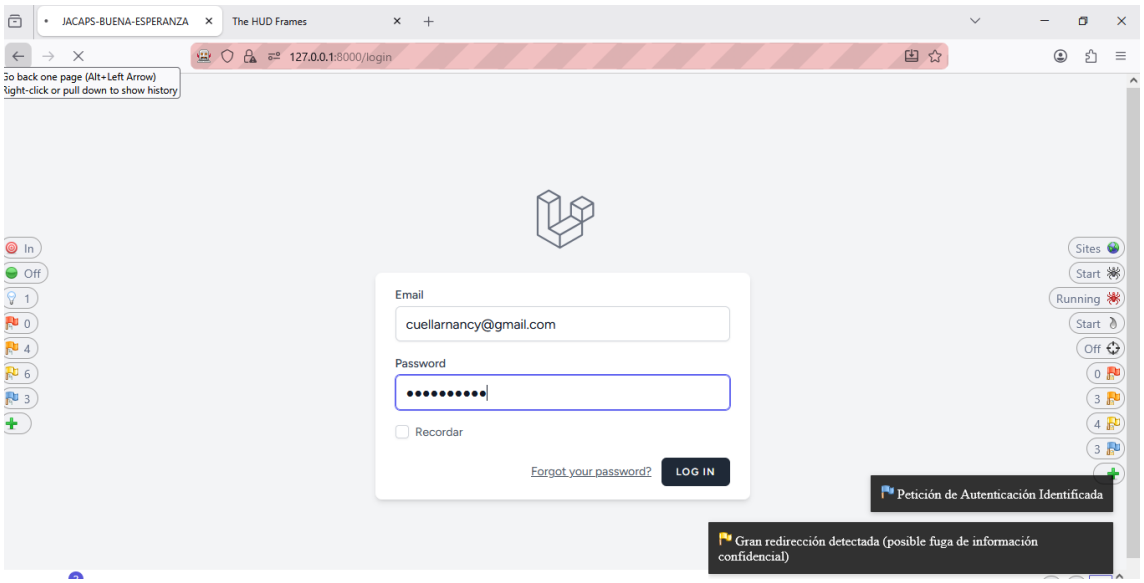


Figura 54 Ejecución de OWASP ZAP

Tabla 72 Resultado de pruebas realizadas en OWASP ZAP

Vulnerabilidad	Descripción	Recomendación
Cabecera Content Security Policy (CSP) no configurada (3)	La Política de seguridad de contenido (CSP) es una capa adicional de seguridad que ayuda a detectar y mitigar ciertos tipos de ataques, incluidos Cross Site Scripting (XSS) y ataques de inyección de datos. Estos ataques se utilizan para todo, desde el robo de datos hasta la desfiguración del sitio o la distribución de malware. CSP proporciona un conjunto de encabezados HTTP estándar que permiten a los propietarios de sitios web declarar fuentes de contenido aprobadas que los navegadores deberían poder cargar en esa página; los tipos cubiertos son JavaScript, CSS, marcos HTML, fuentes, imágenes y	Asegúrese de que su servidor web, servidor de aplicaciones, balanceador de carga, etc. esté configurado para establecer la cabecera Content-Security-Policy.

	objetos incrustados como applets de Java, ActiveX, archivos de audio y video.	
Configuración Incorrecta Cross-Domain (8)	La configuración incorrecta de CORS en el servidor web permite solicitudes de lectura entre dominios de terceros arbitrarios, utilizando API no autenticadas en este dominio. Sin embargo, las implementaciones de los navegadores web no permiten que terceros arbitrarios lean la respuesta de las API autenticadas. Esto reduce un poco el riesgo. Esta configuración errónea podría ser utilizada por un atacante para acceder a datos que están disponibles de forma no autenticada, pero que utilizan alguna otra forma de seguridad, como la lista blanca de direcciones IP.	Asegúrese de que los datos confidenciales no estén disponibles de forma no autenticada (por ejemplo, mediante listas blancas de direcciones IP). Configure el encabezado HTTP "Access-Control-Allow-Origin" a un conjunto más restrictivo de dominios, o elimine todos los encabezados CORS por completo, para

		permitir que el navegador web aplique la Política del Mismo Origen (SOP) de una manera más restrictiva.
Transición Insegura de HTTP a HTTPS en la Publicación del Formulario	La respuesta a la siguiente solicitud a través de HTTP incluía un valor de atributo de acción de etiqueta de formulario HTTPS: http://127.0.0.1:8000/login	Usar HTTPS para páginas de destino que alojan formularios seguros.
El servidor divulga información mediante un campo(s) de encabezado de respuesta HTTP ""X-Powered-By"" (4)	El servidor de la web/aplicación está divulgando información mediante uno o más encabezados de respuesta HTTP ""X-Powered-By"". El acceso a tal información podría facilitarle a los atacantes la identificación de otros marcos/componentes de los que su aplicación web depende y las vulnerabilidades a las que pueden estar sujetos tales componentes.	Asegúrese de que su servidor web, servidor de aplicaciones, balanceador de carga, etc. está configurado para suprimir las cabeceras "X-Powered-By".

Divulgación de información - Comentarios sospechosos (3)	La respuesta parece contener comentarios sospechosos que pueden ayudar a un atacante. Nota: Las coincidencias realizadas dentro de los scripts o archivos se refieren a todo el contenido, no solo a los comentarios.	Eliminar todos los comentarios que muestren información que pueda ayudar a un atacante y solucionar el problema al que se refieren.

Según el análisis de OWASP ZAP nos permitió analizar varios aspectos de la aplicación web, y se evidenció varios problemas relacionados con el manejo de datos sensibles del usuario. En base a esto nos damos cuenta de los puntos de mejora y nos ayudan a prevenir las vulnerabilidades para aumentar la seguridad del sistema.

c. Pruebas de integración

Las pruebas de integración permiten verificar que los distintos componentes de un sistema interactúan correctamente entre sí. este tipo de pruebas resulta fundamental, ya que procesos como el registro de lecturas y la generación de comprobantes, y generación de multas involucran múltiples modelos, reglas de negocio y operaciones sobre la base de datos.

A continuación, se van a describir las pruebas de integración realizadas al módulo de **Gestión de Lecturas, Gestión de Multas**, implementado en el framework Laravel, con el fin de garantizar la confiabilidad y correcto funcionamiento del sistema.

Las pruebas de integración se aplicaron al controlador **LecturaController,** específicamente a los métodos:

create()

store()

ultimaLectura()

Estas pruebas cubren el flujo completo desde la recepción de una solicitud HTTP hasta la persistencia de datos y la generación de registros relacionados en la base de datos.

Herramientas Utilizadas

Lenguaje: PHP

Framework: Laravel

Herramienta de pruebas: PHPUnit (Laravel Testing)

Base de datos de pruebas: SQLite en memoria

Entorno: XAMPP

Configuración del Entorno de Pruebas

Para evitar afectar los datos reales del sistema, se utilizó un entorno de pruebas con base de datos en memoria.

Archivo .env.testing:

DB_CONNECTION=sqlite

DB_DATABASE=:memory:

Además, se utilizó el trait RefreshDatabase, el cual reinicia la base de datos antes de cada prueba, garantizando resultados confiables y repetibles.

Metodología de Pruebas

Las pruebas de integración fueron implementadas en la carpeta **tests/Feature,** simulando el comportamiento real del usuario mediante solicitudes HTTP (POST y GET). Cada prueba valida tanto la respuesta del sistema como los cambios realizados en la base de datos.

Se verificó la interacción entre los siguientes componentes:

- Controlador (LecturaController)
- Modelos (Medidor, Lectura, Tarifa)
- Base de datos
- Validaciones y reglas de negocio

Casos de Prueba de Integración

Tabla 73 Tabla de Casos de Prueba módulo de lecturas

ID	Caso de prueba	Descripción	Resultado esperado	Resultado obtenido
P-01	Registro de lectura válida	Registrar una lectura correcta y generar comprobante	Lectura y comprobante creadas correctamente	Correcto
P-02	Consulta última lectura	Obtener última lectura registrada	Se retorna valor correcto en JSON	Correcto

Implementación de las Pruebas de Integración

Las pruebas se implementaron en el archivo:

tests/Feature/LecturaIntegrationTest.php

Evidencia de Ejecución

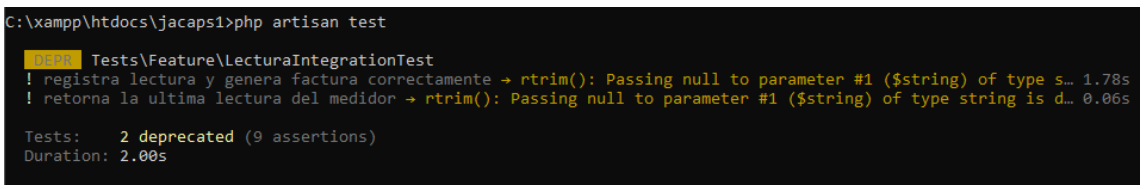


Figura 55 Ejecución de pruebas

Código de las pruebas

```php
<?php

namespace Tests\Feature;

use Tests\TestCase;
use Illuminate\Foundation\Testing\RefreshDatabase;
use App\Models\User;
use App\Models\Medidor;
use App\Models\Tarifa;
use App\Models\Lectura;
use App\Models\Factura;

class LecturaIntegrationTest extends TestCase
{


    public function
test_registra_lectura_y_genera_factura_correctamente()
    {
        $user = User::factory()->create([
            'role' => 'administrador',
        ]);

        $this->actingAs($user);

        // Medidor
        $medidor = Medidor::factory()->create([
            'user_id' => $user->id,
        ]);
        Tarifa::create([
            'nombre' => 'Residencial',
            'consumo_min' => 0,
            'consumo_max' => 15,
            'tarifa_base' => 2,
            'valor_excedente' => 0.15,
            'alcantarillado' => 1,
        ]);

        // Request
        $response = $this->post(route('admin.lecturas.store'), [
            'medidor_id' => $medidor->id,
            'lectura_actual' => 20,
        ]);

        $response->assertRedirect(route('admin.lecturas.create'));
```

```
        $this->assertDatabaseCount('lecturas', 1);
        $this->assertDatabaseCount('facturas', 1);

        $lectura = Lectura::first();

        $this->assertEquals($medidor->id, $lectura->medidor_id);
        $this->assertEquals(20, $lectura->lectura_actual);
        $this->assertEquals(20, $lectura->consumo); // primera lectura
    }

    public function test_retorna_la_ultima_lectura_del_medidor()
    {

        $admin = User::factory()->create([
            'role' => 'administrador',
        ]);

        $this->actingAs($admin);

        // Medidor
        $medidor = Medidor::factory()->create();

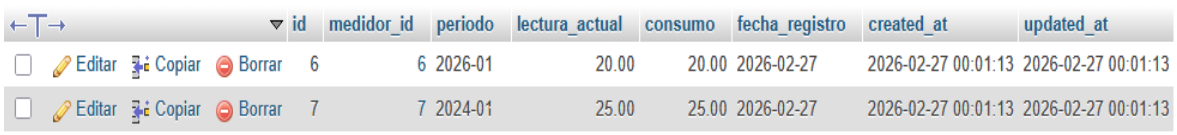
        // Lectura previa
        Lectura::create([
            'medidor_id' => $medidor->id,
            'periodo' => '2024-01',
            'lectura_actual' => 25,
            'consumo' => 25,
            'fecha_registro' => now(),
        ]);

        // Request
        $response = $this->getJson(
            route('admin.lecturas.ultima', $medidor)
        );

        // Assert
        $response
            ->assertStatus(200)
            ->assertJson([
                'lectura_anterior' => 25,
            ]);
    }

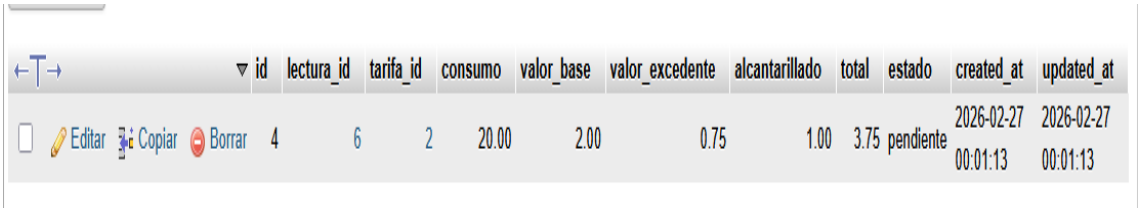
}
```

Base de datos



	id	medidor_id	periodo	lectura_actual	consumo	fecha_registro	created_at	updated_at
Editar Copiar Borrar	6	6	2026-01	20.00	20.00	2026-02-27	2026-02-27 00:01:13	2026-02-27 00:01:13
Editar Copiar Borrar	7	7	2024-01	25.00	25.00	2026-02-27	2026-02-27 00:01:13	2026-02-27 00:01:13

Figura 56 Tabla lecturas con el registro creado



	id	lectura_id	tarifa_id	consumo	valor_base	valor_excedente	alcantarillado	total	estado	created_at	updated_at
Editar Copiar Borrar	4	6	2	20.00	2.00	0.75	1.00	3.75	pendiente	2026-02-27 00:01:13	2026-02-27 00:01:13

Figura 57 Tabla comprobantes con el total calculado

La Figura muestra la ejecución exitosa de las pruebas de integración mediante el comando **php artisan test**, evidenciando el correcto funcionamiento del módulo de lecturas y comprobante

Resultados Obtenidos

Los resultados obtenidos demuestran que el sistema:

- Registra correctamente las lecturas de los medidores.
- Aplica correctamente las reglas de negocio definidas.
- Genera automáticamente los comprobantes correspondientes.
- Evita inconsistencias en los datos mediante validaciones y transacciones.

