



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

ESCUELA DE INGENIERÍA EN SISTEMAS

“DESARROLLO DE UNA APLICACIÓN WEB/MÓVIL PARA EL REGISTRO DE CONSUMO/PAGO DE LOS USUARIOS DE LA JUNTA ADMINISTRADORA DE AGUA POTABLE ANGAHUANA ALTO, APLICANDO TDD”

TRABAJO DE TITULACIÓN

TIPO: PROYECTO TÉCNICO

Presentado para optar al grado académico de:

INGENIERO EN SISTEMAS INFORMÁTICOS

AUTOR: TISALEMA PAOQUIZA TUPAC AMARUC

TUTORA: Ing. ALEJANDRA OÑATE

Riobamba – Ecuador

2019

© 2019, Tupac Amaruc Tisalema Poaquiza

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

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

El Tribunal del Trabajo de Titulación certifica que: El trabajo de investigación: Tipo Proyecto Técnico “DESARROLLO DE UNA APLICACIÓN WEB MÓVIL PARA EL REGISTRO DE CONSUMO, PAGO DE LOS USUARIOS DE LA JUNTA ADMINISTRADORA DE AGUA POTABLE ANGAHUANA ALTO, APLICANDO TDD”, de responsabilidad del señor Tupac Amaruc Tisalema Poaquiza, ha sido minuciosamente revisado por los miembros del Trabajo de Titulación, quedando autorizada su presentación.

FIRMA

FECHA

Dr. Julio Santillán

**VICEDECANO DE LA FACULTAD DE
INFORMÁTICA Y ELECTRÓNICA**

Ing. Patricio Moreno

**DIRECTOR DE LA ESCUELA
DE INGENIERÍA
EN SISTEMAS**

Ing. Alejandra Oñate

**DIRECTORA DE TRABAJO DE
TITULACIÓN**

Ing. Eduardo Villa

MIEMBRO DEL TRIBUNAL

“Yo, Tupac Amaruc Tisalema Poaquiza, soy responsable de las ideas, doctrinas y resultados expuestos en el presente Trabajo de Titulación y el patrimonio intelectual expuesto en el documento pertenece a la Escuela Superior Politécnica de Chimborazo”.

Tupac Amaruc Tisalema Poaquiza

DEDICATORIA

Dedicado a mi familia por ser el pilar fundamental en todo lo que soy, en toda mi educación, tanto académica, como de la vida, a mis amigos los cuales supieron brindarme su amistad sin ningún compromiso, aquellas personas que sin conocerme me dieron palabras de aliento para seguir adelante por su apoyo incondicional perfectamente mantenido a través del tiempo.

Tupac

AGRADECIMIENTO

Agradezco a Dios por haberme acompañado y guiado a lo largo de mi carrera, por ser mi fortaleza en los momentos de debilidad y brindarme una vida llena de aprendizajes, experiencias y sobre todo felicidad. Además le doy gracias a mis padres Enrique y Felipa por apoyarme incondicionalmente en todo momento, por los valores que me han inculcado, y por haberme dado la oportunidad de tener una excelente educación en el transcurso de mi vida. Sobre todo por ser un excelente ejemplo de vida a seguir. A mi hermano Alex por ser parte importante de mi vida y representar la unidad familiar, por sus palabras y apoyo incondicional durante el proceso educativo, y por supuesto a todos mis familiares quienes emocionalmente me apoyaron con sus palabras de aliento.

A la ESPOCH, y a mis maestros por haberme brindando una educación de calidad durante mi carrera estudiantil, lo cual se ha convertido en un factor fundamental para mi vida profesional.

Tupac

ÍNDICE DE ABREVIATURAS

IDE	Integrated Development Environment
GPL	General Public License
HTML	HyperText Markup Language
BD	Base de Datos
IIS	Internet Information Server
IP	Internet Protocol
SSL	Secure Sockets Layer
TSL	Transport Layer Security
JSP	Java Server Pages
JEE	Java Enterprise Edition
API	Application Programming Interface
J2SE	Java 2 Standard Edition
JSE	Java Estándar Edition
J2ME	Java 2 Micro Edition
J2EE	Java 2 Enterprise Edition
KVM	Kilobyte Virtual Machine
CVM	Compact Virtual Machine
CDC	Connected Device Configuration
CLDC	Connected Limited Device Configuration
PDF	Portable Document Format
XLS	Extensión de los Archivos Ficheros
CSV	Comma Separated Values
XML	Extensible Markup Language
JAR	Java Archive
TDD	Test Driven Development

MVCC	Modelo Vista Controlador
FTP	File Transfer Protocol
SCRUM	Desarrollo de Software
HTTPS	Hypertext Transfer Protocol Secure
XHTML	Extensible Hyper Text Markup Language
IMAP	Protocolo de Acceso a Mensajes de Internet
NNTP	Network News Transport Protocol
POP	Protocolo de Oficina Postal
CRUD	Crear, Leer, Actualizar y Borrar
CDC	Connected Device Configuration
CLDC	Connected Limited Device Configuration

TABLA DE CONTENIDOS

RESUMEN.....	xx
ABSTRACT.....	xxi
INTRODUCCIÓN	1

CAPÍTULO I

1.	MARCO TEÓRICO REFERENCIAL	9
1.1.	Fundamentación Teórica.....	9
1.2.	Procesos de control de consumo de agua según la Ley Orgánica de Recursos Hídricos, Usos y Aprovechamiento del Agua (LORHUYA)	9
<i>1.2.1.</i>	<i>Juntas Administradoras de Agua Potable y Saneamiento</i>	<i>9</i>
<i>1.2.2</i>	<i>Consumidor</i>	<i>10</i>
<i>1.2.2.</i>	<i>Tarifario</i>	<i>10</i>
1.3.	Herramientas de Desarrollo.....	10
<i>1.3.1.</i>	<i>Netbeans</i>	<i>11</i>
<i>1.3.1.1.</i>	<i>Definición.....</i>	<i>11</i>
<i>1.3.1.2.</i>	<i>Características.....</i>	<i>11</i>
<i>1.3.2.</i>	<i>Patrón Modelo-Vista-Controlador (MVC)</i>	<i>11</i>
<i>1.3.2.1.</i>	<i>Definición.....</i>	<i>12</i>
<i>1.3.2.2.</i>	<i>Elementos.....</i>	<i>12</i>
<i>1.3.3.</i>	<i>Java Server Faces (JSF)</i>	<i>13</i>
<i>1.3.3.1.</i>	<i>Definición.....</i>	<i>14</i>
<i>1.3.3.2.</i>	<i>Características.....</i>	<i>14</i>
<i>1.3.4.</i>	<i>Primefaces</i>	<i>14</i>
<i>1.3.4.1.</i>	<i>Definición.....</i>	<i>14</i>
<i>1.3.4.2.</i>	<i>Características.....</i>	<i>15</i>

1.3.5.	Android	15
1.3.5.1.	Definición.....	15
1.3.5.2.	Características.....	15
1.3.5.3.	Ventajas	16
1.3.5.4.	Desventajas.....	16
1.3.6.	Android Estudio	16
1.3.6.1.	Definición.....	16
1.3.6.2.	Características.....	17
1.3.6.3.	Requerimientos del Sistema.....	17
1.3.7.	PostgresSQL	18
1.3.7.1.	Definición.....	18
1.3.7.2.	Características.....	19
1.3.8.	Test Driven Development - TDD	20
1.3.8.1.	Definición.....	21
1.3.8.2.	Pilares fundamentales.....	21
1.3.8.3.	Características de TDD	21
1.3.8.4.	Proceso de diseño de software.....	22
1.3.8.5.	Tipos de Test	23
1.3.8.6.	Niveles.....	23
1.3.8.6.1.	Nivel de micro-iteraciones.....	23
1.3.8.6.2.	Nivel Iteración o Funcional.....	25
1.3.8.7.	Herramientas	25
1.3.8.7.1.	xUnit: Familia de marcos para pruebas unitarias	26
1.3.8.7.2.	FIT: Marco para pruebas de aceptación.....	27
1.3.8.8.	Fortalezas y Debilidades	27
1.3.8.9.	Fortalezas	28
1.3.8.10.	Debilidades	29
1.3.8.11.	Ventajas	29

1.3.8.12.	<i>Desventajas</i>	30
1.3.9.	<i>Metodología SCRUM</i>	30
1.3.9.1.	<i>Definición</i>	30
1.3.9.2.	<i>Roles de la metodología Scrum</i>	31
1.3.9.3.	<i>Procesos</i>	31
1.3.9.4.	<i>Beneficios</i>	32
1.3.10.	<i>Norma de evaluación ISO/IEC 25000</i>	33
1.3.10.1.	<i>Definición</i>	33
1.3.10.2.	<i>Actualización ISO/IEC 25000:2014</i>	33
1.3.10.3.	<i>Divisiones familia de normas ISO/IEC 25000</i>	34
1.3.10.4.	<i>Modelo de Calidad - ISO/IEC 25010</i>	38
1.3.10.5.	<i>ISO/IEC 25010 – Seguridad</i>	40
1.3.10.5.1.	<i>Definición</i>	40
1.3.10.5.2.	<i>Subcaracterísticas</i>	41

CAPÍTULO II

2.	MARCO METODOLÓGICO	42
2.1.	Alcance del producto software	42
2.2.	Limitaciones	43
2.3.	Descripción general del producto	43
2.3.1.	<i>Aplicación web</i>	44
2.3.2.	<i>Aplicación móvil</i>	45
2.3.3.	<i>Integración móvil</i>	45
2.4.	Fase de Planificación	45
2.4.1.	<i>Preparación del Proyecto</i>	45
2.4.2.	<i>Reuniones</i>	46
2.4.3.	<i>Procesos a Automatizar</i>	47
2.4.4.	<i>Personas y Roles del Proyecto</i>	47

2.4.5.	<i>Pila del Producto (Product Backlog)</i>	47
2.4.6.	<i>Factibilidad Económica</i>	50
2.4.7.	<i>Factibilidad Técnica</i>	50
2.4.8.	<i>Análisis y Gestión de Riesgos</i>	51
2.4.8.1.	Riesgos del Software.....	51
2.5.	Fase de Diseño	53
2.5.1.	<i>Diagrama de Casos de Uso</i>	53
2.5.1.1.	Procesos de Administrador	54
2.5.1.2.	Procesos de Secretario	54
2.5.1.3.	Procesos de Tesorero	55
2.5.1.4.	Procesos de Plomero.....	56
2.5.2.	<i>Diagrama de Clases</i>	57
2.5.3.	<i>Arquitectura del Sistema</i>	57
2.5.3.1.	Arquitectura de la aplicación móvil	59
2.5.4.	<i>Recursos Necesarios</i>	59
2.5.5.	<i>Estándar de Codificación</i>	60
2.5.6.	<i>Diseño de la Interfaz de Usuario</i>	61
2.5.7.	<i>Diseño de la Base de Datos</i>	62
2.6.	Fase de Desarrollo	62
2.6.1.	<i>Pila de Sprint (Sprint Backlog)</i>	62
2.6.2.	<i>Definición de Historias de usuarios</i>	65
2.6.3.	<i>Tablero de Tareas</i>	66
2.6.4.	<i>Ejecución TDD</i>	68
2.6.4.1.	Plan de Pruebas	68
2.6.4.2.	Pruebas Unitarias	71
2.6.4.3.	Pruebas de Aceptación.....	74
2.6.5.	Implantación	75
2.6.5.1.	Descripción de Recursos Hardware	75

CAPÍTULO III

3.	MARCO DE RESULTADOS	76
3.1.	Gestión del Proyecto	76
3.2.	Evaluación del tiempo de respuesta	77
3.2.6.	<i>Mejora de procesos.....</i>	77
3.2.6.1.	<i>Tamaño de muestra.....</i>	77
3.2.6.2.	<i>Recolección de datos.....</i>	78
3.2.6.3.	<i>Shapiro – Wilk Contraste de normalidad</i>	79
3.2.6.4.	<i>Análisis de datos – T-student</i>	79
3.3.	Evaluación de características de Calidad - Seguridad.....	82
3.3.6.	<i>Modelo de Evaluación de característica de Calidad - Seguridad</i>	83
3.3.6.1.	<i>Lineamientos</i>	83
3.3.6.2.	<i>Descripción de criterios de evaluación (CE).....</i>	85
3.3.6.3.	<i>Métricas para cada subcaracterística</i>	86
3.3.7.	<i>Evaluación - Seguridad</i>	88
3.3.7.1.	<i>Establecer los requisitos de la evaluación.....</i>	88
3.3.7.2.	<i>Especificar la evaluación.....</i>	88
3.3.7.3.	<i>Diseñar la evaluación.....</i>	89
3.3.7.4.	<i>Ejecutar la evaluación</i>	89
3.3.7.5.	<i>Finalización de la evaluación.....</i>	93
3.3.8.	<i>Análisis de la característica – Seguridad</i>	93
	CONCLUSIONES.....	95
	RECOMENDACIONES.....	97
	BIBLIOGRAFÍA	
	ANEXOS	

ÍNDICE DE TABLAS

Tabla 1-1.	Requerimientos para instalar Android.....	17
Tabla 2-1.	Características TDD	21
Tabla 1-2.	Actividades Establecidas en las Reuniones.....	46
Tabla 2-2.	Personas y Roles del Proyecto	47
Tabla 3-2.	Tallas Definidas en T-Shirt	48
Tabla 4-2.	Pila del Producto	48
Tabla 5-2.	Detalle de Impacto en el Riesgo.....	51
Tabla 6-2.	Riegos del Proyecto.....	52
Tabla 7-2.	Recursos Necesarios Para la Implementación.....	60
Tabla 8-2.	Estándar de Codificación CamelCase	61
Tabla 9-2.	Product Backlog.....	63
Tabla 10-2.	Historia de Usuario 07	66
Tabla 11-2.	Tablero de Tareas HU-03.....	67
Tabla 12-2.	Tablero de Tareas HU-04.....	67
Tabla 13-2.	Tablero de Tareas HU-05.....	67
Tabla 14-2.	Caso de prueba PA_05	69
Tabla 15-2:	Prueba de Aceptación.....	74
Tabla 1-3:	Medición de tiempo en el registro de lecturas de medidor.....	78
Tabla 2-3:	Diferencia tiempo SOLRIF - manual.....	80
Tabla 3-3:	Subcaracterísticas de Seguridad	83
Tabla 4-3:	Lineamientos para la característica de Seguridad	83
Tabla 5-3:	Descripción de criterios de evaluación (CE).....	85
Tabla 6-3:	Métricas para cada subcaracterística	87
Tabla 7-3:	Fórmula para cada subcaracterística	88
Tabla 8-3:	Fórmula para característica seguridad.....	88

Tabla 9-3:	Escala de Aceptación	89
Tabla 10-3:	Puntajes obtenidos.....	89
Tabla 11-3:	Resultados subcaracterísticas de seguridad.....	92
Tabla 12-3:	Resultados característica seguridad.....	92
Tabla 13-3:	Resultados de aceptación subcaracterísticas de seguridad	93
Tabla 14-3:	Resultados de aceptación característica seguridad.....	93

ÍNDICE DE FIGURAS

Figura 1-1: Representación, proceso de registro de consumo/pago de agua.	7
Figura 2-1: Interacción de la Arquitectura Modelo Vista Controlador.....	13
Figura 3-1: Componentes de PostgreSQL	18
Figura 4-1: Proceso de Diseño de Software.....	22
Figura 5-1: Proceso en un ciclo TDD	24
Figura 6-1: Test-Driven Development.....	28
Figura 7-1: Método científico	28
Figura 8-1: Proceso de SCRUM	30
Figura 9-1: Modelo de Calidad ISO/IEC 25000	33
Figura 10-1: Modelo calidad del producto software ISO/IEC 25010	34
Figura 11-1: Divisiones familia de normas ISO/IEC 25000.....	34
Figura 12-1: Modelo calidad del producto software ISO/IEC 25010	38
Figura 1-2: Módulos/Funcionalidades SOLRIF	43
Figura 1-2: Diagrama de Caso de Uso Procesos de Administrador.....	54
Figura 2-2: Diagrama de Caso de Uso Procesos de Secretario.....	55
Figura 3-2: Diagrama de Caso de Uso Procesos de Tesorero.....	56
Figura 4-2: Diagrama de Caso de Uso Procesos de Plomero	56
Figura 5-2: Diagrama de Clases.....	57
Figura 6-2: Arquitectura de la aplicación web/móvil	58
Figura 7-2: Arquitectura del Sistema - móvil	59
Figura 8-2: Plan de pruebas del sistema SOLRIF.....	68
Figura 9-2: Plan para pruebas de funcionalidad.	68
Figura 10-2: Ingreso de requerimiento para plan de prueba.	69
Figura 11-2: Caso de prueba PA_05.....	70
Figura 12-2: Ejecución caso de prueba.....	70
Figura 13-2: Código Test.....	71

Figura 14-2: Ejecución Test Fallida.....	72
Figura 15-2: Código a probar.....	72
Figura 16-2: Ejecución Test Exitosa.....	72
Figura 17-2: Refactorización	73
Figura 18-2: Ejecución Test Exitoso	73
Figura 19-2: Método @After	73
Figura 20-2: Método @Before	74

ÍNDICE DE GRÁFICOS

Gráfico 1-3: BurnDown Chart	76
Gráfico 2-3: Tiempo sistema vs tiempo manual	82
Gráfico 3-3: Subcaracterísticas de seguridad del sistema evaluado.....	94

ÍNDICE DE ANEXOS

- Anexo A.** Diagrama de actividades
- Anexo B.** Estimación de costos y esfuerzos
- Anexo C.** Factibilidad Técnica
- Anexo D.** Módulo de Autenticación
- Anexo E.** Diseño de la base de datos
- Anexo F.** Historias de Usuario
- Anexo G.** Plan de Pruebas
- Anexo H.** Pruebas Unitarias
- Anexo I.** Pruebas de Aceptación
- Anexo J.** Shapiro-Wilks - Contraste de normalidad
- Anexo K.** Métricas de Seguridad
- Anexo L.** Lineamientos aplicados
- Anexo M.** Obtención de puntaje
- Anexo N.** Resultados Evaluación

RESUMEN

El presente proyecto de titulación tuvo como objetivo implementar un sistema web y móvil, para mejorar el registro de consumo y cobros por el servicio de agua potable en la Junta Administradora “Angahuana Alto”, ubicada en la parroquia de Santa Rosa, cantón Ambato, provincia de Tungurahua; la institución cuenta aproximadamente con 10 funcionarios, quienes realizan los diferentes procesos operativos y de gestión administrativa. La tecnología Java conjuntamente con PostgreSQL forma una eficiente combinación para el desarrollo de aplicaciones web y móviles, haciendo que el uso de librerías propias del lenguaje de programación facilite el consumo masivo de datos y que la concurrencia sea adecuada, permitiendo así la correcta manipulación de la información. Para desarrollar el proyecto se utilizó SCRUM, metodología que permitió ejecutar las tareas de manera eficaz con participación activa del cliente, que por medio de sus especificaciones acerca de las funcionalidades requeridas se consiguió obtener un producto acorde a las necesidades del mismo; se aplicó Test-Driven Development (TDD), implementando pruebas unitarias en las tareas de programación y pruebas de aceptación para validar funcionalidades, consiguiendo así un código de calidad, claro y simple, exento o con mínimo de errores; asegurando que el software satisfaga los requerimientos cumpliendo con lo necesario. Además se analizó el proceso de registro de lecturas de medidor mediante la aplicación móvil determinando que existe una reducción del 49.25% en relación al tiempo que se emplean en realizar el mismo proceso manualmente, aumentando así la productividad en la institución. Se evaluó también la seguridad del sistema con el estándar ISO/IEC 25000, y se determinó que el sistema implementado está dentro del rango de aceptación, obteniéndose un 53.33%, en cuanto a la calidad de producto; sin embargo, al caer en la escala mínimamente aceptable se determina que existen falencias al respecto.

PALABRAS CLAVE: <INGENIERÍA DE SOFTWARE>, <DESARROLLO DE APLICACIONES WEB>, <DESARROLLO DE APLICACIONES MÓVILES>, <DESARROLLO DIRIGIDO POR TESTS >, <ANDROID (SISTEMA OPERATIVO)>, <POSTGRESQL (GESTOR DE BASE DE DATOS)>, <JAVA SERVER FACES (FRAMEWORK)>, < SCRUM (METODOLOGÍA DE DESARROLLO ÁGIL)>.

ABSTRACT

The objective of this qualification project was to implement a web and mobile system, to improve the consumption and collection records by the potable water service in the management Board “Angahuana Alto”, located in the parish of Santa Rosa, Ambato Canton, Tungurahua Province; the institution has approximately 10 staff members, who carry out the different operational and administrative management processes. The Java technology in conjunction with PostgreSQL forms an efficient combination for the development of web and mobile applications, making the use of libraries of the programming language facilitate the massive consumption of data and that the concurrence is adequate, thus allowing the correct manipulation of the information. To develop the project was used SCRUM, methodology that allowed to execute the tasks in an affective way with active participation of the client, that by means of its specifications on the required functionalities was managed to obtain a product according to the needs of the same; Test-driven Development (TDD) was applied, implementing unit test in programming tasks and acceptance test to validate functionalities, thus achieving a quality code, clear and simple, exempt or with minimum errors; Ensuring that the software meets the requirements by complying with the necessary. In addition, the process of recording meter readings through the mobile application was analyzed, determining that there is a reduction of 49.25% in relation to the time that they are used to perform the same process manually, thus increasing the productivity in the institution. The security of the system was also evaluated with the standard ISO/IEC 25000, and it was determined that the system implementing is within the acceptance range, obtaining a 53.33%, in terms of product quality; However, by falling into the minimally acceptable scale, it is determined that there are flaws.

KEYWORDS: <SOFTWARE ENGINEER>, <WEB APPLICATION DEVELOPMENT>, <MOBILE APPLICATION DEVELOPMENT>, <TEST- DRIVE DEVELOPMENT>, <ANDROID (OPERATING SYSTEM)>, <POSTGRESQL (DATABASE MANAGER)>, <JAVA SERVER FACES (FRAMEWORK)>, <SCRUM (METHODOLOGY OF AGILE DEVELOPMENT) >.

INTRODUCCIÓN

En la sociedad actual existe un amplio vínculo entre calidad de servicio y el usuario, estando presente el impacto que esto genera, aprovechando al máximo el desempeño profesional y administrativo.

El servicio público se considera un sistema generador de recursos monetarios para una población en particular, en primer término, dicho servicio acumula registros de la totalidad de usuarios que forman parte la entidad encargada; en segundo término, nos ayuda a generar y obtener información cualitativa y cuantitativa del consumo total o parcial del agua por parte de los usuarios en las respectivas distribuciones zonales de la comunidad, en tercer término, tenemos el poder contar con un respaldo informático de los datos generados a diario por los cobros realizados por el personal encargado.

