

UNIVERSIDAD TÉCNICA DEL NORTE



Facultad de Ingeniería en Ciencias Aplicadas
Carrera de Ingeniería en Sistemas Computacionales

**SISTEMA DE FACTURACIÓN UTILIZANDO LOS FRAMEWORK ANGULAR 5,
LARAVEL 5.5 PARA LA JUNTA ADMINISTRADORA DE AGUA POTABLE SAN
JOSÉ DE CHORLAVÍ DE LA CIUDAD DE IBARRA – ECUADOR.**

Trabajo de grado previo a la obtención del título de Ingeniero
en Sistemas Computacionales

Autor:

Figuroa Cuaspud Washington Estuardo

Director:

Msc. Granda Gudiño Pedro David

Ibarra, 2018



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

| DATOS DE CONTACTO | |
|----------------------|--|
| CÉDULA DE IDENTIDAD: | 1003833447 |
| APELLIDOS Y NOMBRES: | FIGUEROA CUASPUD WASHINGTON ESTUARDO |
| DIRECCIÓN: | IBARRA – SAN ANTONIO – LOS SOLES |
| EMAIL: | wefigueroa@utn.edu.ec |
| TELÉFONO FIJO: | |
| TELÉFONO MÓVIL: | 0969191290 |

| DATOS DE LA OBRA | |
|-----------------------------|--|
| TÍTULO: | “SISTEMA DE FACTURACIÓN UTILIZANDO LOS FRAMEWORK ANGULAR 5, LARAVEL 5.5 PARA LA JUNTA ADMINISTRADORA DE AGUA POTABLE SAN JOSÉ DE CHORLAVÍ DE LA CIUDAD DE IBARRA – ECUADOR.” |
| AUTOR (ES): | FIGUEROA CUASPUD WASHINGTON ESTUARDO |
| FECHA: DD/MM/AAAA | 21 /09/2018 |
| SOLO PARA TRABAJOS DE GRADO | |
| PROGRAMA: | <input checked="" type="checkbox"/> PREGRADO <input type="checkbox"/> POSGRADO |
| TÍTULO POR EL QUE OPTA: | INGENIERO EN SISTEMAS COMPUTACIONALES |
| ASESOR /DIRECTOR: | MSC. PEDRO GRANDA |

2. CONSTANCIAS

El autor manifiesta que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto la obra es original y que es el titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 21 días del mes de septiembre de 2018

EL AUTOR:

(Firma).....

Figueroa Cuaspud Washington Estuardo

II

II

UNIVERSIDAD TÉCNICA DEL NORTE



FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN DEL DIRECTOR

Por medio del presente yo Msc. Pedro Granda, certifico que el Sr. Washington Estuardo Figueroa Cuaspud, portador de la cédula de identidad Nro. 100383344-7. Ha trabajado en el desarrollo del proyecto de tesis **“SISTEMA DE FACTURACIÓN UTILIZANDO LOS FRAMEWORK ANGULAR 5, LARAVEL 5.5 PARA LA JUNTA ADMINISTRADORA DE AGUA POTABLE SAN JOSÉ DE CHORLAVÍ DE LA CIUDAD DE IBARRA – ECUADOR.”**, previo a la obtención del título de ingeniería en sistemas computacionales, lo cual ha realizado en su totalidad con responsabilidad.

Es todo cuanto puede certificar en honor a la verdad

Msc. Pedro Granda
DIRECTOR DE TESIS



FUNDADA EL 3 DE ABRIL DE 1993

CERTIFICADO

YO Lcdo. Héctor Álvarez PRESIDENTE DE LA JUNTA DE AGUA POTABLE Del Barrio San José de Chorlavi;

CERTIFICO que el Señor Washington Estuardo Figueroa Cuaspud con C.I.100383344-7 ha implementado el SISTEMA DE FACTURACIÓN Y ADMINISTRACIÓN en nuestra Institución quien gracias a sus conocimientos, entrega y dedicación nos ha entregado listo el sistema el día domingo 5 de agosto del año 2018.

Es todo cuanto puedo certificar en honor a la verdad.

El mencionado señor puede hacer uso del presente como estime conveniente.

Atentamente,

Lcdo. Héctor Álvarez

PRESIDENTE DE LA JAAP. CHORLAVÍ.



Dirección: San José de Chorlavi (Casa comunal)
E-mail: juntasanjosedechorlavi@gmail.com

0986411744
0985576513
0988136912

DEDICATORIA

"La determinación es el punto inicial de todo logro". –W. Clement Stone

A Dios

Por haberme permitido tener vida, salud y bien estar para alcanzar este gran objetivo.

A mis padres.

Esteban Figueroa Y Rosita Cuaspud por todo su amor, por cada uno de sus valiosos consejos, principios y valores que me han formado como persona, y principalmente por su ejemplo de lucha perseverancia y superación diaria.

A mis familiares.

Hermanos, tíos, primos por la motivación constante que me ha facilitado el camino para el cumplimiento de cada gran meta y objetivo planteado en mi vida

A mis amigos.

Por su apoyo incondicional, y su compartir diario que ha sido el mayor impulso para el cumplimiento de este gran reto.

En fin, a cada una de aquellas personas que de forma directa o indirecta contribuyeron con mi formación académica que estoy seguro será de gran ayuda para el buen desarrollo de mi vida profesional.

AGRADECIMIENTOS

Atraves de estas líneas quiero agradecer principalmente a Dios por bendecirme con la vida que es esencial para llegar hasta donde eh llegado.

A la Universidad Técnica el Norte, principalmente a quienes son representantes de la Facultad de Ingeniería en Ciencias Aplicadas por permitirme tener una formación académica y profesional que contribuirá con el desarrollo de mi país

A los docentes de mi carrera ingeniería en Sistemas Computacionales por contribuir con sus conocimientos en mi formación profesional que estoy seguro será de gran ayuda al momento de ejercer mi profesión.

A la directiva de la Junta Administradora De Agua Potable San José De Chorlaví de la ciudad de Ibarra por la confianza apertura y apoyo en el desarrollo del presente trabajo de grado

A mis familiares y amigos por su apoyo confianza y cada buen consejo que me a motiva a seguir adelante siempre.

Sin más me gustaría poder extender mi más sincero agradecimiento a cada una de las personas que han contribuido de una u otra forma para el cumplimiento de cada una de las metas y objetivos que se han presentado durante esta mi formación profesional.

RESUMEN

Este documento describe el análisis, diseño e implementación implicados en la creación del sistema “La Sirena” (Sistema de Administración para la junta de agua potable, San José de Chorlaví). El cual fue desarrollado con los framework Angular 5 (FrontEnd) y Laravel 5.5 (BackEnd).

El desarrollo de un sistema Web para la administración de la junta de agua potable de San José de Chorlaví, aparece a través de las necesidades de los señores usuarios del servicio y de la directiva de la comunidad, dando lugar a los problemas en los procesos de, registro de inventario, seguimiento de medidores, registro de consumos mensuales de agua potable, control de ingresos egresos y tiempo en la gestión de procesos operativos.

El sistema automatiza dichos procesos mediante la utilización de herramientas Open Source como Angular 5, Laravel 5.5, PostgreSQL, MDB y más herramientas que son descritas en los siguientes capítulos.

El estudio esencial de este trabajo es analizar el nivel de flexibilidad, compatibilidad y adaptabilidad entre estos dos framework, llevado a cabo el desarrollo del sistema, cumpliendo con los estándares de calidad del software, como son la funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad.

El sistema contribuye al cuidado del medio ambiente mediante la disminución del uso de papel y suministros de oficina.

ABSTRACT

This document describes the analysis, design and implementation involved in the creation of the "La Sirena" system (Administration System for the Potable Water Board, San José de Chorlaví). Which was developed with the framework Angular 5 (FronEnd) and Laravel 5.5 (BackEnd).

The development of a Web system for the administration of the Board of Drinking Water of San José de Chorlaví, appears through the needs of the users of the service and the board of the community, giving rise to the problems in the processes of, inventory record, monitoring of meters, record of monthly consumption of drinking water, control of income, expenses and time in the management of operational processes.

Open source such as Angular 5, Laravel 5.5, PostgreSQL, MDB and more tools described in the following chapters.

The essential study of this work is to analyze the level of flexibility, compatibility and adaptability between these two frameworks, carried out in the development of the system, complying with the software quality standards, such as functionality, reliability, usability, efficiency, updating and portability.

The system contributes to the care of the environment by reducing the use of paper and office supplies.

ÍNDICE DE CONTENIDOS

| | |
|---|-------------------------------|
| CERTIFICACIÓN DEL DIRECTOR | ¡Error! Marcador no definido. |
| CERTIFICACIONES JAAP | ¡Error! Marcador no definido. |
| DEDICATORIA..... | V |
| AGRADECIMIENTOS..... | VI |
| RESUMEN..... | VII |
| ABSTRACT..... | VIII |
| INDICE DE TABLAS | XII |
| ÍNDICE DE FIGURAS..... | XIII |
| INTRODUCCIÓN..... | XV |
| Antecedentes..... | XV |
| Situación actual..... | XV |
| Prospectiva..... | XV |
| Definición del problema..... | XVI |
| Objetivos..... | XVI |
| Objetivo general..... | XVI |
| Objetivos específicos..... | XVI |
| Alcance..... | XVI |
| Justificación..... | XVII |
| CAPÍTULO I..... | 1 |
| MARCO TEÓRICO..... | 1 |
| 1.1 Sistema Web | 1 |
| 1.1.1 Características y beneficios de un sistema Web..... | 2 |
| 1.2 Junta San José de Chorlaví | 3 |
| 1.3 Base de datos..... | 3 |
| 1.3.1 Base de datos de distribución libre..... | 3 |
| 1.3.2 PostgreSQL..... | 4 |
| 1.4 Metodologías | 4 |
| 1.4.1 Metodologías ágiles | 5 |
| 1.4.2 Metodología SCRUM | 12 |
| 1.4.2.1 Fases de Scrum..... | 13 |
| 1.5 Lenguaje PHP | 14 |
| 1.5.1 Framework..... | 14 |
| 1.5.2 Comparativa framework PHP | 16 |
| LARAVEL | 16 |

| | |
|---|----|
| CAKEPHP | 16 |
| CODEIGNITER | 16 |
| SYMFONY | 16 |
| YII2 | 16 |
| 1.6 Laravel | 19 |
| 1.6.1 Laravel 5.5 LTS..... | 20 |
| 1.6.2 Arquitectura MVC | 21 |
| 1.6.3 Arquitectura MVC Laravel | 21 |
| 1.6.2 Diagrama de secuencia de Laravel | 23 |
| 1.6.3 Estructura del proyecto | 24 |
| 1.7 Lenguaje JS | 25 |
| 1.7.1 Framework JS | 25 |
| 1.7.2 Comparativa framework JS..... | 26 |
| 1.7.3 Angular | 27 |
| 1.7 Angular 5 | 28 |
| 1.7.1 Principales características de Angular 5..... | 29 |
| 1.7.1 Data binding en Angular 5..... | 30 |
| 1.7.2 Módulos..... | 30 |
| 1.7.3 Scopes..... | 31 |
| 1.7.4 Controladores..... | 31 |
| 1.7.5 Directivas | 32 |
| 1.8 Gestores de dependencias..... | 32 |
| 1.8.1 Node | 32 |
| 1.8.2 Composer | 33 |
| CAPITULO II..... | 35 |
| 2. DESARROLLO | 35 |
| 2.1 Pre-juego Planificación | 35 |
| 2.1.1 Análisis de Procesos..... | 35 |
| 2.1.2 Simbología utilizada en los diagramas..... | 36 |
| 2.1.3 Alcance del software | 39 |
| 2.1.4 Definición del Backlog del Producto. | 40 |
| 2.2 Juego Desarrollo de Sprint | 41 |
| 2.2.1 Diseño de la base de datos..... | 41 |
| 2.2.2 Sprint1 “Módulo de Administración” | 43 |
| 2.2.3 Análisis | 47 |
| 2.2.3.2 Especificación de casos de Uso | 47 |

| | |
|--|----|
| 2.3 Post-juego | 57 |
| Preparación, pruebas, lanzamiento..... | 57 |
| 2.3.1 Modelos de datos | 57 |
| 2.3.2 Arquitectura | 59 |
| 2.3.3 Construcción y pruebas..... | 60 |
| 2.3.4 Pruebas..... | 62 |
| CAPITULO III | 65 |
| 3.1 Análisis encuesta | 65 |
| 3.2 Análisis de la entrevista | 77 |
| 3.3 Cuadro de resultados de los frameworks utilizados..... | 77 |
| 3.4 Análisis de Impactos..... | 78 |
| 3.4.1 Impacto educativo..... | 80 |
| 3.4.1 Impactos Socio Cultural..... | 81 |
| 3.4.2 Impactos Económicos..... | 82 |
| 3.4.3 Impacto Ambiental..... | 82 |
| Conclusiones..... | 83 |
| Recomendaciones..... | 84 |
| Bibliografía..... | 85 |
| Anexos..... | 89 |

INDICE DE TABLAS

| | |
|---|----|
| Tabla 1. Características del PostgreSQL..... | 4 |
| Tabla 2. Cuadro comparativo de las metodologías..... | 7 |
| Tabla 3. Características de la metodología Scrum | 13 |
| Tabla 4. Versiones de framework Laravel..... | 18 |
| Tabla 5. Estructura de Laravel..... | 24 |
| Tabla 6. Características de framework Java Script..... | 26 |
| Tabla 7. Características de Angular..... | 27 |
| Tabla 8. Simbología utilizada en los diagramas..... | 36 |
| Tabla 9. Requerimientos funcionales y no funcionales | 37 |
| Tabla 10. Backlog del Producto..... | 40 |
| Tabla 11. Requerimientos funcionales y no funcionales de los Sprint | 43 |
| Tabla 12. Historias de Usuario - Sprint 1 | 44 |
| Tabla 13. Ingreso de parámetros Sprint 2 | 44 |
| Tabla 14. Sprint 3 registro de proveedores..... | 45 |
| Tabla 15. Sprint 4 control de ingresos | 46 |
| Tabla 16. Definición del equipo de trabajo..... | 47 |
| Tabla 17. Actores del sistema | 47 |
| Tabla 18. Gestionar Usuarios..... | 48 |
| Tabla 19. Seleccionar Usuario..... | 49 |
| Tabla 20. Asignar/Reasignar Responsable..... | 49 |
| Tabla 21. Seleccionar Perfil..... | 51 |
| Tabla 22. Seleccionar Formulario..... | 51 |
| Tabla 23. Gestionar Perfiles | 52 |
| Tabla 24. Gestionar Formularios | 52 |
| Tabla 25. Gestión de reportes | 54 |
| Tabla 26. Ingreso de proveedores..... | 55 |
| Tabla 27. Ingreso de parámetros..... | 56 |
| Tabla 28. Control de ingresos generales | 57 |
| Tabla 29. Backlog Sprint “Módulo de Administración” | 60 |

ÍNDICE DE FIGURAS

| | |
|--|-----------|
| Figura 1. Arquitectura del sistema | XVII |
| Figura 2. Metodología Scrum | 12 |
| <i>Figura 3. Framework de php</i> | <i>18</i> |
| Figura 4. Modelo, vista, controlador MVC..... | 21 |
| Figura 5. Arquitectura MVC Laravel..... | 22 |
| Figura 6. Diagrama de secuencia de Laravel | 23 |
| Figura 7. Data binding en angular | 30 |
| Figura 8. Controladores..... | 31 |
| Figura 9. Proceso de administración del sistema de facturación | 36 |
| Figura 10. Diseño base de datos..... | 42 |
| Figura 11. Modelo entidad relación..... | 58 |
| Figura 12. Agilizar la etapa de pruebas ZenHub | 63 |
| Figura 13. Implementación del sistema | 68 |
| Figura 14. Eficiencia del sistema de agua potable..... | 69 |
| Figura 15. Solicitud de información en el sistema..... | 70 |
| Figura 16. Valor cancelado por planilla es el correcto | 71 |
| Figura 17. Lectura de medidor..... | 72 |
| Figura 18. Mejoramiento del servicio de cobro de planillas..... | 73 |
| Figura 19. Beneficios de la JAAP. | 74 |
| Figura 20. Genero de los encuestados..... | 75 |
| Figura 21. Instrucción de las personas encuestadas | 76 |
| Figura 22. Diagrama de casos de uso sistema de facturación..... | 89 |
| Figura 23. Validar usuario | 89 |
| Figura 24. Registro de datos | 90 |
| Figura 25. Asignación servicio..... | 90 |
| Figura 26. Registro Consumo..... | 90 |
| Figura 27. Calcular tarifas | 91 |
| Figura 28. Facturas | 91 |
| Figura 29. Generar reportes | 91 |
| Figura 30. Diagrama de estado contribuyente | 92 |
| Figura 31. Diagrama de estado del servicio..... | 92 |
| Figura 32. Diagrama de estado factura | 92 |
| <i>Figura 33. Diagrama de secuencia de usuarios del sistema.....</i> | <i>93</i> |
| <i>Figura 34. Diagrama de secuencia de JAAP</i> | <i>93</i> |

Figura 35. Diagrama de secuencia de Costo Parámetros..... 94

Figura 36. Diagrama de secuencia de Contribuyente 94

Figura 37. Diagrama de secuencia de Activación servicio 95

Figura 38. Diagrama de secuencia de Consumo 95

Figura 39. Diagrama de secuencia tarifas 96

Figura 40. Diagrama de secuencia Emitir consumo..... 96

Figura 41. Diagrama de secuencia de Facturar 97

Figura 42. Diagrama de secuencia de Reportes..... 97

INTRODUCCIÓN

Antecedentes.

La Junta administradora de agua potable de San José de Chorlaví, está ubicado en la parroquia San Francisco del cantón Ibarra perteneciente a la provincia de Imbabura.

Basados en la declaración constitucional de agua como derecho humano fundamental, La junta cuenta con personería jurídica, emitida por SENAGUA, cuya fuente de agua de donde se abastece el sistema está ubicada en las coordenadas E816979 N10036705, con una concesión de 2 litros por segundo de agua potable se constituye como Junta Administradora de agua potable denominada JAAP de San José de Chorlaví, fundada el 03 abril de 1993.

La JAAP de San José de Chorlaví además de trabajar con autonomía administrativa y financiera, La administración de operación y mantenimiento de la infraestructura del sistema de agua potable tiene como objetivo la prestación de los servicios públicos y saneamiento sin fines de lucro y en beneficio de sus consumidores.

Situación actual.

En cuanto al manejo de la administración por parte de la institución se lo realiza a través de forma empírica (escritos), el actual crecimiento de usuarios y adquisición de recursos; el sistema tradicional ha declinado en velocidad, capacidad de almacenamiento y seguridad, poniendo en riesgo a la junta administradora de agua potable.

Observando y analizando estos problemas, los directivos actuales han optado en implementar sistemas informáticos con nueva tecnología, que mejoren la administración en beneficio en usuarios y la JAAP.

Prospectiva.

La Junta administradora de agua potable de San José de Chorlaví, va a contar con un sistema web que permita mejorar los procesos administrativos. Gracias al presente proyecto se mejorará la administración y facturación con beneficio a usuarios y JAAP, reduciendo tiempo y costo; con objeto de brindar un servicio de calidad.

Definición del problema.

La Junta administradora de agua potable de San José de Chorlaví, no cuenta con ningún sistema, ni medio para administrar sus bienes. Según el avance de los tiempos la Junta administradora de agua potable, posee problemas al momento de la lectura, control y entrega de información.

La Junta actualmente realiza lecturas en hojas de papel aumentando el riesgo de pérdidas de datos, errores en la lectura de consumo de agua potable, pérdida de tiempo en la entrega de datos, provocando con ello molestias en los usuarios e insatisfacción en las facturas, entre otras.

Objetivos.

Objetivo general.

Desarrollar un Sistema de facturación utilizando los framework Angular 5, Laravel 5.5 para la junta administradora de agua potable San José de Chorlaví de la ciudad de Ibarra - Ecuador.

Objetivos específicos

- ✓ Recopilar información acerca del manejo del Framework Laravel5.5 como BackEnd y Angular5 como FrontEnd.
- ✓ Desarrollar el sistema de facturación utilizando la metodología Scrum.
- ✓ Socializar los resultados alcanzados.

Alcance.

El sistema informático se desarrollará para una plataforma web Responsiva, a través de las nuevas tecnologías como Laravel 5.5 (BackEnd) creador de servicios y Angular 5 (FrontEnd) frontal del sistema, con la finalidad de satisfacer las necesidades de los usuarios y alcanzar un software de calidad para facturación de la junta de agua San José de Chorlaví.

El sistema contara con los siguientes módulos:

- ✓ Módulo Administración
- ✓ Módulo de Recaudación
- ✓ Módulo de Facturación

Los procesos que se van a llevar a cabo son:

- ✓ Scrum como metodología ágil y flexible para gestionar el desarrollo de software.
- ✓ PostgreSQL 9.4 ya que es dirigido por una comunidad de desarrolladores.
- ✓ Para el desarrollo del sistema web, se lo llevara a cabo con los lenguajes de programación PHP y JavaScript utilizando los siguientes Framework.

Laravel 5.5 y Angular 5; Se desarrollará un sistema en la que Laravel creará las API y Angular consumirá las API.



Figura 1. Arquitectura del sistema

Autor / Fuente: Figueroa Washington

Esta versión entra dentro del plan de Google de lanzar una nueva revisión "grande" cada 6 meses, aunque a veces las novedades sean más bien escasas. No es lo que ocurre en esta ocasión, en la que la versión 5 trae bastantes cosas nuevas y alguna de ellas de cierta importancia.

Justificación.

El desarrollo del sistema de facturación utilizando los framework Angular 5, Laravel 5.5 para la junta administradora de agua potable San José de Chorlaví de la ciudad de Ibarra -

Ecuador, se planificó con objeto de aportar a los comuneros y empresa, reduciendo tiempo y costo en los servicios de facturación y diferentes trámites de la empresa.

Las razones del planteamiento es por los grandes beneficios que tendrán los ciudadanos y directivos de la empresa, tales como:

Ciudadanos.

- ✓ Facturación mensual
- ✓ Disminución de tiempo, en el transcurso de cancelación del servicio
- ✓ Corrección a tiempo de la lectura de consumo de agua potable, cuando exista error en la misma
- ✓ Entrega de la Planilla mensual
- ✓ Entrega de rendición de cuentas mensual

Empresa.

- ✓ Reducción de riesgos en pérdidas de hojas de lectura de consumo de agua potable
- ✓ Corrección a tiempo en la lectura de consumo de agua potable
- ✓ Eliminación de tiempo en digitación de datos del consumo de agua potable al sistema
- ✓ Visita domiciliaria a los deudores, con la factura mensual.
- ✓ Reducción de alto porcentaje de deudores, mediante la aplicación del sistema facturación
- ✓ Facturación de ventas de producto y consumo mensual
- ✓ Informes de actividades

CAPÍTULO I

MARCO TEÓRICO

La secretaría del Agua SENAGUA, consiente de su responsabilidad con la comunidades rurales donde se han formado las Juntas administrativas de Agua Potable y Saneamiento JAAP, que son las organizaciones comunitarias encargadas de la administración y operación de los sistemas de agua potable, con la finalidad de que los miembros de estos entes comunitarios tengan un cabal conocimiento de sus funciones, ha creído conveniente recapitular leyes, reglamentos, disposiciones generales y anexos que regulen el funcionamiento de los mismos, tanto en la parte administrativa, comercial y técnica.

Al no contar la institución con un sistema para el manejo y al llevarlo de forma manual mucha de la información tiende a perderse y eso no es conveniente para la empresa, es de allí que nace la necesidad de la implementación de una plataforma informática para agilizar los procesos de administración de la Junta de Agua Potable.

La propuesta del proyecto es realizar la implementación del sistema de facturación utilizando los framework angular 5, Laravel 5.5 para la junta administradora de agua potable san José de Chorlaví de la ciudad de Ibarra, se desarrollará mediante tecnologías actuales que aporte con rapidez en su desarrollo y eficiencia a la hora de la entrega de información a los usuarios del sistema. El sistema será desarrollado para tecnologías web, a continuación, se detalla las herramientas adecuadas.

1.1 Sistema Web

Un sistema web le permitirá ahorrar costos, tiempo y Recursos humanos. Por tanto, la rentabilidad de su empresa aumentará y gestionará mejor a sus clientes, proveedores, distribuidores, etc. De acuerdo con Websystem (2017) menciona que: “Los Sistemas Web han demostrado mejores resultados para las empresas frente a los Sistemas Tradicionales Cliente/Servidor, pues le brindan beneficios que estos no pueden tales como”:

- ✓ No tiene que pagar licencias por cada computadora con el Sistema instalado pues está en un sólo Servidor Web.
- ✓ Facilidad para acceder al Sistema desde cualquier punto con conexión a Internet
- ✓ Puede Integrar todas sus tiendas y sucursales; manteniendo información actualizada de stock de productos y de los ingresos y los egresos.
- ✓ Le permite comunicarse con sus clientes de manera más fluida las 24 horas del día

1.1.1 Características y beneficios de un sistema Web

Las aplicaciones web son populares debido a la practicidad del navegador web que actualmente está disponible tanto en Equipos de escritorio, Notebooks, Celulares, Tablet, etc. La facilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software en miles de potenciales clientes es otra razón de su renombre.

- ✓ **Compatibilidad multiplataforma:** Una misma versión de la aplicación puede correr sin problemas en múltiples plataformas como Windows, Linux, Mac, Android, etc.
- ✓ **Actualización:** Las aplicaciones web siempre se mantienen actualizadas y no requieren que el usuario deba descargar actualizaciones y realizar tareas de instalación.
- ✓ **Acceso inmediato y desde cualquier lugar:** Las aplicaciones basadas en tecnologías web no necesitan ser descargadas, instaladas y configuradas. Además, pueden ser accedidas desde cualquier computadora conectada a la red en donde se accede a la aplicación.
- ✓ **Menos requerimientos de hardware:** Pueden funcionar en cualquier equipo que disponga de un navegador web. Esto aplica tanto a Celulares, Tablet y otros dispositivos modernos.
- ✓ **Menos Bugs (errores):** Deberían ser menos propensas a tildarse y crear problemas técnicos debido a problemas de software y conflictos de hardware. Otra razón es que con aplicaciones basadas en web todos utilizan la misma versión, y los bugs pueden ser corregidos tan pronto como son descubiertos beneficiando inmediatamente a todos los usuarios del sistema.
- ✓ **Seguridad en los datos:** Los datos se alojan en servidores ubicados en Data Centers con toda la infraestructura necesaria para asegurar la protección de datos y el funcionamiento constante de las aplicaciones.

Plataforma web permitirá a los usuarios tener un acceso del sistema desde diferentes ubicaciones por esta razón se propone la elaboración. La elaboración del sistema web se lo llevara a cabo mediante el uso de nuevas tecnologías y framework que evalúen la calidad del producto.

1.2 Junta San José de Chorlaví

La Junta administradora de agua potable de San José de Chorlaví, está ubicado en la parroquia San Francisco del cantón Ibarra perteneciente a la provincia de Imbabura.

La JAAP cuenta con personería jurídica, emitida por SENAGUA, cuya fuente de agua de donde se abastece el sistema está ubicada en las coordenadas E816979 N10036705, llamada "La Sirena", con una concesión de 2 litros por segundo de agua potable se constituye como (Junta Administradora de Agua Potable) denominada JAAP de San José de Chorlaví, fundada el 03 abril de 1993.

1.3 Base de datos

Una base de datos es un almacén que nos permite guardar grandes cantidades de información de forma organizada para que luego podamos encontrar y utilizar fácilmente.