Pruebas de integración módulo multas

Se implementaron pruebas de integración en el módulo de gestión de multas con el objetivo de validar el correcto funcionamiento de los procesos de registro, pago y anulación de multas. Estas pruebas verifican la interacción entre controladores, modelos, autenticación y base de datos.

Tabla 74 Casos de Prueba módulo multas

Caso de prueba	Descripción	Resultado esperado	Resultado obtenido
CP-M01 – Listado de multas	Verificar que el sistema muestre correctamente el listado de multas registradas	El sistema carga la vista con el listado de multas ordenadas por fecha	Correcto
CP-M02 – Registro de multa	Verificar el registro de una nueva multa con datos válidos	La multa se registra en la base de datos con estado “pendiente”	Correcto
CP-M03 – Anulación de multa	Verificar la anulación de una multa indicando un motivo válido	El estado de la multa cambia a “anulada” y se registra el motivo	Correcto

Implementación de las Pruebas de Integración

Las pruebas se implementaron en el archivo:

tests/Feature/MultaIntegrationTest.php

Evidencia de Ejecución

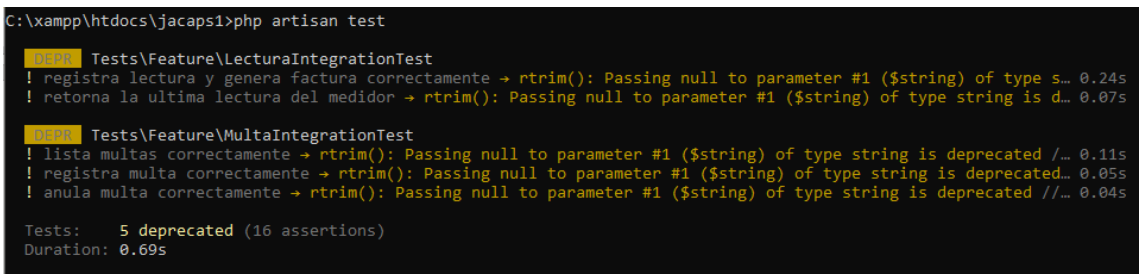


Figura 58 Ejecución de pruebas módulo multas

Código de pruebas

```php
<?php

namespace Tests\Feature;

use Tests\TestCase;
use Illuminate\Foundation\Testing\RefreshDatabase;
use App\Models\User;
use App\Models\Multa;
use App\Models\MotivoMulta;

class MultaIntegrationTest extends TestCase
{


    public function test_lista_multas_correctamente()
{
    $admin = User::factory()->create([
        'role' => 'administrador',
    ]);

    $this->actingAs($admin);
    Multa::factory()->count(3)->create();

    $response = $this->get(route('admin.multas.index'));

    $response->assertStatus(200);
    $response->assertViewIs('admin.multas.index');
}
public function busca_socios_activos_por_ajax()
{
    User::factory()->create([
        'name' => 'Juan Perez',
        'role' => 'socio',
        'status' => 'activo',
    ]);

    $response = $this->getJson(
        route('admin.multas.buscarSocio', ['q' => 'Juan'])
    );

    $response
        ->assertStatus(200)
        ->assertJsonFragment([
            'name' => 'Juan Perez',
        ]);
}
```

```php
public function test_registra_multa_correctamente()
{
    $admin = User::factory()->create([
        'role' => 'administrador',
        'status' => 'activo',
    ]);

    $this->actingAs($admin);
    $socio = User::factory()->create(['role' => 'socio',
    'status' => 'activo'
]);

    $motivo = MotivoMulta::create([
        'id_motivo' => 1,
        'descripcion' => 'Incumplimiento',
    ]);

    $response = $this->postJson(route('admin.multas.store'), [
        'id_socio' => $socio->id,
        'id_motivo' => $motivo->id_motivo,
        'monto' => 10.50,
        'fecha_multa' => now()->toDateString(),
    ]);

    $response->assertJson(['success' => true]);

    $this->assertDatabaseHas('multas', [
        'id_socio' => $socio->id,
        'estado' => 'pendiente',
    ]);
}

public function test_anula_multa_correctamente()
{

    $admin = User::factory()->create([
        'role' => 'administrador',
        'status' => 'activo',
    ]);

    $this->actingAs($admin);

    // Multa pendiente
    $multa = Multa::factory()->create([
        'estado' => 'pendiente',
    ]);


    $response = $this->patchJson(
```

```
        route('admin.multas.anular', $multa),
        ['motivo_anulacion' => 'Error administrativo']
    );

    // Assert
    $response->assertStatus(200)
             ->assertJson(['success' => true]);

    $this->assertDatabaseHas('multas', [
        'id_multa' => $multa->id_multa,
        'estado' => 'anulada',
        'anulada_por' => $admin->id,
    ]);
}
}
```

	id_multa	id_socio	id_motivo	monto	fecha_multa	estado	motivo_otro	motivo_anulacion	anulada_por	fecha_anulacion
Editar Copiar Borrar	52	122	54	10.50	2026-02-27	pendiente	NULL	NULL	NULL	NULL
Editar Copiar Borrar	54	129	56	161.53	2026-02-27	pagada	NULL	NULL	NULL	NULL
Editar Copiar Borrar	55	130	57	55.90	2026-02-27	pagada	NULL	NULL	NULL	NULL
Editar Copiar Borrar	56	131	58	39.30	2026-02-27	pagada	NULL	NULL	NULL	NULL
Editar Copiar Borrar	57	133	59	10.50	2026-02-27	pendiente	NULL	NULL	NULL	NULL
Editar Copiar Borrar	58	135	60	63.16	2026-02-27	anulada	NULL	Error administrativo	134	2026-02-27 01:16:
Editar Copiar Borrar	59	140	61	90.68	2026-02-27	pagada	NULL	NULL	NULL	NULL
Editar Copiar Borrar	60	141	62	92.88	2026-02-27	pagada	NULL	NULL	NULL	NULL
Editar Copiar Borrar	61	142	63	106.54	2026-02-27	pagada	NULL	NULL	NULL	NULL
Editar Copiar Borrar	62	144	64	10.50	2026-02-27	pendiente	NULL	NULL	NULL	NULL
Editar Copiar Borrar	63	146	65	175.85	2026-02-27	anulada	NULL	Error administrativo	145	2026-02-27 01:16:

Figura 59 Resultado de Base de datos

Los resultados obtenidos confirman que el módulo de multas cumple correctamente con las reglas de negocio establecidas, garantizando la integridad de la información y la correcta gestión de estados.

Pruebas de integración módulo reportes

Se implementaron pruebas de integración en el módulo de reportes para validar que los procesos de generación, visualización y descarga de reportes funcionen correctamente. Cada prueba valida el flujo completo desde la recepción del request, pasando por las validaciones y la invocación de los servicios correspondientes, hasta la generación de la vista o archivo PDF.

Tabla 75 Casos de Prueba módulo reportes

Caso de prueba	Descripción	Resultado esperado	Resultado obtenido
CP-R01	Mostrar índice de reportes	La vista de selección de reportes se carga correctamente	Correcto
CP-R02	Generar reporte válido	Se muestra la vista con datos del reporte	Correcto
CP-R03	Validación de request	Retorna errores si faltan campos obligatorios	Correcto
CP-R04	Descargar PDF	Se genera un PDF con nombre y contenido correcto	Correcto

Implementación de las Pruebas de Integración

Las pruebas se implementaron en el archivo:

tests/Feature/ReportIntegrationTest.php

Evidencia de Ejecución

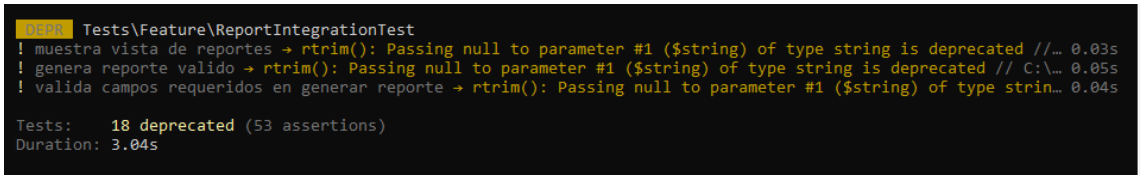


Figura 60 Ejecución de pruebas

Código de Pruebas

```php
<?php

namespace Tests\Feature;

use Tests\TestCase;
use Illuminate\Foundation\Testing\RefreshDatabase;
use App\Services\Reports\PagosPorMetodoReport;
use App\Services\Reports\MultasReport;

class ReportIntegrationTest extends TestCase
{
    use RefreshDatabase;
    public function test_muestra_vista_de_reportes()
    {
        $admin = \App\Models\User::factory()->create([
            'role' => 'administrador', //
            'status' => 'activo',
        ]);

        $response = $this->actingAs($admin)
            ->get(route('admin.reports.index'));

        $response->assertStatus(200);
        $response->assertViewIs('admin.reports.index');
    }
    public function test_genera_reporte_valido()
    {
        $admin = \App\Models\User::factory()->create([
            'role' => 'administrador',
            'status' => 'activo',
        ]);

        $requestData = [
            'report_type' => 'multas',
            'month_year'  => '2024-02',
        ];

        $response = $this->actingAs($admin)
            ->post(route('admin.reports.generate'), $requestData);

        $response->assertStatus(200);
        $response->assertViewIs('admin.reports.index');
        $response->assertViewHas('reportData');
        $response->assertViewHas('reportType', 'multas');
    }
    public function test_valida_campos_requeridos_en_generar_reporte()
```

```
    {
        $admin = \App\Models\User::factory()->create([
            'role' => 'administrador',
            'status' => 'activo',
        ]);

        $response = $this->actingAs($admin)
            ->post(route('admin.reports.generate'), []);

        $response->assertStatus(302); // redirect por validación
        $response->assertSessionHasErrors(['report_type', 'month_year']);
    }
public function descarga_pdf_reporte()
{
    $requestData = [
        'report_type' => 'multas',
        'month_year'  => '2024-02',
    ];

    $response = $this->get(route('admin.reports.download',
$requestData));

    $response->assertStatus(200);
    $response->assertHeader('content-type', 'application/pdf');
}

}
```

Resultados pruebas modulo reportes:

El módulo genera reportes válidos y descarga los pdf de reportes

Pruebas de integración módulo de Pagos

Las pruebas de integración del módulo de pagos permiten verificar que la búsqueda de socios, la obtención de comprobantes pendientes, el registro de pagos y la generación de comprobantes PDF funcionan correctamente, interactuando con la base de datos y controladores de forma integrada.