En la actualidad el proceso de registro de la cantidad de agua consumida por un usuario, en un determinado período de tiempo y el valor de la tarifa a pagar se lo realiza de la siguiente manera; el plomero se encarga de registrar en una hoja de registros las lecturas del medidor recorriendo las rutas establecidas por la Junta de Agua, posteriormente respalda la información recolectada en un documento Excel, que se entrega después al tesorero de la institución quien cumple con el rol de cajero; mediante el documento Excel, el tesorero realiza los cálculos necesarios para determinar los valores a cobrar y procede con las respectivas recaudaciones, donde se registra el pago correspondiente de cada usuario. Debido al excesivo tiempo que implica este proceso, se genera un costo económico alto y un servicio ineficiente por parte del personal administrativo.

Es así que la La Junta Administradora de Agua Potable Angahuana Alto, ubicada en la parroquia de Santa Rosa, cantón Ambato, provincia de Tungurahua; ha emprendido un estudio y planificación para poder implementar determinadas mejoras en cuanto a la calidad de servicio prestada, con el propósito de renovar procesos que conlleven al cambio de gestión y distribución de las tareas que se efectúan, conociendo el impacto que estas puedan llegar a tener en el servicio brindado.

El presente trabajo tiene como propósito, mejorar los procesos de registro de consumo de agua y cobro de rubros generados por la misma, con el finalidad de asegurar y gestionar dichos datos en un sistema web, y permitiendo el acceso al registro de lecturas de medidor desde diferentes puntos geográficos mediante una aplicación móvil.

Antecedentes

La Junta Administradora de Agua Potable Angahuana Alto nació el 29 de enero de 2002 como una organización sin fines de lucro en la comunidad de Angahuana Alto, parroquia de Santa Rosa, cantón Ambato, provincia de Tungurahua; con la finalidad de prestar servicios públicos de agua potable en la comunidad rural. Esta organización está conformada por personas que tienen una línea de agua para su consumo, a los que se considera usuarios o clientes, está encabezado por directivos, quienes se encargan de toda la gestión administrativa y tratan de brindar un servicio de calidad a todos los usuarios. Las instalaciones de esta institución se encuentran ubicado en el centro de la comunidad de Angahuana Alto, en el sector 4, a una cuadra del Estadio Central.

Para un control adecuado se debe llevar un registro de la cantidad de agua consumida por el usuario en un determinado mes, por el cual debe pagar un valor monetario por ese consumo y otros servicios que reciba. Según la LEY ORGÁNICA DE RECURSOS HÍDRICOS, USOS Y APROVECHAMIENTO DEL AGUA en el artículo 135 de la constitución vigente, *“un usuario debe pagar por la prestación de servicios y autorización para usos y aprovechamiento del agua”*. (Tribunal Constitucional de la República de Ecuador, 2014, p. 27)

Actualmente el proceso que se pretende mejorar, se lo realiza de la siguiente manera: el plomero recorre una determinada ruta que definen los dirigentes de la institución, revisa el medidor y en una hoja de registros va registrando la cantidad de metros cúbicos de agua que ha consumido el usuario hasta la fecha. Luego, regresa a las instalaciones de la institución y procede a transcribir la información recolectada en un documento Excel para posteriormente entregar al tesorero.

En base a los datos recopilados en el documento Excel el tesorero realiza los cálculos necesarios y determina los valores que se debe cobrar a cada usuario, generando así información de vital importancia para efectuar las recaudaciones correspondientes; esto a su vez, interpretada de mejor manera como cobro/pago por consumo, se lo va registrando en un determinado documento Excel dispuesto por la entidad, documento que a posterior se entrega al personal de contabilidad.

Este proceso genera un costo, debido a que todo el proceso es manual; el plomero en lugar de estar transcribiendo nuevamente la misma información a un documento Excel, podría estar haciendo otras actividades que aporten a la productividad de la organización, y el tesorero en lugar de estar realizando cálculos que se pueden automatizar solo se dedicaría a recaudar, generar el respectivo reporte y entregar la información correspondiente a sus superiores.

Además, hay ocasiones en que los entes reguladores que supervisan a este tipo de instituciones llegan a realizar auditorías y necesitan que la información que solicitan como; los registros de la cantidad de agua consumida, las recaudaciones realizadas, etc., estén disponible de manera inmediata; y muchas de las veces, en ese momento el personal encargado tiene que estar buscando, recopilando y procesando datos para así generar un reporte con la información requerida, esto provoca la indisponibilidad inmediata de la información generando una pérdida de tiempo.

Presentación del Problema

Actualmente la Junta Administradora de Agua Potable Angahuana Alto, lleva el registro de la cantidad de agua que consume un usuario en un determinado mes y sus respectivos pagos de forma manual y poco organizada; empleando tiempo excesivo en la ejecución de dichos procesos, ocasionando demora para la generación de informes e inconvenientes al momento de auditorías por parte de los entes que regulan a esta institución.

Formulación del problema

La Junta Administradora de Agua Potable Angahuana Alto, no cuenta con un proceso automatizado adecuado que permita realizar de manera ágil el registro de las lecturas de medidor realizadas por el personal correspondiente, así como el control de las recaudaciones realizadas por los rubros generados por el consumo del agua; esto imposibilita al personal operativo y administrativo obtener, generar y brindar información de manera precisa e inmediata, provocando un servicio inoportuno cuando los usuarios se acercan a efectuar los pagos correspondientes.

Hasta el momento el manejo de la información se lo ha venido llevando de forma manual y poco organizada generando el desconcierto de los usuarios y las personas que están a cargo de la prestación del servicio, el no poder procesar y totalizar los datos de manera inmediata y mucho más al requerir de datos históricos en momentos determinados revela que son propensos en un futuro a tener problemas como redundancia de datos o simplemente pérdida de información relevante para el proceso de recaudaciones, que a posterior provocaría descuadres en el ámbito financiero con respecto a los recurso monetarios que se administra.

Se generan problemas como:

- Redundancia de datos.
- Información desactualizada.
- Pérdida y desorganización de registros.

- Falta de control de información de las nuevas adquisiciones.
- Falta de integridad de datos.
- Falta de respaldo de información en una base de datos o similar.

A consecuencia de ello se presenta lo siguiente:

- Descuadres financieros
- Falta de reportes individuales y/o generales.
- No disponibilidad inmediata de la información.
- Retraso en las recaudaciones por la adquisición y consumo del servicio.
- Datos erróneos en el registro de lecturas e inventario de usuarios.
- Desconocimiento de la totalidad de registros históricos y actuales generados mediante los procesos efectuados.

Sistematización del Problema

¿Cuál es el proceso para registrar las lecturas de medidor en la Junta Administradora de Agua Potable Angahuana Alto?

¿Cómo se efectúa los pagos por el consumo del agua en la Junta Administradora de Agua Potable Angahuana Alto?

¿Cómo se puede optimizar los procesos de lectura de medidor y recaudaciones en la Junta Administradora de Agua Angahuana Alto?

¿Con que herramientas tecnológicas se va a desarrollar la aplicación web/móvil?

¿Cuáles son los criterios y aspectos necesarios que se deben considerar para desarrollar la aplicación web/móvil?

¿Cuál es la metodología a utilizar para el desarrollo de la aplicación web/móvil?

¿Qué técnica se va aplicar para la implementación de aplicación web/móvil?

¿Cuál es el aporte que brinda el sistema desarrollado, a la Junta Administradora de Agua Potable Angahuana Alto?

Objetivos

Objetivo General

Desarrollar una aplicación web/móvil para el registro de consumo/pago de los usuarios de la Junta Administradora de Agua Potable Angahuana Alto, aplicando TDD.

Objetivos Específicos

- Recolectar información necesaria para la definición de requerimientos.
- Estudiar las herramientas a utilizar en el desarrollo de la aplicación web/móvil.
- Ejecutar la metodología de desarrollo de software Scrum para el desarrollo de la aplicación web/móvil.
- Aplicar la técnica de diseño TDD (Diseño guiado por pruebas) para el eficiente desarrollo de la aplicación web/móvil.
- Evaluar el tiempo de respuesta y la funcionalidad con respecto al criterio de seguridad del sistema informático desarrollado, utilizando la NORMA DE EVALUACIÓN ISO/IEC 25000, para conocer el aporte que brinda a la Junta Administradora de Agua Potable Angahuana Alto.

Justificación del Trabajo de Titulación

Justificación Teórica

El Plan Nacional de Desarrollo 2017-2021 “Toda una Vida” basado en el marco de lo que ordena la Constitución de la República del Ecuador plantea, generar incentivos que permitan a los distintos niveles de gobierno ampliar la dotación de instalaciones y equipamientos suficientes y eficientes, para la prestación oportuna de servicios de agua y saneamiento, con criterios de sustentabilidad y salubridad. (Secretaría Nacional de Planificación y Desarrollo - Senplades, 2017, p. 56)

La Junta Administradora de Agua Potable Angahuana Alto, como una organización comunitaria sin fines de lucro que tiene la finalidad de prestar servicios públicos de agua potable en las comunidades rurales realiza una gestión administrativa donde gestiona el uso adecuado del agua y las respectivas tarifas por el consumo de la misma, por tanto en esta institución se debe llevar un registro adecuado de la cantidad de litros de agua que consume un cliente o usuario, y de igual manera se debe registrar los respectivos pagos que realice.

Según la LEY ORGÁNICA DE RECURSOS HÍDRICOS, USOS Y APROVECHAMIENTO DEL AGUA, en el **Artículo 44** se expone deberes y atribuciones de las diferentes Juntas Administradoras, a continuación se describe las relacionadas al uso del agua potable:

- *Establecer, recaudar y administrar las tarifas por la prestación de los servicios, dentro de los criterios generales regulados en esta Ley y el Reglamento expedido por la Autoridad Única del Agua.* (Tribunal Constitucional de la República de Ecuador, 2014, p. 13)
- *Rehabilitar, operar y mantener la infraestructura para la prestación de los servicios de agua potable.* (Tribunal Constitucional de la República de Ecuador, 2014, p. 13)

El desarrollo de un sistema informático que ayude a registrar la cantidad de agua que consume un usuario en determinado mes y su respectivo pago facilitará y optimizará el proceso de incrementar el control por parte de la administración del lugar, debido a que el sistema automatizado ofrecerá información relevante, actualizada de forma inmediata; la misma que será necesaria para una óptima gestión administrativa.

Por esta razón se plantea el desarrollo de una Aplicación web/móvil, y para una mejor implementación se ha decidido aplicar la técnica de diseño y desarrollo de software TDD (Test Driven Development), la cual se centra en tres pilares fundamentales: (Ble, 2013)

- El desarrollo de las funciones necesarias que el cliente necesita.
- La disminución de defectos del software en la fase de producción o codificación.
- La codificación de software reutilizable y listo para realizar variantes.

Estos pilares permitirán un desarrollo eficaz y eficiente del software, expresándolo de otra manera, la productividad incrementará y de la misma manera la calidad del software aumentará. Al trabajar con TDD las jornadas de implementación se harán mucho más amenas, siendo esto un factor muy influyente para el vertiginoso desarrollo del sistema informático. (Ble, 2013)

Justificación Aplicativa

En la tarea de registrar el consumo de agua de un determinado mes por un usuario y el pago de un valor monetario por la tarifa de la misma, se requiere de agilidad y disponibilidad inmediata en el proceso, por lo que es necesario un sistema informático que ayude a gestionar esta labor de la manera más fácil y eficiente. La aplicación web y móvil propuesta ayudará a los dirigentes de

la Junta Administradora de Agua Potable Angahuana Alto a llevar un control adecuado, con mayor eficiencia y de forma oportuna, como se puede apreciar en la **Figura 1-1**.

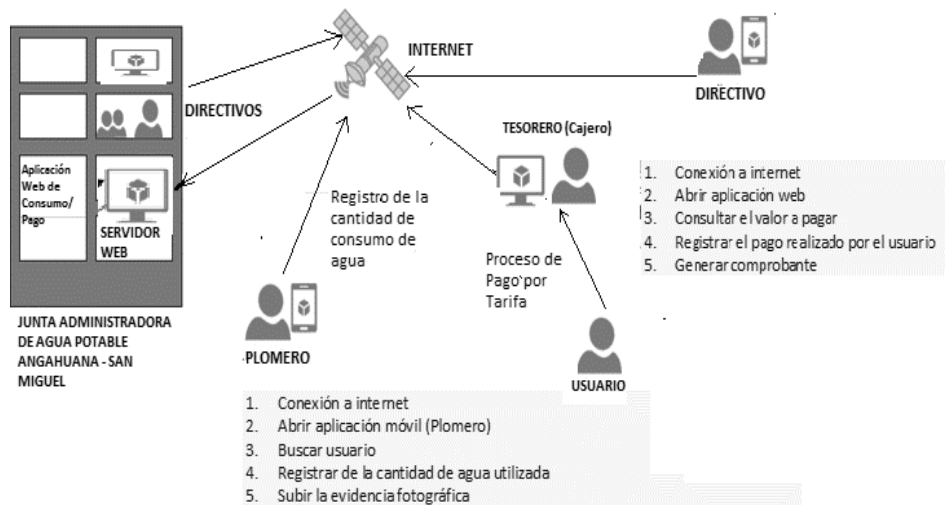


Figura 1-1: Representación, proceso de registro de consumo/pago de agua.

Realizado por: Tupac Tisalema, 2019

La aplicación móvil instalada en el teléfono inteligente que usará el plomero (persona encargada de ir de domicilio en domicilio), ayudará a realizar un registro de forma directa y fácil, quien solo tendrá que conectarse al internet a través del dispositivo móvil, abrir la aplicación, autenticarse en el sistema, buscar al respectivo usuario o cliente de la organización y proceder a introducir la información necesaria como lo es cantidad de agua consumida adjuntando a la misma una evidencia fotográfica del consumo; y si fuera el caso, también se registraría algo extra como por ejemplo un servicio de mantenimiento de la llave de agua potable; todo esto asignándolo al usuario registrado.

La aplicación constará de cinco módulos principales, detallados a continuación:

- **Módulo de parametrización:** Este módulo permitirá la gestión de usuarios, roles, tarifas y servicios; brindando la posibilidad de establecer parámetros necesarios para la ejecución de los procesos internos del sistema.
- **Módulo de contribuyente:** En este módulo se permitirá registrar/actualizar datos de personas, adquirir servicios y registrar las cuentas por cobrar de cada cliente.
- **Módulo de medición:** El presente módulo permitirá registrar las lecturas del medidor, realizadas por el plomero mediante la aplicación móvil y web.

- **Módulo facturación:** Este módulo permitirá efectuar los pagos, generar facturas comprobantes por adquisición de servicios y consumo de los mismos.
- **Módulo reportes:** Este módulo permitirá generar todos los reportes requeridos por el personal involucrado.

Al ser una aplicación para la web y móvil los dirigentes podrán desde cualquier dispositivo con capacidad de procesamiento acceder y visualizar la información requerida. Para esta aplicación se implantará un servidor web en la Junta Administradora de Agua Potable Angahuana Alto.

Alcance del Trabajo

El desarrollo de un sistema web y móvil para la Junta Administradora de Agua Potable Angahuana Alto, ayudará a llevar un control adecuado del consumo de agua mediante el registro de lecturas de medidor correspondientes a cada usuario, proceso que será ejecutado por el plomero, quién tendrá la posibilidad de acceder al sistema desde una aplicación web o móvil.

El sistema permitirá generar facturas y/o comprobantes para efectuar los respectivos cobros por adquisición o consumo de los servicios brindados, tarea que será ejecutado por el tesorero mediante la aplicación web. También contemplará el poder gestionar la privacidad de los datos, y su administración a corto y largo plazo, manejo de credenciales debidamente autorizadas por el personal administrativo y elaboración reportes primordiales para la toma de decisiones.

Para contar con la total funcionalidad, el sistema web deberá estar desplegado el servidor institucional y debidamente hechas las pruebas necesarias para poder realizar la respectiva socialización del sistema con el personal administrativo el cual será el encargado del manejo y gestión del mismo. La aplicación móvil será de uso exclusivo del plomero y específicamente para el registro de lecturas de medidores.

Para el funcionamiento del mismo será indispensable la conexión a internet del dispositivo móvil donde se encuentre instalado la aplicación, lo cual limita el registro de datos desde ubicaciones geográficas remotas donde no existen conectividad a internet.

CAPÍTULO I

1. MARCO TEÓRICO REFERENCIAL

1.1. Fundamentación Teórica

El desarrollo de este sistema viene dado por la utilización de herramientas de libre distribución siendo Java nuestra principal plataforma de construcción y codificación, apoyado por un gestor de base de datos siendo PostgreSQL, mismos que conjuntamente forman un gran equipo de desarrollo. Con estas herramientas como base se implementa un sistema web robusto de gran alcance y fácil manejo para el usuario, en la cual, además se incluye una aplicación móvil desarrolla en Android con el módulo de lectura de medidor, dicha app consume un servicio web del sistema principal para realizar la búsqueda del cliente y registrar la cantidad de agua consumida por el mismo en un determinado periodo.

Utilizando las herramientas tecnológicas mencionada se planifica el desarrollo de un sistema web y móvil que permita el control y manejo de información generada en la institución, siendo las principales funcionalidades el registro de la lectura de medidor por el consumo de agua, generación de comprobantes o facturas, y el registro de cobros; ya que dicha información será relevante para realizar cualquier análisis necesario de forma periódica, mensual o anual según la necesidad de los directivos de la Junta.

1.2. Procesos de control de consumo de agua según la Ley Orgánica de Recursos Hídricos, Usos y Aprovechamiento del Agua (LORHUYA)

1.2.1. Juntas Administradoras de Agua Potable y Saneamiento

Son organizaciones comunitarias, que no tienen fines de lucro con el propósito de prestar el servicio público de agua potable. *Su accionar se fundamenta en criterios de equidad, solidaridad, interculturalidad, eficiencia económica, sostenibilidad de recurso hídrico, calidad en la prestación de los servicios y en el reparto del agua.* (Tribunal Constitucional de la República de Ecuador, 2014, p. 13)

Cuando las juntas presten el servicio de saneamiento se llamarán Juntas Administradoras de Agua Potable y Saneamiento; caso contrario ésta última palabra quedará exenta de su denominación. (Tribunal Constitucional de la República de Ecuador, 2014, p. 13)

1.2.2 Consumidor

Art. 67 de la LORHUyA: “*los consumidores son personas naturales, jurídicas, organizaciones comunitarias que demandan bienes o servicios relacionados con el agua proporcionados por los usuarios*”. (Tribunal Constitucional de la República de Ecuador, 2014, p. 16)

1.2.2. Tarifario

Esta Junta Administradora de Agua potable al igual que otras se somete a las normativas de la AGENCIA DE REGULACIÓN Y CONTROL DE AGUA (ARCA), entidad que tiene como objetivo principal:

“Ejercer la regulación y control de la gestión integral e integrada de los recursos hídricos, de la cantidad y calidad de agua en sus fuentes y zonas de recarga, calidad de los servicios públicos relacionados al sector agua y en todos sus usos y aprovechamientos”. (AGENCIA DE REGULACIÓN Y CONTROL DEL AGUA, 2017)

Rigiéndose a las normativas planteadas por dicha entidad, y basándose en la REGULACIÓN Nro. DIR-ARCA-RG-002-2016, CAPITULO II (CRITERIOS TECNICOS TARIFARIOS) (AGENCIA DE REGULACIÓN Y CONTROL DE AGUA, 2016, p. 9), la Junta Administradora de Agua Potable Angahuana Alto incluye como parte de su reglamento interno, la definición de las tarifas a cobrar por los adquisición y consumo de agua potable; determinado así un costo de 150 y 2 dólares Americanos por cada 10 metro cubico respectivamente.

1.3. Herramientas de Desarrollo

En este apartado se estudiará a más a profundidad sobre las herramientas tecnológicas a utilizar en el desarrollo de la aplicación web/móvil; en la misma se puntualiza las definiciones, funcionalidades, características, ventajas y desventajas de cada herramienta en particular. El presente estudio se realiza con el propósito de adquirir más conocimientos teóricos y prácticos para utilizarlas de manera adecuada, con esto se pretende evitar mayores dificultades al momento de emplearlas y así obtener el máximo rendimiento de cada tecnología; puesto que esto incide de manera directa y con gran impacto en el avance del proyecto.

1.3.1. Netbeans

Para trabajar de manera adecuada con esta herramienta es necesario profundizar conocimientos al respecto. Este apartado tiene como propósito estudiar los aspectos relevantes de Netbeans.

1.3.1.1. Definición

Entorno integrado de desarrollo IDE (Integrated Development Environment), que permite desarrollar aplicaciones Web, Móvil y de Escritorio para distintos lenguajes de programación como Java, C++, Ruby y PHP entre otros, de código abierto, multiplataforma y multilenguaje. (González, 2015, p. 1)

1.3.1.2. Características

A continuación se describe las características más relevantes de Netbeans:

Soporte Java Beans

- Modelos Bean en el Navegador
- Generador de Propiedades Bean
- Editor BeanInfo (Reyes, 2017)

Generador JSF CRUD

- Generador de aplicaciones JavaServer Faces CRUD a partir de clases de entidades.
- Soporta todo tipo de relaciones.
- Soporta todo tipo de claves principales (columna simple, compuesta y generada). (Reyes, 2017)

1.3.2. Patrón Modelo-Vista-Controlador (MVC)

Es un paradigma que permite dividir las partes que componen una aplicación es decir se puede dividir en el Modelo, Vistas y Controladores, permitiendo la implementación por separado de cada componente, facilitando de esta manera el mantenimiento y la actualización de un producto software de forma. Existen diversos frameworks basados en el patrón MVC lo cuales se pueden

utilizar para lograr una mejor organización del trabajo y mayor especialización de los desarrolladores y diseñadores. (Fernández and Díaz, 2012, p. 1)

1.3.2.1. Definición

Es un patrón de arquitectura de software que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones. (Fernández and Díaz, 2012, p. 47)

1.3.2.2. Elementos

Modelo: Representa datos del programa, maneja y controla sus transformaciones. No tiene conocimiento específico de los Controladores y de las Vistas, ni contiene referencias a ellos. El propio sistema tiene la responsabilidad de mantener vínculos entre el Modelo y las Vistas, además de informar a las Vistas cuando se altera el Modelo. (Fernández and Díaz, 2012, p. 49)

Responsabilidades:

- Acceder a la capa de almacenamiento de datos, sin embargo, es recomendable que el modelo sea independiente del sistema de almacenamiento.
- Define reglas de negocio (la funcionalidad del sistema).
- Notificar a las vistas los cambios que se produzcan en los datos por un agente externo si se está ante un modelo activo (ejemplo, un fichero bath que actualiza los datos, un temporizador que desencadena una inserción, entre otros.). (Fernández and Díaz, 2012, p. 50)

Vista: Maneja la presentación visual de los datos representados por la capa del Modelo, crea una representación visual del Modelo para mostrar los datos al usuario. Interactúa específicamente con el Controlador, aunque es posible que trate directamente con el Modelo a través de una referencia al propio Modelo. (Fernández and Díaz, 2012, p. 49)

Responsabilidades:

- Recibir datos procesados por el controlador o del modelo y mostrarlos al usuario.
- Está asociado directamente con un determinado controlador.