El término de bases de datos fue escuchado por primera vez en 1963, en un simposio celebrado en California, USA. Una base de datos se puede definir como un conjunto de información relacionada que se encuentra agrupada o estructurada. Desde el punto de vista informático, la base de datos es un sistema formado por un conjunto de datos almacenados en discos que permiten el acceso directo a ellos y un conjunto de programas que manipulen ese conjunto de datos. (Perez, 2007, pág. 1)

1.3.1 Base de datos de distribución libre.

Los sistemas de gestores libres son aquellos que podemos descargar sin tener que pagar alguna cantidad, por ejemplo:

- ✓ MySQL
- ✓ Sydes (Linux)
- ✓ SQLIFE
- ✓ PostgreSQL
- ✓ Filebird
- ✓ Apache Derby
- ✓ DBD2 express-C DB2

1.3.2 PostgreSQL

PostgreSQL es un sistema de base de datos relacional de objetos de código abierto. Una base de datos de clase empresarial, PostgreSQL cuenta con funciones sofisticadas tales como Multi-Version Concurrency Control (MVCC), recuperación puntual, tables paces, replicación asincrónica, transacciones anidadas (puntos de rescate), copias de seguridad en línea / calientes, un sofisticado planificador / optimizador de consultas y escritura registro anticipado para la tolerancia a fallas.

Admite conjuntos de caracteres internacionales, codificaciones de caracteres multibyte, Unicode, y es compatible con la configuración regional para la clasificación, la distinción entre mayúsculas y minúsculas y el formateo.

Tabla 1. Características del PostgreSQL

| Límite | Valor |
|--------------------------------|---|
| Tamaño de base de datos máximo | Ilimitado |
| Tamaño máximo de la tabla | 32 TB |
| Tamaño máximo de fila | 1.6 TB |
| Tamaño máximo de campo | 1 GB |
| Filas máximas por mesa | Ilimitado |
| Columnas máximas por mesa | 250 - 1600 dependiendo de los tipos de columnas |
| Índices máximos por tabla | Ilimitado |

Fuente: Autor

1.4 Metodologías

“Una metodología de software es un enfoque, una manera de interpretar la realidad o la disciplina en cuestión, que en este caso particular correspondería a la Ingeniería de Software”. (Gómez, 2017). Igualmente Gómez (2017) menciona que, “la metodología destinada al desarrollo de software se considera como una estructura utilizada para planificar y controlar el procedimiento de creación de un sistema de información especializada” (p.1).

1.4.1 Metodologías ágiles

Dentro de las metodologías se puede mencionar a las más utilizadas para el desarrollo de software las cuales menciono a continuación:

Modelo de Cascada

En este modelo cada etapa representa una unidad de desarrollo con un pequeño descanso en el medio. Por lo tanto, cada siguiente etapa inicia tan pronto como la anterior haya culminado, y esos descansos son usados para confirmaciones del lado del cliente. (Gómez, 2017)

Modelo de Espiral

La metodología espiral refleja la relación de tareas con prototipos rápidos, mayor paralelismo y concurrencia en las actividades de diseño y construcción. El método en espiral debe todavía ser planificado metódicamente, con las tareas y entregables identificados para cada paso en la espiral. (Gómez, 2017)

Metodología de Prototipo

Con el objetivo de otorgar resultados rápidos, se trata de un enfoque que está destinado a proporcionar un excelente proceso de desarrollo con la ayuda de otros enfoques, pero además, está diseñado para aumentar la viabilidad de todo el procedimiento de desarrollo de software para resaltar la participación de un usuario activo. (Gómez, 2017).

Dicho esto, algunas de las ventajas a destacar de este tipo de desarrollo son las siguientes:

- ✓ Hace todo el proceso de desarrollo sin esfuerzo.
- ✓ Asiste al cliente en la realización de revisiones rápidas.
- ✓ Alienta la retroalimentación de los clientes para su mejora.

Metodología de Programación Extrema (XP)

Como metodología ágil de ingeniería de software, la metodología de programación extrema se conoce actualmente como metodología de XP (eXtreme Programming). Esta metodología, se utiliza principalmente para evitar el desarrollo de funciones que actualmente no se necesitan, pero sobre todo para atender proyectos complicados. Sin embargo, sus métodos peculiares pueden tomar más tiempo, así como recursos humanos en comparación con otros enfoques. (Gómez, 2017)

Metodología SCRUM

A pesar de que la metodología XP recibe la mayor atención bibliográfica, las organizaciones están enfocando su atención en la metodología ágil denominada SCRUM (Schwaber & Shuterland, 2011). La cual aplica las mismas premisas conceptuales que XP, pero para resolver un problema ligeramente distinto como es el de desarrollo evolutivo de aplicaciones. SCRUM es una metodología ágil y flexible que sirve para gestionar el desarrollo de software, cuyo principal objetivo es maximizar el retorno de la inversión para su empresa. Se basa principalmente en construir la funcionalidad de mayor valor para el cliente y en los principios de inspección continua, adaptación, autogestión e innovación. (Cendejas, 2014)

Las metodologías utilizadas para el desarrollo de software vienen siendo la metodología en cascada, espiral, XP y scrum a continuación se muestra un cuadro comparativo de las metodologías:

Tabla 2. Cuadro comparativo de las metodologías

| Metodología | Descripción | Ventajas | Inconvenientes |
|--------------------|---|--|---|
| Cascada | Es un proceso de desarrollo secuencial, en el que el desarrollo de software se concibe como un conjunto de etapas que se ejecutan una tras otra. Se le denomina así por las posiciones que ocupan las diferentes fases que componen el proyecto, colocadas una encima de otra, y siguiendo un flujo de ejecución de arriba hacia abajo, como una cascada. | <p>El tiempo que se pasa en diseñar el producto en las primeras fases del proceso puede evitar problemas que serían más costosos cuando el proyecto ya estuviese en fase de desarrollo. (Dominguez, 2017)</p> <p>La documentación es muy exhaustiva y si se une al equipo un nuevo desarrollador, podrá comprender el proyecto leyendo la documentación. (Dominguez, 2017)</p> <p>Al ser un proyecto muy estructurado, con fases bien definidas, es fácil entender el proyecto. (Dominguez, 2017)</p> <p>Ideal para proyectos estables, donde los requisitos son claros y no van a cambiar a lo largo del proceso de</p> | <p>En muchas ocasiones, los clientes no saben bien los requisitos que necesitarán antes de ver una primera versión del software en funcionamiento. Entonces, cambiarán muchos requisitos y añadirán otros nuevos, lo que supondrá volver a realizar fases ya superadas y provocará un incremento del coste. (Dominguez, 2017)</p> <p>No se va mostrando al cliente el producto a medida que se va desarrollando, si no que se ve el resultado una vez ha terminado todo el proceso. Esto cual provoca inseguridad por parte del cliente que quiere ir viendo los avances en el producto. (Dominguez, 2017)</p> <p>Los diseñadores pueden no tener en cuenta todas las dificultades que se encontrarán cuando estén diseñando un software, lo que conllevará rediseñar el proyecto para solventar el problema. (Dominguez, 2017)</p> |

| | | | |
|----------------|--|--|--|
| | | desarrollo. (Dominguez, 2017) | Para proyectos a largo plazo, este modelo puede suponer un problema al cambiar las necesidades del usuario a lo largo del tiempo. Si, por ejemplo, tenemos un proyecto que va a durar 5 años, es muy probable que los requisitos necesiten adaptarse a los gustos y novedades del mercado. (Dominguez, 2017) |
| Espiral | Proporciona el potencial para el desarrollo rápido de versiones incrementales del software que no se basa en fases claramente definidas y separadas para crear un sistema. | El modelo en espiral puede adaptarse y aplicarse a lo largo de la vida del software de computadora. Como el software evoluciona a medida que progresa el proceso, el desarrollador y el cliente comprenden y reaccionan mejor ante riesgos en cada uno de los niveles evolutivos. | Resulta difícil convencer a grandes clientes de que el enfoque evolutivo es controlable. Debido a su elevada complejidad no se aconseja utilizarlo en pequeños sistemas. Genera mucho tiempo en el desarrollo del sistema Modelo costoso |
| | | El modelo en espiral permite a quien lo desarrolla aplicar el enfoque de construcción de prototipos en cualquier etapa de evolución del producto. | Requiere experiencia en la identificación de riesgo |

| | | | |
|-----------|---|--|---|
| | | El modelo en espiral demanda una consideración directa de los riesgos técnicos en todas las etapas del proyecto y si se aplica adecuadamente debe reducir los riesgos antes de que se conviertan en problemas. | |
| | | En la utilización de grandes sistemas ha doblado la productividad | |
| XP | Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, | Programación organizada. Menor tasa de errores. Satisfacción del programador | Es recomendable emplearlo solo en proyectos a corto plazo. Altas comisiones en caso de fallar. |

comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.

Scrum

Scrum es una metodología ágil y flexible para gestionar el desarrollo de software, cuyo principal objetivo es maximizar el retorno de la inversión para su empresa (ROI). Se basa en construir primero la funcionalidad de mayor valor para el cliente y en los principios de inspección continua, adaptación, autogestión e innovación. (Dominguez, 2017)

Entregables en tiempo y forma, puedes ir enviando entregables al cliente mientras vas atacando los objetivos más sencillos, eso te hace ganar tiempo para atacar los objetivos más complejos.

El ScrumMaster tiene el conocimiento necesario para lograr el objetivo primario y secundario por lo cual puede ir controlando el proyecto y delegando roles.

Algunos miembros de tu equipo pueden saltar pasos importantes en el camino rápido para llegar al "sprint" final.

El cliente siempre va a esperar los informes con la fecha exacta, y muchas veces los va a pedir antes, cuando capaz no pudiste avanzar en nada.

Demasiadas Reuniones para poco avance, a veces es muy cansado y estresante reunirse demasiadas veces por el mismo tema, algunos van perdiendo el interés en el proyecto.

Cada persona sabe que es lo que tiene que hacer y no es necesario estar reorganizando una y otra vez los Tracks de cada persona.

Se involucra desde un principio y se da un rol a todos los stakeholders (personas que van a participar en el proyecto incluyendo cliente final, QA, Testers, etc.)

Si una persona renuncia o hay algún cambio es complicado remplazar ese rol ya que es la persona que se lleva el conocimiento específico y afecta a todo el proyecto.

No es aplicable a grandes escalas o cuando el sector IT es variado.

Fuente: Autor

Luego de revisar el cuadro se puede apreciar con claridad que la metodología scrum es la adecuada para llevar a cabo este proyecto. Ya que el proceso de la metodología Scrum se necesitará una cantidad mínima de elementos para sacar adelante el proyecto. Se empieza con la elaboración del Product Backlog, un listado que recoge características, funcionalidades y requerimientos del proyecto a desarrollar. El Product Backlog es mantenido y priorizado por el Producto Owner, sólo este puede agregar prioridades. La segunda etapa es el Sprint Backlog, que viene siendo la lista de tareas que elabora el equipo, se asignaran tareas a cada persona y el tiempo que queda para terminarlas, de esta manera el proyecto se puede descomponer en unidades más pequeñas pudiéndose determinar en donde no se está avanzando e intentar solucionar el problema. Permite crear ciclos breves para el desarrollo, las llamadas iteraciones, que en Scrum se conocen como "Sprint". Sin duda la metodología Scrum aporta grandes beneficios para el proyecto que repercutirán en el conjunto de una organización.

1.4.2 Metodología SCRUM

Scrum permite al software realinearse en cualquier momento según los objetivos del negocio propuestos por la empresa, ya que puede introducir cambios funcionales o de prioridad en cada nueva iteración.

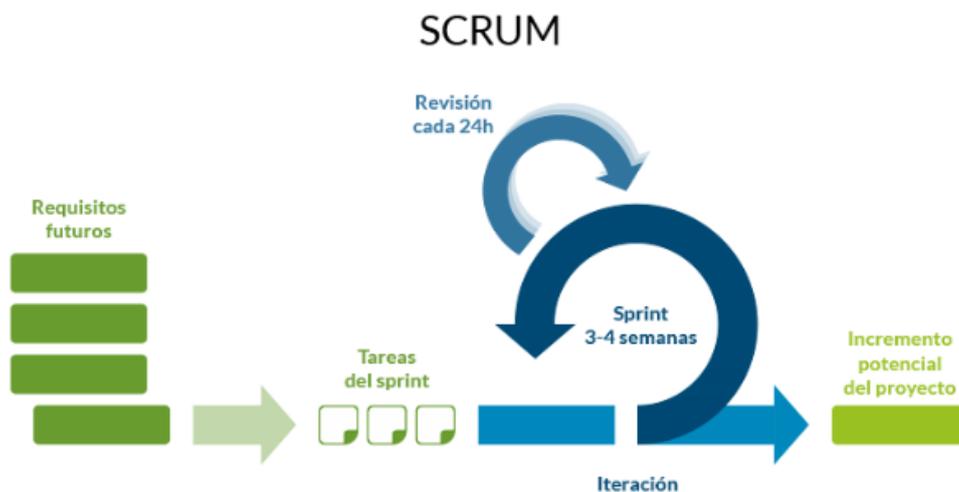


Figura 2. Metodología Scrum

Fuente: <https://es.linkedin.com/pulse/metodologías-agile-cuál-es-la-mejor-rogelio-toledogarcía>

Scrum es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos. (Proyectosagiles, 2017)

Es parte esencial del desarrollo ágil, se centra en las funcionalidades con más prioridades y que pueden ser ejecutadas en un periodo corto de tiempo, Los ciclos de desarrollo, llamados sprint, producen un incremento de funcionalidad terminado y operativo.

A continuación presentamos las características principales que trabaja la metodología scrum:

Tabla 3. Características de la metodología Scrum

| CARACTERÍSTICAS DE LA METODOLOGÍA SCRUM |
|---|
| ✓ Enfatiza valores y prácticas de gestión, sin pronunciarse sobre requerimientos, prácticas de desarrollo, implementación y demás cuestiones técnicas. |
| ✓ Hace uso de Equipos autodirigidos y auto organizados. |
| ✓ Puede ser aplicado teóricamente a cualquier contexto en donde un grupo de gente necesita trabajar junta para lograr una meta común. |
| ✓ Desarrollo de software iterativo incremental basado en prácticas ágiles. |
| ✓ Iteraciones de treinta días; aunque se pueden realizar con más frecuencia, estas iteraciones, conocidas como Sprint. |
| ✓ Dentro de cada Sprint se denomina el ScrumMaster al Líder de Proyecto quien llevará a cabo la gestión de la iteración. |
| ✓ Se convocan diariamente un “Scrum Daily Meeting” el cual representa una reunión de avance diaria de no más de 15 minutos con el propósito de tener realimentación sobre las tareas de los recursos y los obstáculos que se presentan. En la cual se responden preguntas como: ¿Qué has hecho desde el último encuentro? ¿Qué obstáculos hay para cumplir la meta? ¿Qué harás antes del próximo encuentro? |

Fuente: (Universidad de Barcelona, 2018)

Las características Scrum es la adecuada para llevar a cabo este proyecto. Ya que permitirá obtener resultados de forma inmediata, y a su vez permite llevar un trabajo más organizado. Para agilizar los procesos de programación del sistema.

1.4.2.1 Fases de Scrum.

Pre-juego

Planificación: Desarrollo de una nueva versión basada en la actividad actual, junto a una estimación de coste y prolongación de tiempo del proceso, En un sistema nuevo, esta fase

abarca la visión del análisis, y en un caso de mejoras del sistema comprende un análisis de alcance limitado.

Arquitectura: Diseño de la implementación de las funcionalidades de las tareas, incluyendo la modificación de la arquitectura y diseño general.

Juego

Desarrollo de sprint: Desarrollo de las actividades de la nueva versión con respecto a las variables de tiempo, requisitos, costo y competencia.

La iteración de variables define el final de esta fase, en donde el sistema va evolucionando mediante las múltiples iteraciones de desarrollo de sprint.

Post-juego

Test: Pruebas antes del lanzamiento de la versión del sistema.
Preparación para el lanzamiento de la versión final, incluyendo la documentación.

1.5 Lenguaje PHP

PHP identifica a un lenguaje de programación que nació como Personal Home Page (Página personal). Fue desarrollado por el programador de origen danés Rasmus Lerdorf en 1994 con el propósito de facilitar el diseño de páginas web de carácter dinámico.

Sin embargo, PHP en la actualidad está vinculado Hipertexto Pre-Processor (Hipertexto pre-procesado). El lenguaje es desarrollado hoy en día por The PHP Group, aunque carece de una normativa formal. La Free Software Foundation, por lo tanto, considera la licencia PHP como parte del software libre.

1.5.1 Framework

El concepto framework se emplea en muchos ámbitos del desarrollo de sistemas software, no solo en el ámbito de aplicaciones Web. Podemos encontrar framework para el desarrollo de aplicaciones médicas, de visión por computador, para el desarrollo de juegos, y para cualquier ámbito que pueda ocurrirnos. En general, con el término framework, nos estamos refiriendo a una estructura software

compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, un framework se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta. (Javier, 2017)

A continuación se muestra un cuadro comparativo de sus características principales en los últimos tiempos:

1.5.2 Comparativa framework PHP

| LARAVEL | CAKEPHP | CODEIGNITER | SYMFONY | YII2 |
|---|---|---|--|--|
| <p>Laravel fue introducido en el mundo del desarrollo hace justo 4 años, y solo en un lapso corto de tiempo capturó la atención de los programadores de todo el mundo debido a su extrema elegancia.</p> <p>Pros</p> <p>Reducción de costos y tiempos en el desarrollo y posterior mantenimiento de la aplicación creada.</p> <p>Uso del sistema de plantillas Blade para acelerar las tareas de compilación, y los usuarios pueden incluir nuevas características de una manera sencilla.</p> <p>El "Bundled modularity" permite que el código sea</p> | <p>Cuando se lanzó allá por el 2005, CakePHP era sólo un proyecto modificado en Ruby on Rails. Ahora se ha convertido en un gran framework para PHP con características espléndidas.</p> <p>Pros</p> <p>Gran apoyo de la comunidad, que puede despejar cualquier duda sobre el desarrollo de cualquier web de una manera eficaz.</p> <p>ORM muy útil que ayuda a los desarrolladores a crear excelentes consultas, así como códigos.</p> <p>Disponibilidad de fantásticos plugins para mantener los códigos</p> | <p>CodeIgniter se lanzó sólo un año después del lanzamiento de CakePHP. Al igual que Cake, se ha ganado muchos seguidores en un corto periodo de tiempo.</p> <p>Pros</p> <p>Es un framework que es simple de entender. Incluso un principiante encuentra esta plataforma ideal para el desarrollo de una tarea.</p> <p>Muy buena documentación.</p> <p>Alta velocidad, no hay retrasos.</p> <p>Como CakePHP, CodeIgniter cuenta con</p> | <p>Aunque fue lanzado a finales de 2005, Symfony como plataforma llegó a tener protagonismo después de 2007. Se convirtió lentamente en un amplio framework MVC para construir aplicaciones web.</p> <p>Pros</p> <p>Flexibilidad en la creación de proyectos.</p> <p>Los desarrolladores cuentan con la opción de elegir su propio ORM.</p> <p>Los componentes Symphony se pueden incorporar a proyectos mucho más grandes como Drupal.</p> <p>Contras</p> | <p>Similar a Laravel, Yii tardó mucho tiempo en convertirse en todo un referente dentro del mercado de framework para PHP. Ahora se presenta como una plataforma perfecta para muchos desarrolladores con la que crear aplicaciones web.</p> <p>Pros</p> <p>Para hacer tareas como un buscador, una paginación o grids, los desarrolladores pueden recurrir a las funciones Ajax incorporadas.</p> <p>Códigos poco pesados.</p> <p>Seguridad excelente, y grandes extensiones.</p> |

| | | | | |
|--|--|---|--|--|
| reutilizado sin muchas molestias. | limpios, así como elegantes. | un gran apoyo de la comunidad. | La documentación carece de amplias referencias. | Framework muy fácil de aprender. |
| ORM que es fácil de entender. Es por eso que, la creación de relaciones de bases de datos es tan simple. | Contras Códigos extensos. Cuando comparamos CakePHP con otros framework, los códigos para hacer una tarea en particular se ven enormes. | Contras La librería no es tan exhaustiva o refinada como en otras plataformas. | El mecanismo de seguridad de Symfony es difícil de utilizar. Parsear archivos en Symfony es un pelín complicado. | Contras Le falta el apoyo de extensiones muy necesarias para crear aplicaciones complejas. |
| Un CLI Artisan que comprende herramientas avanzadas para hacer tareas y migraciones. | Es una de las plataformas más lentas en cuanto a rendimiento. | La falta de separación modular de códigos. Es difícil mantener o modificar códigos. | | La disponibilidad de expertos en este framework no es tan alta si la comparamos con otras plataformas. |
| Documentación espléndida | | | | El apoyo de la comunidad no es tan masivo como el que los desarrolladores obtienen con CakePHP. |
| Característica adicional para el enrutamiento inverso. | La carga de los códigos. | | | |
| Contras | | | | |
| Los desarrolladores amateurs suelen liarse extendiendo códigos y clases. | | | | |

Fuente: Autor

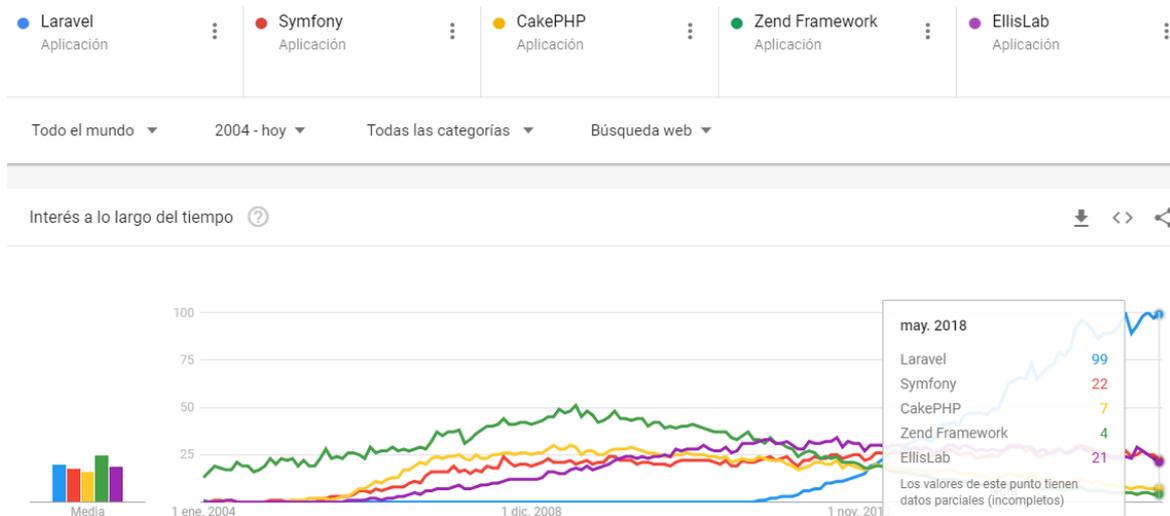


Figura 3. Framework de php
Fuente:<https://trends.google.es/trends/explore?>

Luego de revisar el cuadro comparativo y la figura 3 de los framework podemos apreciar que Laravel es el framework más adecuado para llevar a cabo este proyecto debido a las propiedades que nos brinda.

A continuación, se menciona la evolución del framework Laravel en sus distintas versiones:

Tabla 4. Versiones de framework Laravel

| VERSIONES | CARACTERÍSTICAS |
|-----------------|---|
| Laravel 4.1 | <ul style="list-style-type: none"> ✓ Posee un poderoso conjunto de librerías ✓ Eloquent ORM ✓ Soporte cache ✓ Soporte para MVC ✓ Autenticación simple ✓ Sistema de ruteo (RESTfull) ✓ Blade, Motor de plantillas |
| Laravel 5.0 LTS | <ul style="list-style-type: none"> ✓ Peticiones Fluent ✓ Adopta las especificaciones PSR-2 y PSR-4 ✓ Almacenamiento en cache de rutas y middleware ✓ Inyección de dependencias en rutas y controladores ✓ Autenticación Scaffolding ✓ Socialite |

| | |
|-----------------|--|
| Laravel 5.2 | <ul style="list-style-type: none"> ✓ Estructura de carpetas ✓ Cambios en Blade ✓ Contracts ✓ Comandos y eventos ✓ Soporte para multiples drivers de autenticación ✓ Model binding implícito ✓ Global scopes de Eloquent mejorados ✓ Generador del sistema de autenticación incluido por defecto en Laravel 5.0 (login, logout, etc.) ✓ Middleware para limitar el número de peticiones por dirección IP por tiempo ✓ Posibilidad de agrupar varios middlewares ✓ Mejoras en la validación de array de formularios |
| Laravel 5.5 LTS | <ul style="list-style-type: none"> ✓ Frontend Presents ✓ Utiliza PHP 7 ✓ Incorpora Whoops ✓ Automatic Package Discovery ✓ Validacion del Request simplificado ✓ Customizables Reglas de validación ✓ Debuggear con dump ✓ Generar Factory por terminal ✓ Route::view y Route::redirect |

Fuente: (Gonzales, 2017)

Se puede evidenciar que la versión 5.5 LTS es una versión estable y cuenta con un gran potencial para llevar a cabo el presente proyecto debido a la variedad de procesos ágiles que se puede manejar al momento de desarrollar la aplicación.

1.6 Laravel

Laravel es un framework de código abierto más fáciles de interpretar para el lenguaje de PHP, Es simple, potente y cuenta con una interfaz elegante de usar, fue creado en el año 2011, creado por Taylor Otwell, facilitando a los desarrolladores a la creación de nuevas aplicaciones Web de forma rápida con un código limpio, ordenado y estructura.

Laravel surge rescatando a PHP, aprovechando el máximo potencial de las características de las últimas versiones, como el uso del paradigma POO (programación orientada a objetos), Laravel se adapta tanto a proyectos sencillos y robustos, de forma ordenada y estructurada, haciendo que las aplicaciones sean más fáciles de administrar, destacando su profesionalidad y velocidad en el desarrollo, frente al resto de opciones como: routes, closures, helpers, middlewares, internacionalización, objetos ORM, motor de plantillas Blade.

1.6.1 Laravel 5.5 LTS

De acuerdo con Laravel (2017) menciona que:

Con Laravel 5.5 LTS unos siete meses después del lanzamiento de la versión anterior 5.4, apareció una actualización para el popular framework y, por primera vez desde la versión 5.1, una nueva versión LTS. La atención se centra en una serie de nuevas características, mejoras y correcciones de errores, tales como:

- ✓ Paquete Whoops: rastreos de pila para la depuración.
- ✓ Collection Dumping: también una herramienta auxiliar de depuración.
- ✓ Renderización de excepciones: las excepciones ahora pueden representar una respuesta en la medida en que definen un método de "respuesta" pública, es decir, en laravel 5.5 simplemente puede lanzar la excepción y puede responder sin ninguna lógica adicional anclada en el controlador.
- ✓ Responsable de interfaz: un nuevo Tipo de retorno permite que los objetos que son devueltos por un controlador o un cierre de ruta para convertir una instancia HTTP Request Validación de Método: En versiones anteriores laravel era la instancia solicitud al `$this->validate()` método en un controlador se da a conocer; Ahora el objeto de solicitud puede ser validado.
- ✓ Objetos y cierres de reglas de validación personalizadas: una buena alternativa para crear reglas personalizadas con las directivas Validator :: extend Auth y Guest Blade: las "innovaciones de Blade" agregan soporte para -ifdeclaraciones personalizadas simplistas en vistas.
- ✓ Ajustes preestablecidos frontend: al crear un nuevo proyecto, Vue.js ahora está configurado como el dispositivo frontend por defecto.

1.6.2 Arquitectura MVC

Es uno de los patrones fundamentales en el diseño de aplicaciones web es. Este patrón propone la separación en tres capas:

- ✓ Vista: Es todo lo que se refiera a la visualización de la información: el diseño, colores, estilos y la estructura visual en sí de nuestras páginas.
- ✓ Modelo: Es el responsable de la conexión a la base de datos y la manipulación de los datos mismos.
- ✓ Controlador: Su responsabilidad es procesar y mostrar los datos obtenidos por el Modelo. Es decir, trabaja de intermediario entre los otros dos, encargándose también de la lógica de negocio.