Tabla 76 Casos de prueba de integración

Caso de prueba	Descripción	Resultado esperado	Resultado obtenido
CP-P01	Vista principal	La vista de pagos se carga correctamente	Correcto
CP-P02	Búsqueda de socios	Retorna JSON con los socios/medidores que coinciden	Correcto
CP-P03	Facturas pendientes	Retorna JSON con comprobantes pendientes	Correcto
CP-P04	Pago exitoso	Comprobante cambia a pagada, se registra pago y retorna URL de comprobante	Correcto
CP-P05	Pago ya realizado	Retorna error y no cambia estado de factura	Correcto
CP-P06	Validación de campos	Retorna errores de validación si faltan campos	Correcto
CP-P07	Comprobante PDF	PDF generado en storage y redirige a URL pública	Correcto

Implementación de las Pruebas de Integración

Las pruebas se implementaron en el archivo:

tests/Feature/PagoIntegrationTest.php

Evidencia de Ejecución

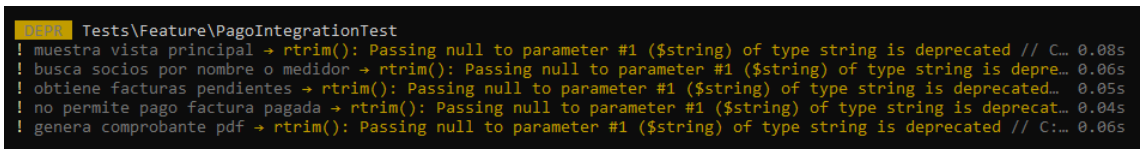


Figura 61 Ejecución de pruebas

Código de Prueba

```php
<?php

namespace Tests\Feature;

use Tests\TestCase;
use Illuminate\Foundation\Testing\RefreshDatabase;
use App\Models\User;
use App\Models\Medidor;
use App\Models\Lectura;
use App\Models\Factura;
use App\Models\Pago;

class PagoIntegrationTest extends TestCase
{
    use RefreshDatabase;
    public function test_muestra_vista_principal()
    {
        $admin = User::factory()->create([
            'role' => 'administrador',
            'status' => 'activo',
        ]);

        $response = $this->actingAs($admin)
            ->get(route('admin.pagos.index'));

        $response->assertStatus(200);
        $response->assertViewIs('admin.pagos.index');
    }
    public function test_busca_socios_por_nombre_o_medidor()
    {
        $secretario = User::factory()->create([
            'role' => 'secretario',
            'status' => 'activo',
        ]);

        $user = User::factory()->create([
            'name' => 'Juan Perez'
        ]);

        $medidor = Medidor::factory()->create([
            'user_id' => $user->id,
            'numero_medidor' => '12345'
        ]);

        $response = $this->actingAs($secretario)
```

```
            ->getJson(route('admin.pagos.search-socios', ['query' =>
'Juan']));

        $response->assertStatus(200)
                 ->assertJsonFragment([
                     'numero_medidor' => '12345',
                     'socio' => 'Juan Perez'
                 ]);
    }
    public function test_obtiene_facturas_pendientes()
    {
        $tesorero = User::factory()->create([
            'role' => 'tesorero',
            'status' => 'activo',
        ]);

        $user = User::factory()->create();
        $medidor = Medidor::factory()->create(['user_id' => $user->id]);
        $lectura = Lectura::factory()->create([
            'medidor_id' => $medidor->id,
            'consumo' => 10
        ]);

        $factura = Factura::factory()->create([
            'lectura_id' => $lectura->id,
            'estado' => 'pendiente'
        ]);

        $response = $this->actingAs($tesorero)
            ->getJson(route('admin.pagos.facturas-pendientes', $medidor-
>id));

        $response->assertStatus(200)
                 ->assertJsonFragment([
                     'id' => $factura->id,
                     'nombre_socio' => $user->name,
                 ]);
    }
    public function test_no_permite_pago_factura_pagada()
    {

        $tesorero = User::factory()->create([
            'role' => 'tesorero',
            'status' => 'activo',
        ]);

        $factura = Factura::factory()->create([
            'estado' => 'pagado',
        ]);
```

```
        $response = $this->actingAs($tesorero)
            ->postJson(route('admin.pagos.process-payment'), [
                'factura_id' => $factura->id,
                'metodo_pago' => 'efectivo',
            ]);

        $response->assertStatus(400)
                 ->assertJsonFragment([
                     'success' => false,
                     'message' => 'Esta factura ya fue pagada',
                 ]);
    }
public function test_genera_comprobante_pdf()
{
    $user = User::factory()->create();
    $medidor = Medidor::factory()->create(['user_id' => $user->id]);
    $lectura = Lectura::factory()->create(['medidor_id' => $medidor-
>id]);
    $factura = Factura::factory()->create(['lectura_id' => $lectura->id,
'estado' => 'pagado']);
    $pago = Pago::factory()->create(['factura_id' => $factura->id]);

    $response = $this->get(route('admin.pagos.comprobante', $pago->id));
    $response->assertStatus(302); // Redirección a URL del PDF
}
}
```

←T→	id	lectura_id	tarifa_id	consumo	valor_base	valor_excedente	alcantarillado	total	estado	created_at	updated_at
Editar Copiar Borrar	1	1	1	20.00	2.00	0.75	1.00	3.75	pendiente	2026-02-27 09:55:56	2026-02-27 09:55:56

Figura 62 Base de datos

CONCLUSIONES

- La recolección de información fue una etapa clave para conocer los problemas reales que la junta administradora comunitaria de agua potable y saneamiento la buena esperanza, enfrenta con la gestión de cobros, gracias a esto se pudo identificar y establecer los requerimientos necesarios para que la aplicación cumpla con las funcionalidades necesarias que resulten útiles para resolver los procesos actuales.
- El diseño de la arquitectura de la aplicación web junto con la base de datos es la parte más importante ya que debe ser coherente con los requerimientos, caso contrario tendremos datos mal almacenados, y el sistema será propenso a muchos errores.
- Las pruebas realizadas fueron importantes para comprobar que el sistema está cumpliendo con los objetivos definidos, está agilizando los procesos que conforman la gestión de información de la junta de agua y que cumple con la seguridad y confiablidad requerida.

RECOMENDACIONES

- En el caso de los pagos es recomendable hacer auditorías de seguridad y seguir aplicando nuevos métodos de seguridad, esto para garantizar que la información financiera de los socios este completamente segura ya que se manejan datos sensibles.
- Aunque la aplicación es intuitiva es recomendable estar en constante capacitación ya que algunos usuarios talvez no están muy familiarizados con las tecnologías digítales, esto permitirá generar más confianza en el uso de la aplicación y con eso aprovechar todas las funcionalidades.
- Se debe realizar mantenimiento y actualizaciones periódicas para asegurar que el sistema continúe funcionando correctamente y se adapte a cambios futuros ya que a medida que la junta vaya creciendo pueden surgir nuevas necesidades.

BIBLIOGRAFÍA

Salazar, H. (2022). *La facturación electrónica como un mecanismo de control para el cumplimiento tributario en las empresas constructoras del cantón Ambato* [Tesis de Ingeniería, UNIVERSIDAD TÉCNICA DE AMBATO]. https://doi.org/https://repositorio.uta.edu.ec/server/api/core/bitstreams/d6e1fcdb-3ff8-4437-ac35-ec9bb1ca2277/content

Boada, D. (2025, enero 16). *¿Qué es un servidor web y cómo funciona? - Guía completa*. https://www.hostinger.com/es/tutoriales/que-es-un-servidor-web

Borovskoy, D. (2023, octubre 10). *Desarrollo de Sistema Web - Perú | LinkedIn*. https://www.linkedin.com/pulse/desarrollo-de-sistema-web-per%C3%BA-denis-borovskoy/

Cimas Cuadrado, G. (2020, julio 16). *Qué es un lenguaje de programación: Guía para principiantes | OpenWebinars*. https://openwebinars.net/blog/que-es-un-lenguaje-de-programacion/

Deyimar A. (2023, enero 11). *¿Qué es Bootstrap? - Todo lo que necesitas saber*. https://www.hostinger.com/es/tutoriales/que-es-bootstrap#Funciones_basicas_de_Bootstrap

Ferreira, A. C. (2023, julio 11). *▷ Qué es el CSS, para qué sirve y mucho más | InboundCycle*. https://www.inboundcycle.com/blog-de-inbound-marketing/que-es-el-css-para-que-sirve-y-mucho-mas

Flores Chango, E. L. (2024). *Aplicación web progresiva para la gestión de cobranza del servicio del agua de regadío para la junta modular de riego y drenaje "Quillán*

Loma". [Tesis de Ingeniería, Universidad Técnica de Ambato]. https://repositorio.uta.edu.ec/handle/123456789/40812

Folgueiras Bertomeu, P. (2016). *La entrevista*.

García de Zuñiga, F. (2025, febrero 3). *¿Qué es un framework y para qué sirve? | Blog de Arsys*. https://www.arsys.es/blog/que-es-un-framework-en-programacion-y-para-que-sirve

Gillis, A. S. (2024, octubre 22). *¿Qué son las pruebas de aceptación? | Definición de TechTarget*. https://www-techtarget-com.translate.goog/searchsoftwarequality/definition/acceptance-test?_x_tr_sl=en&_x_tr_tl=es&_x_tr_hl=es&_x_tr_pto=tc

Guamán Arenillas, K. R. (2023). *Propuesta y desarrollo de un prototipo para el registro de pagos en la Junta de Agua Monseñor «Leónidas Proaño» de la Comunidad de Guartanchún utilizando almacenamiento en la nube* [Tesis de Ingeniería, Universidad Católica de Cuenca]. https://dspace.ucacue.edu.ec/handle/ucacue/14307

LEY DE COMERCIO ELECTRONICO, FIRMAS Y MENSAJES DE DATOS. (2002). www.lexis.com.ec

López, M. (2021, octubre 7). *Base de datos ¿Qué es y para qué sirve?* https://immune.institute/blog/base-de-datos-que-es-usos/

Mancuzo, G. (2020, agosto 8). *Metodología XP: La Mejor Vía para el Desarrollo de Software*. https://blog.comparasoftware.com/metodologia-xp/

Mendoza López, M. (2020, septiembre 18). *Extreme Programming: Qué es y cómo aplicarlo | OpenWebinars*. https://openwebinars.net/blog/extreme-programming-que-es-y-como-aplicarlo/

Ortega, K. (2024, mayo 29). *¿Qué es el lenguaje de programación PHP?* https://worldcampus.saintleo.edu/blog/sistemas-computacionales-que-es-el-lenguaje-de-programacion-php

Perandones, S. M. (2024, julio 31). *Aprende todo lo que necesitas saber sobre HTML*. https://www.cursosfemxa.es/blog/estudiar-html

Ridge, B. V. (2023, agosto 20). *La Importancia de una Aplicación Web en la Era Digital*. https://www.mediummultimedia.com/web/cual-es-la-importancia-de-una-aplicacion-web/

Semblantes Quinche, W. D. (2021). *Desarrollo de un sistema web para el proceso de facturación, toma de lectura y consulta de planillas, en la Junta Administradora de agua potable Mollepamba de la parroquia Picaihua* [Tesis de Ingeniería, UNIVERSIDAD TÉCNICA DE AMBATO]. https://repositorio.uta.edu.ec/handle/123456789/33671

Sotomayor, S. G. (2024). Metodologías ágiles: ¿Qué son y cuáles son más utilizadas? *Thinking for Innovation*. https://www.iebschool.com/blog/que-son-metodologias-agiles-agile-scrum/

Yelenevych, A. (2023, julio 21). *Metodologías de desarrollo de software*. https://codegym.cc/es/groups/posts/es.185.metodologias-de-desarrollo-de-software

Zelaya Reyes, C. J. (2020, diciembre 2). *Nuevas Tendencias en Desarrollo Web*.