- Puede brindar un servicio de actualización, para que sea invocado por el controlador o por el modelo informando los cambios producidos en los datos por otros agentes. (Fernández and Díaz, 2012, p. 50)

Controlador: Proporciona significado a las órdenes del usuario (a los eventos de la vista), actuando sobre los datos representados por el Modelo, centra toda la interacción entre la Vista y el Modelo. (Fernández and Díaz, 2012, p. 49)

Responsabilidades:

- Recibir eventos que se generan en la Vista (un clic, un cambio en un campo de texto, etc.).
- Contiene reglas de gestión de eventos. Estas acciones pueden suponer peticiones al modelo o a las vistas. (Fernández and Díaz, 2012, p. 50)

Para mejor explicación se ilustra la **Figura 2-1**, donde se puede apreciar todo lo descrito anteriormente.

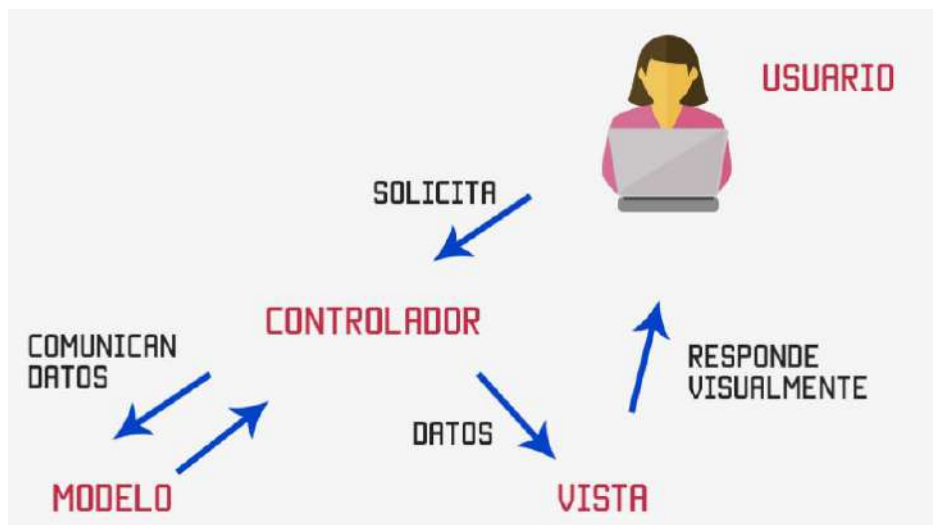


Figura 2-1: Interacción de la Arquitectura Modelo Vista Controlador

Fuente: (Hernandez, 2015)

1.3.3. *Java Server Faces (JSF)*

JSF es una especificación de desarrollo para aplicaciones web que está basada en el patrón Modelo Vista Controlador (MVC). Está encaminado a mejorar la interfaz gráfica del usuario, manteniendo el concepto fundamental del patrón MVC, es decir separando el comportamiento y la presentación. (Rios, 2015)

1.3.3.1. Definición

“Es un framework MVC (Modelo-Vista-Controlador) basado en el API de Servlets que proporciona un conjunto de componentes en forma de etiquetas definidas en páginas XHTML mediante el framework Facelets”. (Dept. Ciencia de la Computación e IA, 2014)

1.3.3.2. Características

- Permite la definición de las interfaces gráficas de usuario mediante vistas que agrupan componentes gráficos.
- Conexión de los componentes visuales con los datos de la aplicación a través de los Beans gestionados.
- Validación automática de la entrada del usuario.
- Navegación entre vistas.
- Conversión de datos introducidos por el usuario.
- A partir de la especificación 2.0 un modelo estándar de comunicación Ajax entre la vista y el servidor. (Dept. Ciencia de la Computación e IA, 2014)

1.3.4. Primefaces

A continuación se expone la definición y características referentes a la tecnología primefaces, la cual ayudará a desarrollar de manera ágil y oportuna la interfaz de la aplicación web.

1.3.4.1. Definición

Es una librería de componentes visuales para Java Server Faces 2.0, tiene como objetivo primordial de proveer un conjunto de componentes visuales para facilitar la creación y posterior diseño de aplicaciones web. (Pech *et al.*, 2011, p. 8)

Los componentes de PrimeFaces cuentan con soporte nativo de Ajax, además dispone de un módulo denominado TouchFaces que permite el desarrollo de aplicaciones web para dispositivos móviles con navegadores basados en WebKit. (Pech *et al.*, 2011, p. 8)

1.3.4.2. Características

- Soporte nativo de Ajax.
- Posee un conjunto de herramientas que facilitan la creación de aplicaciones web para móviles.
- Es compatible con otras librerías similares, por ejemplo, RichFaces
- Uso de JavaScript no intrusivo. (Pech *et al.*, 2011)

1.3.5. Android

Para desarrollar aplicaciones móviles es importante familiarizarse con algunas herramientas que permitan su implementación, y en este proyecto, para desarrollar la aplicación móvil se ha optado por utilizar la plataforma Android. Por lo tanto en el siguiente apartado se pretende conocer las especificaciones relevantes en relación a esta plataforma.

1.3.5.1. Definición

Es un sistema operativo para dispositivos móviles propiedad de Google, al igual que iOS de Apple, Symbian y Blackberry OS, está basado en Linux, un núcleo de sistema operativo libre, gratuito y multiplataforma. (GONZALEZ, 2011)

1.3.5.2. Características

- Framework de aplicaciones: Permite el reemplazo y la reutilización de los componentes.
- Posee un navegador web integrado que está basado en los motores open Source Webkit.
- Cuenta con SQLite, que es una base de datos que se integra directamente con las aplicaciones.
- Cuenta con soporte para multimedia como audio, video e imágenes planas (H.264, MP3, JPG, PNG, GIF, MPEG4, etc).
- Telefonía GSM: dependiente del terminal.
- Bluetooth, EDGE, 3g y Wifi.
- Cámara, GPS, brújula y acelerómetro: Dependiente del terminal
- La pantalla es táctil.

1.3.5.3. Ventajas

- El código de Android es abierto dado que cuenta con una licencia Apache. Lo que significa que cualquier persona puede realizar una aplicación para este sistema operativo.
- Es uno de los sistemas operativos más utilizados en la actualidad, dado que existen una gran variedad de aplicaciones gratuitas.
- Es un sistema operativo que soporta multitarea.
- Cuenta con una variedad de servicios incorporados tales como GPS, bluetooth, etc.
- Bajo consumo de batería ya que es capaz de gestionar varias aplicaciones y dejar en modo suspensión si no se utilizan e incluso cerrarlas si no se está utilizando.

1.3.5.4. Desventajas

- Existen ocasiones en la que las aplicaciones inactivas consumen demasiada energía sin que el sistema operativo lo cierre, provocando un consumo excesivo de batería.
- Para muchas personas que no están familiarizados con este sistema operativo resulta ser poco intuitivo. Por ejemplo, configurar el teléfono, puede llevar mucho tiempo, y esto es generado por la interfaz de Android, aunque con las últimas versiones esto está cambiando.

1.3.6. Android Estudio

Anteriormente se explica que para desarrollar la aplicación móvil se opta por la plataforma Android, por lo tanto este apartado tiene como propósito dar a conocer las especificaciones necesarias en relación al entorno de desarrollo Android Estudio. A continuación se expone la definición, características y requerimientos de instalación para trabajar con este IDE.

1.3.6.1. Definición

Es un entorno de desarrollo integrado (IDE) oficial para el desarrollo de aplicaciones Android, fundamentado en IntelliJ IDEA; además ofrece mejoras con respecto al plugin ADT (Android Developer Tools) para Eclipse. Está programado en Java, utiliza licencia de software libre Apache 2.0, y es multiplataforma. (academia android, 2014)

1.3.6.2. Características

- Tiene soporte para programar aplicaciones para dispositivos corporales como por ejemplo un reloj a través de Android.
- Posee Lint, el cual es una herramienta que permite detectar código no compatible entre arquitecturas diferentes, errores de sintaxis o código confuso.
- Utiliza ProGuard para mejorar y reducir el código del proyecto al exportar a APK lo cual aporta un beneficio cuando se requiere ejecutar una determinada aplicación en dispositivos de gama baja con limitaciones de memoria interna.
- Integración de la herramienta Gradle encargada de gestionar y automatizar la construcción de proyectos, como pueden ser las tareas de testing, compilación o empaquetado.
- Cuenta con Gradle que es un sistema de compilación muy potente.
- Posibilita el control de versiones accediendo a un repositorio desde el que poder descargar Mercurial, Git, Github o Subversion.
- Permite crear dispositivos virtuales en diferentes dispositivos y resoluciones para mostrar una vista previa, de lo que se está construyendo.
- Facilita la integración con Google Cloud Platform, con el fin de permitir el acceso a los diferentes servicios que provee Google en la nube.
- Editor de diseño que muestra una vista previa de los cambios realizados directamente en el archivo xml. (academia android, 2014)

1.3.6.3. Requerimientos del Sistema

Tabla 1-1. Requerimientos para instalar Android

Windows	Mac OS	Linux
Microsoft Windows 8/7/Vista/2003 (32 o 64 bit)	Mac OS X 10.8.5 o superior, hasta la 10.9 (Mavericks)	GNOME o entorno de escritorio KDE
Mínimo de 2 GB de RAM, recomendado 4 GB de RAM		
400 MB de espacio en disco		
Necesita de al menos 1 GB para Android SDK, emulador de imágenes del sistema, y cachés		
Resolución mínima de pantalla de 1280 x 800		
Java Development Kit (JDK) 7 o superior		

Realizado por: Tupac Tisalema. 2019

1.3.7. PostgreSQL

En la actualidad para desarrollar cualquier aplicación o sistema informático es primordial disponer de un motor de base de datos, ya que este permite el almacenamiento y gestión de toda la información manejada. Para la implementación de esta aplicación web/móvil se decide trabajar con PostgreSQL.

1.3.7.1. Definición

Sistema de gestión de bases de datos de código abierto más potente del mercado y en sus últimas versiones no tiene nada que envidiarle a otras bases de datos comerciales. (Guerrero, 2016)

Utiliza un modelo cliente/servidor y usa multiprocesos en lugar de multihilos con la finalidad de garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afecta el resto del sistema por lo que continuará funcionando (Martin, 2011).

A continuación, tenemos un gráfico **Figura 3-1** se ilustra de manera general los componentes más importantes en un sistema PostgreSQL.

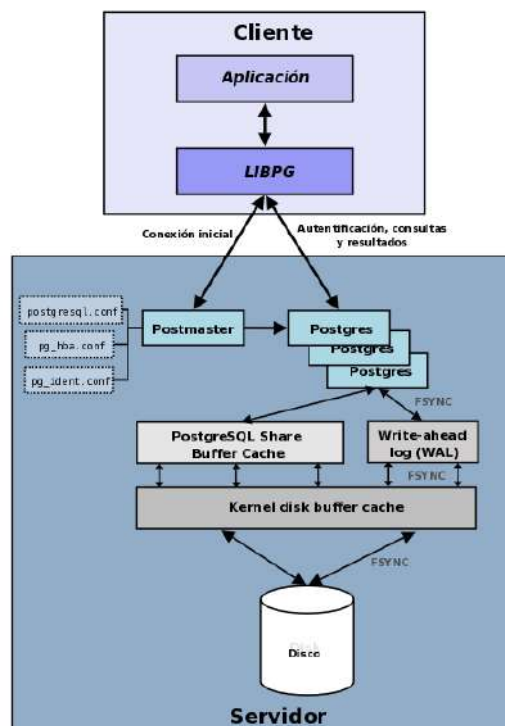


Figura 3-1: Componentes de PostgreSQL
Fuente: (González, 2014)

- **Aplicación cliente:** Aplicación que permite utilizar PostgreSQL desde el cliente y administrador de bases de datos remotamente, la conexión puede ocurrir vía TCP/IP o sockets locales.
- **Demonio postmaster:** Encargado de escuchar por un puerto/socket por conexiones entrantes de clientes, además de crear los procesos hijos o subprocesos y autenticar las peticiones entrantes, y por último gestiona las consultas y manda los resultados a las aplicaciones que lo solicitaron (clientes).
- **Ficheros de configuración:** Existen 3 ficheros de configuración principales estos son, pg_hba.conf , postgresql.conf y pg_ident.conf
- **Procesos hijos postgres:** Procesos hijos que se encargan de autenticar a los clientes, de gestionar las consultas y mandar los resultados a las aplicaciones clientes.
- **PostgreSQL share buffer cache:** Es una memoria compartida usada para almacenar datos en caché.
- **Write-Ahead Log (WAL):** Componente del sistema encargado de asegurar la integridad de los datos mediante la recuperación de tipo REDO.

1.3.7.2. Características

En el presente apartado se expone las características relevantes con respecto a este motor de base de datos, para una mejor interpretación estas características se clasifican en generales, programación/desarrollo y SQL, esto con el propósito de conocerlas de manera puntual.

A continuación se presenta las siguientes características consideradas generales (González, 2014).

Generales

- Integridad referencial
- Replicación asincrónica/sincrónica / Streaming replication - Hot Standby
- Copias de seguridad en caliente (backups)
- Múltiples métodos de autenticación
- Acceso encriptado vía SSL

- Actualización in-situ integrada (pg_upgrade)
- Completa documentación
- Licencia BSD
- Disponible para Linux y UNIX en todas sus variantes y Windows 32/64bit.

Seguidamente tenemos las características específicas en relación a la programación en este motor de base datos (González, 2014).

Programación / Desarrollo

- Funciones/procedimientos almacenados (stored procedures) en numerosos lenguajes de programación, entre otros PL/pgSQL (similar al PL/SQL de oracle), PL/Perl, PL/Python y PL/Tcl.
- Numerosos tipos de datos y posibilidad de definir nuevos tipos. Además de los tipos estándares en cualquier base de datos, tenemos disponibles, entre otros, tipos geométricos, de direcciones de red, de cadenas binarias, UUID, XML, matrices, etc.
- Soporta el almacenamiento de objetos binarios grandes (gráficos, videos, sonido, ...)

Para finalizar con este tema, también se da a conocer las características fundamentales con respecto al lenguaje SQL (González, 2014).

SQL

- Llaves primarias (primary keys) y foráneas (foreign keys)
- Check, Unique y Not null constraints
- Restricciones de unicidad postergables (deferrable constraints)
- Índices compuestos, únicos, parciales y funcionales en cualquiera de los métodos de almacenamiento disponibles, B-tree, R-tree, hash ó GiST
- Consultas recursivas
- Joins, Vistas (views), Disparadores (triggers) comunes, por columna, condicionales.

1.3.8. Test Driven Development - TDD

Al hablar de TDD se cree que es una buena práctica para hacer que el código tenga una excelente cobertura de pruebas, pero indiscutiblemente TDD es una herramienta de diseño que ayuda al programador a convertirse en un desarrollador de alto nivel, sin definirse como una metodología de pruebas se apoya vigorosamente en las pruebas unitarias y también en pruebas de aceptación;

pero esto no quiere decir que se tenga que generar pruebas de manera excesiva, sino de diseñar apropiadamente según los requerimientos definidos.

El presente temario tiene la finalidad de estudiar de manera adecuada todo lo referente a esta herramienta, para emplearla de manera correcta en la implementación de la aplicación web/móvil.

1.3.8.1. Definición

TDD, es una técnica de diseño e implementación de software. (Ble, 2013)

1.3.8.2. Pilares fundamentales

- Implementación de las funciones justas que el cliente necesita y no más.
- Minimizar del número de defectos que llegan al software en fase de producción.
- Producción de software modular, reutilizable y preparado para cambios. (Ble, 2013)

1.3.8.3. Características de TDD

En la **Tabla 2-1** se expone las características de TDD con su respectiva descripción.

Tabla 2-1. Características TDD

Característica	Descripción
Evitar escribir código innecesario	Consiste en escribir lo mínimo posible, si el código implementado por el programador pasa y falla, a partir de ahí se tiene una idea de cómo modificar los requerimientos y de esa manera también agregan nuevos.
Generar pruebas para cada funcionalidad	Permite que el desarrollador confíe en el código implementado, ya que con ello se facilitará futuros cambios, porque si se logra pasar todas las pruebas es seguro que se obtendrá un código que funcione correctamente.
TDD requiere que haga fallar los casos de prueba	Con esto se pretende asegurar el correcto funcionamiento de los casos de prueba para de esta manera se pueda detectar errores.
El desarrollo guiado por pruebas	Esta práctica provee un valor agregado en el desarrollo de software, facilitando así obtener aplicaciones de calidad cumpliendo con los tiempos planificados.
TDD tiene la capacidad de avanzar en pequeños pasos	Permite que el desarrollador se enfoque en la tarea actual, haciendo que primero pase el test para poder continuar.

Fuente: (Cuenca *et al.*, 2014)

Realizado por: Tupac Tisalema. 2019

1.3.8.4. Proceso de diseño de software

1. Cliente escribe su historia de usuario.
2. Escriben junto con el cliente los criterios de aceptación de esta historia, desglosándolos mucho para simplificarlos todo lo posible.
3. Escoger el criterio de aceptación más simple y se traduce en una prueba unitaria.
4. Comprobar que esta prueba falla.
5. Escribir el código que hace pasar la prueba.
6. Ejecutar todas las pruebas automatizadas.
7. Re factorizar y limpiar el código.
8. Pasar nuevamente las pruebas automatizadas para comprobar que todo sigue funcionando.
9. Regresamos al punto 3 con los criterios de aceptación que falten y repetimos el ciclo una y otra vez hasta completar nuestra aplicación. (Herranz, 2011)

A medida que se van añadiendo nuevas funcionalidades, se genera nuevas pruebas, las cuales (naturalmente) fallan al no estar implementadas, posterior se desarrolla el software necesario para aprobar el caso de prueba hasta que todos tengan éxito. Ver. **Figura 4-1**.

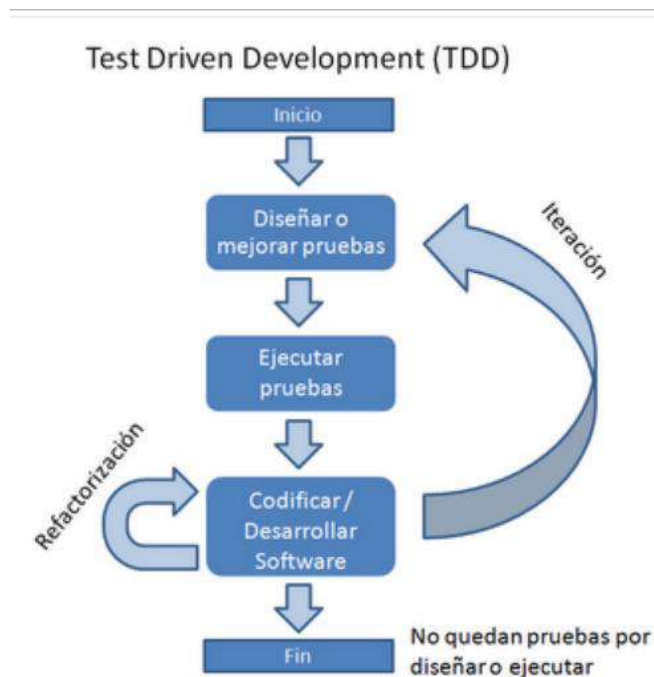


Figura 4-1: Proceso de Diseño de Software

Fuente: (PMOinformatica, 2012)

1.3.8.5. *Tipos de Test*

De aceptación: Se verifica que se cumpla con un requerimiento del negocio, escritas en lenguaje del cliente pero que puede ser ejecutado con la máquina. Permite probar que el software en desarrollo satisface las expectativas del cliente y de los usuarios. (Jurado *et al.*, 2010, p. 70)

Funcionales: Esta expresión se emplea para determinar pruebas que agrupan a varios tests de aceptación y prueban alguna funcionalidad del negocio. (Jurado *et al.*, 2010, p. 71)

De sistema: Integra varias partes del sistema, inclusive puede probar toda la aplicación o varias funcionalidades en conjunto. Estos tests se comportan de manera similar y buscan emular el comportamiento de los usuarios del sistema. (Jurado *et al.*, 2010, p. 71)

Unitarias: Tests ineludibles, necesarios y los más relevantes para los programadores. Para que un test se considere unitario debe cumplir con cuatro premisas importantes que son; ser rápido, atómico, inocuo e independiente. Tiene que ser rápido porque se ejecutan varios de ellos todo el tiempo, tiene que ser inocuo, ya que no debe alterar el estado del sistema, para ser atómico debe probar la menor cantidad posible de código y para ser independiente no debe depender de que otros tests unitarios sean exitosos o fallen para ser él mismo exitoso. (Jurado *et al.*, 2010, p. 74)

1.3.8.6. *Niveles*

TDD puede ser aplicado a dos niveles: (Araújo, 2007)

1.3.8.6.1. *Nivel de micro-iteraciones*

Nivel el desarrollo guiado por pruebas unitarias escritas por el propio programador, propone el cumplimiento de determinados pasos a ser ejecutados antes y durante la implementación del sistema, dada por el desarrollador, según se detalla a continuación: (Araújo, 2007, p. 3)

- Agregar una prueba.
- Ejecutar todas las pruebas y verificar que solo la nueva falla.
- Escribir código mínimo y suficiente para hacer que la prueba pase.
- Ejecutar todas las pruebas y verificar que pasan exitosamente.
- Refactorizar para eliminar duplicaciones.

Lo explicado anteriormente se lo interpreta en el diagrama de actividad de la **Figura 5-1**.