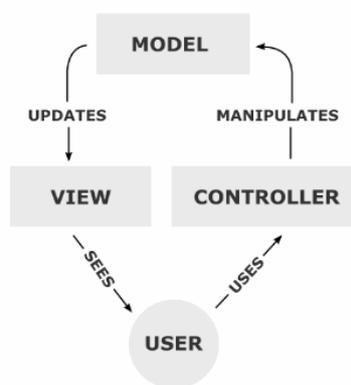


Figura 4. Modelo, vista, controlador MVC

Fuente: https://ajgallego.gitbooks.io/laravel-5/content/images/web_laravel/esquema_mvc.png

El cliente envía una petición, ésta es receptada por el Controlador quien ejecuta las validaciones necesarias, procesa los datos y lógica de negocio asociadas a esa petición del cliente. Luego el Controlador envía datos al Modelo, para que interactúe con la base de datos, dependiendo de la solicitud del usuario, para finalmente enviarlos a la Vista a fin de ser mostrados nuevamente al cliente a través de una respuesta.

1.6.3 Arquitectura MVC Laravel

“Trygve Reenskaug inventó el patrón Model View Controller (en adelante MVC) a finales de los 70 para su posterior implementación en Smalltalk-80. La idea de este patrón era ser usado en pequeños componentes. Imaginemos un checkbox, sí, uno como éste”. (Palacios, 2017)

De acuerdo con Palacios (2017) menciona que este checkbox posee:

- ✓ Un modelo: que se encarga de almacenar los datos del checkbox, en este caso un solo dato que puede ser verdadero (checked) o falso (not checked).
- ✓ Una vista: la representación gráfica del checkbox, es decir una cajita, la vista por supuesto toma los datos del modelo para saber cómo debe ser representada (con o sin marca). El usuario por supuesto ve e interactúa con la aplicación a través de la vista.
- ✓ Un controlador: El controlador se encarga de tomar una acción del usuario (por ejemplo, el usuario presionó el checkbox) y actualizar el modelo (en este caso ahora el modelo contendrá “verdadero” porque el checkbox está presionado o volverá a “falso” si se presiona otra vez).
- ✓ Una vez que se realicen cambios en el modelo, éste actualiza la vista.

A continuación, se muestra la arquitectura del MVC Laravel:

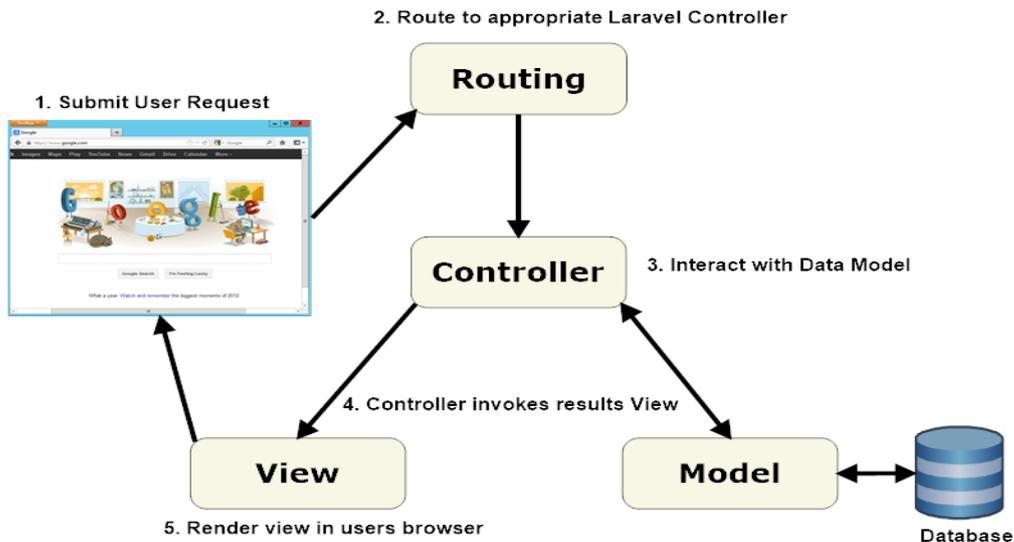


Figura 5. Arquitectura MVC Laravel

Fuente: <https://stackoverflow.com/questions/33148455/mvc-in-php-framework>

De acuerdo con Stackoverflow (2016) menciona que:

Laravel usa el patrón MVC (Modal View Controller) en arquitectura de software, que separa los datos y la lógica de negocios.

1.6.2 Diagrama de secuencia de Laravel

Se propone el uso del patrón MVC (Modelo-Vista-Controlador) para el desarrollo de este proyecto. El modelo MVC ayuda a crear aplicaciones que separan los diferentes aspectos de la aplicación (lógica de entrada, lógica de negocios y lógica de la interfaz de usuario), a la vez que proporciona un bajo acoplamiento entre estos elementos. Este patrón especifica dónde se debería encontrar cada tipo de lógica en la aplicación. La lógica de la interfaz de usuario pertenece a la vista. La lógica de entrada pertenece al controlador. La lógica de negocios pertenece al modelo. Esta separación nos ayuda a administrar la complejidad al compilar una aplicación, ya que le permite centrarse en cada momento en un único aspecto de la implementación.

El acoplamiento bajo entre los tres componentes principales de una aplicación MVC también favorece el desarrollo paralelo. Por ejemplo, un desarrollador de software puede trabajar en la vista, un segundo desarrollador puede ocuparse de la lógica del controlador y un tercero se puede centrar en la lógica de negocios del modelo.

Como el framework de desarrollo propuesto para este proyecto es Laravel, el uso de este patrón parece ser la decisión más correcta. El modelo de secuencia de la aplicación en Laravel es el siguiente:

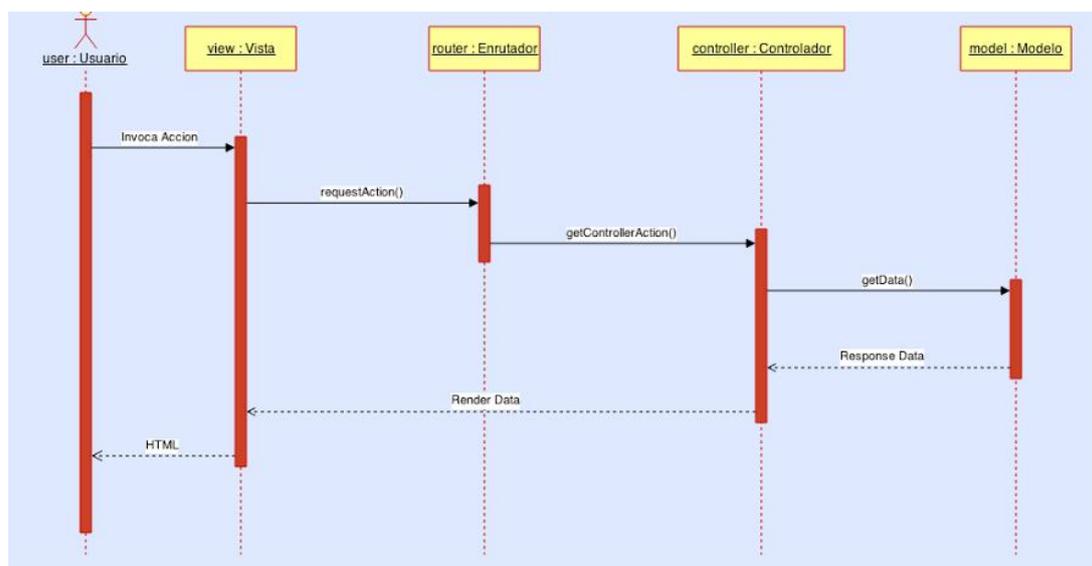


Figura 6. Diagrama de secuencia de Laravel

Fuente: <https://projectmdd.files.wordpress.com/2014/12/vista-1c3b3gica.png>

1.6.3 Estructura del proyecto

La estructura del proyecto está basada en carpetas y archivos que son creados al momento de crear la aplicación, los directorios más utilizados son:

- ✓ **App:** Lugar para desarrolladores BackEnd, donde poseemos el control de Rutas, Controladores que mantienen la lógica del negocio, Modelos que hacen las peticiones a la base de datos, Middlewares permitiendo la filtración de peticiones http con seguridad.
- ✓ **Resources:** Lugar de los desarrolladores FrontEnd, vistas HTML, motor de plantillas Blade para interactuar con PHP, proyectando información necesaria a la aplicación o de la base de datos.

En la tabla 5 se presentará la estructura completa de Laravel 5.5:

Tabla 5. Estructura de Laravel

| Carpeta | Descripción |
|----------------|--|
| App | BackEnd: routes, controllers, models, middleware's, requests, providers, services. |
| Bootstrap | Archivos de arranque |
| Config | Archivos de configuración |
| Database | Migrations y seeders |
| Public | CSS, JavaScript, imágenes, index.php |
| Resources | FrontEnd: Vistas, plantillas, idioma |
| Storage | Archivos temporales, caché, log de errores |
| Test | Pruebas con PHPUnit o PHPSec |
| Vendor | Librerías |

Fuente: (Laravel, 2017)

Como podemos observar la estructura que maneja Laravel es completamente robusta porque mejora enormemente el rendimiento de los servidores. y en la velocidad reduciendo la carga del CPU al 50%.

1.7 Lenguaje JS

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas. Una página web dinámica es aquella que incorpora efectos como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario. (LibrosWeb, 2017).

“Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios”. (LibrosWeb, 2017).

1.7.1 Framework JS

Los framework web de JavaScript pueden ser una solución muy útil para el desarrollo rápido de aplicaciones web. Sirven de estructura para aplicaciones de una sola página (SPA), permiten a los desarrolladores preocuparse menos de la estructura del código y el mantenimiento, y centrarse en la funcionalidad. (CampusMVP, 2017)

Las ventajas de usar framework de JavaScript de acuerdo a Campus MVP (2017) son:

- ✓ Eficiencia y velocidad de desarrollo: antiguamente crear un proyecto desde cero podría llevar varios meses y, ahora gracias al uso de funcionalidades pre programadas, estructuras predefinidas y patrones probados (como MVVM), el desarrollo puede ser mucho más rápido. (CampusMVP, 2017)
- ✓ Mantenimiento e incorporación de miembros a los equipos: gracias a que el uso de un framework fuerza a usar ciertas arquitecturas y formas de proceder, es mucho más fácil incorporar personas a un equipo de trabajo, pues sabrán entender el código y retomarlo de manera más rápida. Algunos, no obstante, son más "estrictos" que otros en cuanto a lo que dejan hacer o no a los programadores. Por ejemplo, Angular es muy estricto en cuanto cómo desarrollar, por lo que resta algo de libertad a los programadores, y al mismo tiempo facilita la incorporación al proyecto, por lo que puede ser ideal para las empresas. (CampusMVP, 2017)
- ✓ Seguridad: los mejores framework de JavaScript están diseñados con la seguridad en mente, y facilitan que nuestras aplicaciones sean más seguras. Por

supuesto, la última palabra la tiene el desarrollador, que debe también pensar en la seguridad todo el tiempo mientras programa. Pero nos facilitan mucho la vida a este respecto también. (CampusMVP, 2017)

✓ Coste: la práctica totalidad de los framework son de código abierto y gratuito. Dado que ayuda a los programadores a crear soluciones personalizadas más rápidamente, el precio final de cualquier aplicación web también será menor. (CampusMVP, 2017)

1.7.2 Comparativa framework JS

A continuación se presenta la tabla 6 la cual resume las características de los framework Java Script:

Tabla 6. Características de framework Java Script

| Características | Backbone.js | Ember.js | Angular |
|---------------------------------|--|------------------------------------|------------------------------|
| Multi-navegador | IE 6+, Firefox 2+, chrome, Opera, Safari | IE 9+ FF 11+ Chrome 17+, Safari 5+ | IE8+ Chrome, Safari, Firefox |
| Diseño web adaptable | x | x | x |
| Open Source | x | x | x |
| Integración con otros framework | x | x | |
| Soporte a versión | x | x | x |
| Tamaño | 50 KB 120KB | 308 KB | 81 KB |
| Estándares HTML5 | x | x | x |
| MVC | x | x | x |
| Comunidad | x | no | x |
| Documentación | x | Muy poca documentación | x |
| Curva de aprendizaje | x | Curva de aprendizaje abrupta | x |
| Navegador Móvil | x | x | x |

Fuente: (JavaScript, 2018)

De acuerdo la comparativa anterior podemos observar que angular es un framework con gran capacidad y a la vez más liviano Angular.

1.7.3 Angular

A continuación se muestra en la tabla las principales características que llevaron a angular un nivel superior:

Tabla 7. Características de Angular

| VERSIONES | CARACTERÍSTICAS |
|------------|--|
| Angular JS | <ul style="list-style-type: none"> ✓ Usa el patrón de diseño habitualmente encontrado en el desarrollo web MVC, aunque en una variante llamada a veces MV* y a veces MVVM. Esto, junto con otras herramientas disponibles en Angular nos permite un desarrollo ordenado, sencillo de realizar y sobre todo más fácil de mantener en un futuro. Angular JS está apoyado por Google y cada día más desarrolladores están adoptándolo, lo que nos da una idea del prometedor futuro de la librería. |
| Angular 2 | <ul style="list-style-type: none"> ✓ Desarrollo Móvil: El desarrollo de aplicaciones de escritorio es mucho más fácil cuando primero se manejan los problemas de rendimiento en el desarrollo móvil. ✓ Modularidad: Para desarrollar una nueva funcionalidad esta se empaqueta en un módulo, produciendo un núcleo más ligero y más rápido. ✓ Compatibilidad: Es compatible con los navegadores más modernos y recientes. ✓ Angular 2 recomienda usar TypeScript un lenguaje de programación libre y de código abierto desarrollado y mantenido por Microsoft, considerado como un superconjunto de JavaScript, que esencialmente añade tipado estático y objetos basados en clases. |
| Angular 4 | <ul style="list-style-type: none"> ✓ Módulo de animación separado ✓ Uso de StrictNullChecks de TypeScript ✓ Integración de Angular Universal como módulo de Angular ✓ Mejoras de rendimiento gracias a FESM |

| | |
|-----------|--|
| Angular 5 | <ul style="list-style-type: none">✓ Transformaciones de TypeScript✓ Preservar los espacios en blanco✓ Build Optimizer✓ API Angular Universal State Transfer y soporte DOM✓ Internacionalización de tuberías para números, fechas y monedas✓ Se reemplaza ReflectiveInjector con StaticInjector✓ Mejoras de velocidad de zonas✓ exportas✓ Angular Forms añade updateOn Blur / Submit✓ Formularios basados en plantillas✓ Formularios reactivos✓ RxJS 5.5 |
|-----------|--|

Fuente: (Angular, 2017)

De acuerdo a la tabla anterior podemos notar que la versión de angular 5 es una de las herramientas más potentes ha lado de FrontEnd. Que junto con otras herramientas disponibles en Angular nos permite un desarrollo ordenado, sencillo de realizar y sobre todo más fácil de mantener en un futuro.

1.7 Angular 5

Angular 5 es el framework para JavaScript estándar para crear webs SPA. Es uno de los framework más populares para desarrollar aplicaciones modernas y escalables en el lado del cliente. Si quieres ser desarrollador FrontEnd o full-stack, o incluso si ya te dedicas al desarrollo web, es casi obligatorio aprender Angular 5. (Robles, 2018)

Angular 5 ha dado un salto de calidad con respecto a las versiones anteriores del framework, como Angular JS. Ahora utilizaremos TypeScript para definir nuestras clases, propiedades y métodos, el código es mucho más limpio y se ha mejorado el funcionamiento general del framework.

1.7.1 Principales características de Angular 5

Velocidad y rendimiento

- ✓ “Generación de código: Angular convierte tus plantillas en código altamente optimizado para las máquinas virtuales de JavaScript de hoy en día, ofreciéndote todas las ventajas del código escrito a mano con la productividad de un framework”. (Angular, 2017)
- ✓ “Universal: Ejecuta la primera vista de tu aplicación en node.js, .NET, PHP, y otros servidores para renderizado de forma casi instantánea obteniendo solo HTML y CSS. También abre posibilidades para la optimización del SEO del sitio”. (Angular, 2017)
- ✓ “División del código: Las aplicaciones de Angular se cargan rápidamente gracias al nuevo enrutador de componentes. Éste ofrece una división automática de códigos para que los usuarios sólo carguen el código necesario para procesar la vista que solicitan”. (Angular, 2017)

Productividad

- ✓ “Plantillas: Permite crear rápidamente vistas de interfaz de usuario con una sintaxis de plantilla simple y potente”. (Angular, 2017)
- ✓ “Angular CLI: Las herramientas de línea de comandos permiten empezar a desarrollar rápidamente, añadir componentes y realizar test, así como pre visualizar de forma instantánea la aplicación”. (Angular, 2017)
- ✓ “IDEs: Obtén sugerencias de código inteligente, detección de errores y otros comentarios en la mayoría de los editores populares e IDEs”. (Angular, 2017)

Historia completa del desarrollo

- ✓ “Testing: Utiliza Karma para realizar pruebas unitarias, y Protractor para realizar pruebas end-to-end de forma rápida y estable”. (Angular, 2017)
- ✓ “Animación: Permite crear animaciones complejas y de alto rendimiento con muy poco código a través de la intuitiva API de Angular”. (Angular, 2017)
- ✓ “Accesibilidad: Posee características para crear aplicaciones accesibles con los componentes disponibles para ARIA”. (Angular, 2017)

1.7.1 Data binding en Angular 5

El enlace de datos en una aplicación Angular es la conexión automática de datos entre el modelo y los componentes de la vista, permitiéndole al modelo tratar como la única fuente verídica en la aplicación, y la vista es una proyección del módulo en todo momento ya sea cuando el modelo cambia, la vista refleja el cambio y viceversa.

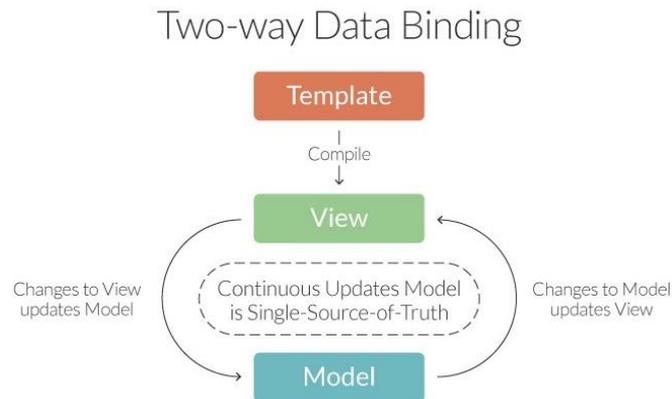


Figura 7. Data binding en angular

Autor: <https://www.ondho.com/wp-content/uploads/2015/03/two-way.jpg>

Angular crea plantillas que realizan un refresh apenas guarda algún cambio en las líneas de comandos o componentes, dando lugar a vistas dinámicas. Ejemplo:

```
<input ng-model="laSirena" type="text" placeholder="La Sirena">
<h1>Hello {{laSirena}} </h1>
```

1.7.2 Módulos.

Un módulo es un contenedor diseñado para diferentes partes de una aplicación: como son los controladores, servicios, filtros, directivas, etc.

La API module de Angular permite declara un módulo usando el método Angular. Module ('myAppLaSirena', []);

Los módulos especifican de forma declarativa como se debe iniciar una aplicación, ventajas del enfoque:

- ✓ Proceso declarativo sencillo de entender
- ✓ Empaquetamiento de módulos reutilizables.
- ✓ Ejecución de módulos en cualquier orden o incluso en paralelo.

- ✓ Test unitarios que cargan módulos relevantes, mejorando la velocidad.
- ✓ Test de extremo a extremo pueden usar módulos para anular la configuración.

1.7.3 Scopes

Es un objeto que se refiere al modelo de la aplicación, en un contexto de ejecución para expresiones Scopes, los cuales se organizan en una estructura jerárquica parecida a la de DOM de la aplicación.

Scope propiedad de datos que se utiliza para fines de depuración, la ubicación de la directiva define el lugar donde está adjunto el ámbito de DOM, sirven como conexión entre el controlador y la vista, permitiendo que la aplicación presente los resultados al usuario.

1.7.4 Controladores

Según Angular JS (2018) expresa que: “Un controlador se define mediante una función de constructor de JavaScript que se utiliza para aumentar el alcance de Angular” (p.1).

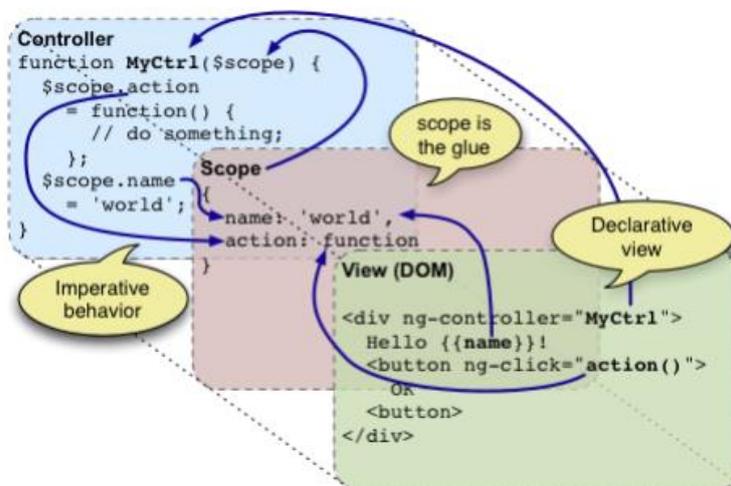


Figura 8. Controladores

Los controladores son los encargados de iniciar y modificar la información que contienen los Scopes en función de las necesidades de la aplicación, Cuando se crea un nuevo controlador en la aplicación Angular le adjunta un nuevo Scope.

1.7.5 Directivas

Las directivas son los marcadores en un elemento DOM, el cual adjunta un comportamiento específico a ese elemento DOM como por ejemplo a través de llamadas a los eventos o incluso para transformar el elemento DOM y sus hijos.

Las directivas son formas de extender la sintaxis del HTML y darle el comportamiento que se desee. Estas directivas se pueden crear a nivel de un elemento, atributo, clase y comentario.

Ejemplo:

```
<div ng-app='MyAppLaSirena'>  
  <div> <input type='text' placeholder='La Sirena' focusable /></div>  
</div>
```

1.8 Gestores de dependencias

1.8.1 Node

Según Node (2018) menciona que:

Node.js es un entorno de ejecución para JavaScript constituido con el motor de JavaScript V8 de Chrome. Node.js usa un modelo de operaciones E/S sin bloqueo y orientado a eventos, que lo hace liviano y eficiente. El ecosistema de paquetes de Node.js, NPM, es el ecosistema más grande de librerías de código abierto del mundo. (p.3)

Las características más importantes de Node.js:

- ✓ JavaScript sin restricciones, debido a que su ejecución está a lado del servidor y no tenemos preocuparnos de si nuestro código será compatible o no con los clientes.
- ✓ Programación Asíncrona, la cual permite liberar procesos de modo que los recursos estén disponibles para hacer otras ejecuciones durante el tiempo de espera de un proceso o hilos de ejecución.

1.8.2 Composer

Según Getcomposer (2018) menciona que: “Composer es una herramienta para la administración de dependencias de PHP. Le permite declarar las bibliotecas de las que depende su proyecto y las administrará (las instalará / actualizará” (p.1). Composer es un gestor de paquetes, capaz de instalar librerías que requiera un proyecto con las versiones que se demanden, en caso de que las librerías tengan dependencia con otra este se encarga de descargar todo lo necesario para que la nueva librería sea instalada.

CAPITULO II

2. DESARROLLO

2.1 Pre-juego Planificación

Las etapas será analizada una vez durante el desarrollo de cada Sprint. Sin embargo, antes de iniciar con las iteraciones, es necesario realizar una definición macro de las iteraciones, esto es, realizar una planificación inicial, que permita identificar el propósito de cada iteración y definir en forma global lo que se realizará en cada sprint. Este análisis inicial se denomina "Sprint 0".

Antes de definir el número y los objetivos de cada Sprint a desarrollar durante la ejecución del presente proyecto, es necesario definir ciertas actividades

2.1.1 Análisis de Procesos

Es necesario definir los alcances funcionales del sistema informático que se implementará como parte de la ejecución del proyecto. Para definir los alcances funcionales, se analizarán los procesos ya definidos internamente por la Junta de agua.

- ✓ Macroproceso – Administración: Es el conjunto de procesos operativos que la junta de agua ofrece a sus contribuyentes. Estos pueden ser: Ingreso de materiales, mantenimiento de la materia prima, control de sanciones, Ingreso de lecturas de los medidores, entrega de facturas de consumo (Reparte) y facturación, entre otros.
- ✓ Proceso – Lectura de medidores de consumo: Proceso mediante el cual se recolecta información correspondiente al consumo registrado en los medidores de los abonados al servicio brindado por los clientes de la Junta de Agua, principalmente empresas de servicios públicos (JAAP).

A continuación en la siguiente figura se da a conocer la administración completa del sistema de facturación para la Junta de agua san José de Chorlaví:

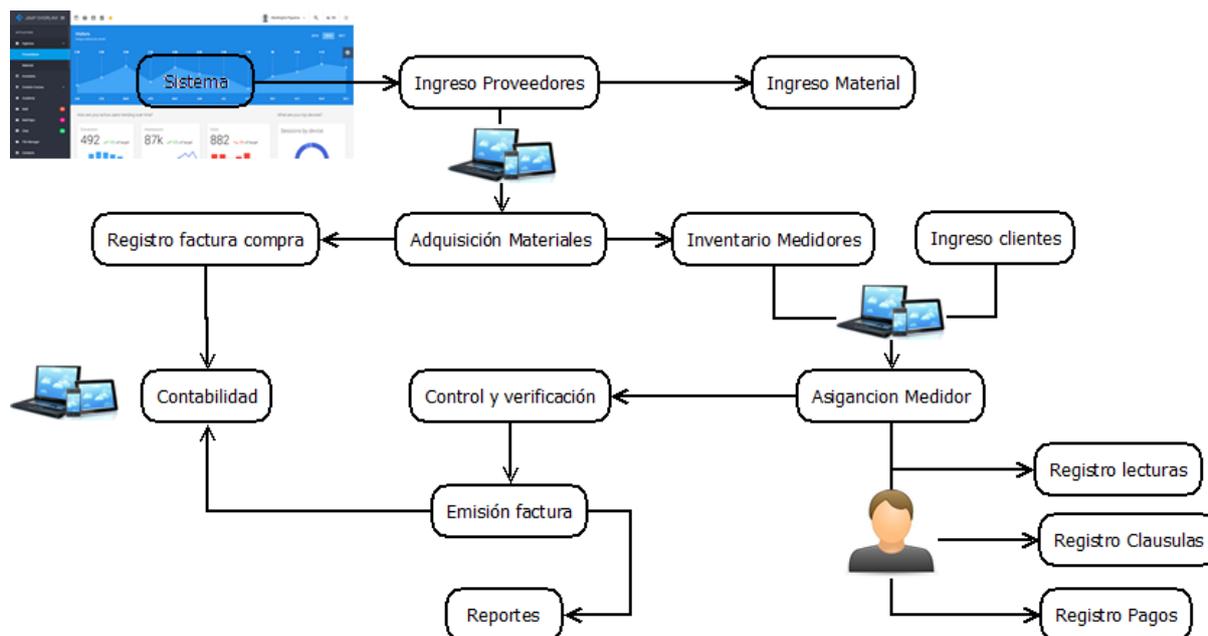


Figura 9. Proceso de administración del sistema de facturación

En la Figura 9 se muestran los procesos de administración del sistema de facturación y de igual manera los subprocesos se encuentra descompuestos en un diagrama que permita analizarlos detalladamente a cada uno de ellos.

2.1.2 Simbología utilizada en los diagramas

En la tabla 8 daremos a conocer la simbología para el desarrollo de los diagramas de caso de uso, relación de cada uno de los roles del sistema de Facturación para la Junta de Agua de San José de Chorlaví.