ANEXOS

Anexo 1. Entrevista para la presidenta de la junta de agua

ENTREVISTA PARA EL SEÑOR PRESIDENTE DE LA JUNTA ADMINISTRADORA COMUNITARIA DE AGUA POTABLE Y SANEAMIENTO LA BUENA ESPERANZA.		
Cargo del Entrevistado: Presidenta de la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza.		
Preguntas	**Respuestas**	**Observación**
¿Cuáles son las funciones que usted desempeña en la Junta de Agua?	Ser responsable de la gestión y administración de la Junta de Agua.	Encargado de la Junta de Agua
¿Cuántas personas conforman la Directiva de la junta de agua?	5 personas	Presidente, Secretario, Tesorero, Primer Vocal, Segundo Vocal

¿Cuántos socios pertenecen a la Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza?	205	Usuarios que se benefician del servicio de agua potable
¿Cómo se maneja la información de los usuarios de la Junta de Agua?	La información se opera manualmente en hojas excel	
¿Cómo se lleva a cabo los cobros del servicio de Agua Potable a los usuarios?	Los usuarios deben acercarse a la junta, hacer fila y al llegar a la ventanilla debe proporcionar su nombre y posteriormente cancelar el valor establecido	
¿Quién se encarga de realizar los cobros de agua?	La tesorera de la Junta	La tesorera cumple el rol de recaudadora.

¿Qué sucede con los usuarios que no cumplen con el pago del servicio?	Se les cobra una tarifa adicional por reconexión, si excede los 6 meses de mora, se procede a quitar la acometida	Al tercer mes que no haya pagado se le cobra una tarifa de 8 dólares.
¿La junta de agua cuenta con algún sistema informático?	No	
¿Cómo es el proceso de toma de lecturas de los medidores?	La persona encargada va por cada domicilio revisando los medidores y anotando en un cuaderno el registro que marque el medidor.	
¿Cuál es la tarifa básica del consumo de agua?	2 dólares	
¿El valor a pagar siempre es el mismo, o depende del consumo que se haya hecho en el medidor?	Dependerá del consumo, ya que la tarifa básica de 2 dólares solo aplicará si el consumo del agua no pasa de los 15 metros cúbicos.	Pasado de los 15 metros cúbicos de consumo el valor irá incrementando

¿Qué piensa usted de la implementación de una aplicación web que facilite los cobros del servicio de agua potable?	Que es primordial ya que los usuarios podrán agilitar los pagos sin necesidad de hacer largas filas y esperar mucho tiempo.	Está de acuerdo que se lleve la gestión de pagos a través de una aplicación web.
Conclusión: La Junta de Agua está dispuesta a brindar toda la información necesaria para desarrollar la aplicación web, para agilizar los procesos de cobros.		

Anexo 2 Matriz de Observación

MATRIZ DE OBSERVACIÓN

Tipo de observación: Indirecta

Fecha:

Lugar: Junta Administradora Comunitaria de Agua Potable y Saneamiento la Buena Esperanza.

Observador: Marco Josue Quishpe Cuellar

Objetivo: Realizar observaciones dentro de la Junta de Agua, para tener una idea general de los procesos de pagos, toma y registro de lecturas.

Escala de valoración

5= Excelente, 4= Muy Bueno, 3= Bueno, 2=Regular, 1=Insuficiente

N°	Descripción	1	2	3	4	5	Observación

1	¿La toma de lecturas es adecuada?		X				Hay medidores que están desgastados y no se puede observar bien el consumo que marca.
2	¿El registro de las lecturas es adecuado?		X				Al hacerlo de manera manual y en hojas puede haber equivocaciones al momento de anotar.
3	Los usuarios tienen acceso a sus lecturas y consumos históricos	X					Únicamente cuando pagan el servicio pueden ver sus lecturas en la factura.
4	¿Hay agilidad en el proceso de cobros?		X				Existe un exceso de tiempo en los cobros, y al estar solo una persona encargada se generan largas filas.

Anexo 3: Manual Técnico

APLICACIÓN WEB PARA LA GESTIÓN DE PAGOS EN LA JUNTA ADMINISTRADORA DE AGUA POTABLE Y SANEAMIENTO LA BUENA ESPERANZA, PARROQUIA CANGAHUA, CANTÓN CAYAMBE.

AUTOR:

MARCO JOSUE QUISHPE CUELLAR

FECHA: 10/01/2026

Introducción

El siguiente manual técnico tiene como objetivo documentar todo el funcionamiento interno de “APLICACIÓN WEB PARA LA GESTIÓN DE PAGOS EN LA JUNTA ADMINISTRADORA DE AGUA POTABLE Y SANEAMIENTO LA BUENA ESPERANZA, PARROQUIA CANGAHUA, CANTÓN CAYAMBE” indicando todos los recursos necesarios para ejecutarla. Por lo tanto, este documento va dirigido para el personal de la junta de agua y así puedan entender la estructura que hay en la aplicación.

Objetivo

Facilitar una guía completa sobre la aplicación web que ayude al personal a comprender, manipular, y mantener el sistema para que vaya adaptándose a nuevos cambios futuros.

Creador

Participante	Marco Josue Quishpe Cuellar
Área:	Carrera de Software- UEB
Teléfono:	0992851530
Rol:	Desarrollador

Tecnología utilizada

Sistema Operativo:	Windows 11
Servidor Web	Apache
Framework:	Laravel 12
Gestor de Base de Datos:	MySQL

Extensiones de PHP	Versión 8.2.12
Gestor de Dependencias:	Composer versión 2.9.3
Control de versiones	Git Hub

Requerimientos Técnicos

Los requisitos mínimos para el funcionamiento de la aplicación se basan en dos categorías: Hardware y Software.

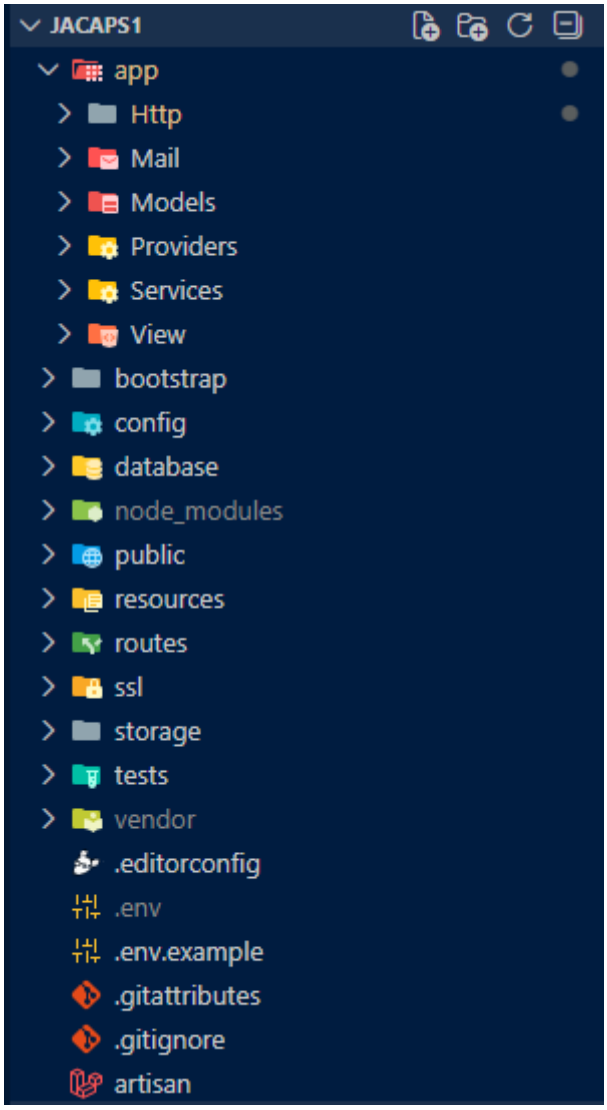
Hardware

- Laravel versión 10
- Laptop o PC con procesador Intel i7 de 11va generación o superior
- Composer
- PHP versión 8.2.12
- Bootstrap
- JS

Software

- Sistema operativo de Windows 10 en adelante
- Navegador web: Firefox, Edge, Brave, Chrome

Estructura de carpetas y archivos



Esta estructura maneja el framework laravel versión 12, para poder conectarse a la base de datos se debe acceder al archivo .env y colocar nombre, usuario, contraseña, número de puerto.

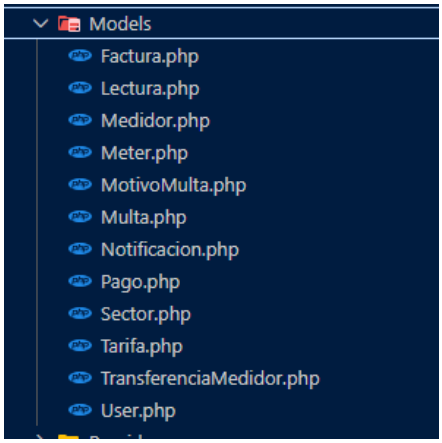
```
.env
1   APP_NAME=JACAPS-BUENA-ESPERANZA
2   APP_ENV=local
3   APP_KEY=base64:UgX7vRkH18Gf0kgt8ymZEwh7DPubDTpimKhK0d5eekY=
4   APP_DEBUG=true
5   APP_URL=http://localhost:8000
6
7   APP_LOCALE=en
8   APP_FALLBACK_LOCALE=en
9   APP_FAKER_LOCALE=en_US
10
11  APP_MAINTENANCE_DRIVER=file
12  # APP_MAINTENANCE_STORE=database
13
14  # PHP_CLI_SERVER_WORKERS=4
15
16  BCRYPT_ROUNDS=12
17
18  LOG_CHANNEL=stack
19  LOG_STACK=single
20  LOG_DEPRECATIONS_CHANNEL=null
21  LOG_LEVEL=debug
22
23  DB_CONNECTION=mysql
24  DB_HOST=localhost
25  DB_PORT=3306
26  DB_DATABASE=bejunta2026
27  DB_USERNAME=root
28  DB_PASSWORD=
29
30  SESSION_DRIVER=file
31  SESSION_LIFETIME=120
32  SESSION_ENCRYPT=false
```

Estructura del Sistema

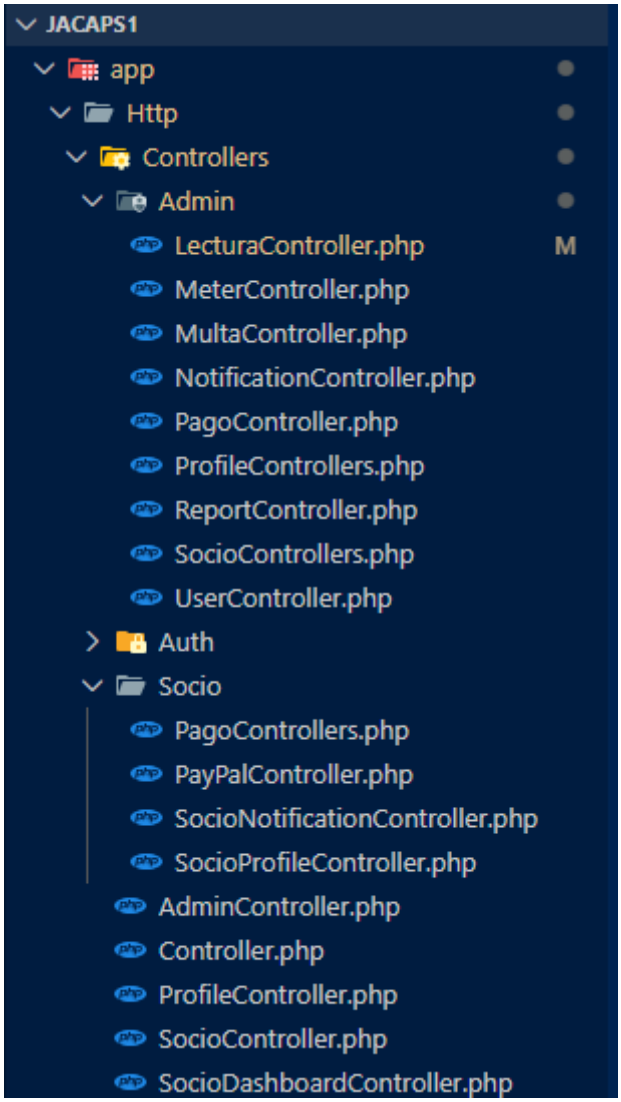
Ya que el framework laravel utiliza el Modelo Vista Controlador (MVC), ayuda a tener una organización limpia del código y así tener eficiencia al momento de desarrollar.