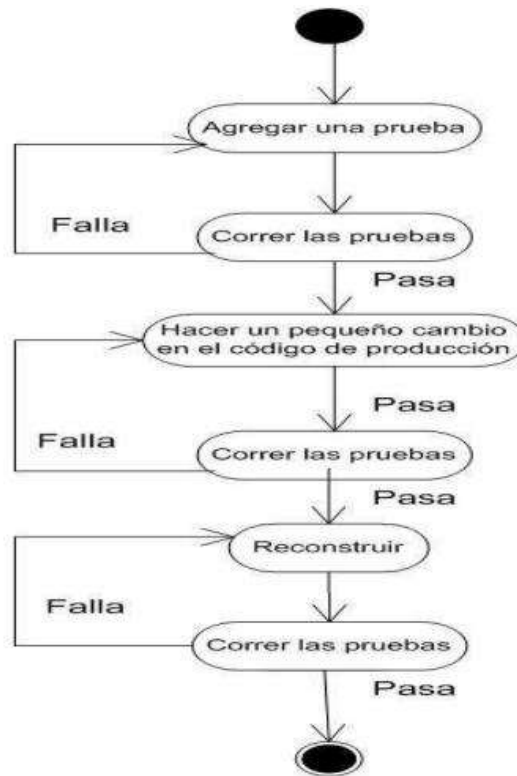


Figura 5-1: Proceso en un ciclo TDD

Fuente: (Araújo, 2007)

Además, al escribir las pruebas en primer lugar, el autor del código no se encuentra condicionado por el código a probar, por otra parte permite también especificar el comportamiento esperado sin restringirse a una implementación en particular. (Vaca *et al.*, 2015)

El algoritmo TDD

En base a proceso descrito anteriormente, el algoritmo TDD se reduce a tres pasos concretos:

- Escribir la especificación del requisito (el ejemplo, el test).
- Implementar el código según dicho ejemplo.
- Refactorizar para eliminar duplicidad y hacer mejoras. (Ble, 2013)

Escribir las especificaciones

Las pruebas se escriben antes de implementar el código por los propios programadores, esto pretende que los mismos desarrolladores logren entender y comprender de manera clara de lo que deben implementar mediante la construcción del código que lo va a probar. (Vaca *et al.*, 2015)

Implementar el código que hace funcionar el ejemplo

En este paso las pruebas deben ser escritas en código, para permitir que se ejecuten automáticamente las veces que sea necesario, y el solo hecho de ejecutar las pruebas debe mostrar si la ejecución fue exitosa o no. (Vaca *et al.*, 2015)

Refactorizar

Permite conservar la calidad de la arquitectura, se modifica el diseño sin cambiar la funcionalidad, conservando las pruebas como reaseguro. (Vaca *et al.*, 2015)

1.3.8.6.2. Nivel Iteración o Funcional

En este nivel el desarrollo es guiado por pruebas de aceptación o ATDD (Acceptance TDD), y como se expone en el Reporte Técnico de Araújo (2007, p. 5) consta de los siguientes pasos:

- El usuario define las pruebas antes que las funcionalidades sean desarrolladas.
- Una vez escrito el código la prueba sirve como un criterio de validación.

La creación de pruebas es un proceso evolutivo con retornos que se dan en ventanas de tiempo más extendidas, a nivel de iteración; esto depende principalmente del usuario apoyado por verificadores y programadores. Dado que se necesita la automatización de las mismas se lo conoce también como “Executable Acceptance TDD” (EATDD) o “StoryTest-Driven Development”. (Araújo, 2007, p. 5)

1.3.8.7. Herramientas

Para aplicar TDD es primordial disponer de herramientas que permitan la automatización de las pruebas, para tales efectos se han desarrollado marcos que permiten definir y ejecutar los test tanto unitarias como de aceptación. (Araújo, 2007, p. 5)

En este ámbito la automatización se considera imprescindible, permite la ejecución rápida y eficaz de todas las pruebas existentes con inmediato retorno al programador, brindándole las necesarias certezas y posibilitar la prueba de regresión, evitando además el ciclo inadecuado de más presión, menos pruebas, más fallas, provocado por el apremio en los plazos. (Araújo, 2007, p. 5)

Las herramientas disponibles están pensadas en general para pruebas unitarias o pruebas de aceptación, aunque la división no es estricta y podrían emplearse en ambos niveles, ya sea mediante la incorporación de extensiones y bajo ciertas restricciones. (Araújo, 2007, p. 5)

De manera particular se han desarrollado varios marcos para pruebas open-source, con colaboración en su inicio de los proponentes principales de TDD. (Araújo, 2007, p. 5)

1.3.8.7.1. xUnit: Familia de marcos para pruebas unitarias

xUnit como tal no se considera como una herramienta, la letra x es un prefijo para referenciar de manera genérica a la variedad de frameworks basados en el original SUnit, que fue creado por Kent Beck para la plataforma SmallTalk, la cual se ha portado a una gran variedad de lenguajes y plataformas como Java (JUnit), .Net (NUnit), Python (PyUnit), Ruby (Rubyunit), PHP (PHPUnit), Perl (PerlUnit), C++ (CppUnit), etc. (Araújo, 2007)

JUnit: Es un conjunto de bibliotecas que se emplean al momento de programar para realizar las pruebas unitarias de aplicaciones Java, ejecuta las clases para comprobar que el funcionamiento de éstas es totalmente correcto. (Peñarrocha, 2016)

TestNG: Al igual que JUnit, se utiliza con Java para pruebas y testeo, se basa en JUnit y en NUnit (para .NET) y consta de nuevas funcionalidades que se describen: (Peñarrocha, 2016, p. 21)

- Permite anotaciones.
- Pruebas muy configurables.
- Soporta el paso de parámetros.
- Permite distribuir las pruebas en máquinas esclavas.
- Dependencia entre test, si un test falla, el resto de test que dependan de este no se ejecutan.
- Soportado por herramientas de desarrollo y plug-ins potentes e importantes como Eclipse o Maven.

CPPUnit: Basado en XUnit para pruebas unitarias, pero enfocado al lenguaje de programación C++, se encuentra en desarrollo activo y podemos encontrarlo en varias distribuciones Linux. Como los anteriores frameworks, también se puede integrar en Eclipse para poder manejarlo desde el mismo entorno y tiene la posibilidad de ejecutarse tanto en Windows como en Linux ya que está bien modulada. (Peñarrocha, 2016)

Visual Studio Unit Testing: Es un framework exclusivo de Visual Studio disponible para este entorno con diferentes versiones desde el año 2005, puede ejecutarse desde línea de comandos usando MSTest permitiendo realizar pruebas unitarias sin ninguna complicación y para el lenguaje que deseemos, lo que más usan en este entorno de desarrollo es C#. (Peñarrocha, 2016)

1.3.8.7.2. FIT: Marco para pruebas de aceptación

Es una herramienta creada por Ward Cunningham con el objetivo de automatizar las pruebas de aceptación y optimizar tanto la comunicación como la colaboración en el desarrollo de software, dirigidas por palabras claves y datos. Consta de los siguientes pasos:

- Lee tablas que contienen información para la prueba y los resultados esperados (ejemplos).
- Interpreta cada tabla con una clase (conocida como “Fixture”).
- Compara el resultado de correr el programa bajo prueba con los ejemplos suministrados en las tablas.

Las “fixtures” son escritas por los programadores, en el mismo lenguaje que el código de producción. Son pequeños fragmentos de código que se encargan de actuar como el pegamento entre las “fixtures” suministradas por el marco y las tablas con la especificación de las pruebas entradas por el usuario. (Araújo, 2007)

En general extienden las “fixtures” del marco o ejecutan bajo ellas. Al igual que los marcos Xunit busca utilizando “Reflection” dentro de los ejecutables las clases, variables y el resto de la información que necesita para ejecutar las pruebas, notificando al programador del resultado con similar código de colores. (Araújo, 2007)

El desarrollador puede agrupar las pruebas en conjuntos (“Suites”) para organizar su ejecución en una corrida, no se detiene la ejecución de las pruebas ante un error. Dado que FIT no provee una interfase gráfica para su utilización se han elaborado extensiones para cargarlo en IDE’s, tales como Eclipse [Ecl07], de los cuales el más conocido es FitNesse [MMW06]. (Araújo, 2007)

1.3.8.8. Fortalezas y Debilidades

TDD como toda disciplina tiene sus pros y contras, defensores y detractores, fortalezas y debilidades las cuales se presentan a continuación. (Araújo, 2007)

1.3.8.9. Fortalezas

- **Al elaborar las pruebas primero** el programador no concibe que la prueba destruye su código. Al invertir la posición, es el código el que salva la prueba, convirtiéndose el acto destructivo de la prueba en una forma de evitar obstáculos, paso a paso. (Araújo, 2007)
- **Al elaborar sus propias pruebas** no somete su código a pruebas ajenas, se pone la carga en la responsabilidad de elaborar buenos obstáculos para un código robusto que deberá sortearlos, sin la presión de la competencia. (Araújo, 2007)
- **Al automatizar** el programador inmediatamente obtiene resultados acerca del estado de calidad de la aplicación; si introduce mejoras en el código sabrá inmediatamente si pasan todas las pruebas. La automatización se escribe en el mismo lenguaje en el que se programa, y acompaña al código. No hay necesidad de aprender lenguajes nuevos. (Araújo, 2007)
- **En cuanto al método**, puede ser explorado desde el valor del método científico como una metáfora para TDD, especialmente al considerar la evolución de la teoría y el rol de la reproducibilidad en la experimentación. En las siguientes figuras se representa dicha metáfora alineando las prácticas con el método científico: (Araújo, 2007)

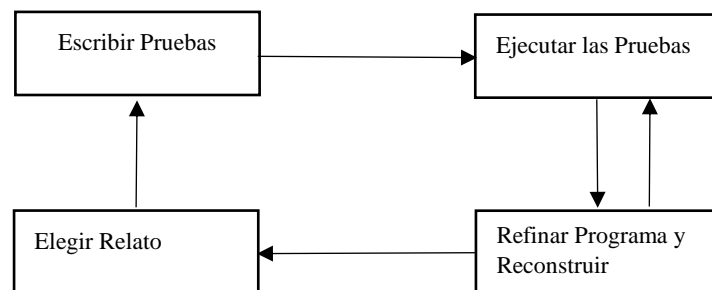


Figura 6-1: Test-Driven Development

Fuente: (Araújo, 2007)

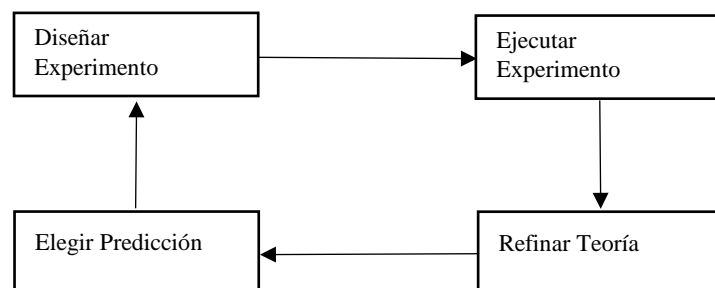


Figura 7-1: Método científico

Fuente: (Araújo, 2007)

- **Al prevenir defectos**, mediante el diseño continuo de pruebas.

1.3.8.10. Debilidades

- **Los ciclos extremadamente cortos y el cambio entre pruebas y codificación** de producción provocan que la mayoría de los programadores lo sientan como contrario a su intuición y desgastante. Exige disciplina, responsabilidad y coraje. (Araújo, 2007)
- **La imposibilidad de adopción** cuando se utilizan lenguajes de programación que no cuentan con el marco de pruebas unitarias, o cuando se utilizan herramientas de alto nivel (CASE) para la generación de código y asistencia al desarrollo que no lo integran. Si bien esto puede resolverse en muchos casos con desarrollo propio, no siempre es posible por problemas de costo y tiempo. (Araújo, 2007)
- **Las dificultades o imposibilidad** para su adopción cuando se utilizan herramientas de alto nivel (CASE) para la asistencia al desarrollo o lenguajes de programación que no adhieren al paradigma de la orientación a objetos. (Araújo, 2007)
- **En las pruebas de aceptación**, es más dificultosa la automatización, específicamente cuando la prueba puede ser asimilada a un ciclo funcional, lo que acrecienta la complejidad de los casos, con altamente probable intervención de interfaces, especialmente gráficas (GUI), desde las cuales la aplicación interactúa con el usuario. También es de particular importancia a este nivel el tiempo que transcurre entre la elaboración de la prueba y su ejecución, que ya no puede medirse en minutos, sino en días o iteraciones, lo cual quita ritmo al proceso y disminuye los aspectos positivos del rápido retorno. (Araújo, 2007)

1.3.8.11. Ventajas

Desde el punto de vista de Kent, su creador, TDD tiene los siguientes beneficios al emplear la metodología. (Torres, 2017)

- Aumenta la calidad del software, porque se elimina duplicidad y se comprende ampliamente desde los pasos iniciales.
- Se obtiene un código altamente reutilizable, debido a que se crean funcionalidades altamente reutilizables en diferentes clases.
- La jornada se hace mucho más agradable y facilita la relación de las personas.

- Trabajar en equipo es más sencillo, aunque tengamos compañeros con menor experiencia es más fácil confiar en ellos, ya que la comunicación entre los miembros, es más fluida.
- Multiplica la comunicación entre los miembros del equipo.
- Las personas encargadas de la garantía de calidad adquieren un rol más inteligente.
- Escribir el ejemplo (test) antes que el código nos obliga a escribir el mínimo de funcionalidad necesaria, evitando sobre diseñar.
- Un buen conjunto de pruebas de un proyecto desarrollado mediante TDD valen más que una documentación técnica.
- Incrementa la productividad del equipo desarrollador.
- Descubre y afrontar más casos de uso en tiempo de diseño.

1.3.8.12. Desventajas

- Incrementan el tiempo de desarrollo
- La curva de aprendizaje al principio es una desventaja, ya que adoptar dicha forma de trabajo al principio cuesta demasiado. (Peñarrocha, 2016)

1.3.9. Metodología SCRUM

1.3.9.1. Definición

Scrum es una metodología ágil y flexible e desarrollo de software para la gestión y el desarrollo de software, su finalidad 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, auto-gestión e innovación.*” (Clarís, 2018).

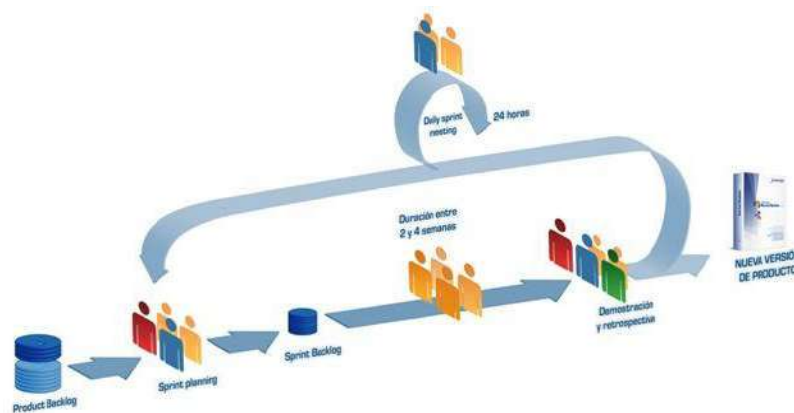


Figura 8-1: Proceso de SCRUM

Fuente: (Clarís, 2018)

1.3.9.2. Roles de la metodología Scrum

- **Product Owner:** Representa al cliente y a los que no están implicados en el proyecto de manera directa, se encarga de determinar los objetivos del proyecto con el propósito de garantizar que el equipo trabaje de manera adecuada para poder lograr los objetivos planteados. (Martínez, 2013)
- **Scrum Master:** Se encarga de asegurar que el resto del equipo no tenga ninguna dificultad para cumplir con sus tareas y funciones, además permite que el equipo se mantenga activo y productivo. (Martínez, 2013)
- **Scrum Team:** Se encarga de desarrollar y entregar el producto. Su trabajo es imprescindible, y presenta una estructura horizontal auto-organizada capaz de auto-gestionarse a sí misma. (Martínez, 2013)

1.3.9.3. Procesos

- **Product Backlog:** Conjunto de requisitos denominados historias, que son descritos en un lenguaje no técnico y priorizados de acuerdo a las necesidades. Los requisitos y prioridades se revisan y ajustan durante el curso del proyecto a intervalos regulares.
- **Sprint Planning:** Es una reunión durante la cual el Product Owner o dueño del producto presenta las historias del backlog por orden de prioridad y el equipo determina la cantidad de historias con la que se comprometen a completar en ese sprint o iteración, para en una segunda parte de la reunión, decidir y organizar cómo lo va a conseguir.
- **Sprint:** Iteración de duración prefijada durante la cual el equipo trabaja para convertir las historias del Product Backlog con las que se han comprometido, en una nueva versión del software totalmente funcional.
- **Sprint Backlog:** Lista de las historias necesarias para llevar a cabo las historias del sprint.
- **Daily sprint meeting:** Reunión diaria que dura como máximo 15 minutos, el equipo se coordina, cada miembro comenta que hizo el día anterior, que hará hoy y si hay impedimentos.
- **Demo y retrospectiva:** Reunión que se celebra al final de cada sprint, en la que el equipo presenta las historias conseguidas mediante una demostración del producto. Posteriormente,

en la retrospectiva, el equipo analiza qué se hizo bien, qué procesos serían mejorables y discute acerca de cómo perfeccionarlos (Dimes, 2015).

1.3.9.4. Beneficios

- **Cumplimiento de expectativas:** Cliente establece sus expectativas indicando el valor que le aporta cada requisito / historia del proyecto, el equipo los estima y con esta información el Product Owner establece su prioridad.
- **Flexibilidad a cambios:** Capacidad de reacción ante los cambios de requerimientos generados por necesidades nuevas o prioritarias del cliente o del mercado; diseñada para adaptarse a los cambios de requerimientos que conllevan los proyectos complejos.
- **Reducción del Time to Market:** Antes que el producto software esté finalizado por completo, el cliente puede empezar a utilizar sus funcionalidades ya implementadas.
- **Mayor calidad del software:** La metódica de trabajo y la necesidad de obtener una versión funcional después de cada iteración, ayuda a la obtención de un software de calidad superior.
- **Mayor productividad:** Se consigue, gracias a la eliminación del papeleo y a la motivación del equipo que proporciona el hecho de que sean autónomos para organizarse.
- **Maximiza el retorno de la inversión (ROI):** Producción de software con las prestaciones que aportan mayor valor de negocio gracias a la priorización por retorno de inversión.
- **Predicciones de tiempos:** Dado que el producto software se presenta cada determinado período de tiempo (2 semanas) se conoce la velocidad media del equipo por sprint, con lo que consecuentemente, es posible estimar fácilmente para cuando se dispondrá de una determinada funcionalidad que todavía está en el Backlog.
- **Reducción de riesgos:** Dado que las funcionalidades de más prioridad se desarrollan antes y se conoce la velocidad con que el equipo avanza en el proyecto, se puede reducir riesgos eficazmente de manera anticipada (Clarís, 2018).

1.3.10. Norma de evaluación ISO/IEC 25000

En el desarrollo de software la calidad es uno de los aspectos más importantes, motivo por el cual surge la familia de las normas ISO/IEC 25000, que tiene como propósito la creación de un cuadro de trabajo común para evaluar la calidad del software. Facilita una guía para el uso de la nueva serie de estándares internacionales denominada Requisitos y Evaluación de Calidad de Productos Software (*SQuaRE - System and Software Quality Requirements and Evaluation*). (Perez, 2015)

1.3.10.1. Definición

Se deriva de la evolución de otras normas anteriores, específicamente de las normas ISO/IEC 9126, que se refiere a las características de un modelo de calidad del producto software, e ISO/IEC 14598, que abordaba el proceso de evaluación de productos software y cuyo objetivo principal es guiar el desarrollo de software mediante la **especificación de requisitos y evaluación de características de calidad**. (ISO 25000, 2018b)

1.3.10.2. Actualización ISO/IEC 25000:2014

Anteriormente, en el **Modelo de Calidad** de la ISO/IEC 25000 la seguridad no se consideraba como una métrica, se basaba concretamente en seis métricas, se puede apreciar en la **Figura9-1**.

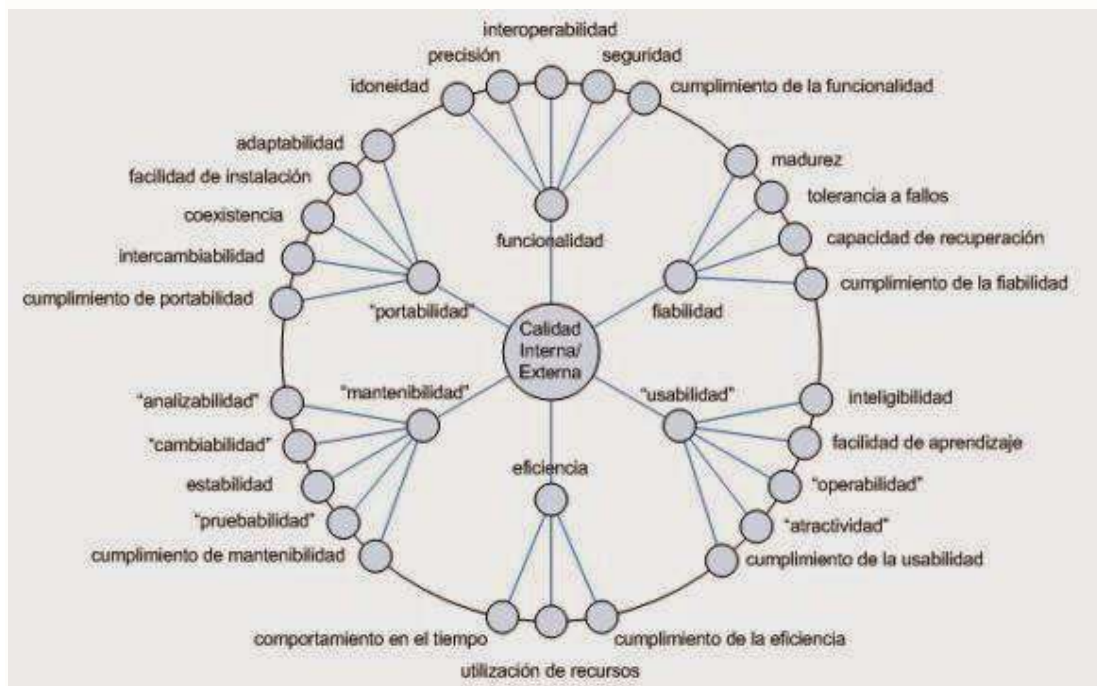


Figura 9-1: Modelo de Calidad ISO/IEC 25000

Fuente: (Rojo, 2015)

La seguridad se interpretaba como uno de los parámetros de la funcionalidad propia del software, esto cambió en el año 2014 con la actualización de la ISO/IEC 25000:2014, en la que se añadieron dos métricas al modelo anterior: Compatibilidad y Seguridad, obteniendo como resultado ocho métricas de calidad como se puede observar en la **Figura 10-1**.



Figura 10-1: Modelo calidad del producto software ISO/IEC 25010

Fuente: (Calabrese *et al.*, 2017)

En esta actualización la seguridad dentro del software pasa a ser una métrica con valor propio y que debe cumplir con algunas subcaracterísticas. (Rojo, 2015)

1.3.10.3. Divisiones familia de normas ISO/IEC 25000

Actualmente la familia de normas ISO/IEC 25000 se está compuesta por cinco divisiones, como podemos ver en la **Figura 11-1**.