Tabla 8. Simbología utilizada en los diagramas

| Nombre | Símbolo | Función |
|----------|------------|--|
| Terminal | | Representa el inicio y fin de un programa. También puede representar una parada o interrupción programada que sea necesaria realizar en un programa. |
| Proceso | Process_01 | Cualquier tipo de operación que pueda originar cambio de valor, formato o posición de la información. |
| Decisión | | Indica operaciones lógicas o de comparación entre datos. |

| | | |
|---|--|--|
| Indicador de dirección o línea de flujo |  | Indica el sentido de la ejecución de las operaciones. |
| Indicador de dependencia |  | Indica que un proceso depende de un recurso (Salida). |
| Salida |  File_01 | Recurso utilizado para la ejecución de un proceso o generado durante la ejecución del proceso. |
| Datos Almacenados | Resource_0 1 | Representa un conjunto de datos, generalmente en Base de datos. |

En la tabla 9. Describiremos requisitos funcionales y no funcionales para el sistema a implementar con cada uno de los procesos.

Tabla 9. Requerimientos funcionales y no funcionales

| Módulo | Requisitos funcionales | Requisitos no funcionales |
|--------|--|--|
| 1.1 | | El sistema brindará seguridad a los usuarios registrados, con diferentes niveles de acceso. |
| 1.2 | El sistema permitirá registrar a los usuarios del sistema. | El sistema funcionará en equipos con cualquier navegador. |
| 1.3 | El sistema permitirá registrar los equipos utilizados para los procesos de recolección de datos en campo (celulares). | |
| 1.4 | El sistema permitirá asignar al usuario responsable para la lectura de medidores. | |
| 2.1 | El sistema permitirá cargar los datos de un plan a la base de datos desde un archivo de texto (Archivo plano separado por comas [columna1], [columna2]). | Este proceso debe ser eficiente, debido a que actualmente toma demasiado tiempo. Es necesario implementar un método rápido para cargar los datos en la DB. |

- 2.2 El sistema permitirá consultar los datos del plan cargado.
- 2.3 El sistema permitirá sincronizar las lecturas asignadas a un usuario, para su respectiva gestión en campo.
- 2.4 El sistema permitirá sincronizar las lecturas gestionadas por un usuario para actualizar los datos requeridos en la base de datos.
- 3.1 El sistema permitirá recolectar la información en campo, utilizando la aplicación web. La aplicación web funcionará cualquier dispositivo móvil.

| Módulo | Requisitos funcionales | Requisitos no funcionales |
|--------|--|--|
| 2.6 | El sistema permitirá sincronizar los datos de las lecturas gestionadas y por gestionar utilizando la interfaz USB. | |
| 4.1 | El sistema permitirá sincronizar los datos de las lecturas gestionadas y por gestionar utilizando la red de datos. | La sincronización de datos se realizará utilizando un plan de datos de internet de los disponibles por las operadoras de telefonía celular, el proveedor no debe ser un limitante. |
| 2.7 | El sistema permitirá identificar quien gestiona las lecturas. | |
| 2.8 | El sistema permitirá modificar las lecturas gestionadas en campo desde la aplicación de escritorio. | |
| 2.9 | El sistema almacenara los datos de los planes gestionados permanentemente. | |
| 5 | El sistema permitirá obtener los siguientes reportes. | |
| 5.1 | *Reportes de asignación de lecturas (Distributivo Plan Actualizado con número de lecturas por lectrista) | |

| | |
|-----|---|
| 5.2 | *Reportes de asignación de lecturas (histórico del anterior) |
| 5.3 | *Reportes de lecturas realizadas por cada lectorista. (Detalle del Distributivo Plan Actualizado con número de lecturas por lectorista) |
| 5.4 | *Reporte "Plan Rendido" |
| 5.5 | Estado de sincronización (Resumen del plan rendido) |

Fuente: Autor

La lista de requerimientos ha sido socializada con los beneficiarios del sistema.

La columna identifica tentativamente el módulo en el cual se incluirán la funcionalidad. Los módulos serán a su vez tentativamente el alcance propuesto para los Sprint. Los módulos identificados durante el análisis son los siguientes:

- 1.-M. Administración
- 2.-M. Secretaria (Procesamiento de la información)
- 3.-M. Gestión en campo o toma de lecturas (Celulares)
- 4.-M. Sincronización en línea. (Celulares)
- 5.-M. Reportes.

2.1.3 Alcance del software

De acuerdo con el análisis de los procesos y las funcionalidades identificadas, el software a desarrollarse tendrá los siguientes alcances.

- ✓ **Módulo de administración permitirá:**
 - Crear, eliminar, modificar usuarios, equipos y perfiles para controlar el acceso a los diferentes módulos del sistema.
- ✓ **Módulo de secretaria permitirá:**
 - Cargar y procesar la información proporcionada y requerida por el cliente de la junta de agua.
 - Ingresar los datos brindados por parte del lector, los cuales fueron tomados del consumo del servicio por medio del medidor.
 - Consultar reportes de asignación (Requerimientos Funcionales/No

Funcionales), los cuales permitan dar seguimiento del proceso.

- Reporte de planes Rendidos

✓ **Módulo de Gestión en campo permitirá:**

- Llevar a cabo el proceso de recolección y validación de información con dispositivos móviles.

✓ **Módulo de Sincronización en línea permitirá:**

Utilizar la red de datos proporcionada por una operadora de telefonía celular local para:

- Descargar los datos de las lecturas asignadas al lectorista, para su respectiva gestión.
- Actualizar los datos de las lecturas gestionas por el Lectorista, en el servidor central de la JAAP.
- Consultar reportes de estado de sincronización los cuales permitan dar seguimiento del proceso.

2.1.4 Definición del Backlog del Producto.

El producto final debería tener. Como dice la metodología, para el presente proyecto se ha elaborado el Backlog del producto, identificando las funcionalidades, realizando una estimación del tiempo requerido para su implementación.

De acuerdo con el estudio de los procesos y los requerimientos identificados durante la etapa de análisis, el Backlog del Producto para el presente proyecto se encuentra definido en la siguiente tabla:

Tabla 10. Backlog del Producto

| ID | Nombre | Importancia | Tiempo o estima do (semana s) | Comentarios |
|-----------|--------------------------|--------------------|--|--|
| 1 | Módulo de Administración | 700 | 3 | Funcionalidad para administración de recursos. |

| | | | | |
|---|----------------------------|------|---|---|
| 2 | Módulo de Secretaría | 800 | 3 | Funcionalidad para gestionar información (in-out) de los clientes de JAAP. |
| 3 | Módulo de Gestión en Campo | 900 | 3 | Aplicación para dispositivos móviles, con funcionalidad para recolección y almacenamiento de información en campo con dispositivos móviles. |
| 4 | Sincronización en línea | 1000 | 3 | Agilizar el proceso de sincronización de datos. |

Fuente: Autor

La importancia está estimada de acuerdo con las necesidades del Product y está cuantificada con números enteros entre 0 y 1000.

El tiempo requerido para el desarrollo está dado principalmente por dos factores: El tiempo recomendado por SCRUM para la ejecución de cada iteración y la experiencia en la implementación de componentes de software del equipo de trabajo. En este caso cada módulo será implementado en una iteración independiente.

2.2 Juego Desarrollo de Sprint

En esta etapa se definirá en forma general, el modelo de datos que serán implementados en los siguientes Sprints. Este modelo será la guía de referencia para la implementación detallada de cada módulo.

2.2.1 Diseño de la base de datos

La administradora de agua potable requiere contar con una base de datos confiable y segura, ya que en la misma cuenta con la información financiera de todos los contribuyentes de la institución.

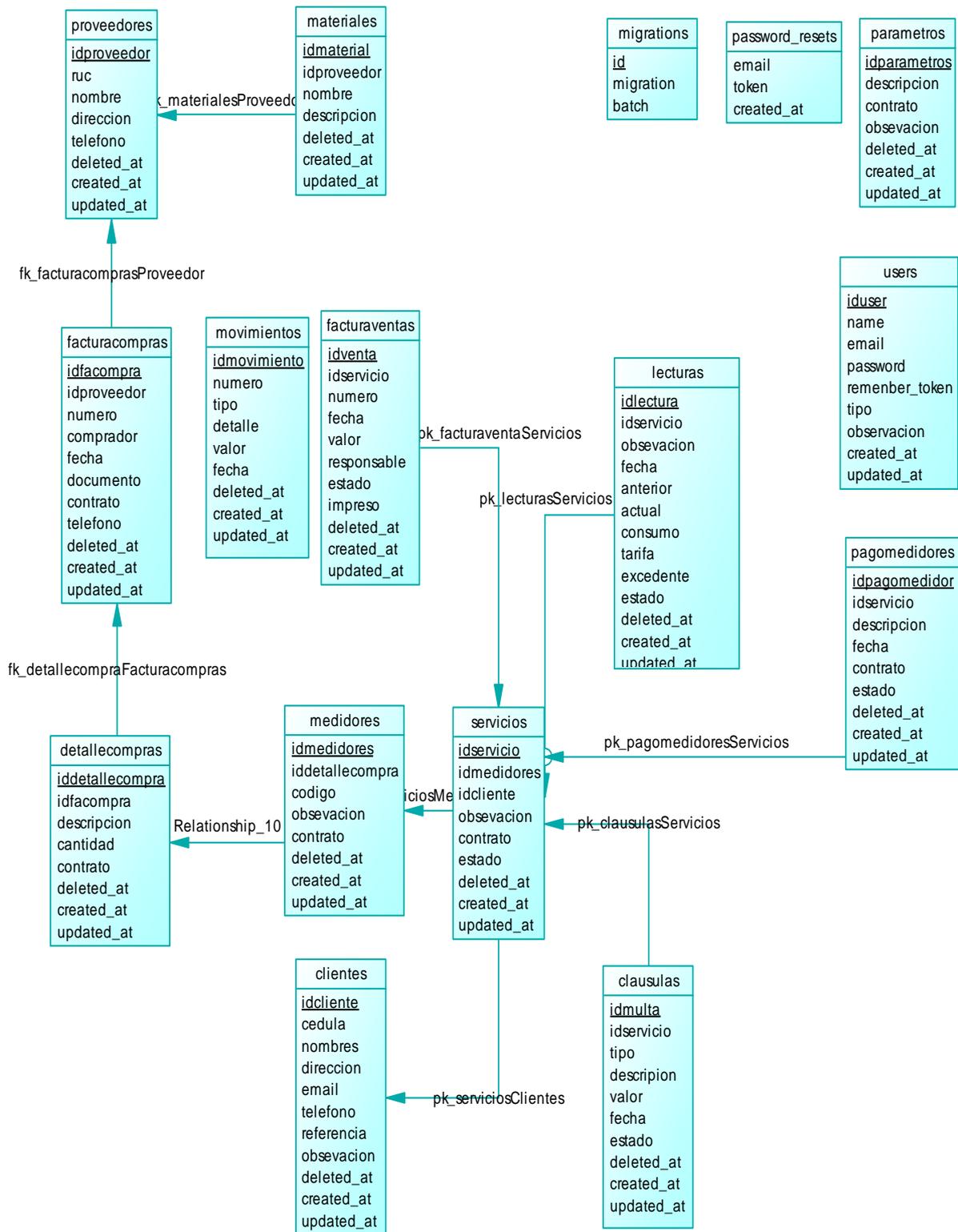


Figura 10. Diseño base de datos

2.2.2 Sprint1 “Módulo de Administración”

Implementa funciones del sistema sistema, esto es; Crear, eliminar, modificar usuarios, equipos y perfiles para controlar el acceso a los diferentes módulos del sistema.

La siguiente tabla describe los requerimientos funcionales y no funcionales de los Sprint del programa a desarrollar.

Tabla 11. Requerimientos funcionales y no funcionales de los Sprint.

| REQUISITOS FUNCIONALES | REQUISITOS NO FUNCIONALES |
|--|--|
| Los usuarios no deben ser eliminados del sistema para mantener consistencia en la información histórica. Un usuario puede tener más de un equipo asignado y un equipo solo puede estar asignado a un usuario. El sistema debe manejar niveles de seguridad para diferentes grupos de usuarios. Ingreso de parámetros (valor del metro3, valor multas, valor excedentes) Registro de proveedores para la compra de material (Medidores, tubos, llaves, permatex, teflón, etc.) Pago a los compañeros (secretaria, lector) Registro pago de la instalación del medidor Facturar venta total del consumo | El sistema brindará seguridad para ser utilizado por usuarios registrados del sistema, con diferentes niveles de acceso. |

Fuente: Autor

De acuerdo con las funcionalidades identificadas para el módulo de administración, se identifican los casos para los diferentes Sprint.

La siguiente tabla describe las historias de los usuarios del sistema.

Tabla 12. Historias de Usuario - Sprint 1.

| ID | Historia de usuario | Importancia producto | Importancia Técnica | Descripción |
|-----------|------------------------------|-----------------------------|----------------------------|--|
| 1 | Creación de usuarios | 500 | 1000 | Es necesario manejar niveles de acceso para los usuarios. |
| 2 | Iniciar sesión en el sistema | 500 | 1000 | Consiste en brindar seguridad al acceso de la aplicación, permitiendo que únicamente usuarios autorizados puedan tener acceso al mismo. |
| 3 | Administración de usuarios | 800 | 800 | El sistema requiere de un número variable de usuarios con diferentes funciones, por lo que es necesario administrar los usuarios y agruparlos en diferentes niveles de acceso a la aplicación. |

Fuente: Autor

La importancia está estimada de acuerdo con las necesidades del Producto y está cuantificada con números enteros entre 0 y 1000. La importancia técnica está basada en las necesidades no funcionales desde el punto de vista del usuario, pero necesarias para un funcionamiento de la aplicación desde el punto de vista técnico.

La siguiente tabla describe el ingreso de los parámetros de los ingresos del sistema.

Tabla 13. Ingreso de parámetros Sprint 2

| ID | Historia de usuario | Importancia producto | Importancia Técnica | Descripción |
|-----------|------------------------------------|-----------------------------|----------------------------|--|
| 1 | Ingreso de valores de metro cubico | 500 | 1000 | Es necesario manejar niveles de ingreso a la aplicación |
| 2 | Ingresar el valor de multas | 500 | 1000 | Consiste en ingresar un valor porcentual de multa en caso de no cancelar su pago a tiempo. |

| | | | | |
|---|--|-----|-----|--|
| 3 | Administra ción de valores excedentes | 800 | 800 | El sistema requiere de la administración de valores excedentes en caso de realizar una mala lectura por parte del encargado del trabajo. |
|---|--|-----|-----|--|

Fuente: Autor

La importancia está estimada de acuerdo con las necesidades del Producto y está cuantificada con números enteros entre 0 y 1000.

La importancia técnica está basada en las necesidades no funcionales desde el punto de vista del usuario, pero necesarias para un funcionamiento de la aplicación desde el punto de vista técnico.

La siguiente tabla describe el registro de proveedores.

Tabla 14. Sprint 3 registro de proveedores

| ID | Historia de usuario | Importancia producto | Importancia Técnica | Descripción |
|-----------|-----------------------------------|-----------------------------|----------------------------|---|
| 1 | Ingreso proveedore s | 500 | 1000 | Es necesario realizar el proveedor de los implementos de la empresa. |
| 2 | Ingresar el número de medidores | 500 | 1000 | Consiste en ingresar el número de medidores con que cuenta la Junta de Agua Potable. |
| 3 | Ingresar los materiales de bodega | 800 | 800 | El sistema requiere de la administración de las exigencias de bodega para realizar las conexiones del servicio en el sector |

Fuente: Autor

La importancia está estimada de acuerdo con las necesidades del Producto y está cuantificada con números enteros entre 0 y 1000.

La importancia técnica está basada en las necesidades no funcionales desde el punto de vista del usuario, pero necesarias para un funcionamiento de la aplicación desde el punto de vista técnico.

La siguiente tabla describe el ingreso de las rutas.

Tabla 15. Sprint 4 control de ingresos

| ID | Historia de usuario | Importancia producto | Importancia Técnica | Descripción |
|-----------|---|-----------------------------|----------------------------|---|
| 1 | Ingreso de rutas, cálculo de beneficios | 500 | 1000 | Es necesario ingresar de las rutas para conocer a que beneficios están sujetos. |
| 2 | Ingresar el pago de instalación medidor | 500 | 1000 | Consiste en ingresar el pago por cada medidor instalado. |
| 3 | Ingresar cálculo de consumo total. | 800 | 800 | El sistema requiere del ingreso del consumo total para efectuar el cobro total del servicio brindado. |
| 4 | Ingreso de pago a compañeros | 500 | 1000 | Se ingresa los pagos realizados a los compañeros que laboran en la Junta de agua potable |

Fuente: Autor

Definición del equipo de trabajo

El equipo de trabajo para la implementación de las funcionalidades del módulo de administración esta descrito en la siguiente tabla:

Tabla 16. Definición del equipo de trabajo.

| ROL | Persona | Descripción/Tareas |
|-----------------------|---------------------------------------|---|
| Producto para la JAAP | Washington Figuroa | Administración del proyecto desde la perspectiva del negocio. |
| ScrumMaster | Washington Figuroa | Asegurar que el proceso SCRUM se lleve a cabo. |
| Team | Codificación Washington Figuroa | Codificación de las funcionalidades identificadas. |
| | Pruebas Washington Figuroa | Pruebas de las funcionalidades codificadas. |

Fuente: Autor

2.2.3 Análisis

Una vez identificadas las historias de usuario, se identifican los actores y se diagrama los casos de uso identificados para la implementación del Sprint 1.

Actores del sistema

La siguiente tabla describe los actores que participan en los casos de uso identificados para el módulo de administración.

Tabla 17. Actores del sistema

| Actor | Descripción |
|---------------|--|
| Presidente | Usuario con privilegios de administrador en el sistema. |
| Administrador | |
| JAAP | Este tipo de usuario puede gestionar todos los recursos utilizados por el sistema. |

Fuente: Autor

2.2.3.2 Especificación de casos de Uso

Los casos de uso son útiles para capturar requerimientos, ayudar a definir la arquitectura, establecer las pautas para el diseño y las pruebas funcionales. Los CU son

una guía de los elementos que serán incluidos en los documentos de usuarios para las aplicaciones, así como la forma en como éstos deben ser empleados. Los CU también establecen las bases para los protocolos de comunicación entre aplicaciones y el diseño de las interfaces gráficas, entre otros.

La siguiente tabla describe la parte de gestionar usuarios.

Tabla 18. Gestionar Usuarios

| ID | RF-01 |
|-----------------|--|
| Descripción | Proporciona funcionalidades para dar mantenimiento a los usuarios del sistema. |
| Precondición | |
| Postcondición | Información actualizada de los usuarios del sistema. |
| Flujo Normal | <ol style="list-style-type: none"> 1 El usuario registrado utiliza el formulario para mantenimiento de usuarios. 2 El sistema solicita los datos del nuevo usuario. 3 El sistema almacena los datos proporcionados. |
| Flujos Alternos | <p>* Usuario ya registrado</p> <p>Si el usuario ya existe en el sistema y es necesario modificar la información actual, el sistema proporciona la funcionalidad para realizarlo.</p> |
| Excepciones | <ol style="list-style-type: none"> 1 Si se intenta registrar un usuario con Login, C.I. o Correo. la entidad que ya se encuentren registrados, el sistema debe advertir la situación y evitará que se registre el usuario. |
| Notas: | <ol style="list-style-type: none"> 1 Se requiere los siguientes datos del usuario: <ol style="list-style-type: none"> 1.1. Login (*): Identificador único de cada usuario con el cual se autentificará en el sistema. 1.2. Clave de acceso (*): código para acceso al sistema. 1.3. Correo (*). 1.6 Identificación |

-
- 1.6.1 Tipo de identificación
 - 1.7. Dirección de domicilio.
 - 1.8. Dos teléfonos de contacto, y de un numero celular (*).
 - 1.9. Dirección de correo electrónico.
-

Fuente: Autor

La siguiente tabla describe la parte de selección usuarios.

Tabla 19. Seleccionar Usuario

| ID | RF-03 |
|---------------|---|
| Descripción | La asignación de lecturas requiere de la selección de un usuario registrado en el sistema, por tanto, se debe poder seleccionar un usuario para utilizar en los procesos mencionados. |
| Precondición | 1 Usuarios registrados en el sistema. |
| Postcondición | ? Lecturas asignadas al usuario seleccionado. ? Equipo asignado al usuario seleccionado. |
| Flujo Normal | 1 Consulta de todos los usuarios registrados en el sistema y permite filtrar de acuerdo con los perfiles existentes. 2 Filtrar información. 3 Seleccionar el usuario para utilizar en la tarea requerida. |

Fuente: Autor

La siguiente tabla describe la asignación y reasignación de los responsables.

Tabla 20. Asignar/Reasignar Responsable

| ID | RF-04 |
|--------------|--|
| Descripción | Proporciona funcionalidades para asignar al usuario responsable de un Equipo (PK). |
| Precondición | |

| | | |
|-----------------|---|---|
| | 1 | Usuarios registrados en el sistema. |
| | 2 | Equipos registrados en el sistema en estado activo. |
| Postcondición | | Información actualizada de los usuarios responsables de las lecturas |
| Flujo Normal | | |
| | 1 | El usuario registrado selecciona un equipo de los registrados en el sistema. |
| | 2 | El equipo no tiene un usuario asignado. |
| | 3 | Se selecciona un usuario del sistema. |
| | 4 | El sistema registra la asignación del equipo al usuario. |
| Flujos Alternos | | |
| | * | Reasignación |
| | 1 | El usuario registrado selecciona un equipo de los registrados en el sistema y este tiene asignado un usuario. |
| | 2 | Se selecciona otro usuario del sistema. |
| | 3 | El sistema actualiza la asignación del equipo al usuario. |
| | * | Liberación |
| | 1 | El usuario registrado selecciona un equipo de los registrados en el sistema y este tiene asignado un usuario. |
| | 2 | Se elimina la asignación, dejando el equipo libre para asignar a otro usuario. |
| | 3 | Se guardará un log de los eventos (asignación/reasignación). |
| Notas: | | |
| | | 1. Un equipo debe ser asignado únicamente a un usuario. |
| | | 2. Un usuario puede tener asignado más de un equipo. |
| | | 3. El proceso de asignación debe ser lo más fácil posible, debido a la alta frecuencia con que los usuarios. |

Fuente: Autor

La siguiente tabla describe la selección del perfil

Tabla 21. Seleccionar Perfil

| ID | RF-05 |
|---------------|---|
| Descripción | Consiste en consultar y seleccionar un perfil determinado, para asignar usuarios o asignar formularios. |
| Precondición | 1 Perfiles registrados en el sistema. |
| Postcondición | 1 Perfil seleccionado para ser utilizado en diferentes procesos. |
| Flujo Normal | 1 Consulta de todos los perfiles registrados en el sistema. 2 Filtrar información. 3 Seleccionar el perfil para utilizar en la tarea requerida. |

Fuente: Autor

La siguiente tabla describe la selección del formulario de perfil del usuario a manipular el sistema.

Tabla 22. Seleccionar Formulario

| ID | RF-06 |
|---------------|--|
| Descripción | Consiste en consultar y seleccionar un formulario (s), para asignar a un determinado perfil. |
| Precondición | 1 Formularios registrados en el sistema. |
| Postcondición | ? Formulario asignado al perfil. |
| Flujo Normal | 1 Consulta de todos los formularios registrados en el sistema. 2 Filtrar información. 3 Seleccionar formularios a asignar a un perfil. |

Fuente: Autor

La siguiente tabla describe la gestión de perfiles del sistema.

Tabla 23. Gestionar Perfiles

| ID | RF-07 |
|-----------------|---|
| Descripción | Proporciona funcionalidades crear, modificar perfiles de usuario, los cuales proporcionarán diferentes niveles de acceso a la aplicación. |
| Precondición | |
| Postcondición | Información actualizada de los perfiles disponibles en el sistema. |
| Flujo Normal | <ol style="list-style-type: none"> 1 El usuario registrado ingresa los datos de un perfil. 2 El sistema guarda la información de nuevo perfil. |
| Flujos Alternos | <ul style="list-style-type: none"> * El perfil ya existe 1 El usuario registrado selecciona un perfil existente 2 Se modifica la información del perfil 3 El sistema actualiza la información del perfil modificado. * Eliminar perfil 1 El usuario registrado selecciona un perfil existente 2 Se solicita la eliminación del perfil. 3 El sistema elimina de la DB los datos del perfil seleccionado. |

Fuente: Autor

La siguiente tabla describe la gestión de formularios para el ingreso al sistema.

Tabla 24. Gestionar Formularios

| ID | RF-08 |
|-------------|--|
| Descripción | Proporciona funcionalidades para registrar información de los formularios (pantallas) existentes en el sistema. Estas son posteriormente asignadas a los perfiles existentes, para brindar acceso a las pantallas requeridas en un perfil. |

Precondición

- 1 Desarrollo de los componentes GUI a través del framework Angular 5.

Postcondición

Información actualizada con los datos de los formularios (pantallas) disponibles en el sistema.

Flujo Normal

- 1 El usuario selecciona los componentes GUI (Ejemplo, Login, dashboard).
- 2 El sistema muestra todos los formularios disponibles en el componente.
- 3 El usuario selecciona un formulario.

- 4 El usuario ingresa una descripción que permita identificar la utilizad que tiene el formulario dentro del sistema, el módulo en el cual es utilizado y el estado (Activo/Inactivo).
- 5 El sistema registra el nuevo formulario disponible en el sistema.

Flujos Alternos

- * El formulario ya existe
- 1 El usuario logeado selecciona un formulario existente.
 - 2 Se modifica la información del formulario.
 - 3 El sistema actualiza la información del formulario modificado.
- * Eliminar Formulario
- 1 El usuario logeado selecciona un formulario existente.
 - 2 Se solicita la eliminación del formulario.
 - 3 El sistema elimina de la DB los datos del formulario seleccionado.

Fuente: Autor

La siguiente tabla describe la gestión de los diferentes reportes del sistema.

Tabla 25. Gestión de reportes

| ID | RF-11 |
|-----------------|---|
| Descripción | Proporciona funcionalidades para crear, modificar reportes utilizados en el sistema. Esto permite agregar o eliminar dinámicamente listas de valores utilizados como reportes. |
| Precondición | |
| Postcondición | |
| Flujo Normal | <ol style="list-style-type: none"> 1 Reportes Actualizados 2 Detalles de reportes actualizados. <ol style="list-style-type: none"> 1 Ingresar los datos del reporte (Descripción de las listas de valores). 2 El sistema guarda los datos del reporte. 3 Ingresar detalles del reporte (Los detalles corresponden a la lista de valores del catálogo). <hr/> <ol style="list-style-type: none"> 4 El sistema almacena la lista de detalles del reporte. |
| Flujos Alternos | <ol style="list-style-type: none"> * El reporte ya existe 1 El usuario dashboard selecciona un reporte existente. 2 Se modifica la información del reporte y el sistema almacena la información modificada (opcional) 3 Se actualiza la información de los detalles del reporte. La actualización puede incluir la adición, eliminación o actualización de los detalles (Opcional). |

Fuente: Autor

La siguiente tabla describe el ingreso de los proveedores del sistema.