BackEnd

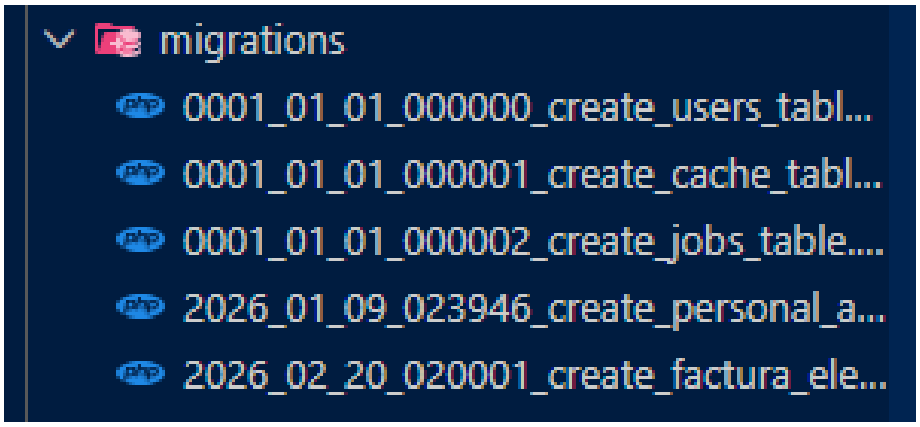
Modelo: En laravel los modelos son los archivos que interactúan con la base de datos.



Controladores: Son los archivos que manejan toda la lógica del sistema, ya que son los responsables de interactuar entre los modelos y las vistas. Estos controladores reciben las solicitudes del usuario, procesan y luego devuelven las respuestas a la vista y así se pueda mostrar al usuario.



Migraciones: Permiten a los desarrolladores establecer y modificar el esquema de la base de datos de manera eficiente.

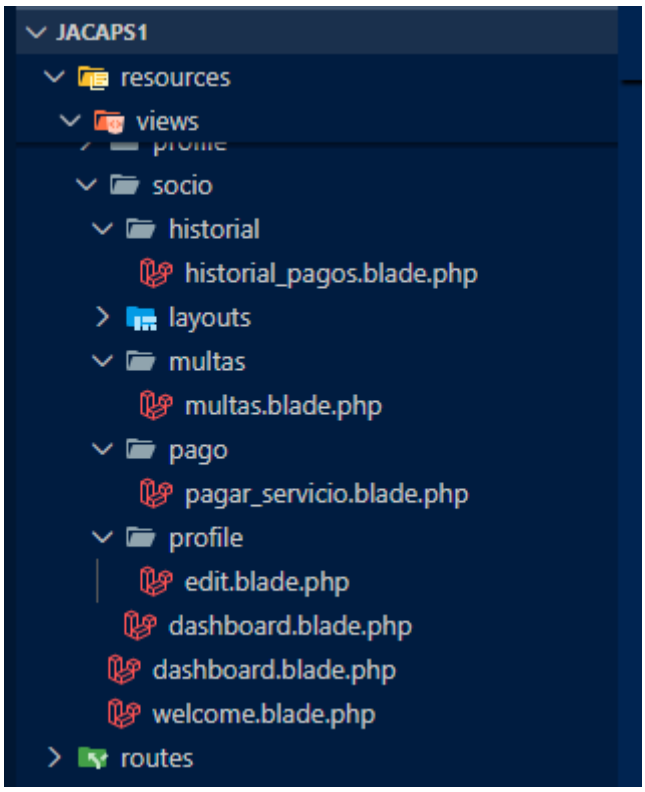


Rutas: Las rutas son las encargadas de dirigir y manejar las solicitudes HTTP que ingresan a la lógica de la aplicación. Es decir que realizan la conexión entre una URL y una acción específica dentro de un controlador.

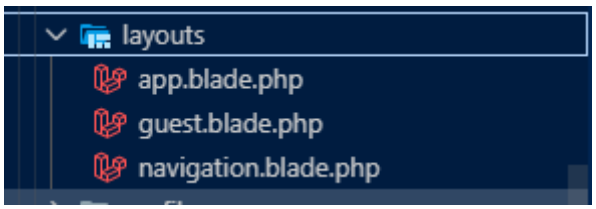


FrontEnd

Vista: Son las encargadas de la interfaz de usuario. Controlan lo que el usuario ve en el navegador, al trabajar con laravel todas las vistas tienen un formato “Blade.php”



Layout: es un vista que se usa como plantilla para otras vistas, es decir que define una estructura base que utilizan los demás archivos.



Copias de Seguridad de la Base de Datos

Es importante realizar copias de seguridad de la base de datos mensualmente ya que es ahí cuando más datos procesa y almacena.

Para eso la aplicación estará configurada para que automáticamente me genere un backup mensualmente.

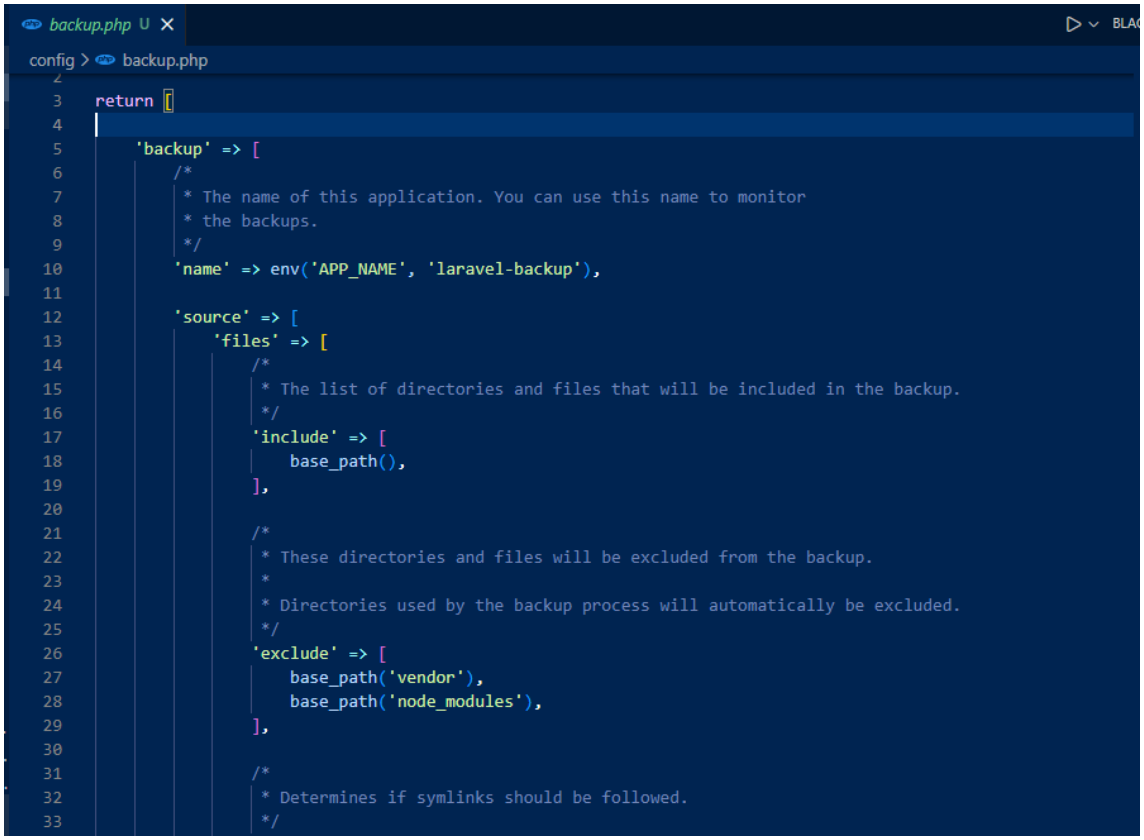
Primero se instala el paquete de backup

```bash
composer require spatie/laravel-backup
```

Publicar el archivo de configuración

```bash
php artisan vendor:publish --provider="Spatie\Backup\BackupServiceProvid
```

Esto va a crear un archivo llamado backup.php



Y aquí configuramos si queremos respaldar todo el proyecto, solo la base de datos o solo el storage, además configuramos donde se va a guardar el backup y adicionalmente podemos indicar el dia y hora en el que se desea hacer la copia de seguridad.

Anexo 4:Manual de Usuario

APLICACIÓN WEB PARA LA GESTIÓN DE PAGOS EN LA JUNTA ADMINISTRADORA DE AGUA POTABLE Y SANEAMIENTO LA BUENA ESPERANZA, PARROQUIA CANGAHUA, CANTÓN CAYAMBE.

AUTOR:

MARCO JOSUE QUISHPE CUELLAR

FECHA: 10/01/2026

Descripción del sistema

Esta aplicación web tiene como objetivo gestionar de manera eficiente todos los procesos relacionados con los pagos, es decir: creación de usuarios, creación de socios, creación de medidores, registro de lecturas, generación de reportes.

Objetivo

Facilitar el manejo del sistema web permitiendo al usuario aprender a manejarlo correcta y eficientemente.

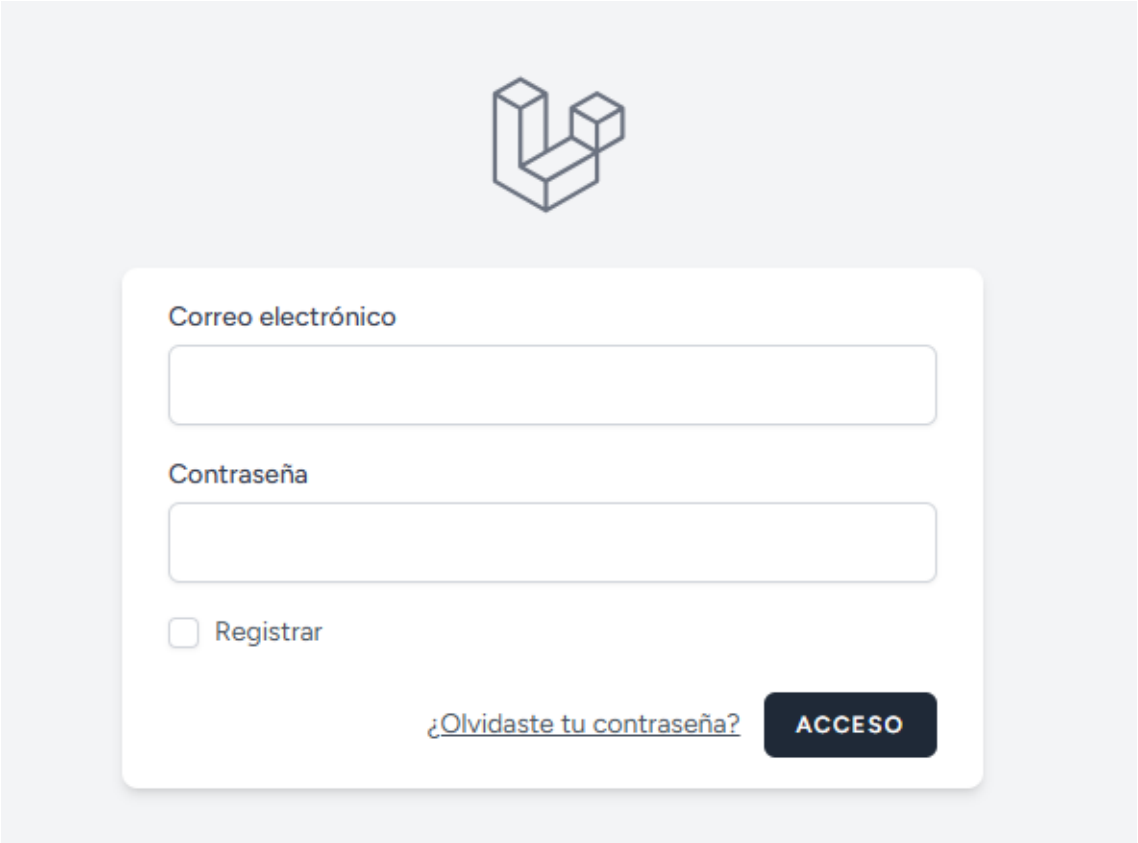
Acceso al Sistema:

Para acceder al sistema se debe seguir los siguientes pasos:

- Abrir el navegador web
- Ingresar la dirección URL en la barra de direcciones.
- En la página de bienvenida encontrará un botón de “Iniciar Sesión”, dar clic ahí.