Figura 11-1: Divisiones familia de normas ISO/IEC 25000

Fuente: (ISO 25000, 2018a)

ISO/IEC 2500n – División de Gestión de Calidad

Se conforma de:

- ISO/IEC 25000 - *Guide to SQuaRE*: incluye el modelo de la arquitectura de SQuaRE, la terminología de la familia, un resumen de las partes, los usuarios previstos y las partes asociadas, así como los modelos de referencia. (ISO 25000, 2018a)
- ISO/IEC 25001 - *Planning and Management*: define requerimientos y orientaciones para gestionar la evaluación y especificación de los requisitos del software. (ISO 25000, 2018a)

ISO/IEC 2501n – División de Modelo de Calidad

Presentan modelos de calidad detallados incluyendo características para calidad interna, externa y en uso del software, está conformada por:

- ISO/IEC 25010 - *System and software quality models*: describe el modelo de calidad para el producto software y para la calidad en uso. Expone las características y subcaracterísticas de calidad frente a las cuales evaluar el producto software. (ISO 25000, 2018a)
- ISO/IEC 25012 - *Data Quality model*: define un modelo general para la calidad de los datos, aplicable a aquellos datos que se hallan almacenados de forma estructurada y son parte de un Sistema de Información. (ISO 25000, 2018a)

ISO/IEC 2502n – División de Medición de Calidad

Incluyen un modelo de referencia de la medición de la calidad del producto, definiciones de calidad (interna, externa y en uso) y guías prácticas para su aplicación, se conforma de:

- ISO/IEC 25020 - *Measurement reference model and guide*: presenta una explicación introductoria y un modelo de referencia común a los elementos de medición de la calidad. También proporciona una guía para que los usuarios seleccionen o desarrollen y apliquen medidas propuestas por normas ISO. (ISO 25000, 2018a)
- ISO/IEC 25021 - *Quality measure elements*: define y especifica un conjunto recomendado de métricas base y derivadas que puedan ser usadas a lo largo de todo el ciclo de vida del desarrollo software. (ISO 25000, 2018a)

- ISO/IEC 25022 - *Measurement of quality in use*: define específicamente las métricas para realizar la medición de la calidad en uso del producto. (ISO 25000, 2018a)
- ISO/IEC 25023 - *Measurement of system and software product quality*: define específicamente las métricas para realizar la medición de la calidad de productos y sistemas software. (ISO 25000, 2018a)
- ISO/IEC 25024 - *Measurement of data quality*: define específicamente las métricas para realizar la medición de la calidad de datos. (ISO 25000, 2018a)

ISO/IEC 2503n – División de Requisitos de Calidad

Consiste en especificar requisitos de calidad que pueden ser utilizados en el proceso de elicitación de requisitos de calidad del software a desarrollar o como entrada del proceso de evaluación, se compone de:

- ISO/IEC 25030 - *Quality requirements*: provee de un conjunto de recomendaciones para realizar la especificación de los requisitos de calidad del software. (ISO 25000, 2018a)

ISO/IEC 2504n – División de Evaluación de Calidad

Incluye normas que proporcionan requisitos, recomendaciones y guías para llevar a cabo el proceso de evaluación del producto software, está formada por:

- **ISO/IEC 25040 - *Evaluation reference model and guide***: plantea un modelo de referencia general para realizar una evaluación, que considera las entradas al proceso de evaluación, las restricciones y los recursos necesarios para conseguir las correspondientes salidas. (ISO 25000, 2018a)

Reemplaza a la ISO/IEC 14598-1:1999. La nueva versión define 13 procesos en cinco etapas (Calabrese *et al.*, 2017, p. 662):

1. Establecer los requisitos de la evaluación:
 - a. Establecer el propósito de la evaluación.
 - b. Obtener los requisitos de calidad del producto.
 - c. Identificar las partes del producto que se deben evaluar.

- d. Definir el rigor de la evaluación.
2. Especificar la evaluación:
 - a. Seleccionar los módulos de evaluación.
 - b. Definir los criterios de decisión para las métricas.
 - c. Definir los criterios de decisión de la evaluación.
 3. Diseñar la evaluación:
 - a. Planificar las actividades de la evaluación.
 4. Ejecutar la evaluación:
 - a. Realizar las mediciones.
 - b. Aplicar los criterios de decisión para las métricas.
 - c. Aplicar los criterios de decisión de la evaluación.
 5. Finalizar la evaluación:
 - a. Revisar los resultados de la evaluación.
 - b. Crear el informe de evaluación.
 - c. Revisar la calidad de la evaluación y obtener feedback.
 - d. Tratar los datos de la evaluación.
- **ISO/IEC 25041 - *Evaluation guide for developers, acquirers and independent evaluators***: describe los requisitos y recomendaciones para la implementación práctica de la evaluación del producto software desde el punto de vista de los desarrolladores, de los adquirentes y de los evaluadores independientes. (ISO 25000, 2018a)
 - **ISO/IEC 25042 - *Evaluation modules***: define lo que la Norma considera un módulo de evaluación y la documentación, estructura y contenido que se debe utilizar a la hora de definir uno de estos módulos. (ISO 25000, 2018a)
 - **ISO/IEC 25045 - *Evaluation module for recoverability***: define un módulo para la evaluación de la subcaracterística Recuperabilidad (Recoverability). (ISO 25000, 2018a)

1.3.10.4. Modelo de Calidad - ISO/IEC 25010

El modelo de calidad representa la piedra angular en torno a la cual se establece el sistema para la evaluación de la calidad del producto. En este modelo se establecen las características de calidad a considerar a la hora de evaluar las propiedades de un determinado producto software.

El modelo de calidad del producto determinado por la ISO/IEC 25010 se halla compuesto por las ocho características de calidad que se exponen en la siguiente imagen, **Figura 12-1**:



Figura 12-1: Modelo calidad del producto software ISO/IEC 25010

Fuente: (ISO 25000, 2018a)

Adecuación Funcional

Representa la capacidad del producto software para proporcionar funciones que satisfacen las necesidades declaradas e implícitas, cuando el producto se usa en las condiciones especificadas. Se subdivide a su vez en las siguientes subcaracterísticas (ISO 25000, 2018a):

- Complejidad funcional
- Corrección funcional
- Pertinencia funcional

Eficiencia de desempeño

Representa el desempeño relativo a la cantidad de recursos utilizados bajo determinadas condiciones. Se subdivide a su vez en las siguientes subcaracterísticas (ISO 25000, 2018a):

- Comportamiento temporal
- Utilización de recursos
- Capacidad

Compatibilidad

Capacidad de dos o más sistemas o componentes para intercambiar información y/o llevar a cabo sus funciones requeridas cuando comparten el mismo entorno hardware o software. Se subdivide a su vez en las siguientes subcaracterísticas (ISO 25000, 2018a):

- Coexistencia
- Interoperabilidad

Usabilidad

Capacidad del producto software para ser entendido, aprendido, usado y resultar atractivo para el usuario, cuando se usa bajo determinadas condiciones. Se subdivide en las siguientes subcaracterísticas (ISO 25000, 2018a):

- Capacidad para reconocer su adecuación.
- Capacidad de aprendizaje
- Capacidad para ser usado
- Protección contra errores de usuario
- Estética de la interfaz de usuario
- Accesibilidad

Fiabilidad

Capacidad de un sistema para desempeñar las funciones especificadas, cuando se usa bajo unas condiciones y periodo de tiempo determinados. Se conforma de: (ISO 25000, 2018a):

- Madurez
- Disponibilidad
- Tolerancia a fallos
- Capacidad de recuperación

Seguridad

Capacidad de protección de datos de manera que personas o sistemas no autorizados no tengan acceso a los mismos. Se subdivide en las siguientes subcaracterísticas (ISO 25000, 2018a):

- Confidencialidad
- Integridad
- No repudio
- Responsabilidad
- Autenticidad

Mantenibilidad

Representa la capacidad del producto software para ser modificado efectiva y eficientemente, debido a necesidades evolutivas, correctivas o perfectivas. Se conforma de: (ISO 25000, 2018a)

- Modularidad
- Reusabilidad
- Analizabilidad
- Capacidad para ser modificado
- Capacidad para ser probado

Portabilidad

Capacidad del producto o componente de ser transferido de forma efectiva y eficiente de un entorno hardware, software, operacional o de utilización a otro. Se subdivide a su vez en las siguientes subcaracterísticas (ISO 25000, 2018a):

- Adaptabilidad
- Capacidad para ser instalado
- Capacidad para ser reemplazado

1.3.10.5. ISO/IEC 25010 – Seguridad

1.3.10.5.1. Definición

Capacidad de protección de la información y los datos de manera que personas o sistemas no autorizados no puedan leerlos o modificarlos.” (ISO 25000, 2018a)

1.3.10.5.2. Subcaracterísticas

Se subdivide en las siguientes subcaracterísticas (ISO 25000, 2018a):

1. **Confidencialidad.** Capacidad de protección contra el acceso de datos e información no autorizados, ya sea accidental o deliberadamente.
2. **Integridad.** Capacidad del sistema o componente para prevenir accesos o modificaciones no autorizados a datos o programas de ordenador.
3. **No repudio.** Capacidad de demostrar las acciones o eventos que han tenido lugar, de manera que dichas acciones o eventos no puedan ser repudiados posteriormente.
4. **Responsabilidad.** Capacidad de rastrear de forma inequívoca las acciones de una entidad.
5. **Autenticidad.** Capacidad de demostrar la identidad de un sujeto o un recurso.

CAPÍTULO II

2. MARCO METODOLÓGICO

El sistema web/móvil a implementar se lo denomina “SOLRIF”, el mismo que representa un proyecto relevante para un determinado sector la parroquia Santa Rosa, en la provincia de Tungurahua; por lo tanto el desarrollo de la aplicación se acoge a los reglamentos y ordenanzas que establece dicha institución, con el propósito de aportar mejoras significativas en los procesos de registro de consumo de agua y registro de pagos efectuados por los usuarios.

Para ello, se considera el uso de una metodología ágil, ideal al momento de desarrollar aplicaciones web y móviles, que garantice su construcción mediante pruebas asegurando la calidad y una rápida reacción ante cambios propuestos por el usuario quién, según esta metodología es considerado un pilar fundamental en el equipo de trabajo. La implementación de un sistema informático aplicando una metodología ágil como lo consideramos a SCRUM, permite al equipo de desarrollo ser altamente productivos y obtener el resultado esperado.

En este capítulo se expone el alcance del producto software, sus aspectos limitantes y se pone en ejecución las actividades definidas según la metodología SCRUM, para el desarrollo de la aplicación web/móvil.

2.1. Alcance del producto software

El proyecto a implementar para la Junta Administradora de Agua Potable Angahuana Alto tiene como alcance optimizar los procedimientos implicados en la gestión operaria y administrativa con respecto al registro de consumo de agua y recaudaciones, es por esto que el sistema informático a construir pretende automatizar primordialmente los procesos de registro de lecturas de medidor y el registro de pagos que efectúan los usuarios por concepto de las mismas; para lo cual se definen las siguientes procesos entre las principales:

- Crear usuarios en el sistema
- Registrar datos de personas/clientes
- Realizar proceso de adquisición de servicios de agua potable
- Registrar lectura del medidor mediante la aplicación móvil

- Registrar lectura del medidor mediante la aplicación web
- Generación de facturas y/o comprobantes.
- Efectuar recaudaciones por consumo de agua potable.
- Generar reportes

2.2. Limitaciones

Para el funcionamiento de la aplicación tanto web y móvil, es primordial que el servidor este siempre disponible y en conexión a la red LAN de la Junta, y a su vez contar con conexión a internet, este último para que la aplicación móvil pueda consumir los servicios provisto por el servidor desde cualquier lugar que se encuentre el plomero registrando la lectura del medidor.

La aplicación móvil se prueba con la versión Android 5.0 Lollipop en adelante, por lo tanto, no se garantiza el funcionamiento correcto con versiones anteriores a lo mencionado.

2.3. Descripción general del producto

A nivel general el sistema SOLRIF consta de cinco módulos principales, que a su vez abarcan las diferentes funcionalidades; totalizando así la implementación de 19 funcionalidades. Para mejor comprensión se expone la **Figura 1-2**.

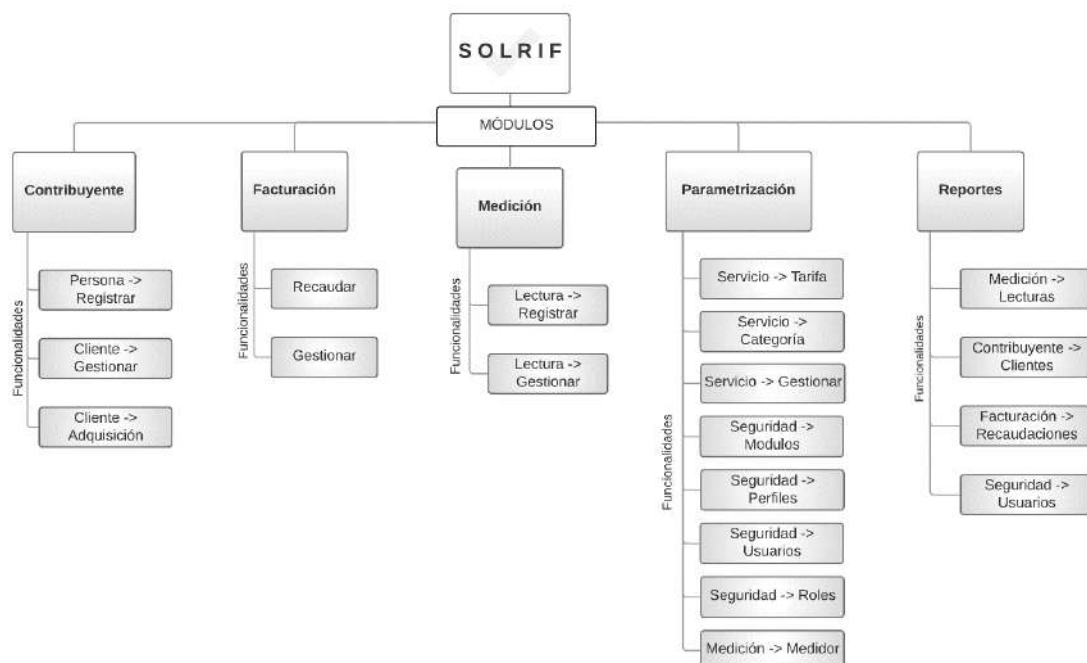


Figura 1-2: Módulos/Funcionalidades SOLRIF
Realizado por: Tupac Tisalema, 2018

- **Módulo parametrización:** Este módulo permitirá la gestión de usuarios, roles, tarifas y servicios; brindando la posibilidad de establecer parámetros necesarios para la ejecución de los procesos internos del sistema.
- **Módulo contribuyente:** En este módulo se permitirá registrar/actualizar datos de personas, adquirir servicios y registrar las cuentas por cobrar de cada cliente.
- **Módulo medición:** El presente módulo permitirá registrar las lecturas del medidor, realizadas por el plomero mediante la aplicación móvil y web.
- **Módulo facturación:** Este módulo permitirá efectuar los pagos, generar facturas comprobantes por adquisición de servicios y consumo de los mismos.
- **Módulo reportes:** Este módulo permitirá generar todos los reportes requeridos por el personal involucrado.

2.3.1. Aplicación web

La aplicación web consta de los 5 módulos mencionados anteriormente, las cuales abarcan procesos primordiales tales como; registrar datos de personas, que son necesarias para agregar nuevos clientes o su vez en caso de ser personal de la institución crear usuarios del sistema, realizar procesos como la adquisición de servicios, registro de lecturas de medidor, generación de facturas y/o comprobantes, registro de pagos efectuados por los usuarios, y adicionalmente también brinda la posibilidad de registrar las deudas por cobrar que mantiene los clientes con la Junta por concepto de adquisición o consumo efectuadas en fechas anteriores.

En cuanto a la administración del sistema, la aplicación web pone a disposición el módulo de parametrización, con funcionalidades que están pensadas específicamente para realizar procesos de gestión para usuarios, roles, tarifas, servicios, etc. y configuración de permisos de acceso al sistema. Se enfoca en la parte configurativa de aplicación para que el uso de la misma posibilite una administración eficiente y adecuada.

Por último, en base las necesidades del área administrativa ofrece un módulo de reportaría, con el propósito de que el personal pertinente tengan acceso a la información requerida y puedan generar informes necesarios para su respectiva gestión.

2.3.2. Aplicación móvil

La aplicación móvil nativa implementada en Android, dispone de una funcionalidad pensada exclusivamente para uso del personal asignado como plomero; la cual permite registrar las lecturas del medidor desde cualquier lugar o punto geográfico en la que se encuentra la persona encargada.

Esta funcionalidad se la utiliza de la siguiente manera; primero se debe buscar un cliente ingresando como criterio de búsqueda el número del medidor, en caso de existir el cliente se presenta los datos correspondientes y un formulario para proceder con el registro de la lectura del medidor, permitiendo también adjuntar una evidencia fotográfica.

2.3.3. Integración móvil

Para la integración de la aplicación móvil se consumen los servicios web basado en la arquitectura REST, que provee la aplicación web denominado SOLRIF, mediante esto se logra la comunicación e interacción entre servidor web y aplicación móvil.

2.4. Fase de Planificación

2.4.1. Preparación del Proyecto

La implementación de un sistema siempre estará sujeto al cumplimiento de actividades previamente definidas por una metodología, donde cada actividad se define en base a reuniones y diálogos propuestos por el Product Owner. El equipo de trabajo acoge las peticiones del cliente, las define como requerimientos funcionales y no funcionales, y obtiene como resultado una lista de requerimientos a ser ejecutadas en un plazo de tiempo planificado.

En cada requerimiento se describe las funcionalidades del sistema a desarrollarse, las cuales se implementa aplicando los diferentes métodos o técnicas existentes para la desarrollo de una aplicación. El desarrollo guiado por pruebas es una tendencia adoptada para la implementación de código limpio y construcción de software de manera ágil, basado en tres pilares fundamentales:

- Desarrollo del requerimiento propuesto por el cliente en su esencia de simpleza y funcionalidad.
- Disminución de errores o defectos de codificación de software a la fase de producción.

- Producción de software modular, código listo para cambios y adaptación en los requerimientos en todas sus fases de desarrollo aprovechando la reusabilidad de código.

Dentro del desarrollo de software se infiere temas sobre reuniones establecidas para la aceptación o modificación del desarrollo antes durante y después de la implementación, la definición de la arquitectura del aplicativo en el entorno de desarrollo, el modelado y diseño de base de datos de forma proporcional al crecimiento, las herramientas de software para el desarrollo, cambios propuestos por el cliente y adaptación del equipo de trabajo al proyecto software.

Teniendo en cuenta estas consideraciones se planifica las actividades a ejecutar, estableciendo plazos de tiempo para entregar, determinando fechas de inicio y fecha fin. Para mejor comprensión puede dirigirse al **Anexo A**.

2.4.2. Reuniones

Las reuniones tuvieron lugar en las instalaciones de la Junta Administra de Agua Potable con el personal administrativo del área y personal técnico cada 2 semanas, en el cual se expusieron requerimientos, los cuales fueron acogidos para posteriormente ejecutarlos. Cabe mencionar que las reuniones son permanentes con el cliente para que acepte el producto según la programación establecida para cada entregable del proyecto.

Tabla 1-2. Actividades Establecidas en las Reuniones

Asistentes	Actividades Propuestas	Resultado Obtenido	Fecha
Ing. Germania Veloz. Sr. Aníbal Guincho. Sr. Tupac Tisalema.	Recopilación de información necesaria para establecer necesidades.	Establecimiento la arquitectura modular del sistema.	04/09/2017 al 15/09/2017
Ing. Germania Veloz. Sr. Aníbal Guincho. Sr. Tupac Tisalema.	Definición de requerimientos funcionales para el sistema.	Creación de la tabla de requerimientos funcionales y sus prioridades de ejecución.	18/09/2017 al 29/09/2017
Ing. Germania Veloz. Sr. Aníbal Guincho. Sr. Tupac Tisalema.	Presentación de los módulos y requerimientos del sistema web propuesto.	Aceptación de los diferentes módulos del software	02/10/2017 al 19/01/2018
Ing. Germania Veloz. Sr. Aníbal Guincho. Sr. Tupac Tisalema.	Presentación de las necesidades del hardware necesario para la implantación del sistema web.	Adquisición del equipo hardware necesario para el despliegue de funciones.	22/01/2018 al 02/02/2018

Realizado por: Tupac Tisalema, 2019

Expuestas las reuniones se llegó a establecer 5 módulos primordiales para cumplirlas durante todas las fases de la metodología, los cuales son:

- Módulo Contribuyente
- Módulo Facturación
- Módulo Medición
- Módulo Parametrización
- Módulo Reportes

2.4.3. *Procesos a Automatizar*

A continuación, se presentan los procesos principales a automatizar dentro de la aplicación web:

- Registro de datos de personas
- Adquisición de servicio de agua potable
- Registro de lecturas de medidor
- Generación de facturas y/o comprobantes
- Registro de pagos
- Generación de reportes

2.4.4. *Personas y Roles del Proyecto*

Para la implementación del proyecto de titulación se establecieron los siguientes roles como lo califica la metodología, estos son:

Tabla 2-2. Personas y Roles del Proyecto

TOTAL	ROL	NOMBRE
1	Product Owner	Sr. Anibal Guincho
1	Stakeholders	Sr. Ernesto Toalombo
1	ScrumMaster	Ing. Germania Veloz
1	Team	Sr. Tupac Tisalema

Realizado por: Tupac Tisalema, 2019

2.4.5. *Pila del Producto (Product Backlog)*

Para la planificación del presente proyecto se hace uso de la técnica T-shirt (tallas de camiseta), la cual nos facilita realizar una estimación de tiempo por tareas; a continuación se muestra la **Tabla 3.2** con las estimaciones definidas.

Tabla 3-2. Tallas Definidas en T-Shirt

TALLA	DESCRIPCIÓN	PUNTOS ESTIMADOS	HORAS DE TRABAJO
XS	Muy pequeña	8	8
S	Pequeña	16	16
XM	Mediana	24	24
M	Grande	32	32
L	Muy grande	40	40

Realizado por: Tupac Tisalema, 2019

Las representaciones del conjunto de historias de usuario vienen dadas por el Product Backlog (Pila del Producto), se detallan a continuación las historias de usuario planteadas por el cliente y el equipo de trabajo.