Tabla 26. Ingreso de proveedores.

| ID | RF-01 |
|-----------------|---|
| Descripción | Se ingresa a los proveedores para proceder a la compra de material medidores, tubos, llaves, permatex, etc. |
| Precondición | |
| Postcondición | Información actualizada de los proveedores del sistema. |
| Flujo Normal | <ol style="list-style-type: none"> 1 El proveedor ingresado. 2 El sistema solicita los datos del nuevo proveedor. 3 El sistema almacena los datos proporcionados. |
| Flujos Alternos | <p>* Proveedor ya registrado</p> <p>Si el proveedor ya existe en el sistema y es necesario modificar la información actual, el sistema proporciona la funcionalidad para realizarlo.</p> |
| Excepciones | <ol style="list-style-type: none"> 1 Si se intenta registrar al proveedor con un código único, C.I. o Correo. la entidad que ya se encuentren registrados, el sistema debe advertir la situación y evitará que se registre el proveedor varias veces. |
| Notas: | <ol style="list-style-type: none"> 1 Se requiere los siguientes datos del proveedor: <ol style="list-style-type: none"> 1.4. Login (*): Identificador único con el cual se autenticará en el sistema. 1.5. Clave de acceso (*): código para acceso al sistema. 1.6. Correo (*). 1.6 Identificación <ol style="list-style-type: none"> 1.6.1 Tipo de identificación <ol style="list-style-type: none"> 1.10. Dirección de domicilio. 1.11. Dos teléfonos de contacto, y de un numero celular (*). 1.12. Dirección de correo electrónico. |

Fuente: Autor

La siguiente tabla describe el ingreso de los parámetros del sistema

Tabla 27. Ingreso de parámetros.

| ID | RF-05 |
|---------------|--|
| Descripción | En realizar el ingreso de los parámetros valor del metro3, valor multas, valor excedente. |
| Precondición | <ol style="list-style-type: none">1 Ingresar parámetros al sistema. |
| Postcondición | <ol style="list-style-type: none">1 Seleccionar el parámetro en diferentes procesos del sistema. |
| Flujo Normal | <ol style="list-style-type: none">1 Consulta de todos los parámetros registrados en el sistema.2 Filtrar información.3 Seleccionar el parámetro para utilizar en la tarea requerida. |

Fuente: Autor

La siguiente tabla describe el control de los ingresos generales del sistema

Tabla 28. Control de ingresos generales

| ID | RF-05 |
|---------------|---|
| Descripción | Ingreso de multas (Ingreso), Ver rutas, Ingreso lecturas Calculo de beneficios en caso que sea mayor de edad, discapacidad, registro pago de la instalación del medidor Calculo del consumo total + multas – beneficios, Facturar venta total del consumo, Reporte de este. |
| Precondición | 1 Realizar todos los ingresos al sistema. |
| Postcondición | 1 Seleccionar el proceso a ejecutar en el sistema. |
| Flujo Normal | 1 Consulta de todos los parámetros registrados en el sistema. 2 Filtrar información. 3 Seleccionar el parámetro para utilizar en la tarea requerida y emitir el reporte de la misma. |

Fuente: Autor

2.3 Post-juego

Preparación, pruebas, lanzamiento.

2.3.1 Modelos de datos

Del análisis de la especificación de los casos de uso, se determina la necesidad de utilizar el modelo de datos descrito en el diagrama de entidad relación de la base de datos. Entidad Relación-Módulo Administración, las mismas que serán implementadas como objetos en la base de datos y como entidades del negocio en la codificación de la aplicación.

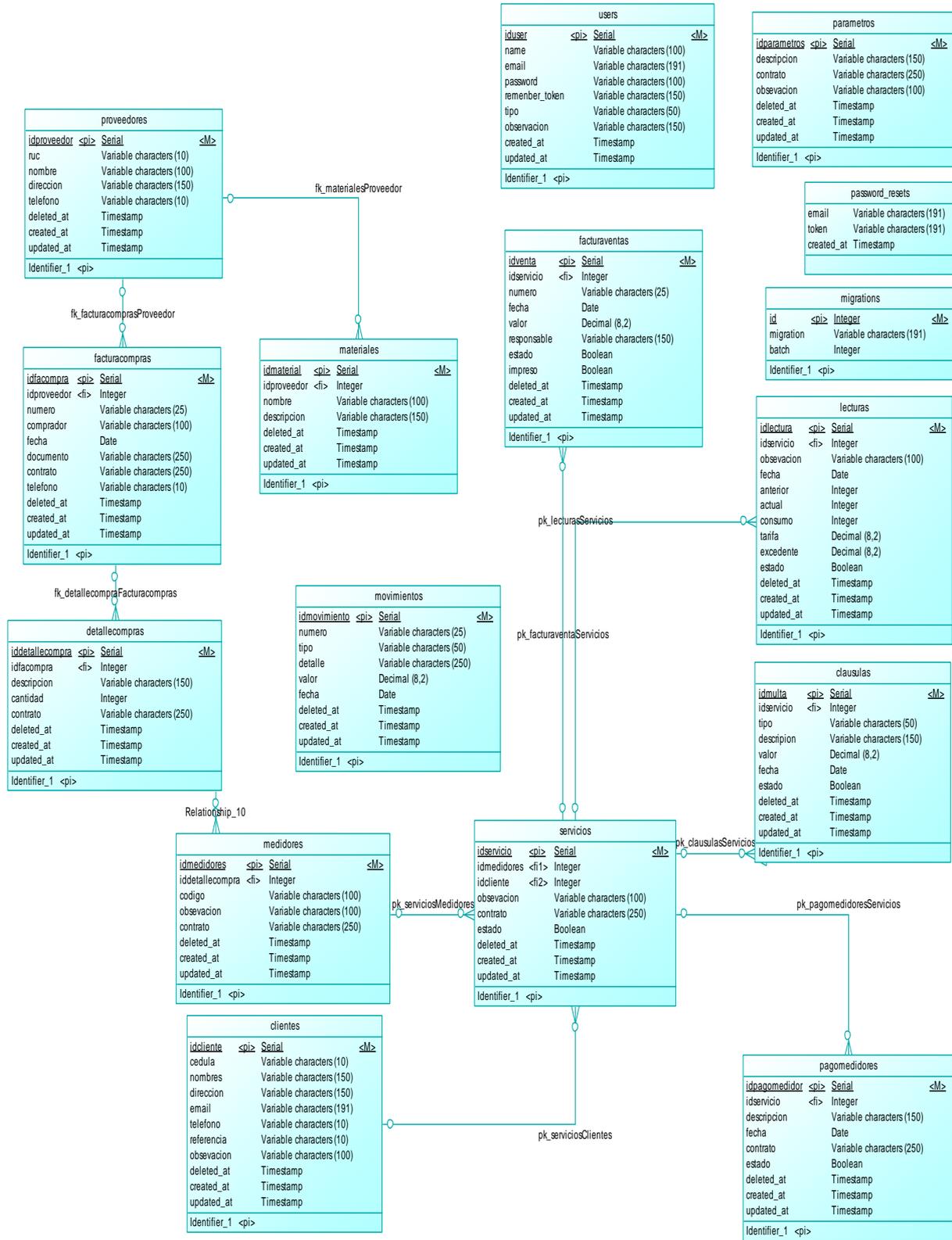


Figura 11. Modelo entidad relación

2.3.2 Arquitectura

De la arquitectura planteada inicialmente para la aplicación completa, la codificación para el módulo de administración se implementará de la siguiente manera:

Formularios (pantallas) para:

Formularios para administración de recursos

- ✓ Administración de Usuarios: Frontal el cual permitirá el control y asignación de privilegios para uso del sistema.
- ✓ Administración de Equipos: Frontal para el control de inventariado.
- ✓ Administración de Multas: Control de responsabilidades por parte de los contribuyentes.

Formularios comunes

- ✓ Principal: Blog informativo de la JAAP.
- ✓ Login: Control de acceso al sistema.

Componentes para controlar la lógica de:

Administración de recursos

- ✓ Administración de Usuarios
- ✓ Administración de Equipos
- ✓ Administración de Multas

Lógica común

- ✓ Seguridad
- ✓ Login

Componente para acceso a la capa de datos (DB)

El ORM Eloquent incluido con Laravel proporciona una implementación de ActiveRecord bella y sencilla para trabajar con su base de datos. Cada tabla de base de datos tiene un "Modelo" correspondiente que se utiliza para interactuar con esa tabla. Los modelos le permiten consultar datos en sus tablas, así como insertar nuevos registros en la tabla.

Componente para administrar las entidades utilizadas en el sistema

Middleware proporciona un mecanismo conveniente para filtrar las solicitudes HTTP que ingresan a su aplicación. Por ejemplo, Laravel incluye un middleware que verifica que el usuario de su aplicación esté autenticado. Si el usuario no está autenticado, el middleware redirigirá al usuario a la pantalla de inicio de sesión. Sin embargo, si el usuario está autenticado, el middleware permitirá que la solicitud proceda más adelante en la aplicación.

2.3.3 Construcción y pruebas

Una vez identificadas las historias de usuario, las cuales se encuentran descritas en los casos de uso y sus respectivas especificaciones, estas serán implementadas en el primer sprint; para ello es necesario definir el Backlog del sprint (pila de tareas) que permitan implementar las funcionalidades especificadas en etapa de análisis, aplicando diseño planteado. La pila de tareas de los diferentes Sprint se muestra en la siguiente tabla:

Tabla 29. Backlog Sprint "Módulo de Administración"

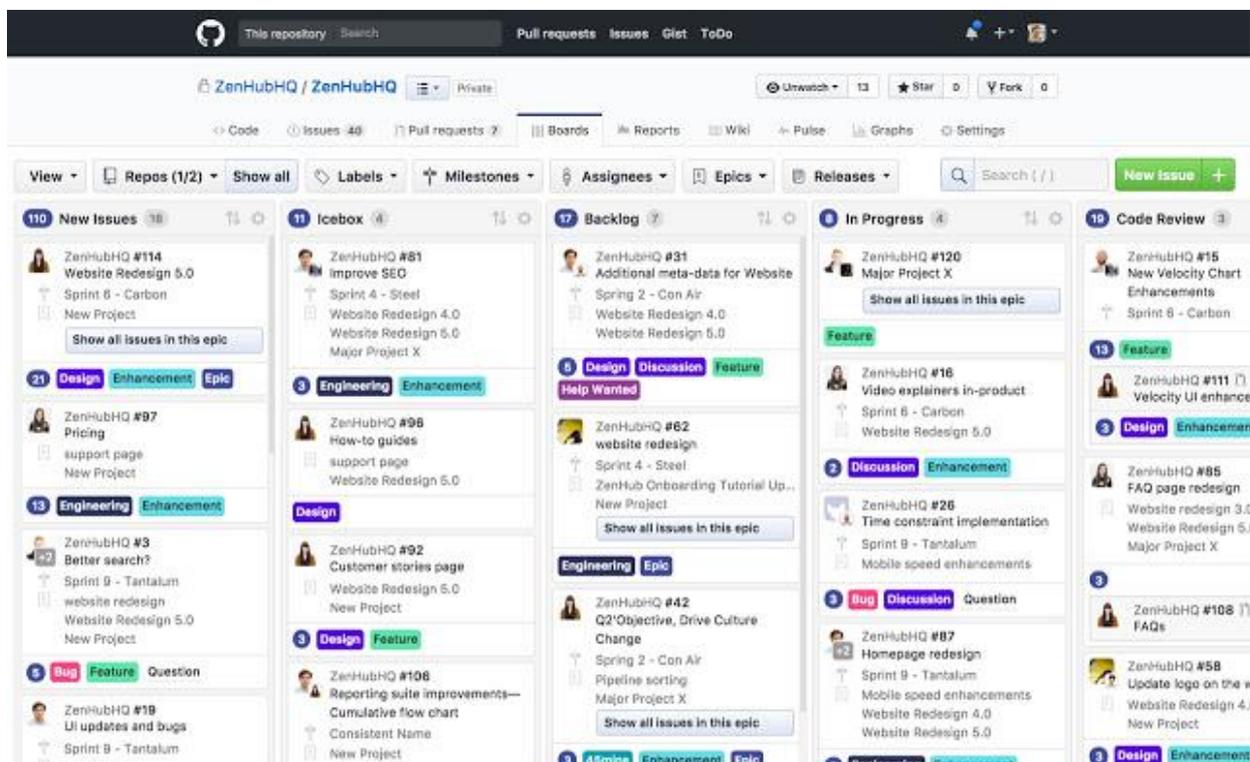
| Story ID | Task# | Story Name, Task Name | Assigned 1 | Estimado (Horas) |
|----------|-------|-----------------------------------|---------------------|------------------|
| 1 | | Iniciar sesión en el sistema | | |
| | 1 | Componentes para Inicio de sesión | Washington Figueroa | 8 |
| | 2 | Componentes para conexión DB | Washington Figueroa | 8 |
| | 3 | Configuración de conexión a DB | Washington Figueroa | 1 |
| | 4 | Componentes base de la aplicación | Washington Figueroa | 8 |

| | | | | |
|----------|---|-----------------------------------|-------------------------|---|
| | 5 | Pruebas de desarrollo | Washington Figuerola | 2 |
| | 6 | Pruebas unitarias | Washington Figuerola | 1 |
| 2 | | Administración de usuarios | | |
| | 1 | Componentes comunes | Washington Figuerola | 8 |
| | 2 | Creación Objetos DB | Washington Figuerola | 5 |
| | 3 | Desarrollo GUI | Washington Figuerola | 8 |
| | 4 | Pruebas de desarrollo | Washington Figuerola | 2 |
| | 5 | Pruebas unitarias | Washington Figuerola | 1 |
| 3 | | Administración de Equipos | | |
| | 1 | Creación Objetos DB | Washington Figuerola | 5 |
| | 2 | Desarrollo GUI | Washington Figuerola | 5 |
| | 3 | Pruebas de desarrollo | Washington Figuerola | 2 |
| | 4 | Pruebas unitarias | Washington Figuerola | 1 |
| 4 | | Asignación de equipos | | |
| | 1 | Creación Objetos DB | Washington Figuerola | 3 |
| | 2 | Desarrollo GUI | Washington Figuerola | 3 |
| | 3 | Pruebas de desarrollo | Washington Figuerola | 1 |
| | 4 | Pruebas unitarias | Washington Figuerola | 1 |
| 5 | | Creación de perfiles | | |
| | 1 | Creación Objetos DB | Washington Figuerola | 4 |

| | | | | |
|--|---|-----------------------|------------------------|---|
| | 2 | Desarrollo GUI | Washington Figueroa | 5 |
| | 3 | Pruebas de desarrollo | Washington Figueroa | 2 |
| | 4 | Pruebas unitarias | Washington Figueroa | 1 |

Fuente: Autor

Para llevar un control integral de los avances de las tareas planteadas para implementar las historias de usuario definida en la planificación del sprint, las tareas son subidas a la herramienta definida. En este caso “ZenHub”.



Fuente: Captura pantalla herramienta “ZenHub”

2.3.4 Pruebas

La sección de Construcción y Pruebas, las pruebas a realizar dependerán del módulo y componentes de estos. Los resultados de las pruebas serán adjuntados como anexos al presente documento.

En ese contexto, los métodos de pruebas utilizados en este módulo son:

Formularios (pantallas) para:

- ✓ Formularios para administración de recursos -> Pruebas de Aceptación
- ✓ Formularios comunes -> Pruebas de Aceptación
- ✓ Formulario para configuración -> Pruebas de Aceptación

Componentes para controlar la lógica de:

- ✓ Administración de recursos -> Pruebas de unidad (Caja Blanca)
- ✓ Lógica común -> Pruebas de unidad (Caja Blanca)

Componente para acceso:

- ✓ Capa de datos (DB) -> Pruebas de unidad (Caja Blanca, Caminos)

BEL: Componente para administrar las entidades. -> Pruebas de unidad

Para agilizar la etapa de pruebas, la persona asignada del equipo utilizará la herramienta ZenHub.

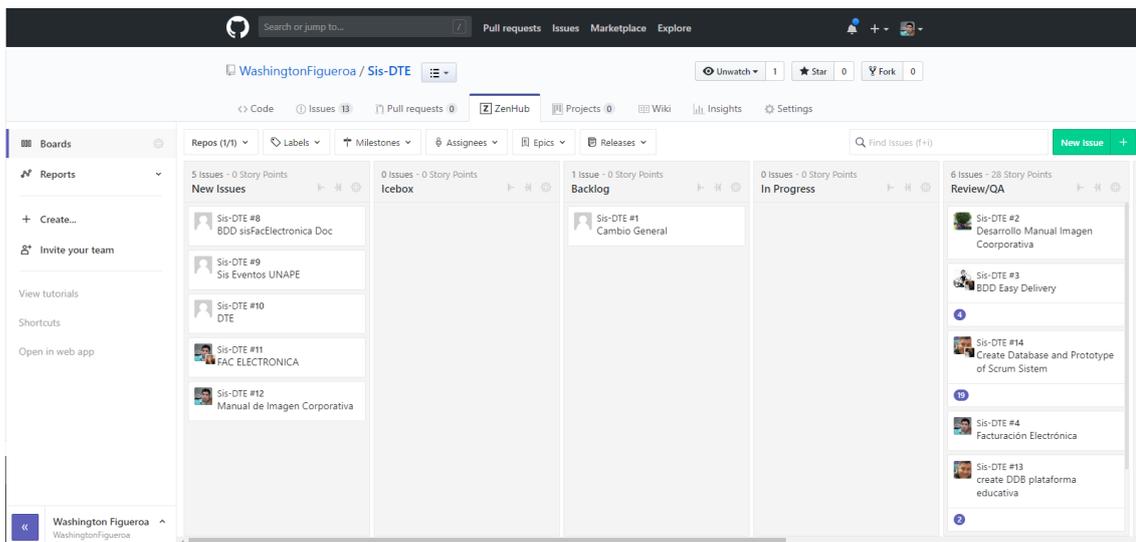


Figura 12. Agilizar la etapa de pruebas Zen

CAPITULO III

3 RESULTADOS

3.1 Análisis encuesta

En el presente análisis de resultados se tomó la población de la junta de agua san José de Chorlaví la cual cuenta con una población de 120 usuarios entre contribuyentes y dirigentes.

Debido a que el número de la población es mínimo para la aplicación de una muestra, se ha dado origen al proceso de un censo.

Las siguientes tablas representan el análisis de las encuestas realizadas a los contribuyentes de la junta de agua san José de Chorlaví.

Tabla 30. Datos para métodos Likert

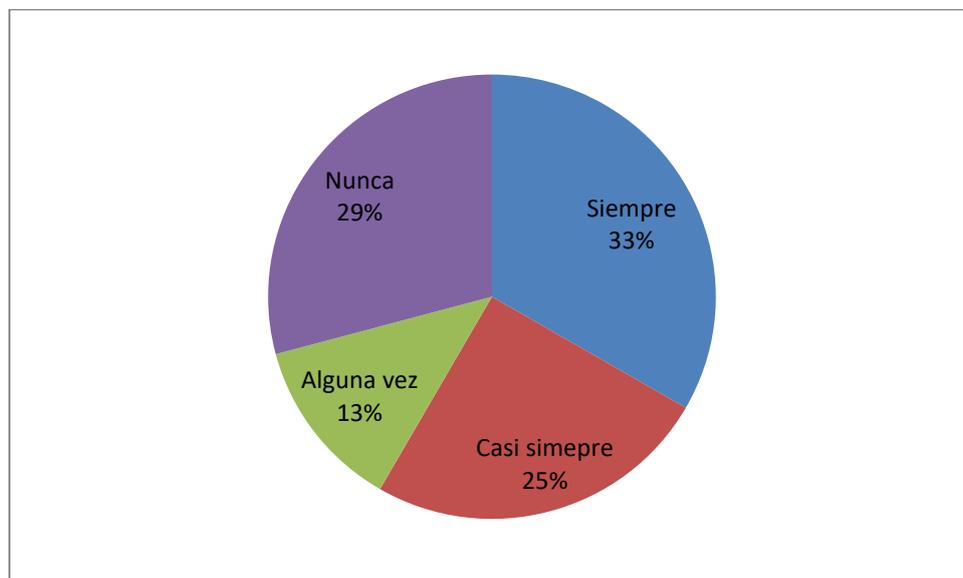
| Pregunta 1 | Pregunta2 | Pregunta3 | Pregunta4 | Pregunta5 | Pregunta6 | Pregunta7 | Pregunta8 |
|------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| 0 | 1 | 0 | 1 | 2 | 1 | 2 | 0 |
| 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 3 | 0 | 0 | 1 |
| 1 | 1 | 0 | 2 | 2 | 0 | 2 | 1 |
| 1 | 1 | 1 | 1 | 2 | 0 | 2 | 1 |
| 1 | 1 | 0 | 3 | 2 | 1 | 3 | 1 |
| 1 | 1 | 0 | 1 | 2 | 0 | 2 | 1 |
| 1 | 0 | 0 | 1 | 1 | 2 | 2 | 3 |
| 0 | 0 | 0 | 2 | 1 | 2 | 2 | 2 |
| 1 | 0 | 0 | 3 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 1 | 1 | 1 | 2 | 0 |
| 0 | 1 | 1 | 1 | 2 | 0 | 2 | 1 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 2 | 2 | 0 | 3 | 2 |
| 1 | 1 | 0 | 1 | 2 | 0 | 2 | 2 |
| 1 | 1 | 0 | 1 | 3 | 1 | 2 | 2 |
| 1 | 0 | 2 | 3 | 2 | 1 | 1 | 1 |
| 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 2 | 1 | 2 | 1 |
| 0 | 1 | 0 | 1 | 3 | 0 | 2 | 2 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 2 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |
| 2 | 1 | 1 | 0 | 2 | 2 | 2 | 2 |
| 2 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 1 | 1 | 1 | 1 | 2 | 1 |
| 1 | 0 | 0 | 1 | 3 | 0 | 3 | 1 |
| 1 | 0 | 0 | 2 | 2 | 1 | 2 | 1 |
| 1 | 1 | 0 | 1 | 3 | 0 | 3 | 2 |
| 2 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 3 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 2 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 2 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 2 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 3 | 0 | 2 | 1 |
| 2 | 1 | 0 | 1 | 3 | 1 | 0 | 2 |
| 1 | 1 | 0 | 2 | 1 | 1 | 2 | 1 |
| 2 | 1 | 1 | 1 | 2 | 1 | 2 | 1 |
| 3 | 1 | 1 | 1 | 2 | 0 | 2 | 1 |
| 2 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 3 | 3 | 0 | 2 | 1 |
| 3 | 1 | 0 | 3 | 2 | 0 | 3 | 1 |
| 2 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 3 | 1 | 0 | 3 | 3 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 2 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 2 | 1 | 0 | 1 | 2 | 1 | 4 | 1 |
| 3 | 1 | 0 | 2 | 2 | 0 | 3 | 1 |
| 2 | 1 | 0 | 2 | 2 | 0 | 0 | 1 |
| 2 | 1 | 0 | 2 | 1 | 0 | 2 | 1 |
| 1 | 1 | 1 | 1 | 1 | 0 | 2 | 1 |
| 3 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 3 | 1 | 0 | 3 | 3 | 0 | 3 | 2 |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 3 | 1 | 0 | 1 | 3 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 3 | 1 | 2 | 2 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 1 | 3 | 0 | 3 | 1 |
| 1 | 1 | 0 | 2 | 2 | 1 | 3 | 1 |

| | | | | | | | |
|--------------|---|---|---|---|---|---|---|
| Siempre | 4 | 4 | 1 | 1 | 4 | 1 | 5 |
| Casi siempre | 2 | 2 | 2 | 2 | 4 | 4 | 2 |
| Alguna vez | 3 | 2 | 1 | 3 | 4 | 1 | 5 |
| Nunca | 1 | 3 | 3 | 1 | 2 | 2 | 2 |

| | Fx | % |
|--------------|-----------|----------|
| Siempre | 40 | 40% |
| Casi siempre | 30 | 30% |
| Alguna vez | 15 | 15% |
| Nunca | 35 | 35% |
| Total | 100 | 100% |



Pregunta 1: ¿Luego de implementar el sistema de la junta de agua está usted de acuerdo con la forma en la que se realiza el cobro de planillas?

| | | Frecuencia | Porcentaje | Porcentaje válido | Porcentaje acumulado |
|--------|----------------|------------|------------|-------------------|----------------------|
| Válido | Muy de acuerdo | 53 | 44,2 | 44,2 | 44,2 |
| | De acuerdo | 67 | 55,8 | 55,8 | 100,0 |
| | Desacuerdo | 0 | 0 | 0 | |
| | Total | 120 | 100,0 | 100,0 | |

luego de implementar el sistema de la junta de agua esta usted de acuerdo con la forma en la que se realiza el cobro de planillas

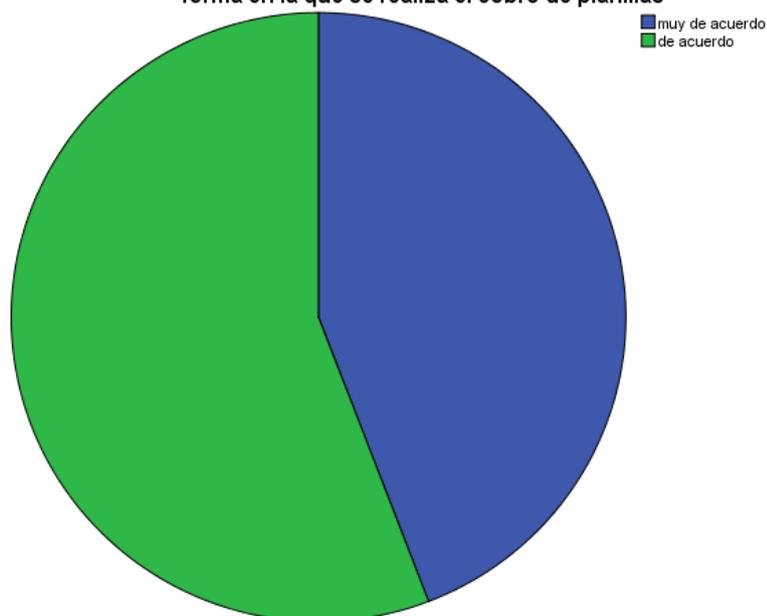


Figura 13. Implementación del sistema

Análisis: La mayoría de la población encuestada menciona que están de acuerdo con la forma de cobro de planillas, afirman que es un método más ágil y conveniente en el cálculo del consumo del agua, el 100% de los encuestados están satisfechos con el nuevo sistema de cobro implementado para agilizar el cobro de panillas de consumo.

Pregunta 2:

¿Considera usted que la atención brindada por la junta de agua es eficiente a raíz de la implementación del sistema?

| | | Frecuencia | Porcentaje | Porcentaje válido | Porcentaje acumulado |
|--------|---------------|------------|------------|-------------------|----------------------|
| Válido | Muy eficiente | 58 | 48,3 | 48,3 | 48,3 |
| | Eficiente | 62 | 51,7 | 51,7 | 100,0 |
| | Deficiente | 0 | 0 | 0 | |
| | Total | 120 | 100,0 | 100,0 | |

considera usted que la atención brindada por la junta de agua es eficiente a raíz de la implementación del sistema

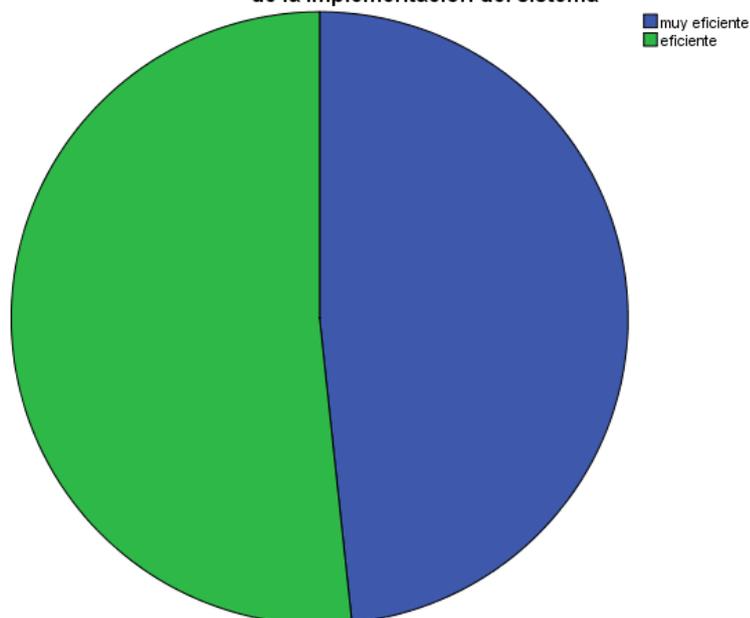


Figura 14. Eficiencia del sistema de agua potable

Análisis: La totalidad de encuestados afirman que el servicio y la atención brindada a partir de la implementación del sistema de cobro es más eficiente y exacta en los cobros, ocasionado una buena atención al cliente y por ende una satisfacción y tranquilidad por parte de los consumidores.