Ingresar correctamente las credenciales enviadas al correo electrónico.



Dar clic en el botón de “Acceder”

Una vez iniciado sesión será dirigido a la página de inicio, esta vista es única para usuarios administrativos.

La vista principal tendrá las siguientes opciones en el menú:

Inicio: vista por defecto

Sección de usuarios: Visualización y gestión de usuarios

Sección de socios: Visualización y gestión de socios

Sección de Medidores: Visualizar y gestionar los medidores activos

Sección de Lecturas: Podrá registrar nuevas lecturas para cada período o mes.

Sección de notificaciones: Podrá enviar recordatorios, comunicados a los socios de la junta.

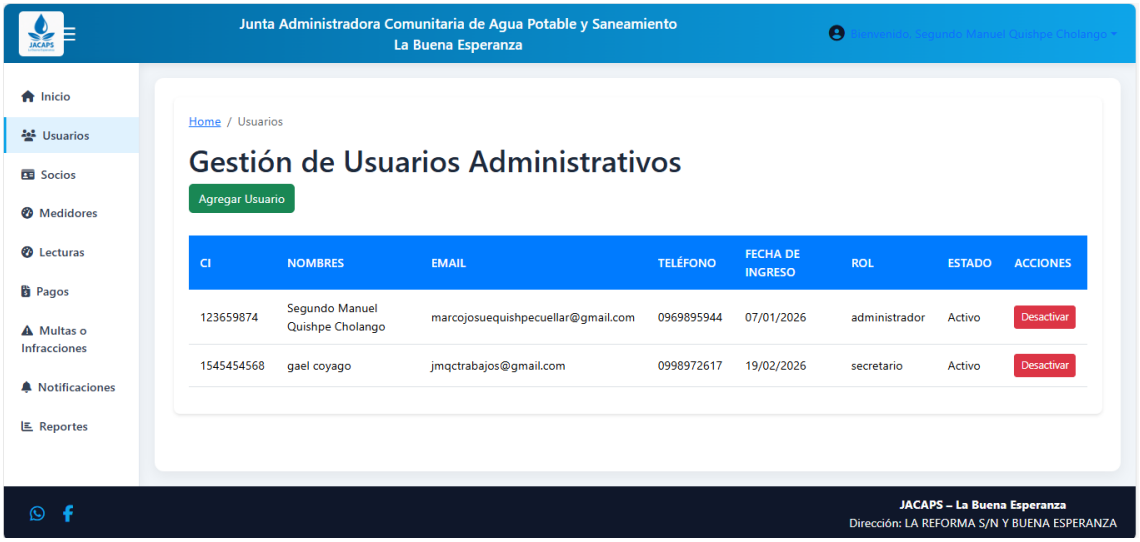
Sección de Reportes: Podrá generar reportes de todo tipo, y descargarlo.



Gestión de Usuarios

En esta sección el administrador podrá ver el listado de usuarios registrados, agregar, desactivar.

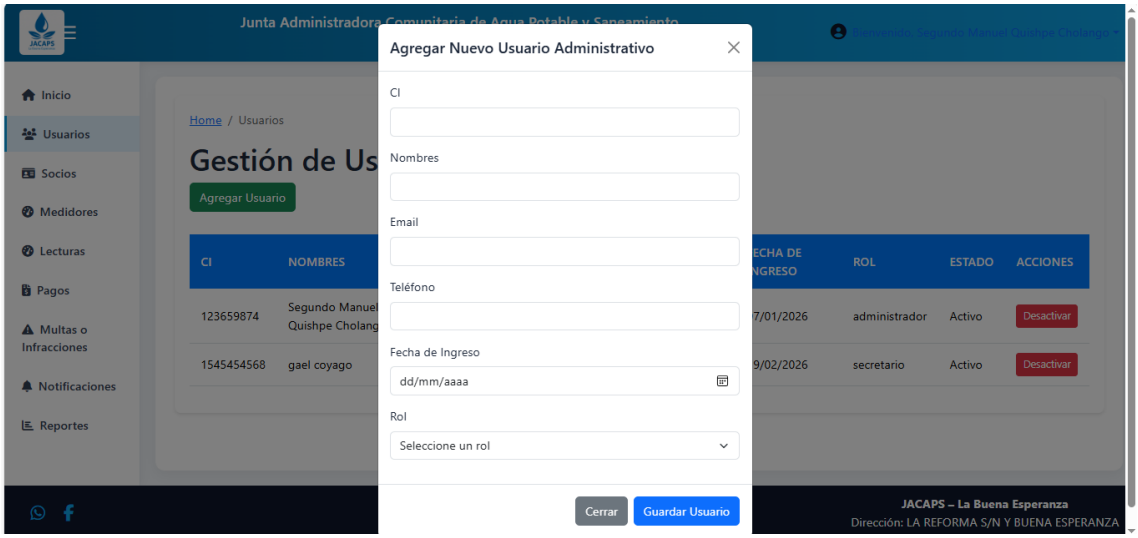
Para esto debe hacer clic en la opción “Usuarios” del menú.



Agregar usuario

Para poder agregar un usuario debe:

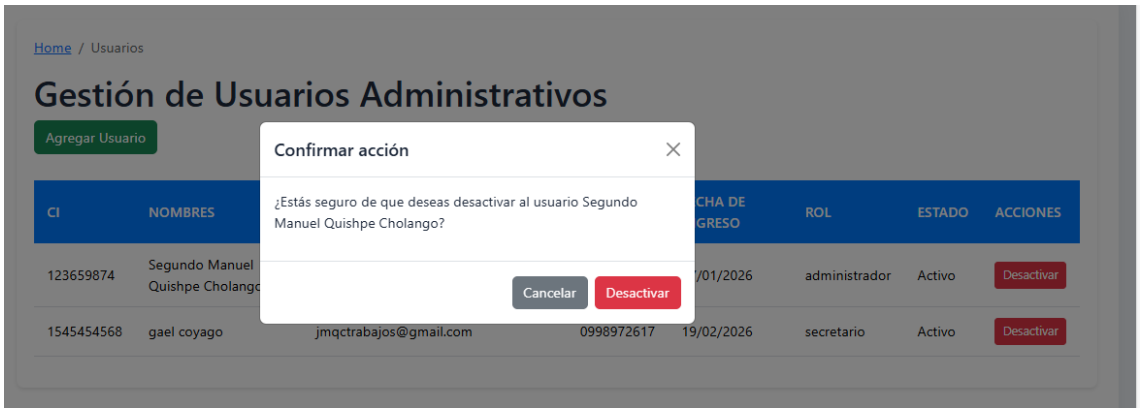
- Hacer clic en el botón de “agregar usuario”
- Se desplegará un modal, ahí llene todos los campos requeridos.
- Haga clic en “Guardar” y el nuevo usuario ya se mostrará en la tabla.



Desactivar Usuario

Si desea desactivar a un usuario debe:

- Dar clic en el botón “Desactivar”
- Se abrirá un modal de confirmación
- Dar clic en “Desactivar” y automáticamente su estado pasara a inactivo



Gestión de Socios

En esta sección el administrador podrá ver el listado de Socios registrados, agregar, desactivar.

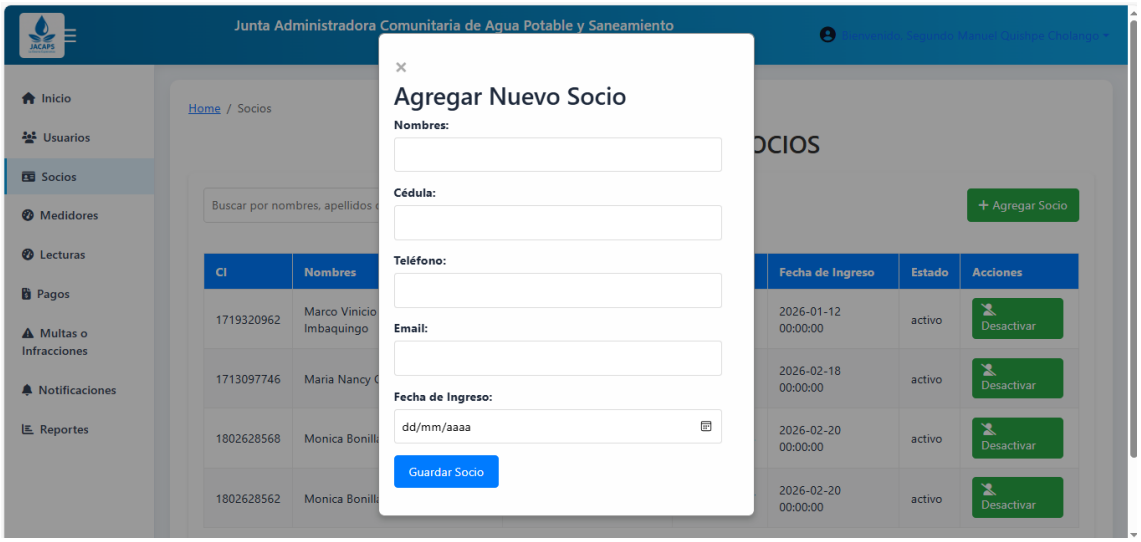
Para esto debe hacer clic en la opción “Socios” del menú.



Agregar Socio

Para poder agregar un socio debe:

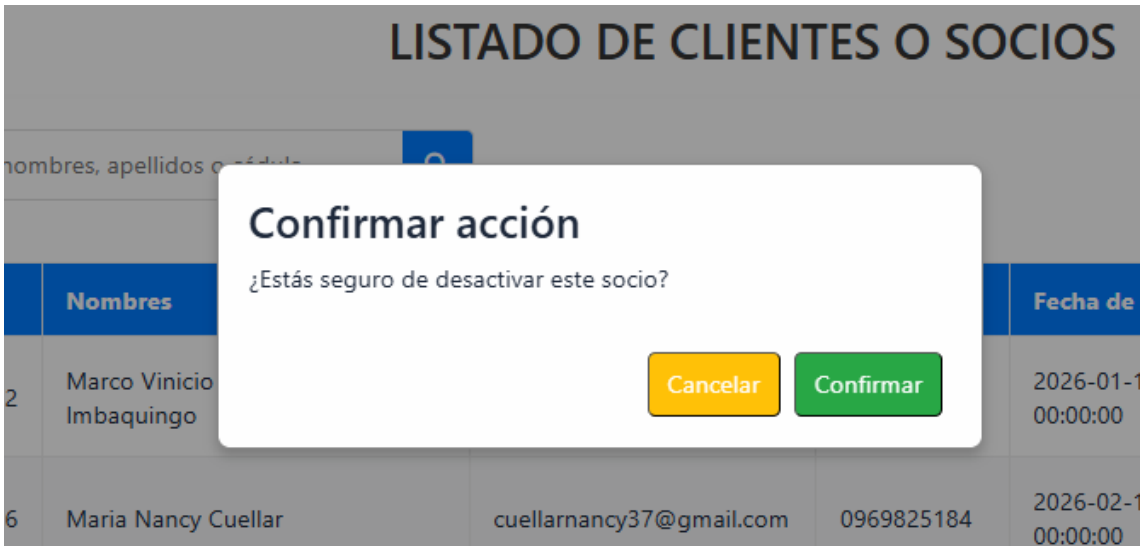
- Hacer clic en el botón de “agregar socio”
- Se desplegará un modal, ahí llene todos los campos requeridos.
- Haga clic en “Guardar” y el nuevo socio ya se mostrará en la tabla.