Tabla 4-2. Pila del Producto

PILA DEL PRODUCTO			
ID.	DESCRIPCIÓN	ESFUERZO	TALLA
HT-01	Como desarrollador necesito obtener los requerimientos establecidos por el cliente.	16	S
HT-02	Como desarrollador necesito diseñar la arquitectura del sistema.	24	XM
HT-03	Como desarrollador necesito definir una estándar de codificación.	8	XS
HT-04	Como desarrollador necesito diseñar la base de datos.	32	M
HT-05	Como desarrollador necesito implementar procesos de conexión a la base de datos.	16	S
HT-06	Como desarrollador necesito implementar las clases en relación a las entidades de la base de datos para el correcto consumo de datos.	24	XM
HU-01	Como administrador necesito una interfaz de autenticación para ingresar las credenciales al identificarme en el sistema.	8	XS
HU-02	Como secretario necesito registrar los valores por cobrar que mantiene los usuarios con las Junta.	32	M
HU-03	Como administrador necesito autenticarme en el sistema para poder acceder.	40	L
HU-04	Como administrador deseo visualizar una interfaz de bienvenida al acceder al sistema.	8	XS
HU-05	Como secretario necesito registrar los datos de los usuarios o clientes de la Junta.	32	M
HU-06	Como secretario necesito realizar el proceso de adquisición del servicio de agua potable.	40	L
HU-07	Como plomero necesito registrar las lecturas del medidor desde el teléfono móvil.	40	L
HU-08	Como plomero necesito registrar las lecturas de medidor desde el sistema.	40	L

HU-09	Como tesorero necesito que se generen las facturas y/o comprobantes de cada usuario, para realizar los respectivos cobros.	40	L
HU-10	Como tesorero necesito realizar los cobros de las facturas generadas.	24	XM
HU-11	Como usuario necesito que se me entregue una factura o comprobante de pago.	16	S
HU-12	Como administrador necesito restringir el acceso a determinadas funcionalidades a los usuarios del sistema.	40	L
HU-13	Como administrador necesito parametrizar las tarifas a cobrar por el servicio de agua potable.	40	L
HU-14	Como administrador necesito que mi sesión caduque en un lapso de tiempo de inactividad.	8	XS
HU-15	Como administrador necesito gestionar los servicios ofertados por la Junta.	32	M
HU-16	Como administrador necesito gestionar los usuarios del sistema.	24	XM
HU-17	Como administrador necesito gestionar los roles existen en la Junta.	16	S
HU-18	Como plomero necesito autenticarme desde el teléfono móvil	24	XM
HU-19	Como administrador necesito gestionar los perfiles determinados para el funcionamiento del sistema.	16	S
HU-20	Como plomero necesito generar un reporte de lecturas de medidor por cliente.	8	XS
HU-21	Como tesorero necesito generar un reporte de las facturas cobradas en el día.	8	XS
HU-22	Como plomero necesito generar un reporte de lecturas de medidor por periodo.	8	XS
HU-23	Como administrador necesito anular las facturas generadas de manera errónea.	16	S
HU-24	Como secretario necesito generar un reporte de todos los clientes activos de la Junta.	8	XS
HU-25	Como administrador necesito generar un reporte de todos los usuarios registrados en el sistema.	32	M
HT-07	Como desarrollador necesito realizar la implantación del sistema.	32	M
HT-08	Como desarrollador necesito realizar una capacitación al personal de la Junta sobre las funcionalidades del sistema.	8	XM
HT-09.	Como desarrollador necesito realizar la documentación del sistema	40	L

Realizado por: Tupac Tisalema, 2019

Para el presente proyecto informático se define 25 historias de usuario y 9 historias técnicas, distribuidas en varios Sprints, mismos que al final componen el producto final entregable para la Junta de Agua Potable.

2.4.6. Factibilidad Económica

Para determinar si es factible el desarrollo del proyecto informático se hace una estimación de costos y esfuerzos requeridos, como se detalla a continuación.

2.4.6.1. Estimación de costos y esfuerzo

En este apartado se realiza una estimación del costo y esfuerzo, utilizando la métrica de puntos de función para del proyecto denominado “SOLRIF” que trata sobre el desarrollo de una aplicación web/móvil, para el registro de consumo de agua y los respectivos cobros.

Para la estimación se ha utilizado la métrica de puntos de función, ya que permitirá determinar el tamaño de la aplicación independientemente del lenguaje de programación o las tecnologías utilizadas.

Es necesario indicar que los puntos de función miden el tamaño de un sistema en términos de la cantidad de funcionalidad del sistema, para lo cual es necesario agrupar los requerimientos del sistema en grupos de cinco tipos: entradas, salidas, interacciones con el usuario, interfaces externas y archivos utilizados por el sistema, cabe recalcar que existen 21 requerimientos funcionales obtenidos para el desarrollo del sistema.

El estudio mencionado se encuentra en el **Anexo B**, y se determina que el tiempo estimado con un solo desarrollador para el finalizar el proyecto es de 7 meses aproximadamente y tendrá un costo estimado de \$2,930.00 dólares americanos.

Esto indica que es factible para la Junta administradora de agua potable dado que cuentas con los económicos para solventar los costos y esfuerzos requeridos para llevar a cabo este proyecto.

2.4.7. Factibilidad Técnica

Para la implantación del sistema web/móvil construido para la Junta de Administración de Agua solo es necesario adquirir la licencia de Windows server 2012 y adquirir un servidor para su implantación, concluyendo que se requiere de 2,970 dólares, en cuanto a la infraestructura requerida y equipos complementarios, para evidenciar diríjase al **Anexo C**.

Dado que la Junta cuenta con los recursos para adquirir los recursos primordiales requeridos, se procede con el desarrollo del sistema informático.

2.4.8. Análisis y Gestión de Riesgos

Gestionar riesgos consiste en un proceso sistemático de identificación, análisis y respuesta inmediata ante posibles problemas a suscitarse, implicando de manera directa a cada una de las fases de la implementación del sistema., el propósito es minimizar la probabilidad existente y las consecuencias de los eventos acontecidos.

Es de mucha relevancia la identificación del riesgo para posteriormente dar paso a un análisis cualitativo, ejecutar un plan de respuesta y establecer un control y seguimiento.

2.4.8.1. Riesgos del Software

Para analizar el riesgo se establece 2 posibilidades que son:

- **Incertidumbre:** el acontecimiento relacionado al riesgo puede ocurrir o no.
- **Pérdida:** el riesgo ocurre, se convierte en una realidad, acarreando consecuencias no deseadas.

Cuando se analice los riesgos se debe cuantificar el nivel de incertidumbre y el grado de pérdida que esta genera, para lo cual se determina 3 tipos de riesgos:

- Riesgo del proyecto
- Riesgos técnicos
- Riesgos del negocio

Tabla 5-2. Detalle de Impacto en el Riesgo

Atributo	Valor	Descripción
Impacto	Insignificantes	No merece ser tenido en cuenta.
	Tolerables	Están dentro de un margen de aceptación, no comprometen el desarrollo del proyecto.
	Graves	Comprometen gravemente el desarrollo consecutivo del proyecto o la organización.

	Catastrófica	Amenaza el avance del proyecto, organización o el producto.
Probabilidad	Muy baja	<10%
	Baja	Existencia del 10 al 25%
	Moderada	Existencia del 25 al 50%
	Alta	Existencia del 50 al 75%
	Muy alta	Existencia >75%
Marco de tiempo	Corto plazo	De 30 días
	Medio plazo	De 1 a 4 meses
	Largo plazo	Más de 4 meses

Realizado por: Tupac Tisalema, 2019

Tabla 6-2. Riesgos del Proyecto

Riesgo	Probabilidad de Ocurrencia	Impacto	Momento	Estrategia
Riesgo del proyecto				
Reducción del presupuesto	Muy baja	Grave	Inicio/durante	Sumar el 20% al costo real del proyecto.
Planificación temporal	Baja	Tolerable	Durante el proyecto	Realizar un descuento del 6% por cada mes de retraso.
Personal (asignación y organización)	Baja	Tolerable	Durante el proyecto	Capacitación el personal.
Recursos	Muy baja	Tolerable	Durante el proyecto	Optimizar los productos.
Cientes y sus requisitos	Moderada	Tolerable	Durante el proyecto	Realizar reuniones con el cliente y que la comunicación sea permanente.
Riesgo Técnico				
Diseño	Moderada	Tolerable	Durante el proyecto	Crear un diseño simple y llamativo.
Mantenimiento	Baja	Tolerable	Durante el proyecto	Fijar fechas para dar un mantenimiento al sistema.
Verificación	Baja	Grave	Durante el proyecto	Establecer periodos de prueba para realizar cambios y las

				posteriores aceptaciones.
Interfaz	Moderada	Grave	Durante el proyecto	Fijar una correcta usabilidad de la interfaz.
Implementación	Baja	Grave	Durante el proyecto	Brindar capacitación al administrador del sistema.
Riesgo del Negocio				
Riesgo estratégico	Muy baja	Tolerable	Inicio/durante	Establecer modificaciones que se ajusten al negocio.
Riesgo de perder contacto personal requirente del software	Muy baja	Grave	Inicio/durante	Firmar un contrato ante un notario estableciendo posibles sanciones por incumplimiento de las partes.
Cierre de operaciones	Baja	Grave	Inicio/durante	Ofrecer a otras empresas el producto sin ninguna limitación del producto.
Rotación del personal	Alta	Tolerable	Inicio/durante	Capacitar al personal permanentemente, nuevos o personal antiguo.

Realizado por: Tupac Tisalema, 2019

2.5. Fase de Diseño

2.5.1. Diagrama de Casos de Uso

Se identificó 4 actores principales para los procesos a automatizar, siendo el administrador encargado del control total del sistema, el plomero cumpliendo con la tarea de tomar las lecturas de medidor, el tesorero quién efectúa recaudaciones y el secretario asignado para registrar datos de clientes y procesar adquisiciones de servicios. Mediante los diagramas de casos de uso se

expone la interacción de los usuarios con el sistema, considerando los diferentes procesos involucrados entre los actores.

2.5.1.1. Procesos de Administrador

El Administrador es la persona encargada de gestionar el sistema, parametriza módulos, funcionalidades, perfiles, usuarios, roles, tarifas, servicios, categorías y también realiza la anulación de facturas generadas y/o emitidas. Ver **Figura 1-2**.

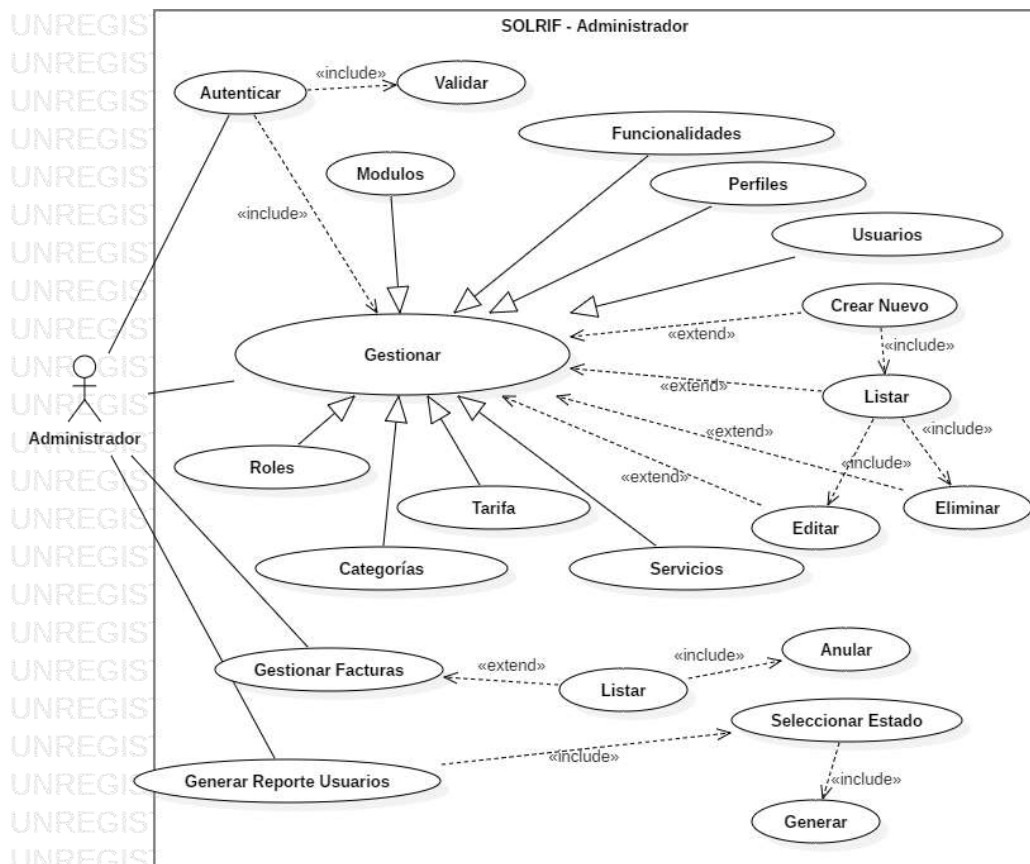


Figura 1-2: Diagrama de Caso de Uso Procesos de Administrador

Realizado por: Tupac Tisalema, 2019

2.5.1.2. Procesos de Secretario

El secretario cumple con la tarea de registrar los datos de los nuevos clientes y gestionar dichos datos en caso de requerirlo, también realiza procesos de adquisición de servicios, en el cual un cliente contrata servicios ofertados por la institución como el servicio de agua potable específicamente.

En el siguiente diagrama de caso de uso se describe los procesos realizados por dicho actor.

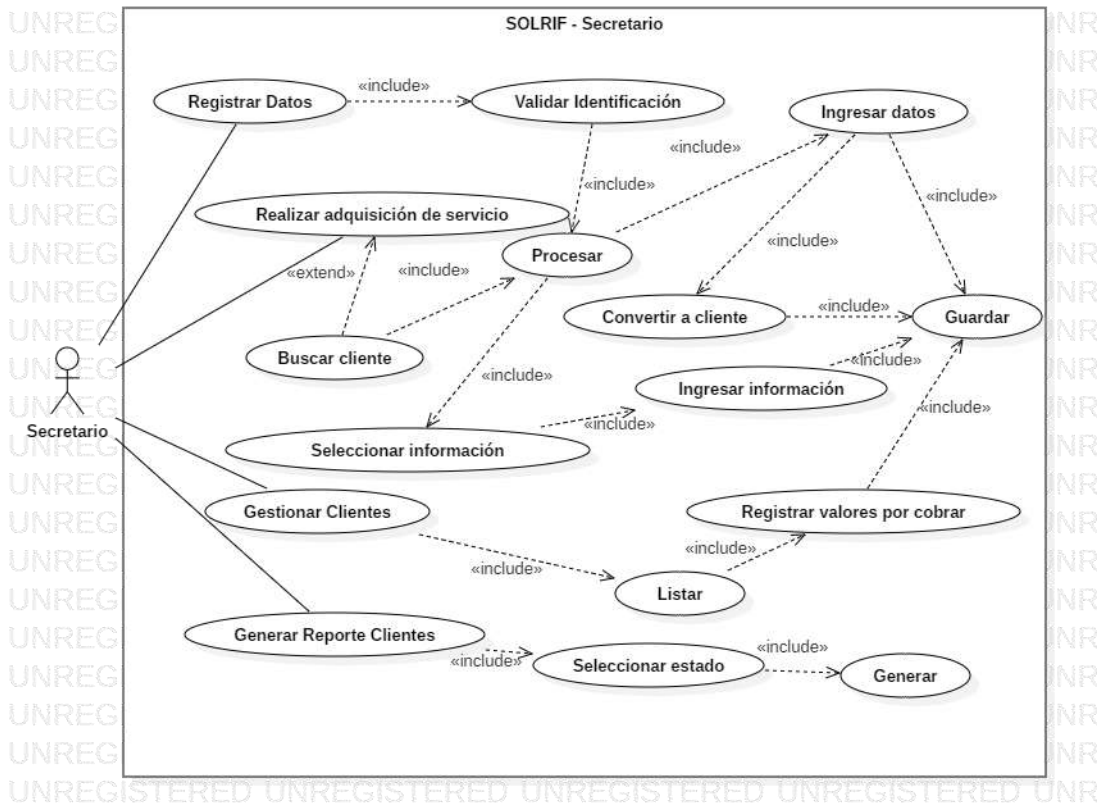


Figura 2-2: Diagrama de Caso de Uso Procesos de Secretario

Realizado por: Tupac Tisalema, 2019

2.5.1.3. Procesos de Tesorero

Cada mes se recauda los valores generados por el consumo de agua potable; una vez que se registrar las lecturas del medidor, el sistema realiza los respectivos procesos de cálculo para obtener los valores a cobrar, y posterior se genera una factura por concepto de agua potable y otros servicios adquiridos por el cliente. Realizar el cobro de estas facturas generadas es una tarea vital del Tesorero.

A continuación en la **Figura 3-2** se detalla los procesos realizados al respecto.

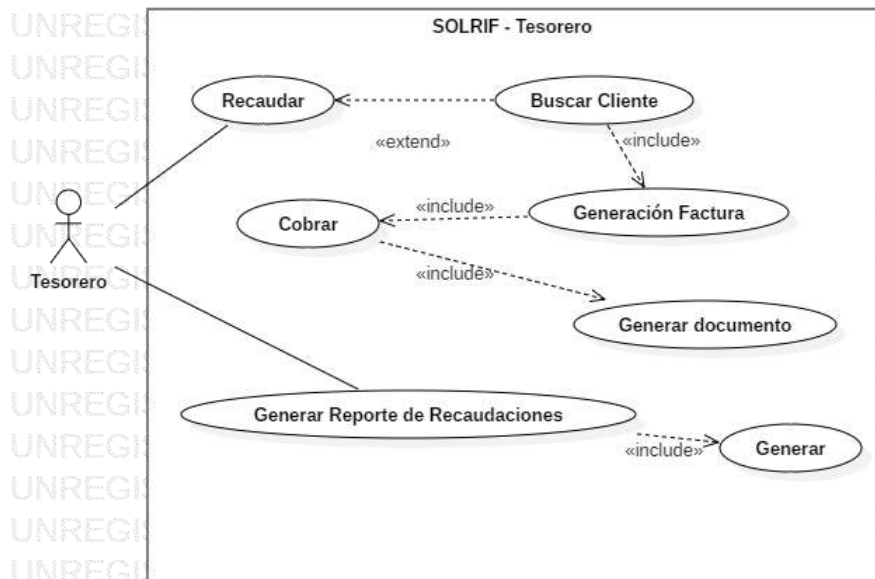


Figura 3-2: Diagrama de Caso de Uso Procesos de Tesorero
 Realizado por: Tupac Tisalema, 2019

2.5.1.4. Procesos de Plomero

Al cumplirse un mes desde el último consumo de agua registrado, el plomero de la institución visita la residencia de cada cliente, toma registro del nuevo consumo hecho durante el mes y sube esa información al sistema mediante una aplicación móvil.

En el siguiente diagrama se expone los procesos efectuados por el Plomero.

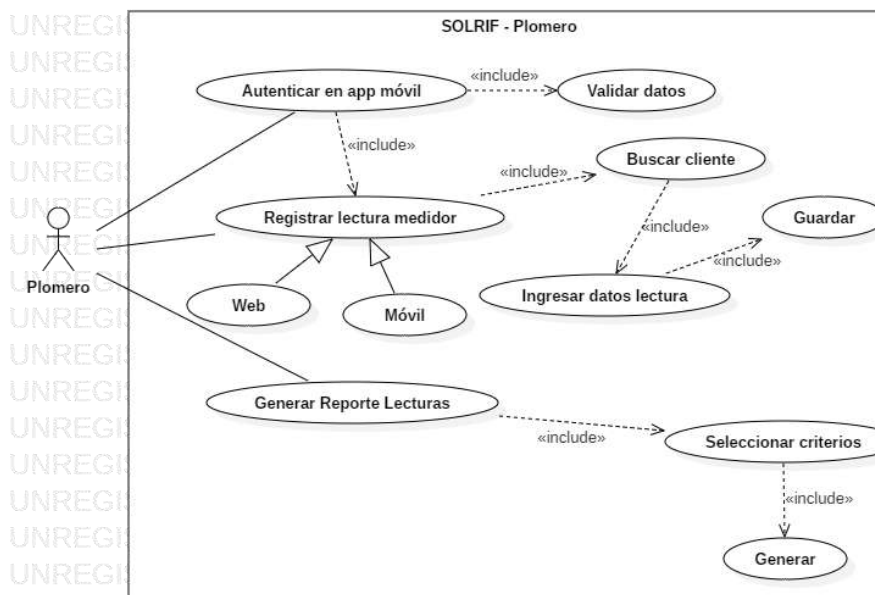


Figura 4-2: Diagrama de Caso de Uso Procesos de Plomero
 Realizado por: Tupac Tisalema, 2019

2.5.2. Diagrama de Clases

Para el sistema se contabilizó un total de 20 clases consideradas principales, posterior, a este diseño se toma en cuenta para el desarrollo de la base de datos, teniendo cada clase sus propios atributos y métodos; entre las cuales tenemos a continuación en la **Figura 5-2**.

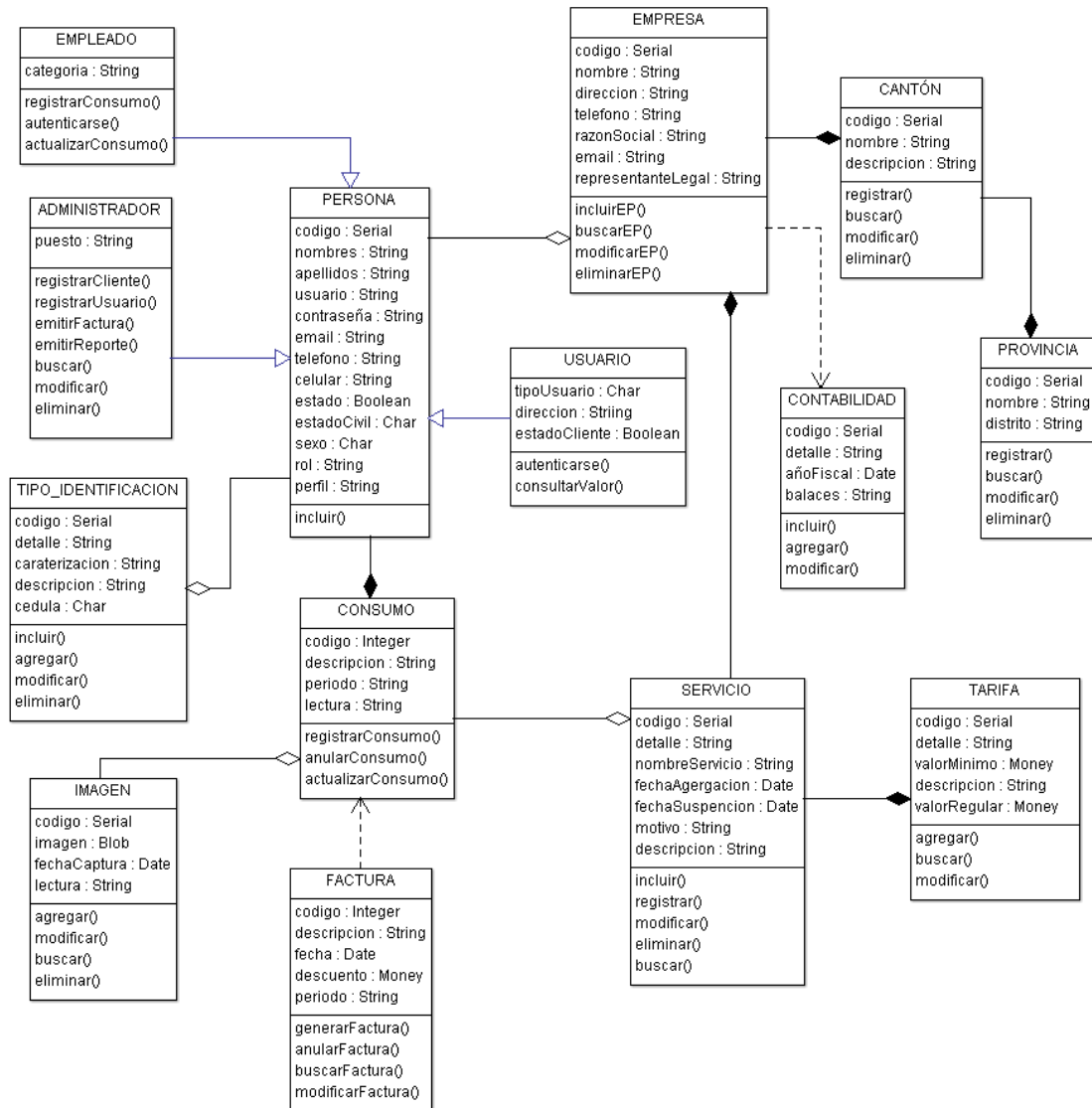


Figura 5-2: Diagrama de Clases

Realizado por: Tupac Tisalema, 2019

2.5.3. Arquitectura del Sistema

De cara al desarrollo de la aplicación web/móvil se establece una arquitectura en 3 capas que son: capa de datos, capa de negocio y capa de presentación. Para establecer estas capas se diseña la

arquitectura de SOLRIF, partiendo del patrón de arquitectura MVC, donde cada componente debe cumplir con tareas específicas que se detallan a continuación:

- **El modelo:** Dentro de la arquitectura MVC tenemos la lógica de negocio establecido, teniendo un acceso de manera directa a los datos, cabe mencionar que los datos dentro de la capa modelo se encuentran encapsulados.
- **El controlador:** El controlador en la arquitectura propuesta es parte fundamental dentro de la composición interna de funcionalidad siendo quien gestiona las peticiones realizadas por uno o varios clientes en la capa intermedia, además de ello es el componente encargado de conectar al resto de componentes presentes.