Pregunta 3:

¿Considera que el tiempo de respuesta por parte de la junta a una solicitud de información detallada sobre las planillas es la adecuada?

| | | Frecuencia | Porcentaje | Porcentaje válido | Porcentaje acumulado |
|--------|-----------|------------|------------|-------------------|----------------------|
| Válido | Inmediata | 119 | 99,2 | 99,2 | 99,2 |
| | Lenta | 1 | ,8 | ,8 | 100,0 |
| | No llega | 0 | 0 | 0 | |
| | Total | 120 | 100,0 | 100,0 | |

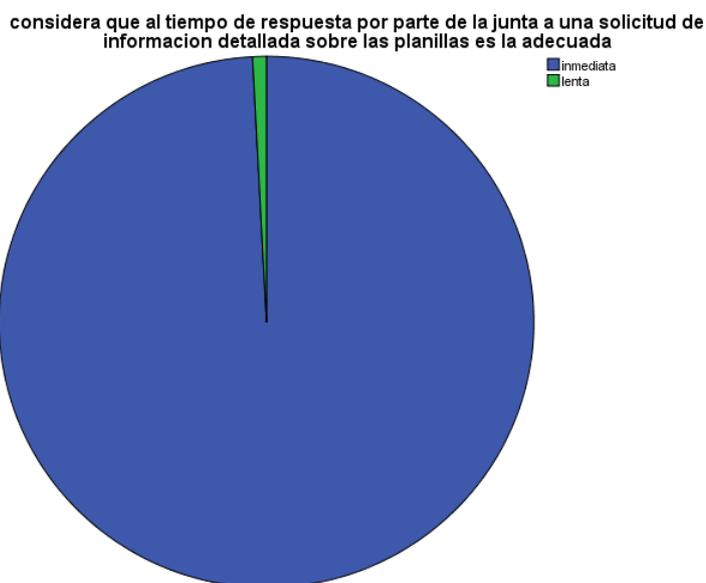


Figura 15. Solicitud de información en el sistema

Análisis: El 100% de los encuestados menciona que el servicio de la junta de agua es más eficiente y rápido, brinda un servicio de calidad al momento de solicitar información del consumo personal de agua, siendo atendido de inmediato en todas las dudas e inquietudes que tengan los ciudadanos de la zona.

Pregunta 4:

¿Considera usted que el valor cancelado por concepto de planillas está acorde con el consumo de agua, ya que las tomas de lectura son registradas en un sistema para calcular el cobro del servicio?

| | | Frecuencia | Porcentaje | Porcentaje válido | Porcentaje acumulado |
|--------|----|------------|------------|-------------------|----------------------|
| Válido | Si | 120 | 100,0 | 100,0 | 100,0 |
| | No | 0 | 0 | 0 | 0 |

considera usted que el valor cancelado por concepto de planillas esta acorde con el consumo de agua, ya que las tomas de lectura son registradas en un sistema para calcular el cobro del servicio

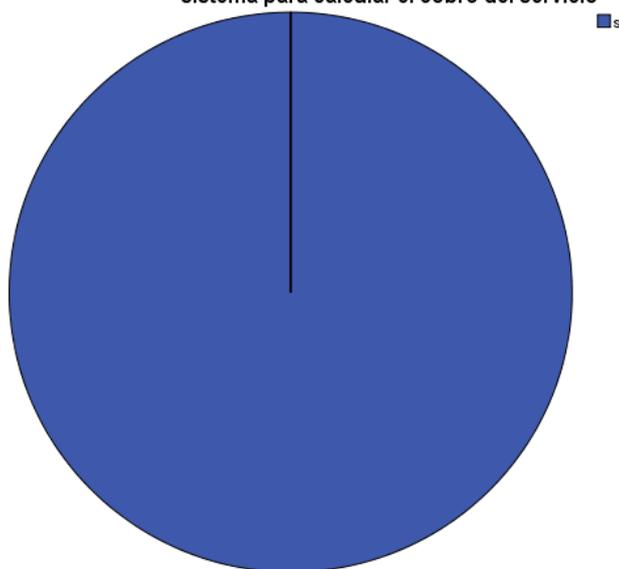


Figura 16. Valor cancelado por planilla es el correcto

Análisis: La totalidad de encuestados de la zona afirman que el valor de pago por consumo está acorde al nivel del consumismo de los mismos, es por eso que los ciudadanos están totalmente satisfechos con el nuevo servicio, ayudando en muchos de los casos y contribuyendo a incentivar el ahorro y evitar el desperdicio de agua.

Pregunta 5:

¿Cuándo proceden a la lectura de su medidor que medio utilizan para registrar el consumo actualmente?

| | Frecuencia | Porcentaje | Porcentaje válido | Porcentaje acumulado |
|----------------|------------|------------|-------------------|----------------------|
| Válido libreta | 3 | 2,5 | 2,5 | 2,5 |
| celular | 65 | 54,2 | 54,2 | 56,7 |
| Tablet | 52 | 43,3 | 43,3 | 100,0 |
| Total | 120 | 100,0 | 100,0 | |

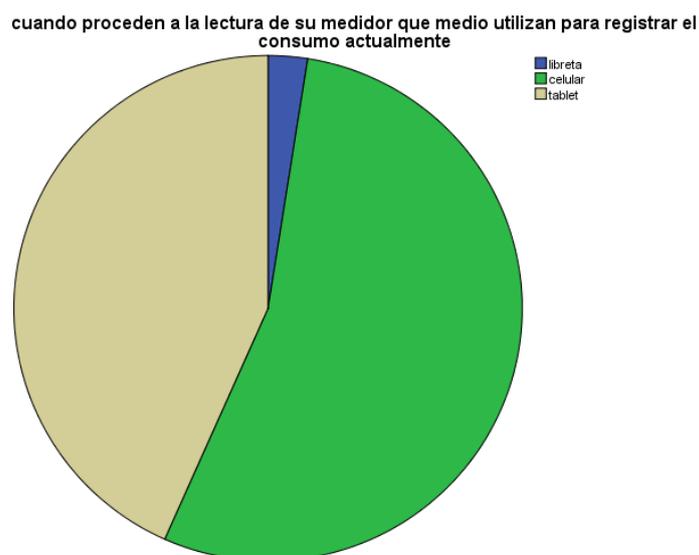


Figura 17. Lectura de medidor

Análisis:

La mayoría de los encuestados afirman que el medio que utilizan para registrar el consumo del medidor de agua es el celular siendo estos el 54,2% de ciudadanos encuestados, de igual manera el 43,3% mencionan que el medio que utilizan para el registro es la tableta, dando una minoría que menciona que en algunos de los casos registran el consumo utilizando un libreta.

Pregunta 6:

¿Considera usted que la implementación del sistema informático para registrar los cobros permitió brindar un mejor servicio y ofrecer información oportuna a todos los usuarios de la junta administradora de agua?

| | | Frecuencia | Porcentaje | Porcentaje válido | Porcentaje acumulado |
|--------|----|------------|------------|-------------------|----------------------|
| Válido | Si | 120 | 100,0 | 100,0 | 100,0 |
| | No | 0 | 0 | 0 | 0 |

considera usted que la implementacion del sistema informatico para registrar los cobros permitio brindar un mejor servicio y ofrecer informacion oportuna a todos los usuarios de la junta administradora de agua

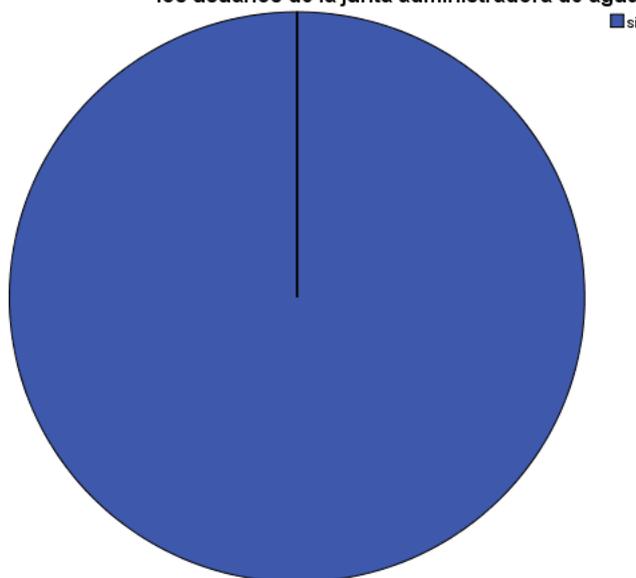


Figura 18. Mejoramiento del servicio de cobro de planillas

Análisis:

La totalidad de los encuestados afirman que a partir de la implementación del sistema para el registro y cobro del consumo del agua, el servicio es más eficiente, brindando un servicio de calidad e información oportuna a todas las inquietudes y solicitudes requeridas por los ciudadanos de la zona.

Pregunta 7:

¿Le parece beneficioso el servicio que ofrece la JAAP, mediante las consultas de planilla a través de internet?

| | | Frecuencia | Porcentaje | Porcentaje válido | Porcentaje acumulado |
|--------|------------------|------------|------------|-------------------|----------------------|
| Válido | Muy beneficioso | 42 | 35,0 | 35,0 | 35,0 |
| | Beneficioso | 78 | 65,0 | 65,0 | 100,0 |
| | Poco Beneficioso | 0 | 0 | 0 | |
| | Total | 120 | 100,0 | 100,0 | |

le parece beneficioso el servicio que ofrece la JAAP, mediante las consultas de planilla a través de internet

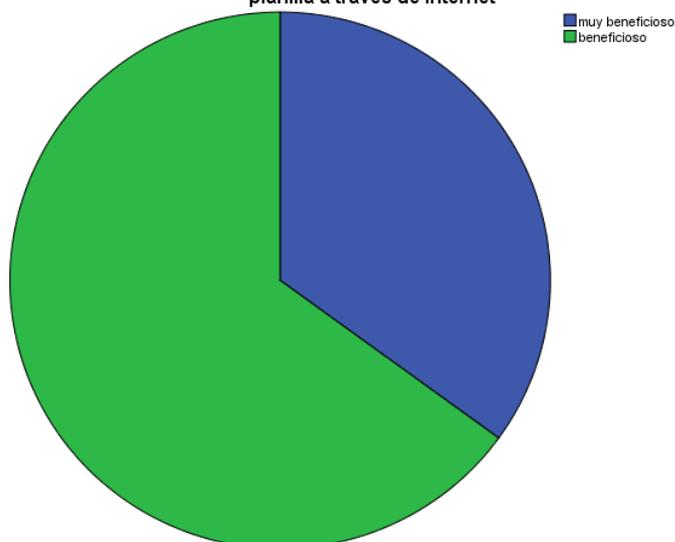


Figura 19. Beneficios de la JAAP.

Análisis:

La mayoría de los ciudadanos de la zona siendo estos el 100% de encuestados afirman que les parece una opción muy alentadora al poder acceder por medio del Internet a la consulta de las planillas de consumo, ya que por medio de este servicio ya tienen la información necesaria, con el fin de realizar los pagos a tiempo, evitando retrasos y en muchos de los casos en corte del servicio.

Pregunta 8:

Genero

| | | Frecuencia | Porcentaje | Porcentaje válido | Porcentaje acumulado |
|--------|-----------|------------|------------|-------------------|----------------------|
| Válido | femenino | 53 | 44,2 | 44,2 | 44,2 |
| | masculino | 67 | 55,8 | 55,8 | 100,0 |
| | Total | 120 | 100,0 | 100,0 | |

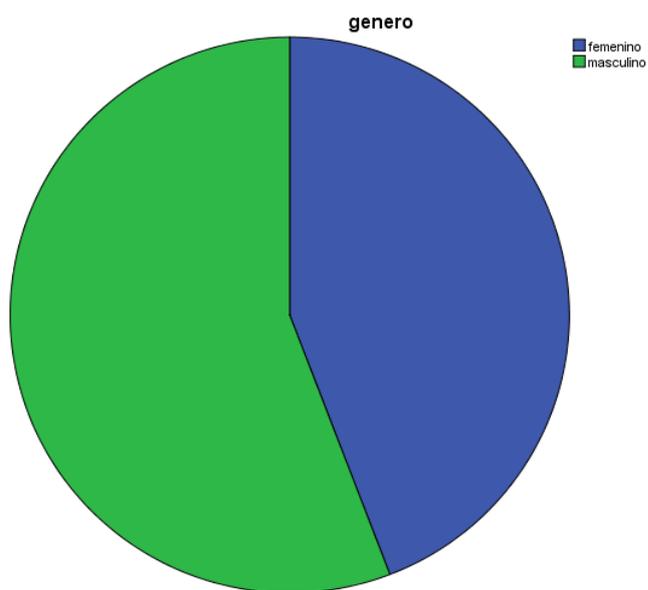


Figura 20. Genero de los encuestados

Análisis:

La mayoría de los encuestados son de género masculino siendo estos el 55.8%, dando una diferencia del 44.2% los cuales son de género femenino. Dando un total de 120 personas encuestadas.

Pregunta 9:

Instrucción

| | | Frecuencia | Porcentaje | Porcentaje válido | Porcentaje acumulado |
|--------|------------|------------|------------|-------------------|----------------------|
| Válido | primaria | 24 | 20,0 | 20,0 | 20,0 |
| | secundaria | 61 | 50,8 | 50,8 | 70,8 |
| | superior | 35 | 29,2 | 29,2 | 100,0 |
| | Total | 120 | 100,0 | 100,0 | |

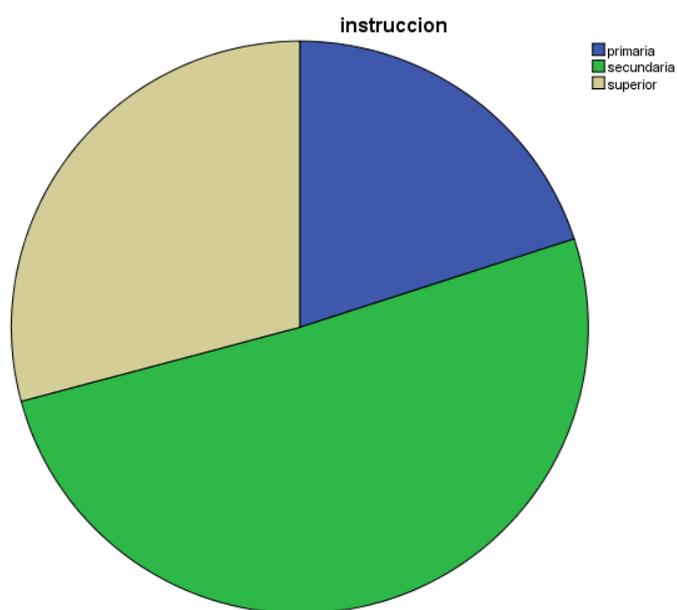


Figura 21. Instrucción de las personas encuestadas

Análisis:

El 50.8% de los encuestados afirman tener estudios que no sobrepasan la secundaria, dedicándose a trabajos de albañilería, carpintería entre otros. El 29.2% menciona tener estudios superiores y ocupando puestos como por ejemplo enfermeros, policías y chef. Dando una minoría del 20% quienes afirman tener estudios de primaria siendo estos la mayoría amas de casa.

3.2 Análisis de la entrevista

Luego de haber implementado el sistema en la junta de agua de Chorlaví el manejo de usuarios se lo maneja de mejor manera debido que se realizan búsquedas rápidas de la información que se requiera y la respuesta es eficiente ahorrando tiempo en el proceso a realizarse, los dirigentes de la junta de agua se encuentran satisfechos en la forma en la cual actualmente se está facturando a los usuarios de la misma, lo que permite llevar un control adecuado del proceso de facturación de la junta, lo que ha favorecido con los usuarios al momento de emitir sus respectivas del consumo del servicio brindando un ambiente amigable con el usuario al momento de realizar los pagos y recibir comprobante que este caso viene siendo la factura, de igual manera los socios consideran oportuno haber realizado la implementación del sistema para la automatización de sus procesos de facturación y registro de ingresos de implementos de la junta de agua, están satisfechos en cuanto a la forma de emisión de reportes de los recibos de consumo y multas para llevar un control adecuado de esta información que es primordial en la empresa. A raíz de la implementación del sistema en la empresa, la misma ha lleva una administración ordenada de cada uno de sus procesos generando satisfacción en los usuarios en los dirigentes de la misma.

3.3 Cuadro de resultados de los frameworks utilizados

Tabla 31. Cuadro de los beneficios de los frameworks utilizados para el desarrollo del sistema

| FRAMEWORKS | PRINCIPALES BENEFICIOS |
|-----------------|---|
| Laravel 5.5 LTS | <ul style="list-style-type: none">✓ Reducción de costos y tiempos en el desarrollo y posterior mantenimiento de la aplicación creada.✓ Curva de aprendizaje relativamente baja si se compara con otros frameworks de PHP.✓ Flexible y adaptable no sólo al uso del sistema MVC tradicional, sino que para reducir las líneas de código propone lo que denomina "Routes with clousures".✓ Modular y con un amplio sistema de paquetes y drivers con el que se puede extender las funcionalidades de forma sencilla, |

| | |
|-----------|--|
| | <p>robusta y segura.</p> <ul style="list-style-type: none"> ✓ Reducción de costos y tiempos en el desarrollo y posterior mantenimiento de la aplicación creada. ✓ Curva de aprendizaje relativamente baja si se compara con otros frameworks de PHP. ✓ Flexible y adaptable no sólo al uso del sistema MVC tradicional, sino que para reducir las líneas de código propone lo que denomina "Routes with clousures". ✓ Modular y con un amplio sistema de paquetes y drivers con el que se puede extender las funcionalidades de forma sencilla, robusta y segura. |
| Angular 5 | <ul style="list-style-type: none"> ✓ Mejorar la productividad: el planteamiento de arquitecturas estándar, MVC, MVP, o cualesquiera que sean, si bien puede suponer una pequeña inversión al principio, repercute directamente en la productividad del proyecto. Si el framework elegido ayuda a conseguirlo, mucho mejor. Aquí es donde Angular es excelente, proporcionando controladores, servicios y directivas para organizar el proyecto. ✓ Menos errores de código: la búsqueda de errores de runtime en JavaScript puede ser una tarea imposible. TypeScript proporciona detección temprana de errores (en tiempo de compilación), y tipado fuerte de clases, métodos, así como de objetos y APIs JavaScript ya existentes. ✓ Calidad del software: Utilizar TypeScript y angular te ayuda a comprender mejor cómo funciona internamente JavaScript y hace que tu código sea mejor. Y sobre todo, facilita labores de testing, y mejora la comprensión y calidad del software en general. ✓ Son proyectos totalmente opensource, publicados en GitHub y abiertos a contribuciones. |

Fuente: Autor

3.4 Análisis de Impactos.

El presente proyecto se sustenta en un análisis técnico de los impactos que en las diferentes áreas o ámbitos que genere el proyecto en el contexto del mismo.

Para el presente análisis de impactos, se ha determinado varias áreas o ámbitos generales en las que el proyecto influirá positiva o negativamente; en el presente proyecto se ha determinado las áreas sociales, económico, ambiental y educativo.

Se selecciona un rango de niveles de impacto positivos y negativos de acuerdo con la siguiente tabla:

| Valor | Impacto |
|--------------|------------------------|
| -3 | Impacto alto Negativo |
| -2 | Impacto medio Negativo |
| -1 | Impacto bajo Negativo |
| 0 | No hay Impacto |
| 1 | Impacto bajo Positivo |
| 2 | Impacto medio Positivo |
| 3 | Impacto alto Positivo |

Se construye para cada una de las áreas una matriz en la que horizontalmente se ubica los niveles de impacto establecido, mientras que verticalmente se determina una serie de indicadores que permitirá tener información específica y puntual del área analizada.

A cada indicador, se asigna un nivel de impacto, positivo o negativo, a continuación, se encuentra la sumatoria de los niveles de impacto, la misma que será dividida para el número de indicadores establecidos en cada área, obteniéndose de esta manera matemáticamente el nivel de impacto del área de estudio.

Bajo cada matriz de área se realiza un breve análisis, en el que se selecciona y argumenta las razones, motivos, circunstancias que conllevaron a asignar el nivel de impacto al indicador de la matriz.

3.4.1 Impacto educativo

Tabla 32. Impacto educativo

| Valores | -3 | -2 | -1 | 0 | 1 | 2 | 3 |
|--|----------|----------|----------|----------|-------------|----------|----------|
| Influencia del comportamiento en el aprendizaje. | | | | | | x | |
| Grado de desconocimiento de las herramientas de desarrollo. | | | | | x | | |
| Mejora en los procesos de desarrollo de software | | | | | | x | |
| Influencia de nuevas herramientas y metodologías para el desarrollo de proyectos informáticos. | | | | | | x | |
| Total | 0 | 0 | 0 | 0 | 1 | 6 | 0 |
| | | | | | Suma | 7 | |

Fuente: Autor

Análisis:

Se puede observar que dentro del análisis el resultado de este impacto corresponderá al medio positivo. Como ya se evidenció en el diagnóstico; la demora de emisión de facturas de consumo del servicio de Agua en la Junta de Agua de Chorlaví genera malestar en los Usuarios de la misma. Esto conlleva también a un desconocimiento respecto al buen manejo de procesos mediante un sistema informático. Es así que esta propuesta en el ámbito Educativo pretende modelar la forma de llevar los procesos de esta Junta de Agua, en base a una serie de procedimientos informáticos, que están constituidos por diferentes módulos para controlar todos los procesos que realiza la Junta de Agua de Chorlaví, complementar y precisar los conocimientos de los usuarios a manipular el sistema y ante todo se suma el aporte metodológico y práctico que genera una nueva práctica en esta área de control.

3.4.1 Impactos Socio Cultural.

Tabla 33. Impactos Socio Culturales

| Valores | -3 | -2 | -1 | 0 | 1 | 2 | 3 |
|-------------------------------------|----------|----------|----------|----------|----------|-------------|----------|
| Transferencia de tecnologías | | | | | | | x |
| Formación | | | | | | x | |
| Bienestar comunitario | | | | | | | x |
| Total | 0 | 0 | 0 | 0 | 0 | 2 | 6 |
| | | | | | | Suma | 8 |

Fuente: Autor

Análisis

En este impacto del proyecto indica que, en corto, mediano y largo plazo, la transferencia tecnológica de los programas de ofimática, mantenimiento y sistemas de informática beneficia a la sociedad de la ciudad de Ibarra, realizando así la reducción de la brecha digital mejorando y agilizando los procesos de las juntas de Agua Potable de la ciudad de Ibarra.

Se establece que el Impacto Socio-Cultural será medio positivo. De esta forma la propuesta ayudará a mejorar la agilidad en los procesos de cobro y manejo de los implementos tanto internos como externos de la Junta de Agua potable de Chorlaví; además las estrategias vienen siendo un aporte para establecerse lineamientos o reglas de comportamiento positivos en el usuario final del servicio.

3.4.2 Impactos Económicos

Tabla 34. Impactos Económicos

| Valores | -3 | -2 | -1 | 0 | 1 | 2 | 3 |
|---|----------|----------|----------|----------|----------|-------------|----------|
| Incremento productividad | | | | | | | x |
| Disminución de pérdidas económicas | | | | | | x | |
| Bienestar comunitario | | | | | | | x |
| Total | 0 | 0 | 0 | 0 | 0 | 2 | 6 |
| | | | | | | Suma | 8 |

Fuente: Autor

Análisis

El proyecto beneficiará económicamente a la Junta ya que incrementa la productividad de ella, de igual manera al llevar el proceso de control de forma automática mediante el sistema informático permite llevar un control eficiente de las existencias, generando de esta manera un ambiente amigable con los usuarios que hacen uso de este servicio.

3.4.3 Impacto Ambiental

| Valores | -3 | -2 | -1 | 0 | 1 | 2 | 3 |
|---|----------|----------|----------|----------|----------|-------------|----------|
| Disminución de útiles de oficina | | | | | | | x |
| Total | 0 | 0 | 0 | 0 | 0 | 0 | 3 |
| | | | | | | Suma | 3 |

Fuente: Autor

El proyecto beneficiará al ambiente debido a que se eliminará los útiles de oficina como papel, grapas, tintas, etc.

Conclusiones

- Los framework utilizados para el diseño y desarrollo son muy confiables y potentes puesto que están siendo probadas y actualizadas constantemente por profesionales y personas naturales con conocimientos de tecnologías de información.
- Las herramientas utilizadas permitieron el desarrollo de un sistema de databinding mediante directivas y filtros, permitiendo crear pequeños componentes que facilitan a la capa de presentación.
- Manejo de una estructura en base a componentes, que básicamente se compone del componente, el selector que parecerá en forma de tag en el código HTML y de la vista y una estructura MVC en el lado del BackEnd.
- Manejo de una sintaxis refinada y expresiva para crear código de forma compleja, evitando el “código espagueti” y permitiendo multitud de funcionalidades.
- Análisis del uso de las librerías a utilizar mediante los gestores de dependencias de cada framework.
- Manejo de rutas a través de almacenamiento en la cache de rutas, middleware.
- Intercambio de recursos de origen cruzado y es una especificación que permite a los navegadores modernos solicitar (y recibir) datos de un dominio distinto al que sirve la página que realizó la solicitud.
- Rendimientos eficaces permitiendo desarrollar gran variedad de aplicaciones, aprovechando herramientas como el templating, el two-way data binding, pluralization, uso de módulos, dependency injection.
- La implementación de estas herramientas permite realizar testing unitarios a través de karma, permitiendo la estabilidad del sistema.
- El desarrollo del presente proyecto fue posible gracias a la información facilitada por los miembros de la junta administradora de agua potable san José de Chorlaví de la ciudad de Ibarra

- La administración del sistema resulta muy versátil y confiable puesto que es una aplicación web, lo que permite reducir recursos como el consumo de memoria, almacenamiento del ordenador y costos.
- El brindar a los operadores una aplicación que permita registrar las lecturas y órdenes de instalación facilitan en gran medida el óptimo desempeño de estos en el campo, puesto que los datos obtenidos son ingresados automáticamente en la base de datos de la Junta.

Recomendaciones

- La curva de aprendizaje se hace un poco dura, sobre todo por la falta de documentación y la elección de nombres, pero luego resulta bastante productivo gracias a cosas como el databinding y el contenedor de inversión de control, así que hay que investigar un poco más respecto a este tema.
- La tecnología que nos proporciona nuestra querida inyección de dependencias y se integra con muchas otras tecnologías, presenta un rendimiento óptimo y mucho más simple y comprensible, nos permite tener todo nuestro código bajo una red de seguridad de tests.
- Angular es una solución completa que incluye prácticamente todos los aspectos que puedes necesitar para crear una aplicación cliente en javascript. Esto incluye la generación de vistas, el uso de databinding, las rutas, la organización de componentes en módulos, la comunicación con el servidor.
- Es recomendable acceder al sitio oficial el cual provee de información bastante útil para conocer el funcionamiento y diseño de sistemas basados en el framework utilizado.
- Es importante en caso de existir duda en cuanto a la emisión de reportes verificar los parámetros de consulta antes de ejecutar la consulta para evitar posibles inconvenientes en cuanto al rendimiento.