Desactivar Socio

Si desea desactivar a un socio debe:

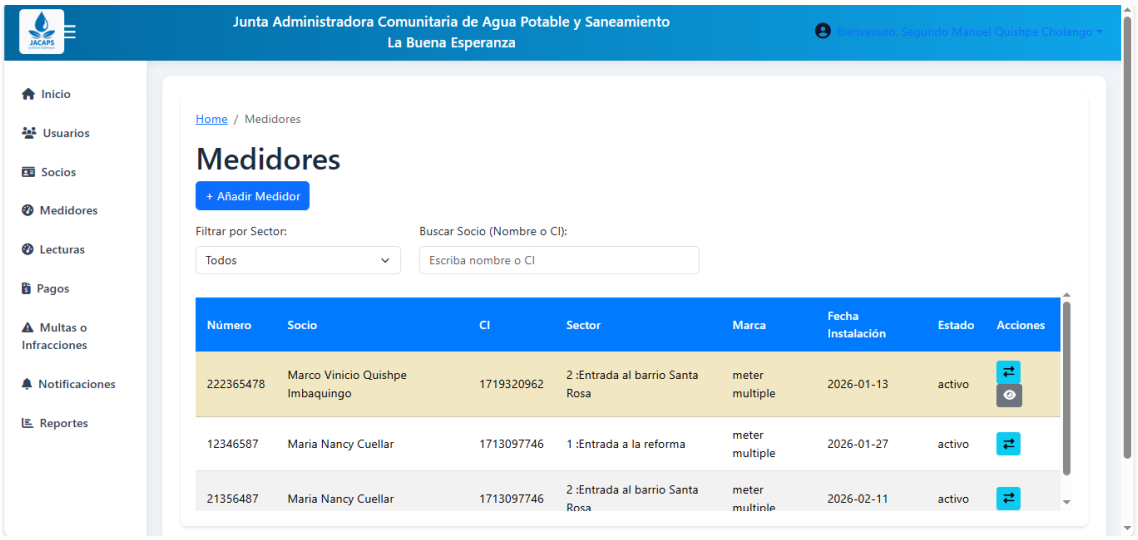
- Dar clic en el botón “Desactivar”
- Se abrirá un modal de confirmación
- Dar clic en “Confirmar” y automáticamente su estado pasara a inactivo



Gestión de Medidores

En esta sección el administrador podrá ver el listado de Medidores registrados, agregar, desactivar.

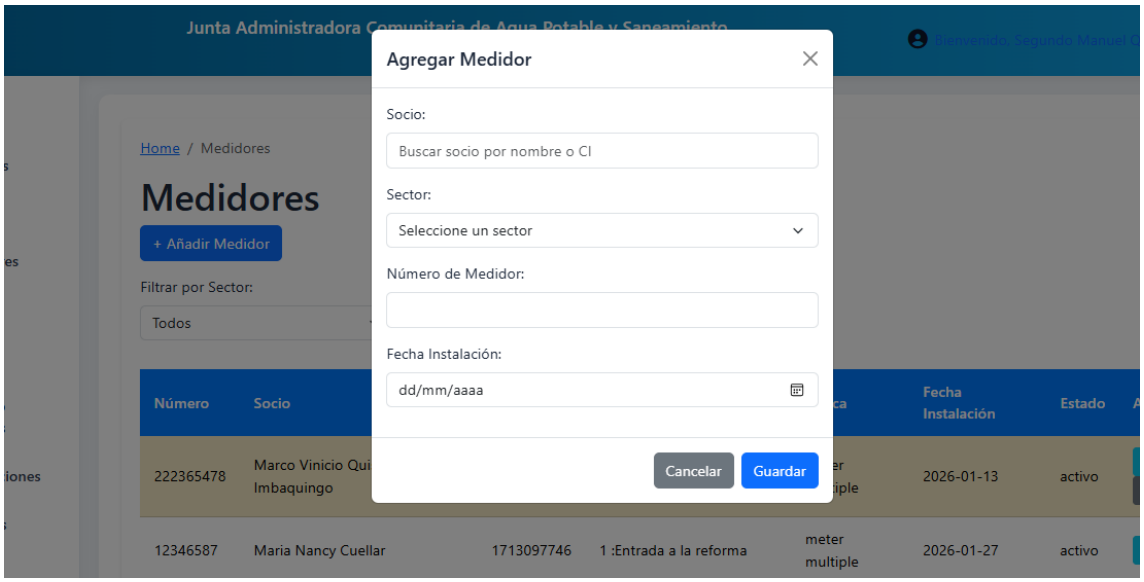
Para esto debe hacer clic en la opción “Medidores” del menú.



Agregar medidor

Para poder agregar un medidor debe:

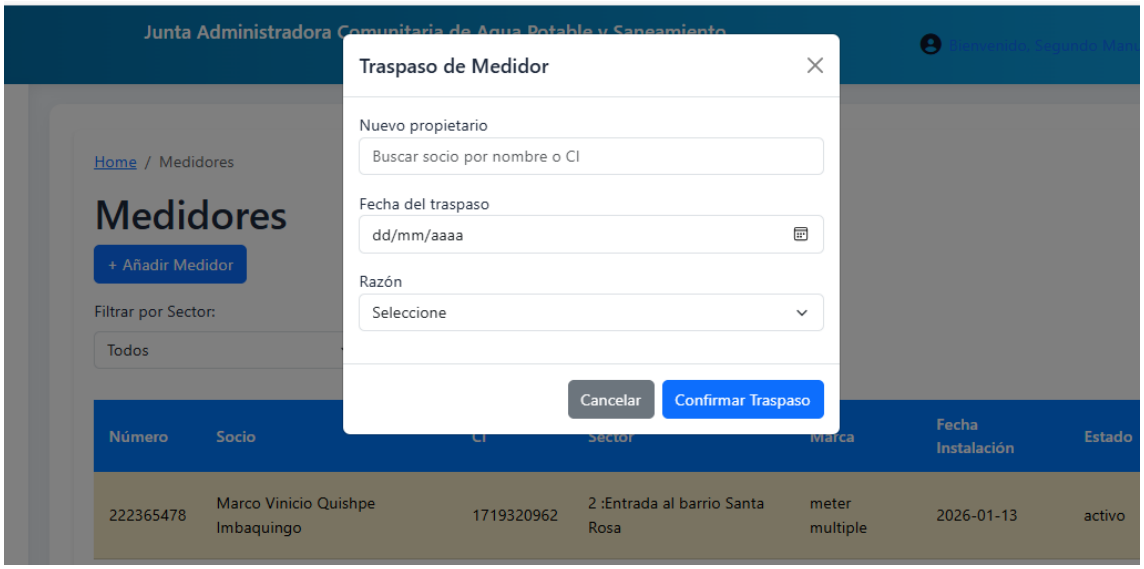
- Hacer clic en el botón de “agregar medidor”
- Se desplegará un modal, ahí llene todos los campos requeridos.
- Haga clic en “Guardar” y el nuevo medidor ya se mostrará en la tabla.



Hacer Traspaso de nombre de Medidor

Si desea hacer un traspaso de nombre de un medidor debe:

- Dar clic en el botón “Traspaso de Nombre” en la columna de acciones
- Se abrirá un modal para ingresar los datos de nuevo propietario, llene correctamente los datos.
- Dar clic en “Confirmar Traspaso” y automáticamente el nombre del propietario de guardará y tendrá la opción de visualizar el Historial de traspasos.

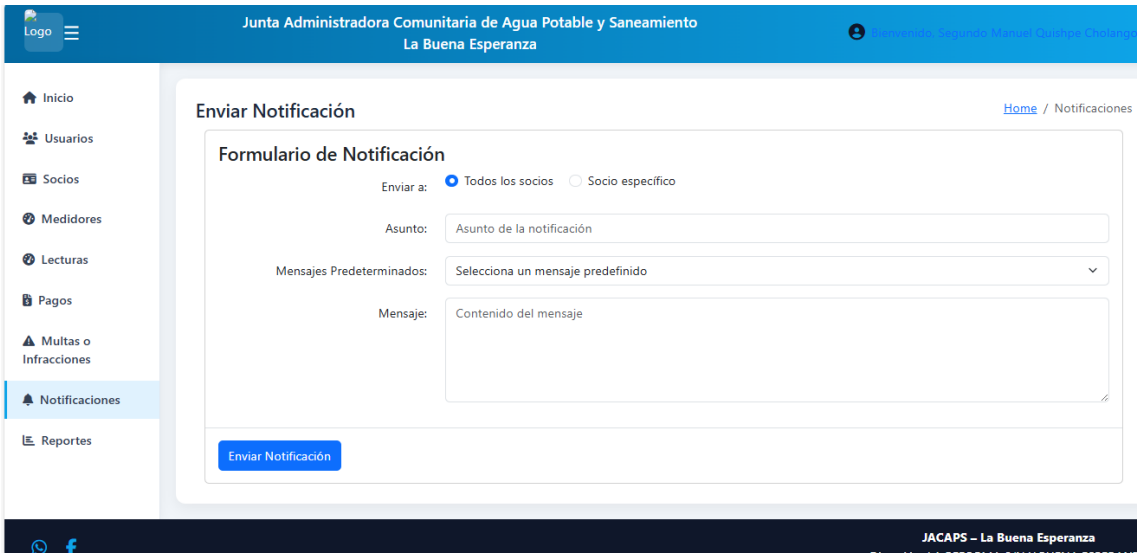




Notificaciones

En esta sección el administrador podrá enviar notificaciones a los socios de la junta.

Para esto debe hacer clic en la opción “Notificaciones” del menú.



Enviar notificación

Para poder enviar una notificación debe

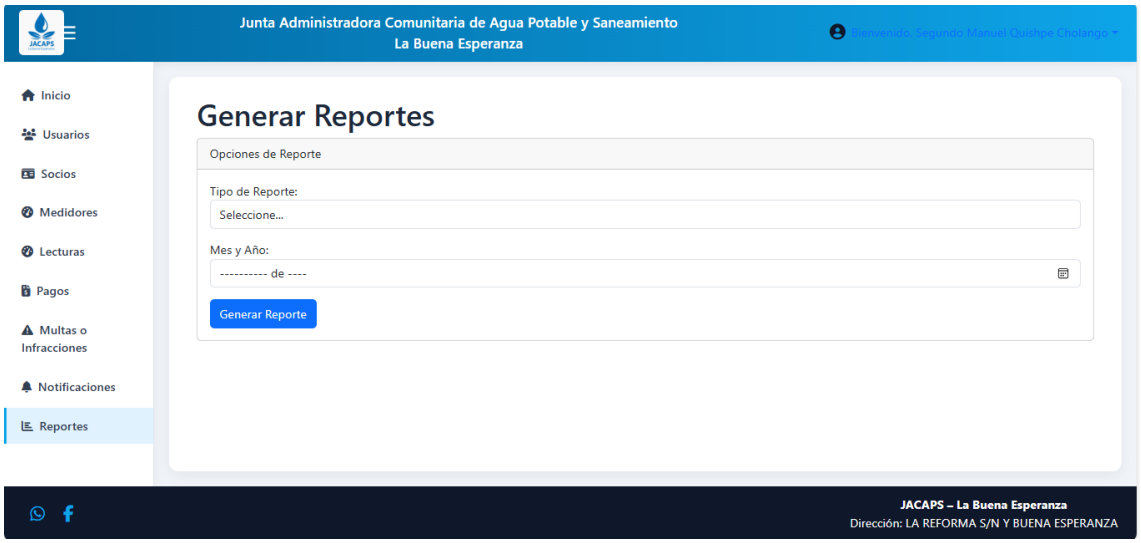
- Seleccionar “enviar a todos” o “socio específico”
- Ingresa el asunto de la notificación
- Ingresa el mensaje
- Dar clic en “enviar notificación” y el mensaje de enviará al socio o socios seleccionados.