Su funcionalidad es la siguiente: Por ejemplo, un usuario del sistema quiere acceder hacia sus datos de consumo de un periodo de tiempo, el controlador es el encargado de solicitar los datos necesarios al modelo, cuando este responde es enviada dicha información a la vista para que sea presentada al cliente con un formato entendible y única respondiendo a la petición realizada.

- **La vista:** La vista es la encargada de realizar las respuestas ante peticiones hechas por el cliente, representado la información en archivos de tipo XML, XHTML que son enviados al cliente cuando la transacción de datos culmino y dio su respuesta el controlador.

En el **Figura 7-2**, se presenta el diseño de la arquitectura y la interacción entre cada una de las capas mencionadas.

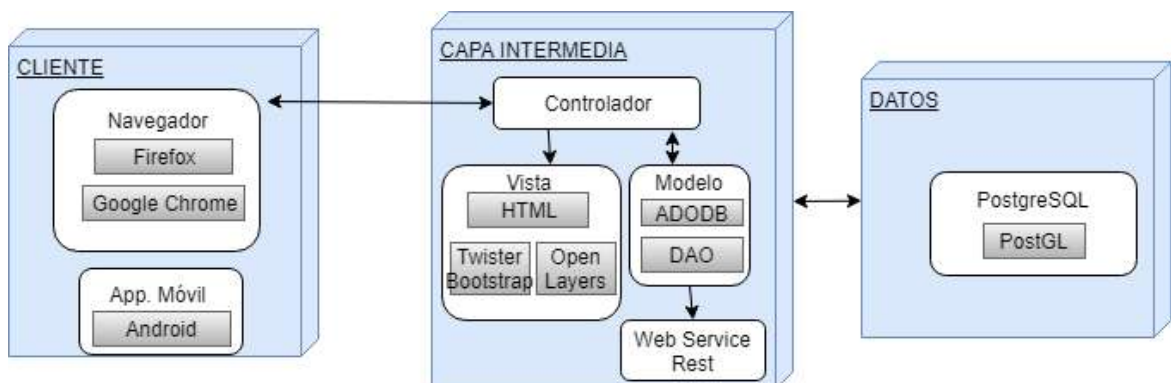


Figura 6-2: Arquitectura de la aplicación web/móvil
Realizado por: Tupac Tisalema, 2019

2.5.3.1. Arquitectura de la aplicación móvil

Para el desarrollo del servicio web se diseña el estilo de arquitectura REST en el lado del servidor y en lado del cliente se usa el patrón de arquitectura MVC; pero cabe mencionar que la mayor parte de la responsabilidad pasa al lado del cliente.

A continuación se describen las tareas que deben cumplir cada uno de los componentes del MVC.

- **Vista:** XML en Android.
- **Controlador:** Gestiona las peticiones realizadas por uno o varios clientes en la capa intermedia.
- **Modelo:** Acceso de manera directa a los datos, mediante los métodos encargados de realizar el consumo de los servicios web que proporciona la aplicación web SOLRIF.

En el **Figura 8-2**, se observa la interacción entre cada una de las capas descritas anteriormente.

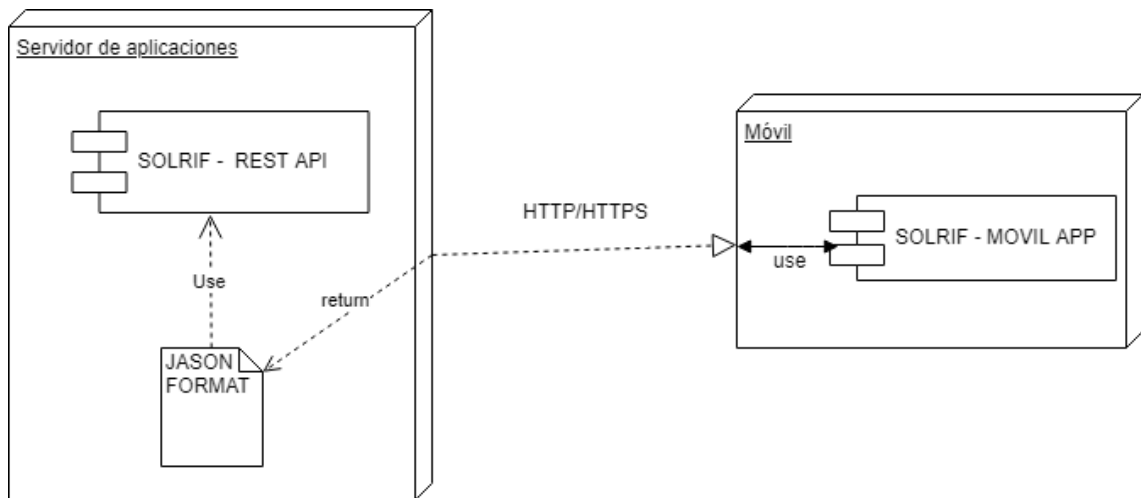


Figura 7-2: Arquitectura del Sistema - móvil

Realizado por: Tupac Tisalema, 2019

2.5.4. Recursos Necesarios

Las siguientes herramientas hardware y software fueron necesarios para la implementación del proyecto software, es así que para la fase de análisis, diseño y construcción del sistema se utilizó en todo momento herramientas debidamente probadas con anterioridad y que por supuesto que aseguren un correcto funcionamiento en cada proceso empleado.

Tabla 7-2. Recursos Necesarios Para la Implementación

TIPO DE RECURSOS	DESCRIPCIÓN
Hardware	DELL Inspiron 14-3467 Intel Core i5 72007 Servidor HPE ML30 Gen9 E3-1220v5-MCA Svr/S/Buy
Software	NetBeans 8.1 PostgreSQL 9.2 MySQL Workbench 5.2 Android Studio 3.0 ArgoUML Microsoft Office 2016
Material software adicional	Disco Duro Samsung 1Tb Flash Memory Kingston 8Gb

Realizado por: Tupac Tisalema, 2019

Con la tabla expuesta de recursos necesarios tenemos, que el software utilizado para el desarrollo del sistema web fue netamente de código libre, no teniendo ningún impedimento al momento de obtenerlos en línea, mismos que cuentan con un gran soporte, avalados y probados por grupos organizados de software libre.

2.5.5. Estándar de Codificación

En la fase de desarrollo tenemos estilos de codificación en todas sus formas, sea a nivel de base de datos o en su estructura de programación, para esto se ha utilizado el estándar CamelCase como base para la composición en la declaración de variables tanto compuesta como independientes.

Conocida como joroba de camello; existiendo tan solo en dos formas de CamelCase; UpperCamelCase y LowerCamelCase, para el proyecto de titulación presentado se utilizó la segunda mencionada, considerando esto tenemos la composición de variables al siguiente aspecto.

La primera letra de cada palabra inicia con minúscula si es una palabra compuesta tenemos; <clasePersona>, si no es compuesta quedará de la siguiente manera <reportes>. Al aplicar el estándar ya definido aplicamos a la construcción del sistema para la Junta de Administración de Agua Potable de la provincia de Tungurahua representado a continuación:

Tabla 8-2. Estándar de Codificación CamelCase

COMPONENTE	APLICADO	DEFINICIÓN
Clases	<claseEmpleado>	La primera letra del nombre de la clase representado al inicio con una letra minúscula seguido de la segunda palabra que comienza con mayúscula.
Variables	<cedulaPersona>	Las variables a ser declaradas dentro del panorama de codificación y al ser compuestas la primera es con minúscula seguido de la siguiente palabra con mayúscula, como podemos evidenciar en el ejemplo expuesto.
Paquetes	<accesoDatos>	Los paquetes a ser convocados dentro de las clases del sistema están compuestos de la manera lowerCamelCase.
Funciones	<registrarCliente>	La misma propuesta que tenemos para la definición de una clase, sabiendo que las funciones creadas nos devuelve un valor.
Métodos	<actualizarRegistro>	Aplicamos la propuesta lowerCamelCase para la puesta en marcha de métodos usados dentro de la estructura del sistema.
TextBox	<txtTelefonoCliente>	Para estos componentes tenemos un prefijo para su iniciación de letra respetando el estándar elegido, empieza con txt...
Botones	<btnIngresarLectura>	Para estos componentes tenemos un prefijo para su iniciación de letra respetando el estándar elegido, empieza con btn...
CheckBox	<chxTipoUsuario>	Para estos componentes tenemos un prefijo para su iniciación de letra respetando el estándar elegido, empieza con chx...
Cuadros de diálogo	<detalle>	El nombre empieza con una letra minúscula de ser simple, al ser compuesta la palabra que le anteponga deberá iniciar con mayúscula.
Menús	<mnConsumo>	Para estos componentes tenemos un prefijo para su iniciación de letra respetando el estándar elegido, empieza con mn...

Realizado por: Tupac Tisalema, 2019

2.5.6. *Diseño de la Interfaz de Usuario*

La interfaz de usuario al ser el medio de comunicación directa con el usuario, esta contener una navegabilidad aceptable mostrando simpleza y funcionalidad que es lo que el cliente prioriza como medio tecnológico de ayuda a sus labores cotidianos, es por ello que se usaron para la realización de este sistema interfaces simples, pero a la vez funcionales y que llamen la atención del cliente; combinando una mezcla de colores y logotipos de la Institución se pudo obtener una aceptación del producto en tanto a su aspecto.

Ya implementado el sistema en su totalidad el cliente realizó la respectiva navegabilidad por los módulos creados, llegando a tener acogida y posterior aceptación, por lo tanto, no se tuvo que realizar modificaciones a las interfaces creadas para la administración y gestión del sistema.

Todo lo expuesto anteriormente se encuentra detallado en el **Anexo D**.

2.5.7. Diseño de la Base de Datos

Se realiza el diseño de la base de datos con el fin de almacenar los datos estructurada y organizadamente, y que se pueda tener disponibilidad de la información de manera inmediata. La base de datos se obtuvo al analizar el proceso del registro de la lectura del medidor de agua de la Junta Administradora del agua, y además del proceso de cobros por servicios, obteniéndose 50 tablas principales las mismas que se evidencian el **Anexo E**.

2.6. Fase de Desarrollo

La fase de desarrollo es la parte central de la construcción del software, la metodología SCRUM permite definir las iteraciones o sprints, donde cada sprint consta de una o más historias de usuario e historias técnicas a implementarse. Para este proyecto se define 9 sprints a realizarse y se definen las fechas de entrega, mismas que fueron aceptadas por el cliente.

Para la implementación del sistema web/móvil se dispone de 25 historias de usuario y 9 historias técnicas distribuidas en los diferentes sprints.

2.6.1. Pila de Sprint (Sprint Backlog)

El distribuir las historias de usuario e historias técnicas según la prioridad de ejecución propuesta por el dueño del sistema, permite asignar tareas al equipo de trabajo y que estas se ejecuten en un determinado plazo de tiempo, para posteriormente entregar el producto con las funcionalidades creadas y así determinar la aceptación o modificación de la funcionalidad desarrollada.

Una vez que se determina los requerimientos con sus respectivas prioridades, se planifica el desarrollo de las historias técnicas y de usuario de acuerdo a la prioridad obtenida en la **Tabla 4-2. Pila del Producto**, además se asigna un responsable para el cumplimiento de las actividades. A continuación se expone lo descrito en la **Tabla 9-2**.

Tabla 9-2. Product Backlog

ITERACIÓN	DESCRIPCIÓN	FEC. INICIO	FEC. FIN	ESFUERZO	RESPONSABLE
Iteración 0	HT-01. Como desarrollador necesito obtener los requerimientos establecidos por el cliente.	04/09/2017	05/09/2017	16	Tupac Tisalema
	HT-02. Como desarrollador necesito diseñar la arquitectura del sistema.	06/09/2017	08/09/2017	24	
	HT-03. Como desarrollador necesito definir una estándar de codificación.	11/09/2017	11/09/2017	8	
	HT-04. Como desarrollador necesito diseñar la base de datos.	12/09/2017	15/09/2017	32	
Iteración 1	HT-05. Como desarrollador necesito implementar procesos de conexión a la base de datos.	18/09/2017	19/09/2017	16	Tupac Tisalema
	HT-06. Como desarrollador necesito implementar las clases en relación a las entidades de la base de datos para el correcto consumo de datos.	20/09/2017	22/09/2017	24	
	HU-01. Como administrador necesito una interfaz de autenticación para ingresar las credenciales al identificarme en el sistema.	25/09/2017	25/09/2017	8	
	HU-02. Como secretario necesito registrar los valores por cobrar que mantiene los usuarios con las Junta.	26/09/2017	29/09/2017	32	
Iteración 2	HU-03. Como administrador necesito autenticarme en el sistema para poder acceder.	02/10/2017	06/10/2017	40	Tupac Tisalema
	HU-04. Como administrador deseo visualizar una interfaz de bienvenida al acceder al sistema.	09/10/2017	09/10/2017	8	
	HU-05. Como secretario necesito registrar los datos de los usuarios o clientes de la Junta.	10/10/2017	13/10/2017	32	
Iteración 3	HU-06. Como secretario necesito realizar el proceso de adquisición del servicio de agua potable.	16/10/2017	20/10/2017	40	Tupac Tisalema
	HU-07. Como plomero necesito registrar las lecturas del medidor desde el teléfono móvil.	23/10/2017	27/10/2017	40	

Iteración 4	HU-08. Como plomero necesito registrar las lecturas de medidor desde el sistema.	06/11/2017	10/11/2017	40	Tupac Tisalema
	HU-09. Como tesorero necesito que se generen las facturas y/o comprobantes de cada usuario, para realizar los respectivos cobros.	13/11/2017	17/11/2017	40	
Iteración 5	HU-10. Como tesorero necesito realizar los cobros de las facturas generadas.	20/11/2017	22/11/2017	24	Tupac Tisalema
	HU-11. Como usuario necesito que se me entregue una factura o comprobante de pago.	23/11/2017	24/11/2017	16	
	HU-12. Como administrador necesito restringir el acceso a determinadas funcionalidades a los usuarios del sistema.	27/11/2017	01/12/2017	40	
Iteración 6	HU-13. Como administrador necesito parametrizar las tarifas a cobrar por el servicio de agua potable.	04/12/2017	08/12/2017	40	Tupac Tisalema
	HU-14. Como administrador necesito que mi sesión caduque en un lapso de tiempo de inactividad.	11/12/2017	11/12/2017	8	
	HU-15. Como administrador necesito gestionar los servicios ofertados por la Junta.	12/12/2017	15/12/2017	32	
Iteración 7	HU-16. Como administrador necesito gestionar los usuarios del sistema.	18/12/2017	20/12/2017	24	Tupac Tisalema
	HU-17. Como administrador necesito gestionar los roles existen en la Junta.	21/12/2017	22/12/2017	16	
	HU-18. Como plomero necesito autenticarme desde el teléfono móvil	25/12/2017	27/12/2017	24	
	HU-19. Como administrador necesito gestionar los perfiles determinados para el funcionamiento del sistema.	28/12/2017	29/12/2017	16	
Iteración 8	HU-20. Como plomero necesito generar un reporte de lecturas de medidor por cliente.	08/01/2018	08/01/2018	8	Tupac Tisalema

	HU-21. Como tesorero necesito generar un reporte de las facturas cobradas en el día.	09/01/2018	09/01/2018	8	
	HU-22. Como plomero necesito generar un reporte de lecturas de medidor por periodo.	10/01/2018	10/01/2018	8	
	HU-23. Como administrador necesito anular las facturas generadas de manera errónea.	11/01/2018	12/01/2018	16	
	HU-24. Como secretario necesito generar un reporte de todos los clientes activos de la Junta.	15/01/2018	15/01/2018	8	
	HU-25. Como administrador necesito generar un reporte de todos los usuarios registrados en el sistema.	16/01/2018	19/01/2018	32	
Iteración 9	HT-07. Como desarrollador necesito realizar la implantación del sistema.	22/01/2018	25/01/2018	32	Tupac Tisalema
	HT-08. Como desarrollador necesito realizar una capacitación al personal de la Junta sobre las funcionalidades del sistema.	26/01/2018	26/01/2018	8	
	HT-09. Como desarrollador necesito realizar la documentación del sistema	29/01/2018	02/02/2018	40	

Realizado por: Tupac Tisalema, 2019

2.6.2. Definición de Historias de usuarios

Las historias de usuario son una representación de los requerimientos que describe que función cumple cada requerimiento en el sistema, debe ser clara y comprensible permitiendo a los usuarios y desarrolladores mantener un control sobre el estado de cada una.

Cuentan con una descripción acerca del número de historia de la que se está tratando, nombre de la historia de usuario que se especificó en el product backlog, puntos estimados en base a la experiencia en desarrollo, puntos reales utilizados para cumplir la actividad, prioridad de ejecución propuesto por el product owner, riesgo que tiene la actividad en el desarrollo, la descripción de la historia de usuario y la observación de funcionalidad en el sistema.

A continuación, tomamos como referencia una historia de usuario con lógica de negocio propuesta por el cliente (ProductOwner) exceptuando la una de la otra ya que como se conoce, tenemos 25 historias de usuario planteadas.

Tabla 10-2. Historia de Usuario 07

Historia de Usuario	
Número: HU_07	Como plomero necesito registrar las lecturas del medidor desde el teléfono móvil.
Modificación de historia de usuario:	
Usuario: Plomero	Iteración Asignada: 1
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 20
Riesgo en el Desarrollo: Alto (Alto / Medio / Bajo)	Puntos Reales: 24
Descripción: Como <i>plomero</i> necesito <i>registrar las lecturas del medidor</i> desde el domicilio de cada usuario mediante el teléfono móvil, para almacenar la respectiva información de manera inmediata en el sistema.	
Observaciones: Pueden acceder a la apelación móvil los usuarios que tengan asignado el rol de plomero.	

Realizado por: Tupac Tisalema, 2019

El resto de las historias de usuario las podemos encontrar en el **Anexo F**.

2.6.3. *Tablero de Tareas*

Se tiene como propósito la creación de un tablero, donde se detalle las tareas desprendidas de cada historia de usuario; la idea es comunicar el progreso paulatino de cada tarea, si aún no se las cumple, si está en progreso o ya se dio por culminada, de esta manera el equipo de trabajo tiene conocimiento del avance que esta tiene y el desempeño de la persona encargada sobre dicha tarea.

A continuación, se analiza las tareas desarrolladas en la iteración 2, la cual dispone de 3 historias de usuario.

HU-03. Como administrador necesito autenticarme en el sistema para poder acceder.

Tabla 11-2. Tablero de Tareas HU-03

H.U.	Pendiente	En Proceso	Terminadas
Como administrador necesito autenticarme en el sistema para poder acceder.			T1-HU-03: Creación de la tabla usuario y tablas relacionadas a la misma.
			T2-HU-03: Creación de la clase modelo mUsuario, con el método iniciarSesion ().
			T3-HU-03: Proceso pruebas unitarias de mUsuario.
			T4-HU-03: Creación del controlador ControladorLogin, con sus atributos y métodos pertinentes incluido el método iniciarSesión().
			T5-HU-03: Proceso pruebas unitarias de ControladorLogin.
			T6-HU-03: Integrar con la interfaz de autenticación implementada en la iteración 1.
			T7-HU-03: Diseño de mensajes informativos.
			T8-HU-03: Pruebas de aceptación.

Realizado por: Tupac Tisalema, 2019

HU-04. Como administrador deseo visualizar una interfaz de bienvenida al acceder al sistema.

Tabla 12-2. Tablero de Tareas HU-04

H.U.	Pendiente	En Proceso	Terminadas
Como administrador deseo visualizar una interfaz de bienvenida al acceder al sistema.			T1-HU-04: Diseño de la interfaz de bienvenida mediante hojas de estilo CSS.
			T8-HU-04: Pruebas de aceptación.

Realizado por: Tupac Tisalema, 2019

HU-05. Como secretario necesito registrar los datos de los usuarios o clientes de la Junta.

Tabla 13-2. Tablero de Tareas HU-05

H.U.	Pendiente	En Proceso	Terminadas
Como secretario necesito registrar los datos de los usuarios o clientes de la Junta.			T1-HU-05: Creación de la tabla cliente, persona, persona natural, persona jurídica y demás tablas relacionadas a la misma.
			T2-HU-05: Creación de la clase mPersona con los métodos pertinentes.
			T3-HU-05: Proceso pruebas unitarias para mPersona.
			T4-HU-03: Creación del controlador ControladorPersona, con sus atributos y métodos necesarios
			T5-HU-05: Proceso pruebas unitarias para ControladorPersona.
			T6-HU-05: Diseño de la interfaz para ingreso de datos del cliente.
			T7-HU-05: Creación de métodos auxiliares para validaciones de datos a ingresar.
			T5-HU-05: Proceso pruebas unitarias para métodos auxiliares de validación.
		T8-HU-05: Pruebas de aceptación.	