Bibliografía

- AngularJS. (2018). Obtenido de <https://docs.angularjs.org/guide/controller>
- Bower. (2018). Obtenido de <https://bower.io/>
- CampusMVP. (2017). Obtenido de <https://www.campusmvp.es/recursos/post/los-5-mejores-frameworks-de-javascript-en-2017.aspx>
- Getcomposer. (2018). Obtenido de <https://getcomposer.org/doc/00-intro.md#system-requirements>
- Git. (2018). Obtenido de <https://git-scm.com/>
- Gómez, K. (2017). Top 5 Metodologías de Desarrollo de Software. Obtenido de <https://www.megapractical.com/blog-de-arquitectura-soa-y-desarrollo-de-software/metodologias-de-desarrollo-de-software>
- Grunt. (2018). Obtenido de <https://gruntjs.com/>
- Laravel. (2017). Obtenido de <https://entwickler.de/online/development/fast-forward-laravel-node-js-579812126.html>
- LibrosWeb. (2017). Obtenido de http://librosweb.es/libro/javascript/capitulo_1.html
- Node. (2018). Obtenido de <https://nodejs.org/es/>
- Palacios, D. (2017). Por qué Laravel NO es un framework MVC y tú deberías olvidarte de MVC. Obtenido de <https://styde.net/porque-laravel-no-es-mvc-y-tu-deberias-olvidarte-de-mvc/>
- Perez, D. (2007). que son las bases de datos . Obtenido de <http://www.maestrosdelweb.com/que-son-las-bases-de-datos/>
- Proyectosagiles. (2017). Obtenido de <https://proyectosagiles.org/que-es-scrum/>
- Robles, V. (2018). Obtenido de <https://www.udemy.com/angular-5-y-typescript-curso-de-introduccion-paso-a-paso/>
- Stackoverflow. (2016). Obtenido de <https://stackoverflow.com/questions/33148455/mvc-in-php-framework>
- Websystem. (2017). Obtenido de <https://websystemperu.com/>
- Yeoman. (2018). Obtenido de <http://yeoman.io/>

- Aguilar, P. A. (01 de 02 de 2013). Repositorio. Obtenido de dspace:
<http://dspace.ups.edu.ec/bitstream/123456789/4211/1/UPS-CT002598.pdf>
- Aguirre, R. G. (11 de Octubre de 2010). es.scribd.com. Obtenido de
<http://es.scribd.com/doc/39106342/Metodologia-Sistemica-de-Klir#scribd>
- Álvarez, K. (2010). Toma de decisiones. Recuperado el 6 de Mayo de 2016, de Toma de decisiones: <http://keytdsig.blogspot.com/2010/03/dss-y-gdss.html>
- Álvarez, M. A. (2 de Enero de 2014). Qué es MVC. Recuperado el 3 de Octubre de 2016, de Desarrollo Web: <http://www.desarrolloweb.com/articulos/que-es-mvc.html>
- Aranibar, N. (2014). Monografias. Recuperado el 1 de 4 de 2016, de
<http://www.monografias.com/trabajos88/mysql-worckbench/mysqlworckbench.shtml>
- Armero, R. (s.f.). Recuperado el 13 de febrero de 2015, de
<http://www.laempresaeninternet.com/contabilidad-y-finanzas/la-contabilidad-en-elcomercio-electronico.html>
- Astros, I.(2010). Monografias. Recuperado el 20 de Febrero de 2014, de
<http://www.monografias.com/trabajos94/sistema-de-informacion-gerencialestrategico/sistema-de-informacion-gerencial-estrategico.shtml#ixzz3TvqzkXHL>
- Cornford, T., & Shaikh, M. (1 de Julio de 2013). is1060_ch1-4: Introduction to information systems. Recuperado el 10 de Mayo de 2016, de University of London:
http://www.londoninternational.ac.uk/sites/default/files/programme_resources/lse/lse_pdf/subject_guides/is1060_ch1-4.pdf
- Domínguez, M. (3 de Agosto de 2016). Qué es Bootstrap y cuáles son sus ventajas. Recuperado el 12 de Octubre de 2016, de Punto Abierto:
<http://puntoabierto.net/blog/que-es-bootstrap-y-cuales-son-sus-ventajas>
- Espinoza, A. V. (15 de Abril de 2008). METODO DEDUCTIVO Y METODO INDUCTIVO. Recuperado el 12 de Abril de 2016, de Colbert Garcia:
<http://colbertgarcia.blogspot.com/2008/04/metodo-deductivo-y-metodoinductivo.html>
- Ebel, S. (2008). Fundamentos de la técnica de automatización. Alemania: Festo Didactic GmbH & Co. KG.
- Gabriela, V. S. (02 de 06 de 2014). Repositorio . Obtenido de espe:
<http://repositorio.espe.edu.ec/bitstream/21000/8873/1/T-ESPE-048054.pdf>

- Sanchez, A. (12 de Noviembre de 2016). Laravel 5. Recuperado el 21 de Noviembre de 2016, de GitBook: <https://ajgallego.gitbooks.io/laravel-5/content/>
- Garcia Arriaza, M. (s.f.). Técnicas de investigación (entrevista). Recuperado el 12 de Abril de 2016, de Monografias: <http://www.monografias.com/trabajos101/tecnicasinvestigacion-entrevista/tecnicas-investigacion-entrevista.shtml>
- Gestiopolis. (16 de Julio de 2002). Encuesta, cuestionario y tipos de preguntas. Recuperado el 12 de Abril de 2016, de Gestiopolos: <http://www.gestiopolis.com/encuesta-cuestionario-y-tipos-de-preguntas/>
- Palacio, J., & Ruata, C. (27 de Enero de 2011). Scrum Manager Gestion de proyectos. Recuperado el 17 de Julio de 2016, de Ministerio de Costa Rica: <http://www.hacienda.go.cr/cifh/sidovih/spaw2/uploads/images/file/Gestión%20de%20proyectos.pdf>
- Patricia, G. (2005). Ecured. Recuperado el 15 de Febrero de 2015, de http://www.ecured.cu/index.php/Sistema_Gestor_de_Base_de_Datos
- Patti, D. (2007). Monografias. Recuperado el 06 de Febrero de 2015, de <http://www.monografias.com/Computacion/Programacion/>
- Perez, P. (2012). DeideaAapp. Recuperado el 03 de Marzo de 2015, de <http://deideaaapp.org/tipos-de-aplicaciones-moviles-y-sus-caracteristicas/>
- Pimienta, P. (5 de Mayo de 2014). Tipos de aplicaciones móviles y sus características. Recuperado el 6 de Mayo de 2015, de De Idea Aapp: <https://deideaaapp.org/tipos-deaplicaciones-moviles-y-sus-caracteristicas/>
- Puente, W. (2006). TÉCNICAS DE INVESTIGACIÓN. Recuperado el 12 de Abril de 2014, de Portal de Relaciones Públicas: <http://www.rppnet.com.ar/tecnicasdeinvestigacion.htm>
- Quesada Allude, X. (8 de Junio de 2009). Introducción a la estimación y planificación ágil. Recuperado el 20 de Junio de 2016, de Proyectos Agiles: <https://proyectosagiles.org/2009/06/08/introduccion-estimacion-planificacion-agil/>
- Raul. (15 de Febrero de 2012). Los 4 tipos de sistemas informaticos. Recuperado el 11 de Abril de 2016, de Sistema Platonico: <http://sistemaplatonico.blogspot.com/2012/02/los-4-tipos-de-sistemas-informaticos.html>

Remon, L. M. (Abril 2013). Desarrollo de Aplicaciones con JAVA. Lima, Peru: Macro EIRL.

Rodríguez, J. M. (2003). Recuperado el 5 de Febrero de 2015, de
<http://www.ual.es/~jmrodri/sistemasdeinformacion.pdf>

Schortborgh, R. J. (Junio de 2010). Recuperado el 29 de febrero de 2015, de
<http://www.monografias.com/trabajos-pdf4/diagramas-secuencia/diagramassecuencia.shtml>

Storti, G., Rios, G., & Campodonico, G. (2007). Recuperado el 6 de Marzo de 2016, de
Colegio Manuel Belgrano:
http://www.belgrano.esc.edu.ar/matestudio/carpeta_de_access_introduccion.pdf

Tonanzin, A. J. (s.f.). Fundamentos de Investigacion. Recuperado el 2 de Abril de 20016, de
shounyalamilla: <http://shounyalamilla.blogspot.com/p/23-tipos-de-metodosinductivo-deductivo.html>

Anexos

(Manuales técnico y usuario)

Anexo A Diagramas de casos de uso del módulo de administración

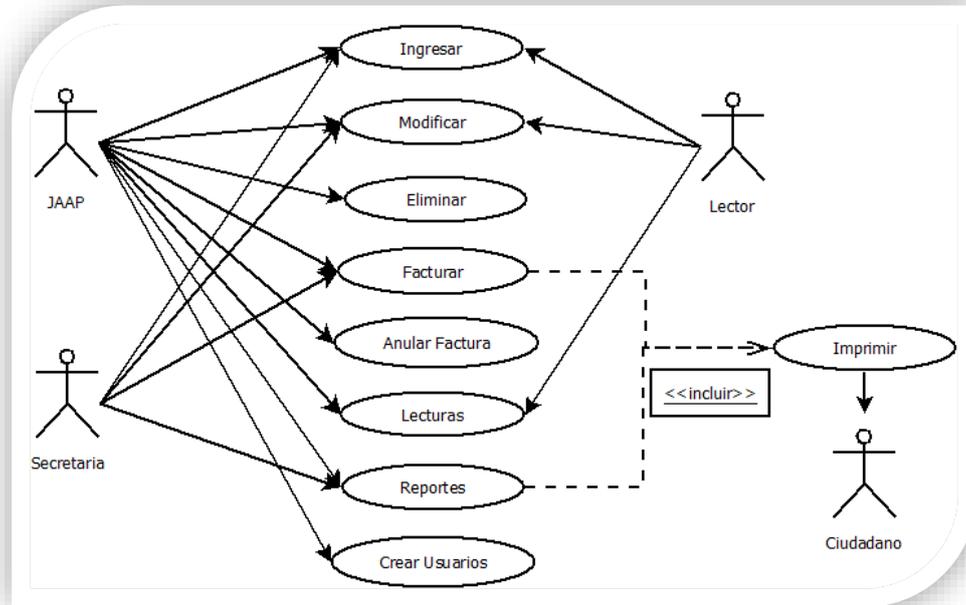


Figura 22. Diagrama de casos de uso sistema de facturación

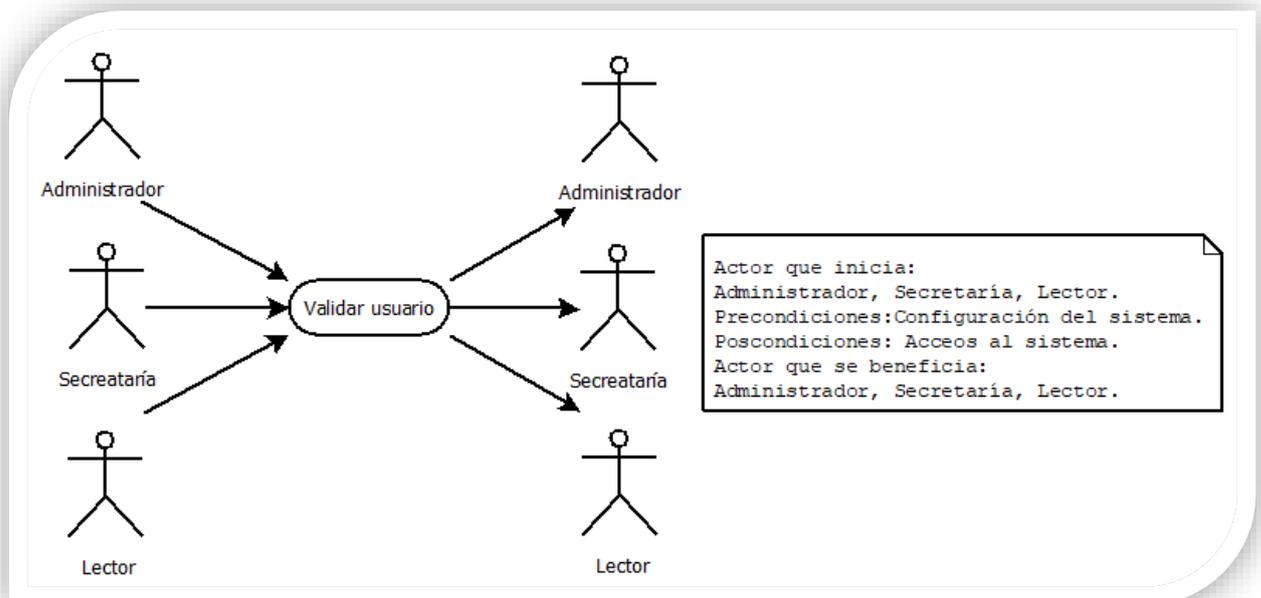


Figura 23. Validar usuario

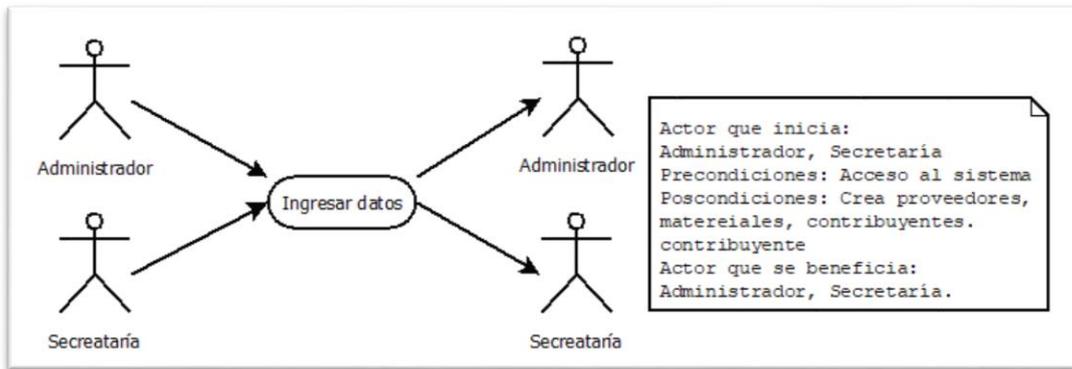


Figura 24. Registro de datos

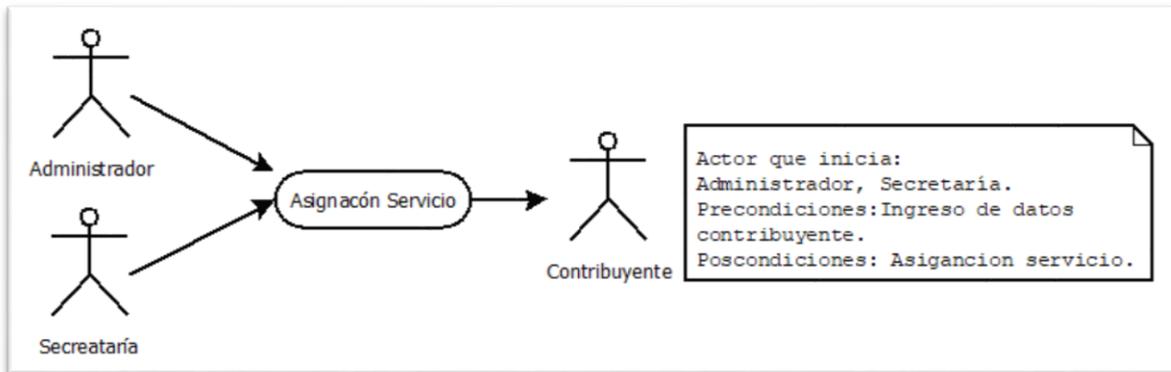


Figura 25. Asignación servicio

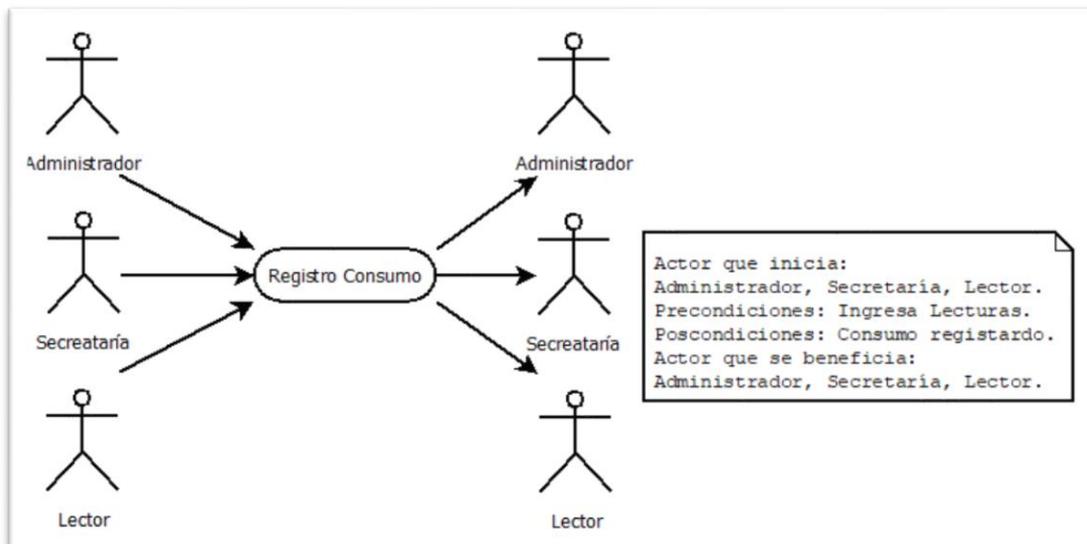


Figura 26. Registro Consumo

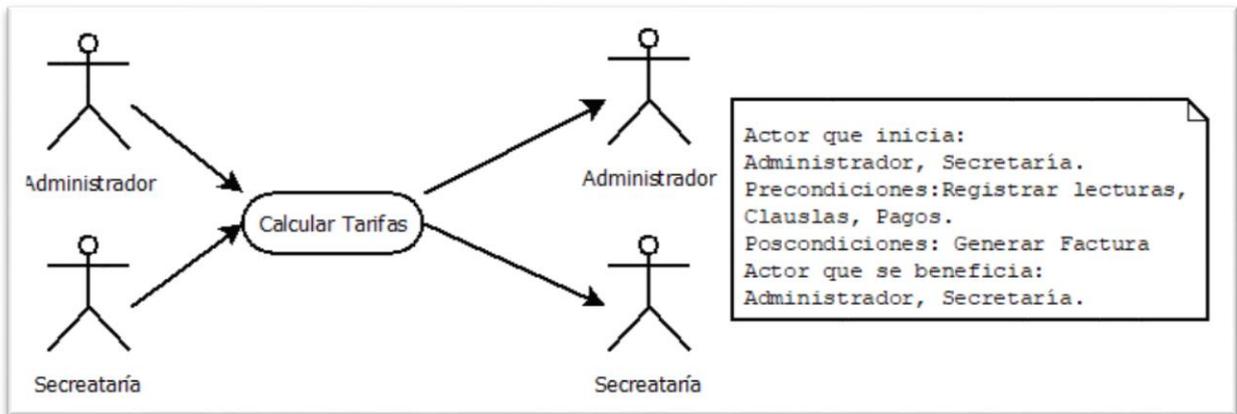


Figura 27. Calcular tarifas

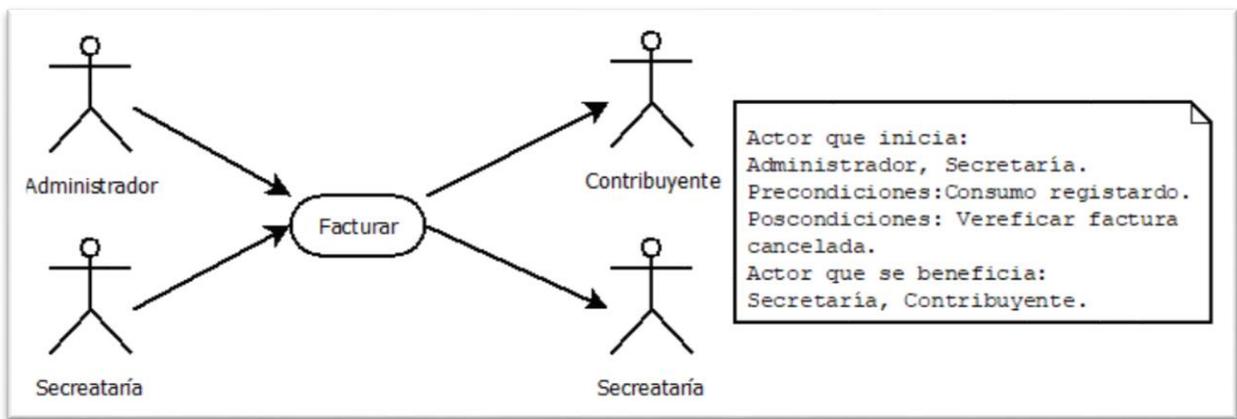


Figura 28. Facturas

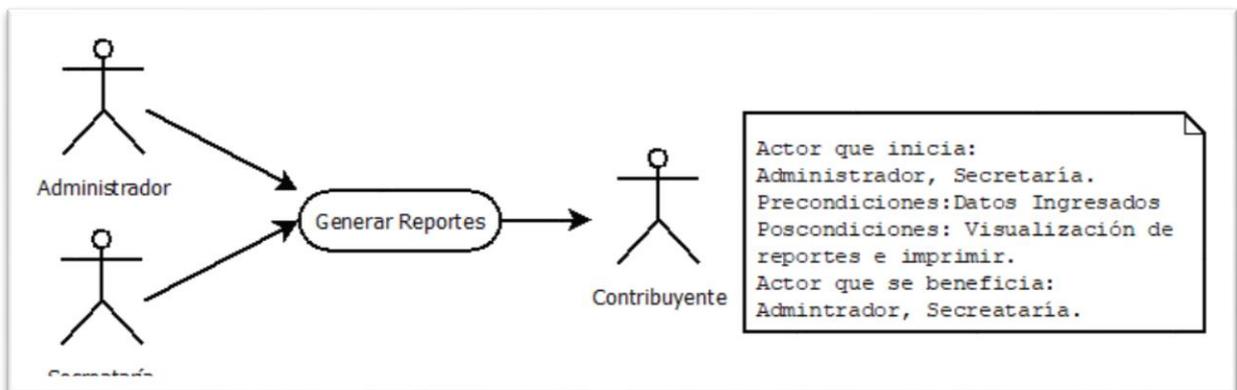


Figura 29. Generar reportes

Anexo B Diagramas de casos de Uso

Contribuyente

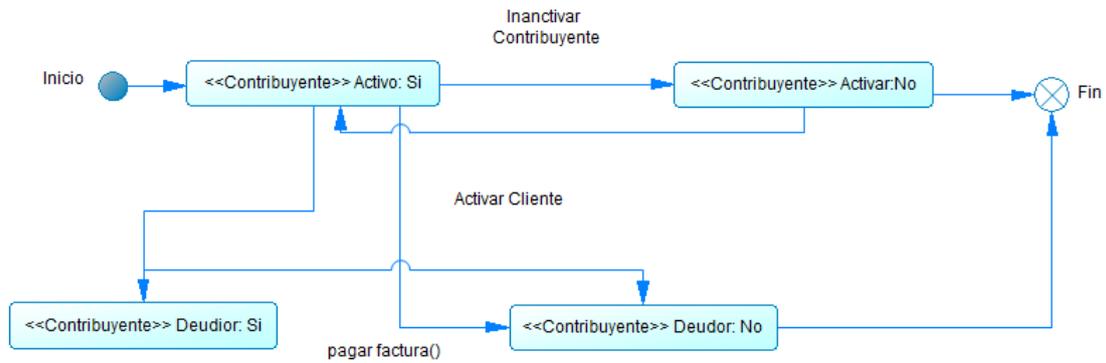


Figura 30. Diagrama de estado contribuyente

Asignación servicio

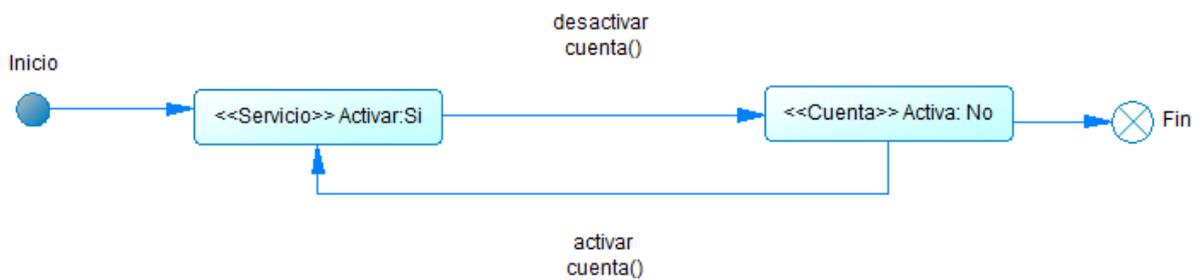


Figura 31. Diagrama de estado del servicio

Factura

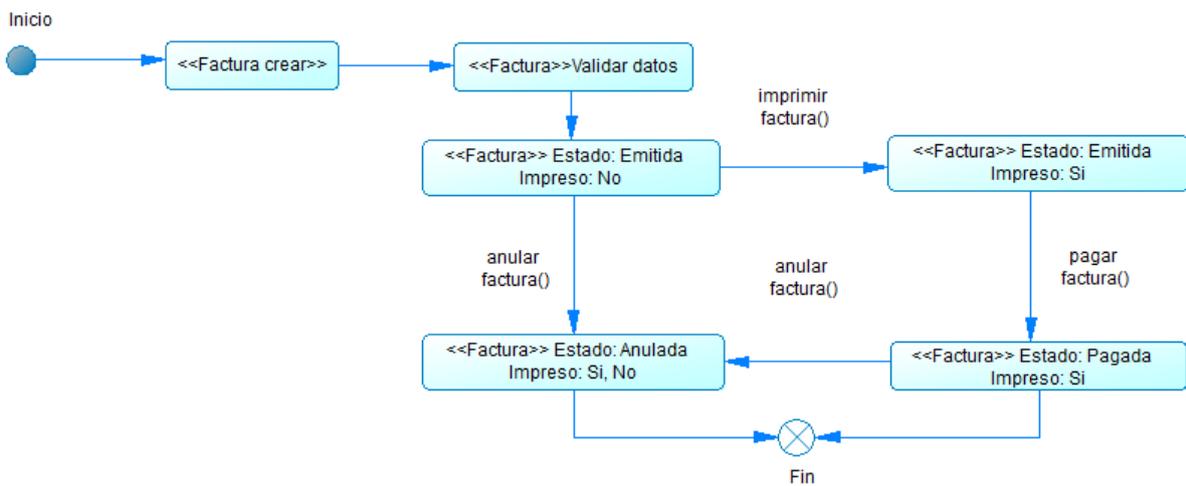


Figura 32. Diagrama de estado factura

Anexo C Diagrama de secuencias

Diagrama de secuencias

Esta muestra la forma en que los objetos se comunican entre sí en transcurrir el tiempo. El diagrama de secuencia es un tipo de diagrama usado para modelar interacción entre objetos en un sistema según UML. En inglés se pueden encontrar como "sequence diagram", "event-trace diagrams".