Reporte

En esta sección el administrador podrá generar todo tipo de reportes para esto debe:

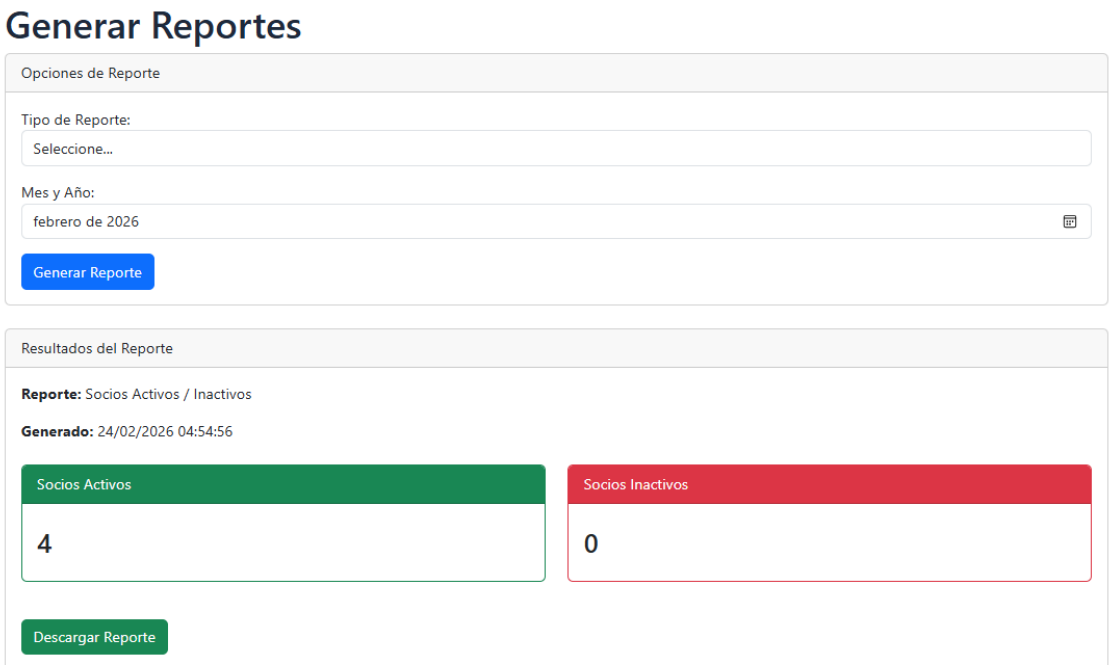
Para esto debe hacer clic en la opción “Reportes” del menú.



Generación de reporte

Una vez ahí, se debe:

- Elegir el tipo de reporte que desea generar
- Seleccionar la fecha a generar
- Dar clic en “generar”
- Y me va a desplegar un preview del reporte seleccionado
- Si desea descargar, dar clic en “descargar reporte” y comenzará la descarga.


Generar Reportes
Opciones de Reporte
Tipo de Reporte:
Seleccione...
Mes y Año:
febrero de 2026
Generar Reporte
Resultados del Reporte
Reporte: Socios Activos / Inactivos
Generado: 24/02/2026 04:54:56
Socios Activos
4
Socios Inactivos
0
Descargar Reporte

Anexo 5: Manual de Usuario Socio



APLICACIÓN WEB PARA LA GESTIÓN DE PAGOS EN LA JUNTA ADMINISTRADORA DE AGUA POTABLE Y SANEAMIENTO LA BUENA ESPERANZA, PARROQUIA CANGAHUA, CANTÓN CAYAMBE.

AUTOR:

MARCO JOSUE QUISHPE CUELLAR

FECHA: 10/01/2026

Descripción

Esta aplicación web tiene como objetivo gestionar de manera eficiente todos los procesos relacionados con los pagos, es decir: creación de usuarios, creación de socios, creación de medidores, registro de lecturas, generación de reportes.

Objetivo

Facilitar el manejo del sistema web permitiendo al usuario aprender a manejarlo correcta y eficientemente.

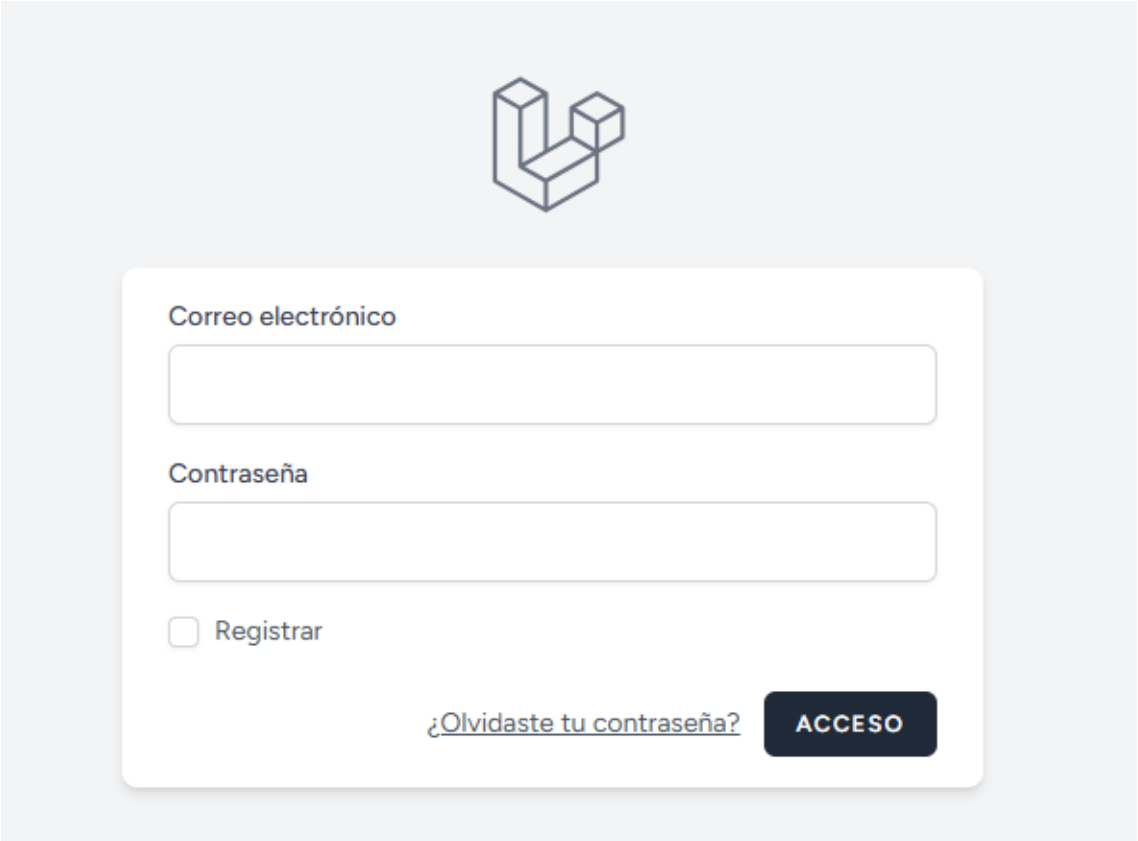
Acceso al Sistema:

Para acceder al sistema se debe seguir los siguientes pasos

- Abrir el navegador web
- Ingresar la dirección URL en la barra de direcciones.
- En la página de bienvenida encontrará un botón de “Iniciar Sesión”, dar clic ahí.



- Ingresar correctamente las credenciales enviadas al correo electrónico.



- Dar clic en el botón de “Acceder”

Una vez iniciado sesión será dirigido a la página de inicio, esta vista es única para socios.

La vista principal del socio tendrá las siguientes opciones en el menú:

- **Inicio:** vista por defecto
- **Sección de Historial de Pagos:** Visualizar el historial de pagos por el servicio
- **Sección de multas:** visualizar y pagar las multas pendientes.
- **Sección de pagos:** visualizar y pagar por el servicio de agua



Historial de Pagos

En esta sección puede visualizar los pagos que ya ha realizado anteriormente.



Multas

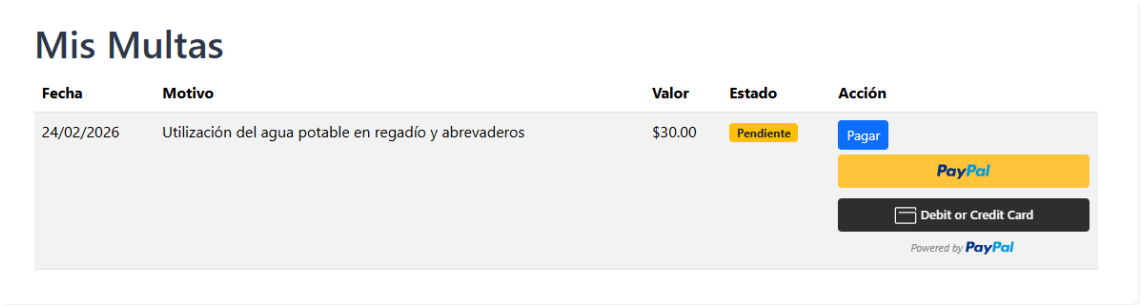
En esta sección se puede visualizar las multas que han sido generadas por cometer alguna infracción.



Pagar multa

Si desea pagar una multa:

- Dar clic en pagar
- Se abrirá el modal para pagar
- Elegir el método, hacer el pago
- Y la multa será marcada como pagada.



Pagar Servicio

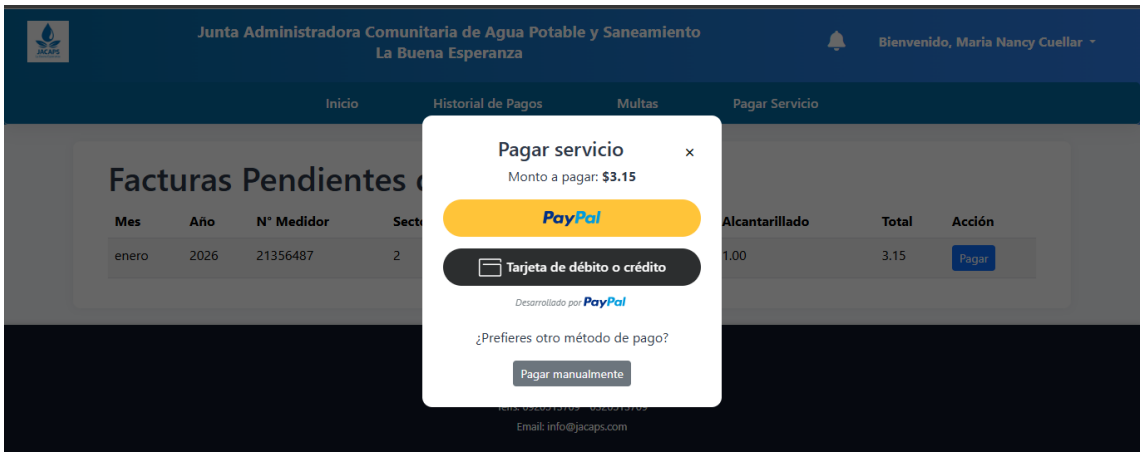
En esta sección se mostrará los valores a pagar por cada periodo o mes



Pagar servicio

Si desea pagar el servicio:

- Dar clic en el botón “pagar”
- Elegir el método de pago
- Realizar el pago
- La factura se marcará como pagada
- Y me genera un comprobante



1793209628001
JAAPS LA BUENA ESPERANZA DE CANGAHUA
LA REFORMA S/N Y BUENA ESPERANZA
Telfs: 0920513709
Email: juntacomunitariadeagua@gmail.com

FACTURA Nro: 0001100-000004764

Identificación: 1713097746 **Fecha:** 23/02/2026 21:45
Cliente: Maria Nancy Cuellar
Dirección: LA BUENA ESPERANZA
Teléfono: 0969825184

Comentario: CONSUMO MES: 2026-01 | SECTOR: 1 | SERVICIO: Residencial | L. ANTERIOR: 0 | L. ACTUAL: 10 | CONSUMO: 10 | MEDIDOR Nro: 12346587 | DIRECCION: COMUNIDAD BUENA ESPERANZA

Cant	Código	Producto	P. Unit	Descuento	Subtotal
1.000	PROD001	CONSUMO AGUA POTABLE	2.00	0.00000	2.00
1.000	PROD002	SERVICIO DE ALCANTARILLADO	1.00	0.00000	1.00

Subtotal	3.00
Base IVA	0.00
Base 0%	3.00
IVA	0.00
Total	**3.00**

Documento sin validez tributaria, revise su correo electrónico
Recaudado por: Tesorería