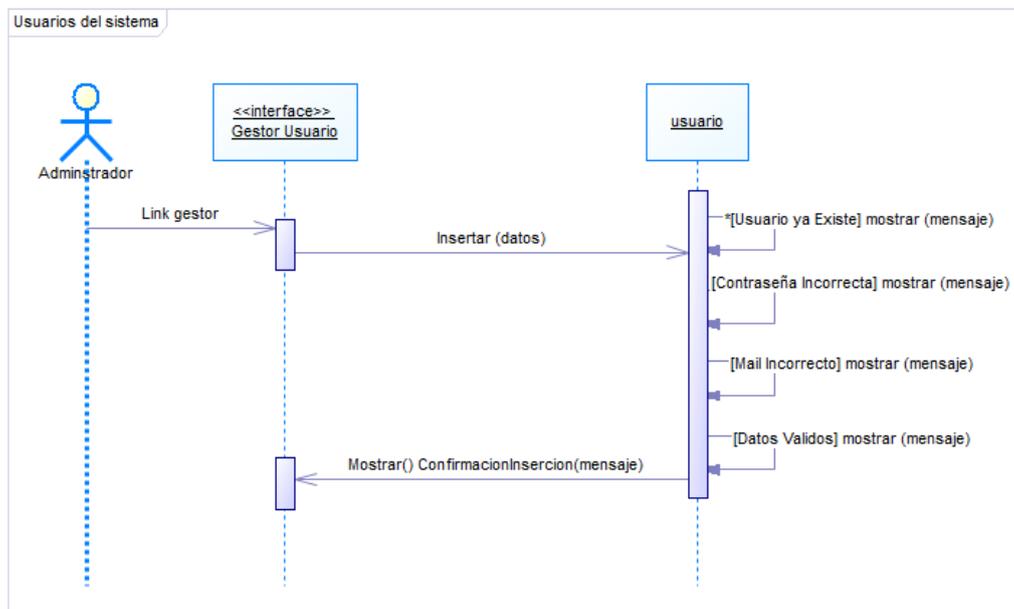


Figura 33. Diagrama de secuencia de usuarios del sistema

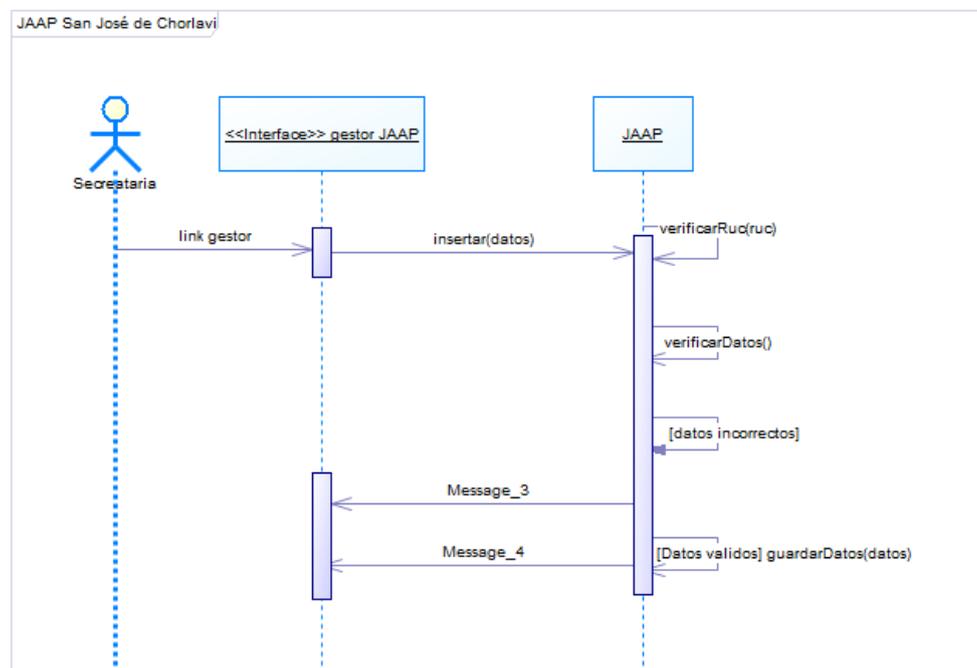


Figura 34. Diagrama de secuencia de JAAP

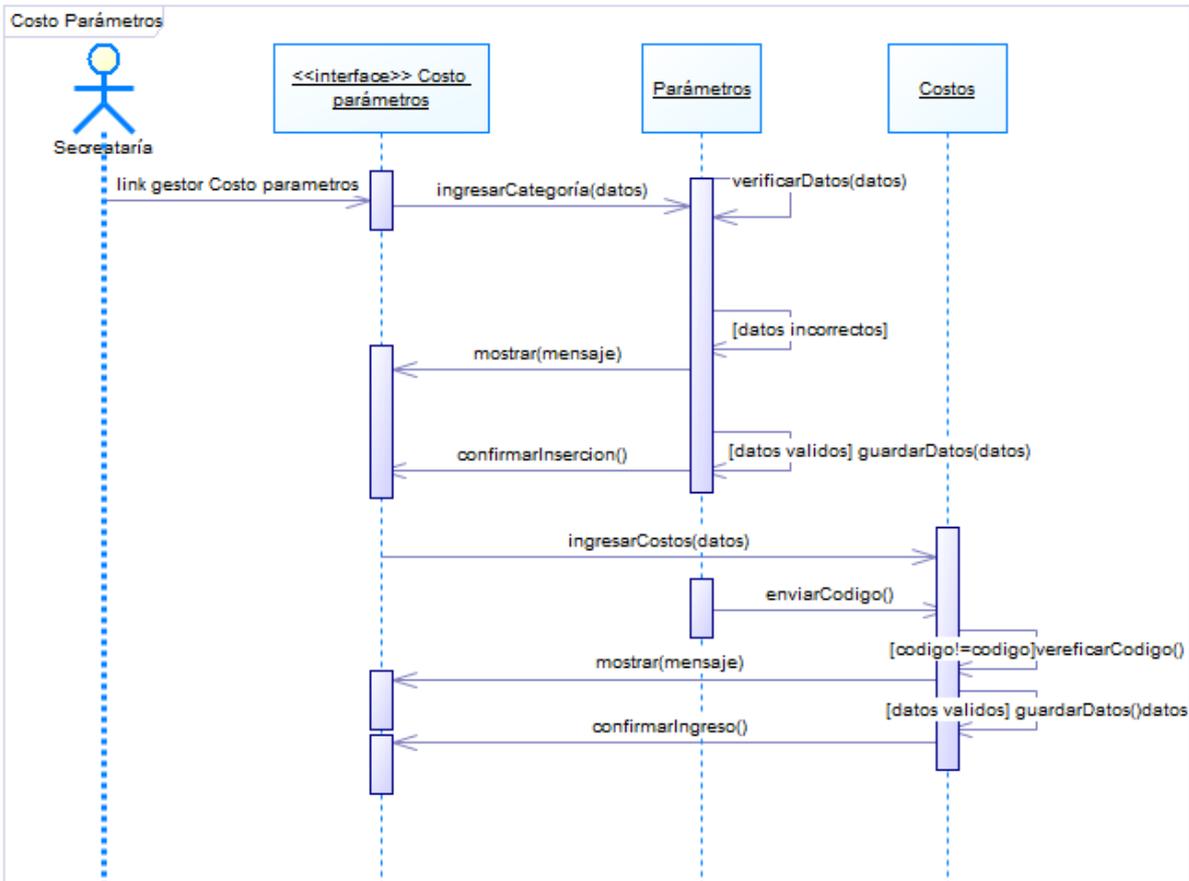


Figura 35. Diagrama de secuencia de Costo Parámetros

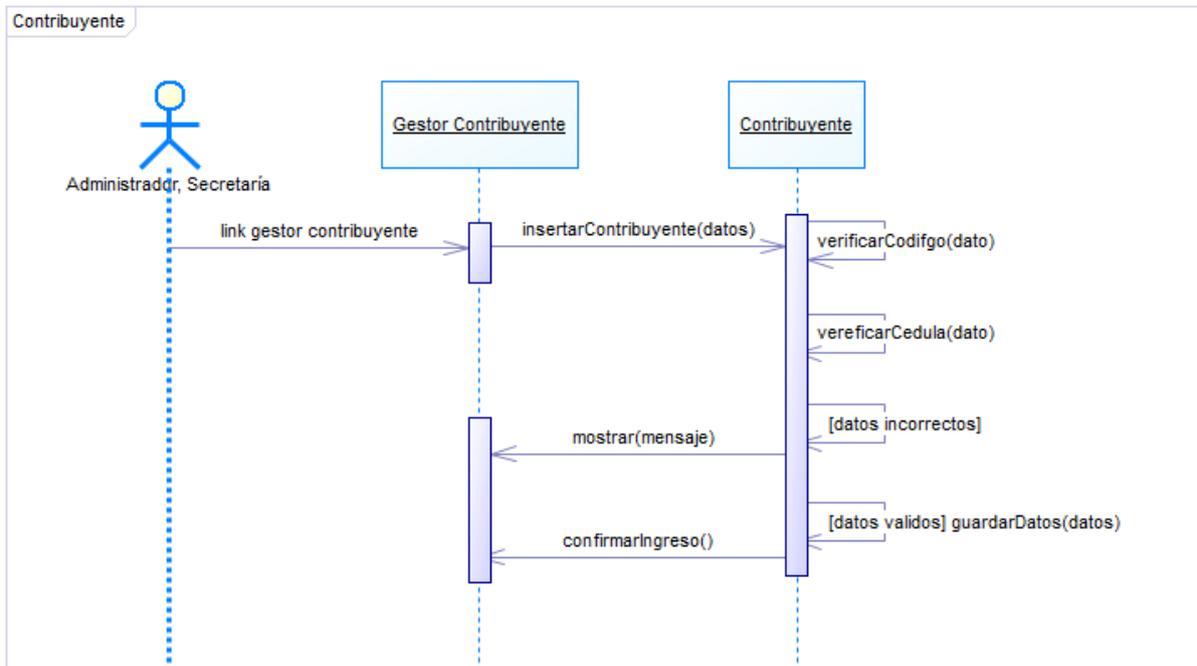


Figura 36. Diagrama de secuencia de Contribuyente

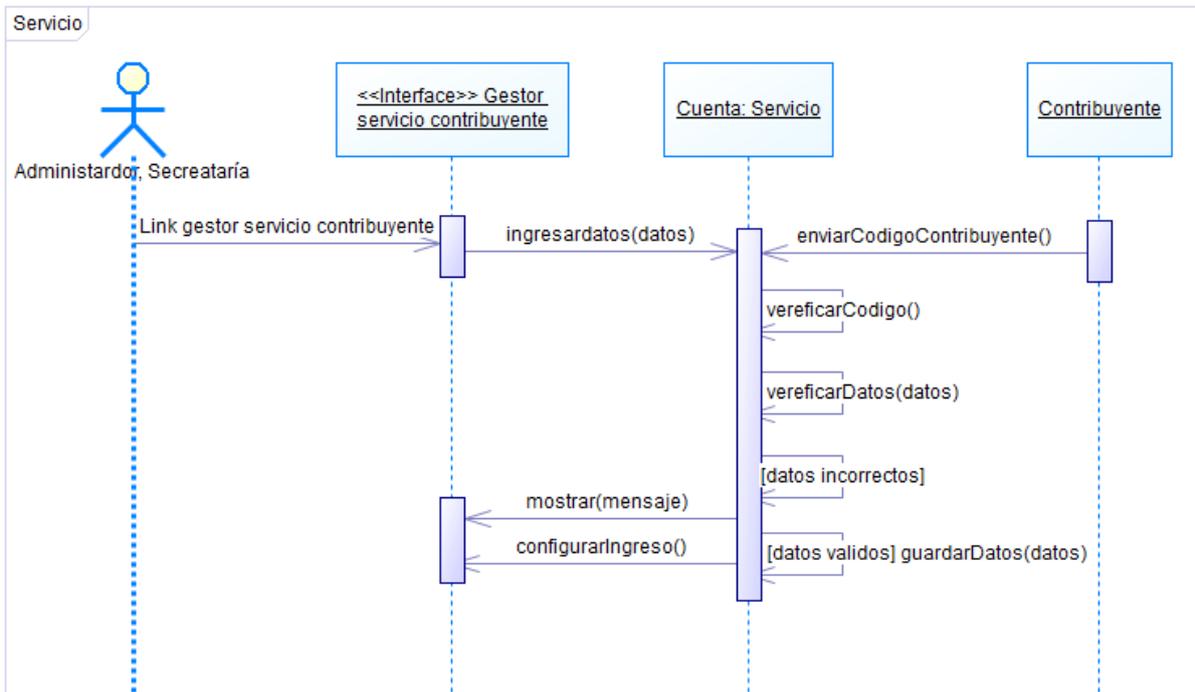


Figura 37. Diagrama de secuencia de Activación servicio

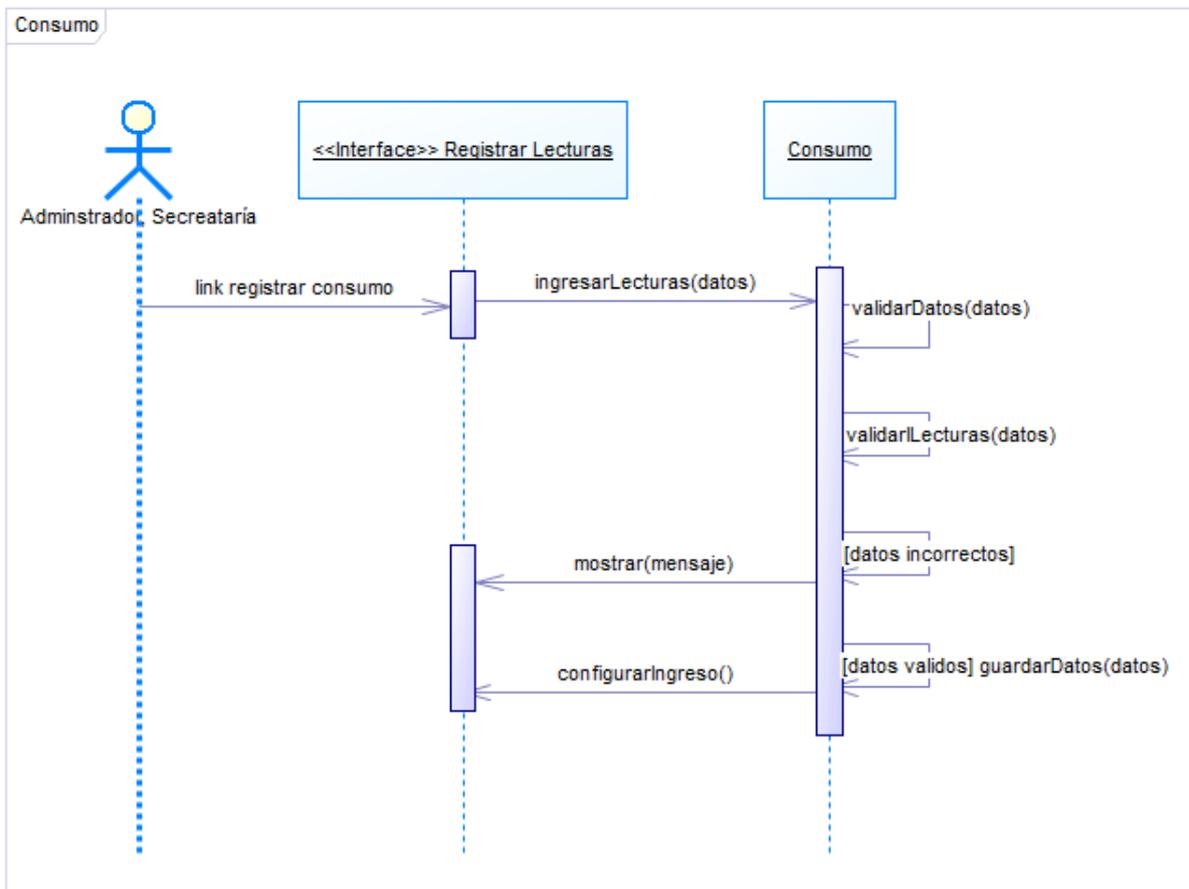


Figura 38. Diagrama de secuencia de Consumo

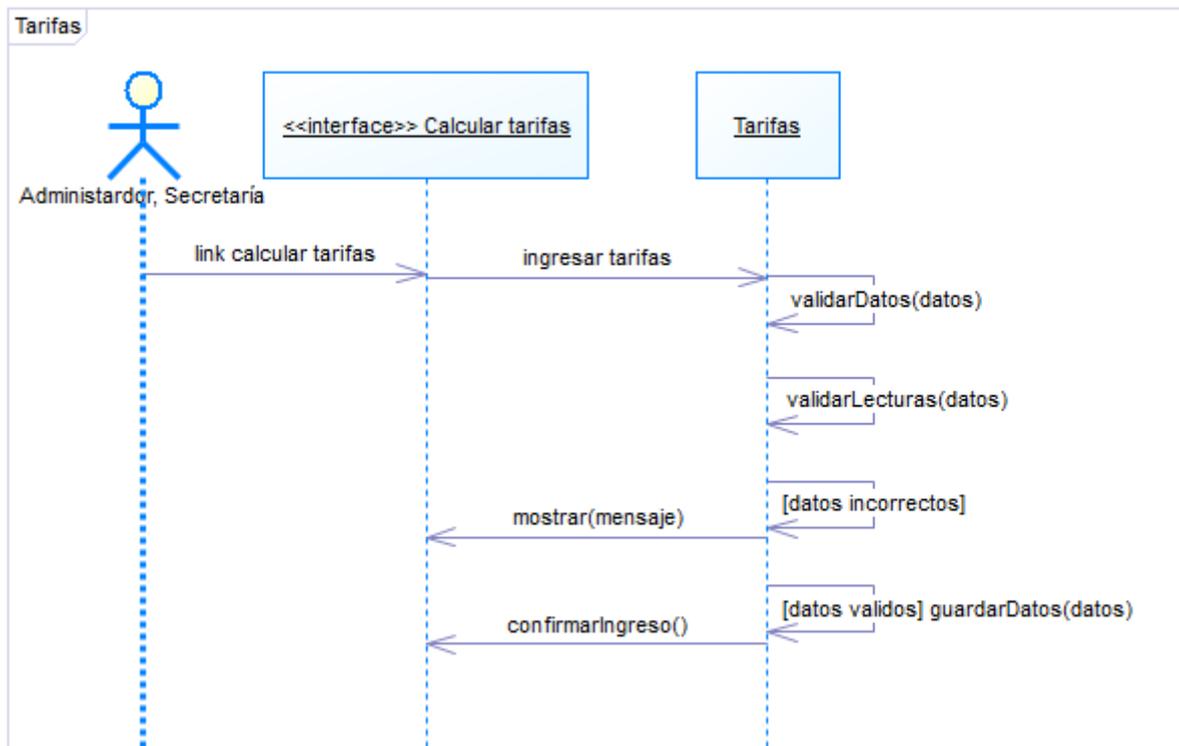


Figura 39. Diagrama de secuencia tarifas

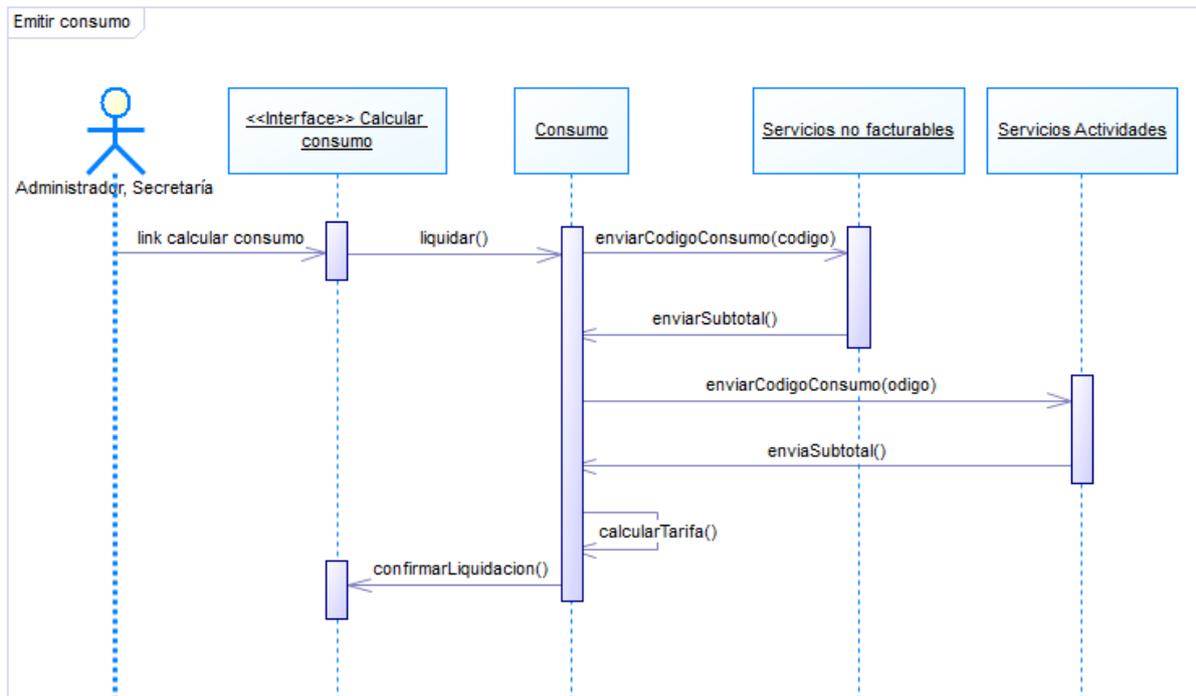


Figura 40. Diagrama de secuencia Emitir consumo

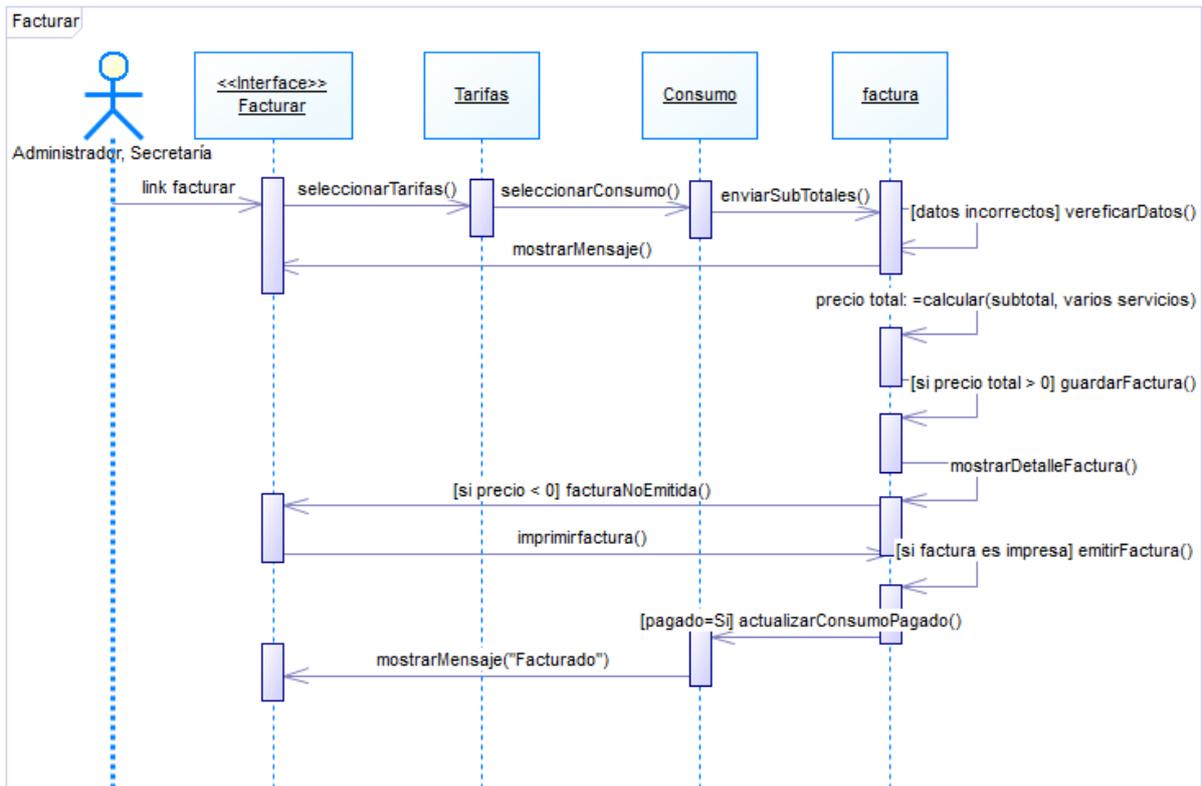


Figura 41. Diagrama de secuencia de Facturar

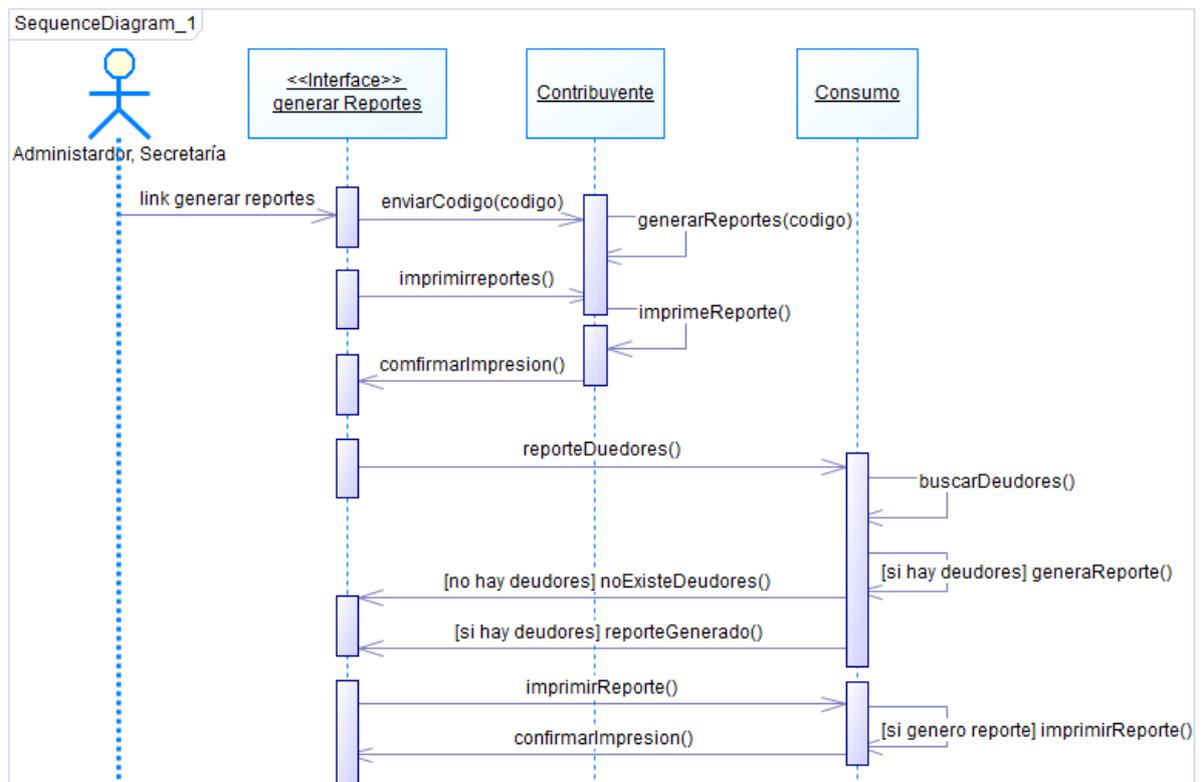


Figura 42. Diagrama de secuencia de Reportes

Formato encuesta



UNIVERSIDAD TÉCNICA DEL NORTE

FORMATO DE ENCUESTA A LOS USUARIOS DE LA JUNTA DE AGUA DE CHORLAVÍ.

1. ¿Luego de Implementar el sistema de la junta de agua está usted de acuerdo con la forma en la que se realiza el cobro de planillas?

Muy de acuerdo De acuerdo Desacuerdo

2. ¿Considera usted que la atención brindada por la Junta de agua es eficiente a raíz de la implementación del sistema?

Muy de acuerdo De acuerdo Desacuerdo

3. ¿Considera que el tiempo de respuesta por parte de la Junta a una solicitud de información detallada sobre las planillas es la adecuada?

Inmediata Lenta No llega

4. ¿Considera usted que el valor cancelado por concepto de planillas está acorde con el consumo de agua ya que las tomas de lecturas son registradas en un sistema para calcular el cobro del servicio?

SI NO

5. ¿Cuándo proceden a la lectura de su medidor qué medio utilizan para registrar el consumo actualmente?

Libreta Celular Tablet

6. ¿Considera usted que la implementación del sistema informático para registrar los cobros permitió brindar un mejor servicio y ofrecer información oportuna a todos los usuarios de la Junta Administradora de Agua?

SI NO

7. ¿Le parece beneficios el servicio que ofrece la JAAP, mediante las consultas de planillas a través de internet?

Muy beneficioso Beneficioso Poco Beneficioso

Formato entrevista



**UNIVERSIDAD TÉCNICA DEL NORTE
FORMATO DE ENTREVISTA A LOS DIRIGENTES DE LA JUNTA DE AGUA DE
CHORLAVÍ.**

1. ¿En la actualidad, con cuántos usuarios cuentan la junta de agua potable?

2. ¿Cuáles son las actividades que se realizan en el proceso de facturación en la junta agua?

3. ¿Cuentan con algún sistema informático que facilite el proceso de facturación?

4. ¿Cómo se realiza el proceso de cobro de planillas de consumos de agua?

5. ¿Considera oportuno automatizar el proceso de la facturación de la junta de agua?

6. ¿Qué procesos considera críticos y que puedan ser automatizados?

7. ¿La junta de agua respalda de la información de los recibos de consumo y multas?

8. ¿Cree usted que sería oportuno generar un sistema para administración de la información de todos los procesos de la Junta de Agua?