Realizado por: Tupac Tisalema, 2019

2.6.4. Ejecución TDD

Para evidenciar el uso de TDD en el desarrollo del presente proyecto, a continuación se expone una prueba unitaria y prueba de aceptación del procesos o funcionalidad de registrar lecturas de medidor.

2.6.4.1. Plan de Pruebas

Para realizar el plan de pruebas se emplea la herramienta TestLink, con el cuál se crea un nuevo proyecto de pruebas denominado “Plan de pruebas del sistema SOLRIF”, se crea un plan con el nombre SOLRIF, se ingresa los requerimientos funcionales, se ingresa los casos de prueba correspondiente a cada requerimiento y se ejecutan las pruebas. A continuación en las siguientes ilustraciones se expone lo descrito.

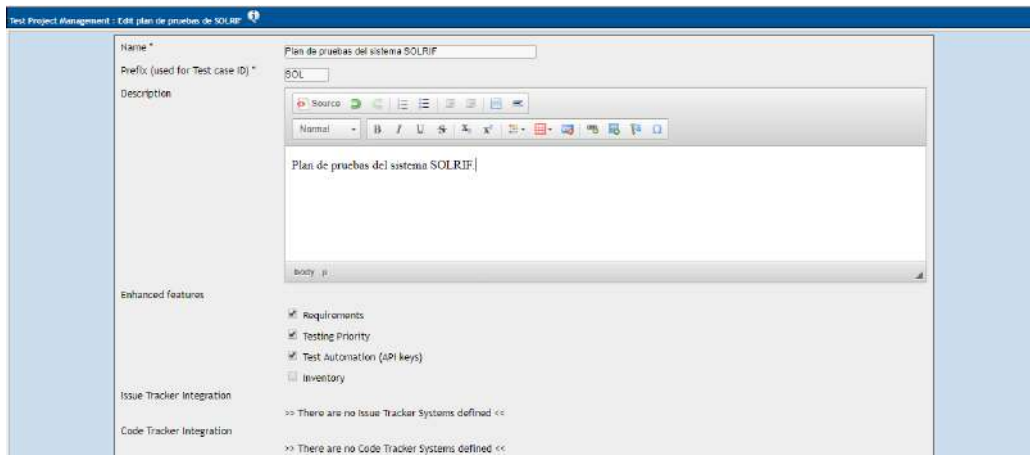


Figura 8-2: Plan de pruebas del sistema SOLRIF

Realizado por: Tupac Tisalema, 2019

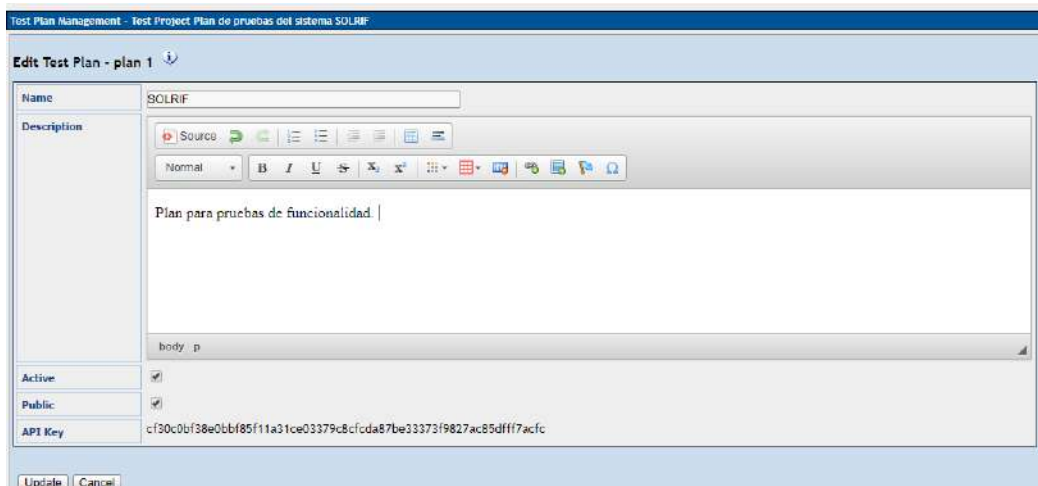


Figura 9-2: Plan para pruebas de funcionalidad.

Realizado por: Tupac Tisalema, 2019

Se ingresa los requerimientos sobre los cuales se va ejecutar los casos de prueba, como se puede observar en la Figura 10-2.

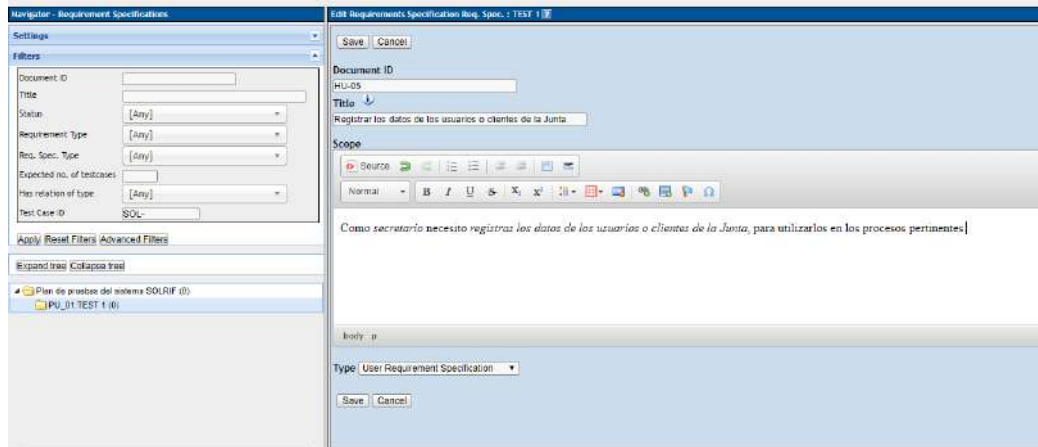


Figura 10-2: Ingreso de requerimiento para plan de prueba.
Realizado por: Tupac Tisalema, 2019

A continuación en la Tabla 14-2 se especifica uno de los caso de prueba que se registran en el plan de pruebas.

Tabla 14-2. Caso de prueba PA_05

< Verificar que se registre correctamente los datos del cliente.>	
Resumen:	
<ul style="list-style-type: none"> Verificar que se registre correctamente los datos del cliente y se emita un mensaje informando que se han guardado correctamente. 	
Pasos:	
<ol style="list-style-type: none"> Ubicarse en el menú contribuyente. Desglosar el menú contribuyente. Seleccionar el sub-ítem Persona. Seleccionar el ítem registrar. Seleccionar el tipo de identificación. Ingresar el número del documento de identificación. Pulsar el botón Procesar Ingresar los datos solicitados. Pulsar el botón guardar. 	
Resultados Esperados:	
<ul style="list-style-type: none"> Datos guardados correctamente y emisión de mensaje informativo. 	
Estado de evolución de la prueba:	
Para verificar defectos Mantis:	

Realizado por: Tupac Tisalema, 2019

A continuación en la figura 11-2, se puede observar cómo se ingresa el caso de prueba que se indica en la tabla anterior.

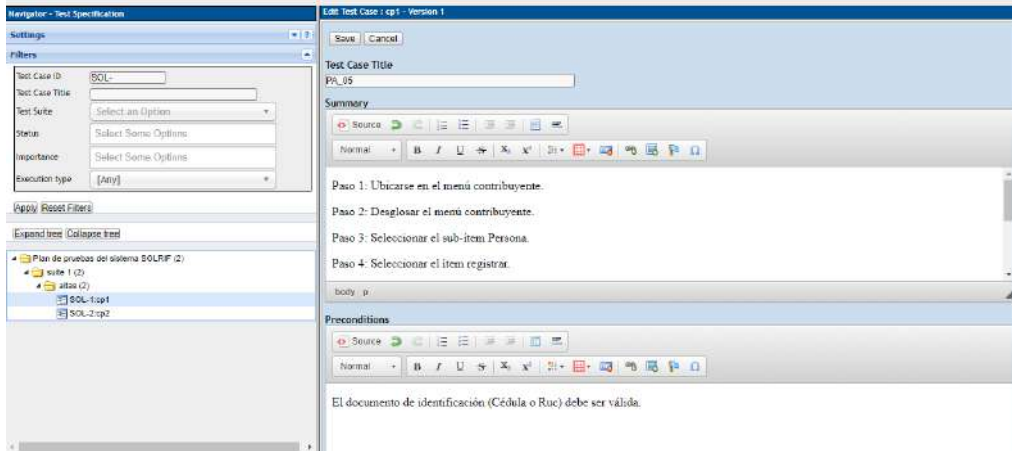


Figura 11-2: Caso de prueba PA_05.

Realizado por: Tupac Tisalema, 2019

Una vez ingresado los casos de prueba, procedemos a ejecutar las pruebas. A continuación se presenta la figura 12-2

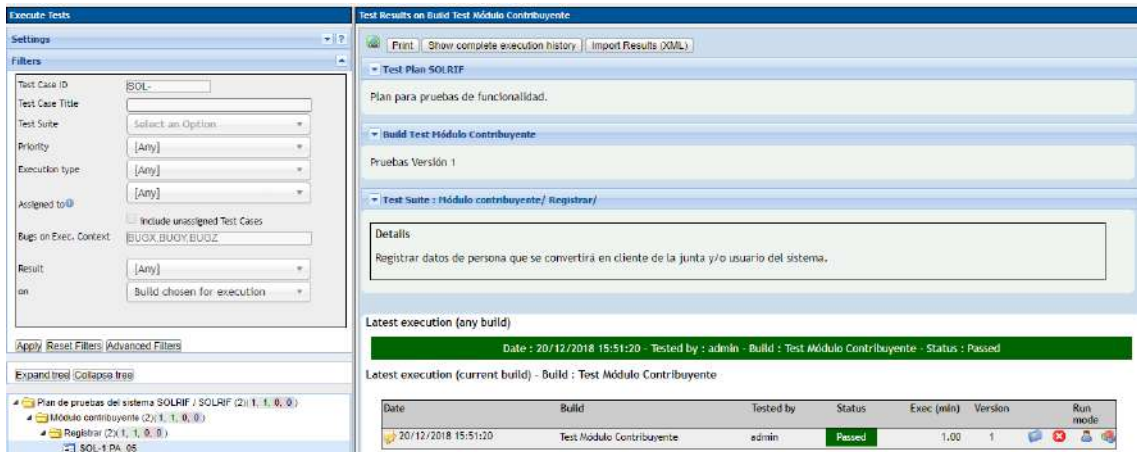


Figura 12-2: Ejecución caso de prueba

Realizado por: Tupac Tisalema, 2019

Una vez concluido el ingreso de requerimientos, ingreso de casos de prueba y ejecución de todos los casos, se puede proceder con la generación del informe de plan de pruebas.

Para visualizar el Plan de Pruebas generado con TestLink, revise en el **Anexo G**.

2.6.4.2. Pruebas Unitarias

Se implementa un código con el propósito de validar una parte de la funcionalidad de registrar lectura de medidor y obtener un código legible del proceso en mención, para lo cual se verifica el correcto funcionamiento del método *insertar* en la clase *mLectura*.

Escribir las especificaciones

Verificar que el método insertar de la clase mLectura retorne true, cuando se haya insertado correctamente los datos.

Clase: mLectura

Método: insertar

Agregar prueba

```
@Before
public void setUp() {
    lectura = new cLectura();
    lectura.setLecturaAnterior((short) 0);
    lectura.setLecturaActual((short) 30);
    lectura.setLecturaConsumo((short) (lectura.getLecturaActual() - lectura.getLecturaAnterior()));
    lectura.setLecturaFechaRegistro(new Timestamp((new Date()).getTime()));
    lectura.setLecturaFechaCorte(new Date());
    lectura.setIdClienteServicio(297); //Servicio Contratado Agua Potable
    lectura.setIdUsuarioRegistra(1);
    lectura.setObjEstadoLectura(new cEstadoLectura((short) 1));
    periodo = 1;
}

@After
public void tearDown() {
    mLectura.eliminar(lectura.getLecturaId());
}

@Test
public void testLecturaInsertar() throws Exception {
    System.out.println("lecturaInsertar");
    cLectura objLectura = lectura;
    long periodoid = periodo;
    boolean expectedResult = true;
    boolean result = mLectura.insertar(objLectura, periodoid);
    assertEquals(expectedResult, result);
    // TODO review the generated test code and remove the default call to fail.
    //fail("The test case is a prototype.");
}
```

Figura 13-2: Código Test

Realizado por: Tupac Tisalema, 2019

Ejecutar prueba y verificar que falle

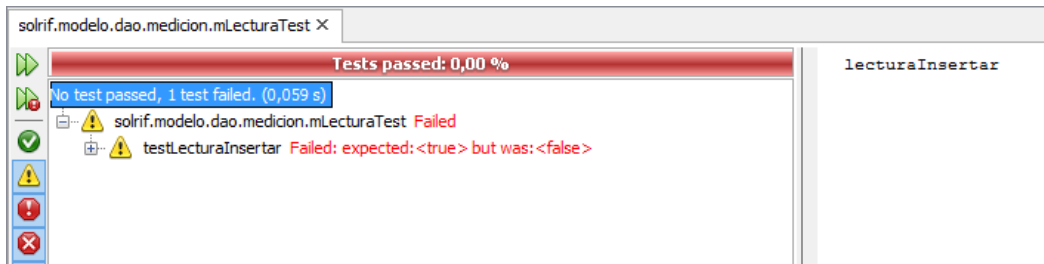


Figura 14-2: Ejecución Test Fallida

Realizado por: Tupac Tisalema, 2019

Implementar el código que hace funcionar el ejemplo

Escribir código mínimo y suficiente para hacer que la prueba pase

```
public static boolean insertar(cLectura objLectura, long periodoid) throws Exception {
    boolean resp = false;
    String sql;

    sql = "SELECT *FROM medicion.fn_insert_lectura(?,?,?,?,?,?,?,?,?,?)";
    ArrayList<Parametro> lstParam = new ArrayList<>();
    lstParam.add(new Parametro(1, objLectura.getLecturaAnterior()));
    lstParam.add(new Parametro(2, objLectura.getLecturaActual()));
    lstParam.add(new Parametro(3, objLectura.getLecturaConsumo()));
    lstParam.add(new Parametro(4, new Timestamp(objLectura.getLecturaFechaCorte().getTime())));
    lstParam.add(new Parametro(5, objLectura.getLecturaFechaRegistro()));
    lstParam.add(new Parametro(6, objLectura.getIdClienteServicio()));
    lstParam.add(new Parametro(7, objLectura.getIdUsuarioRegistra()));
    lstParam.add(new Parametro(8, objLectura.getIdImagen()));
    lstParam.add(new Parametro(9, objLectura.getLecturaObservacion()));
    lstParam.add(new Parametro(10, objLectura.getObjEstadoLectura().getEstadoLecturaId()));
    lstParam.add(new Parametro(11, periodoid));
    ConjuntoResultado rs = AccesoDatos.ejecutarQuery(sql, lstParam);
    while (rs.next()) {
        resp = rs.getBoolean(0);
    }

    return resp;
}
```

Figura 15-2: Código a probar

Realizado por: Tupac Tisalema, 2019

Ejecutar prueba y verificar que pase

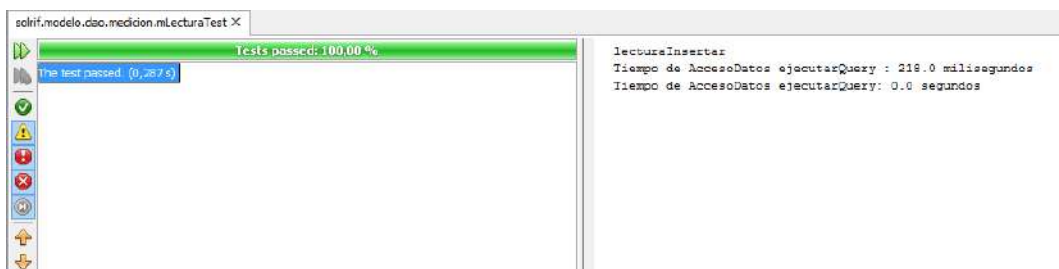


Figura 16-2: Ejecución Test Exitosa

Realizado por: Tupac Tisalema, 2019

Refactorizar

Eliminar duplicidad y hacer mejoras

```
public static boolean insertar(cLectura objLectura, long periodoid) throws Exception {
    boolean resp = false;
    String sql;
    try {
        sql = "SELECT *FROM medicion.fn_insert_lectura(?,?,?,?,?,?,?,?,?)";
        ArrayList<Parametro> lstParam = new ArrayList<>();
        lstParam.add(new Parametro(1, objLectura.getLecturaAnterior()));
        lstParam.add(new Parametro(2, objLectura.getLecturaActual()));
        lstParam.add(new Parametro(3, objLectura.getLecturaConsumo()));
        lstParam.add(new Parametro(4, new Timestamp(objLectura.getLecturaFechaCorte().getTime())));
        lstParam.add(new Parametro(5, objLectura.getLecturaFechaRegistro()));
        lstParam.add(new Parametro(6, objLectura.getIdClienteServicio()));
        lstParam.add(new Parametro(7, objLectura.getIdUsuarioRegistra()));
        lstParam.add(new Parametro(8, objLectura.getIdImagen()));
        lstParam.add(new Parametro(9, objLectura.getLecturaObservacion()));
        lstParam.add(new Parametro(10, objLectura.getObjEstadoLectura().getEstadoLecturaId()));
        lstParam.add(new Parametro(11, periodoid));
        ConjuntoResultado rs = AccesoDatos.ejecutarQuery(sql, lstParam);
        while (rs.next()) {
            resp = rs.getBoolean(0);
        }
    } catch (Exception e) {
        throw e;
    }
    return resp;
}
```

Figura 17-2: Refactorización

Realizado por: Tupac Tisalema, 2019

Ejecutar prueba y verificar que pase

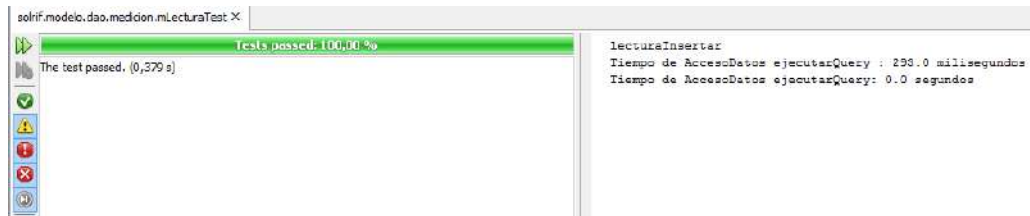


Figura 18-2: Ejecución Test Exitoso

Realizado por: Tupac Tisalema, 2019

Persistencia en la base de datos

Cuando la prueba se cumple satisfactoriamente, se ejecuta el método `@After tearDown`, que se encarga de controlar la persistencia de datos durante la ejecución de una prueba unitaria; en este caso particular en este método se define la eliminación de los datos empleados durante el test.

```
@After
public void tearDown() {
    mLectura.eliminar(lectura.getLecturaId());
}
```

Figura 19-2: Método `@After`

Realizado por: Tupac Tisalema, 2019

Consistencia en la base de datos

Para controlar la consistencia de datos se utiliza el método @Before setUp, en la cual se define los datos a emplearse para la ejecución de una prueba unitaria, según el tipo, formato y longitud especificado en la base de datos

```
@Before
public void setUp() {
    lectura = new cLectura();
    lectura.setLecturaAnterior((short) 0);
    lectura.setLecturaActual((short) 30);
    lectura.setLecturaConsumo((short) (lectura.getLecturaActual() - lectura.getLecturaAnterior()));
    lectura.setLecturaFechaRegistro(new Timestamp((new Date()).getTime()));
    lectura.setLecturaFechaCorte(new Date());
    lectura.setIdClienteServicio(297); //Servicio Contratado Agua Potable
    lectura.setIdUsuarioRegistra(1);
    lectura.setObjEstadoLectura(new cEstadoLectura((short) 1));
    periodo = 1;
}
```

Figura 20-2: Método @Before

Realizado por: Tupac Tisalema, 2019

Este proceso se repite hasta quedar satisfecho con todas las pruebas de la funcionalidad. Para visualizar algunas de las pruebas unitarias revise en el **Anexo_H**.

2.6.4.3. Pruebas de Aceptación

En la **Tabla 14-2**, se aprecia una prueba de aceptación de la historia de usuario HU-07, conformado por el identificador de la prueba, el responsable de la prueba, la fecha de la prueba, pasos para ejecutar la prueba, las condiciones de ejecución, y el resultado de la prueba de aceptación, que en este caso particular es satisfactorio.

Tabla 15-2: Prueba de Aceptación

Prueba de Aceptación	
Código: PA_07	Historia de Usuario: HU-07. Como plomero necesito registrar las lecturas del medidor desde el teléfono móvil.
Nombre: Verificar registro de lecturas de medidor desde el teléfono móvil.	
Responsable: Tupac Tisalema	Fecha: 27/10/2017
Descripción: Verificar que cuando se registre correctamente la lectura del medidor desde el teléfono móvil, se emita un mensaje informativo indicando que los datos se guardaron correctamente.	
Condiciones de ejecución:	
<ul style="list-style-type: none">• Usuario debe tener asignado rol de plomero.• Existen números medidores asignado a cada usuario.	
Pasos de ejecución:	

<ol style="list-style-type: none"> 1. Ejecutar la aplicación móvil. 2. Ingresar credenciales e iniciar sesión. 3. Buscar cliente por número de medidor. 4. Introducir datos solicitados. 5. Pulsar el botón guardar.
<p>Resultado esperado: Se emite un mensaje informativo indicando que los datos se registraron correctamente.</p>
<p>Evaluación de la prueba: Satisfactorio.</p>

Realizado por: Tupac Tisalema. 2019

Para revisar más pruebas aceptación, diríjase al **Anexo I**.

2.6.5. Implantación

Finalizada la etapa de pruebas y posterior aceptación de la aplicación web/móvil se procede a la implantación del sistema en el servidor de la Junta Administrador de Agua Potable Angahuana Alto, para lo que se requiere que el hardware cumpla con ciertas características necesarias para el óptimo desempeño del sistema implementado.

2.6.5.1. Descripción de Recursos Hardware

Servidor HPE ML30 Gen9 E3-1220v5-MCA Svr/S/Buy

- Intel Xeon E3-1220v5 Quad Core (3.00GHz 8MB)
- 4GB (1 x 4GB) DDR4 2133MHz UDIMM
- Integrated Matrox G200eH2 / HP Embedded 1GbE Dual Port 332i
- HDD 1TB (7.2k rpm) Non-Hot Plug 3.5in SATA (4LFF)
- Dynamic Smart Array B140i controller Soporta RAID 0/1/1+0/5 (sólo SATA)

CAPÍTULO III

3. MARCO DE RESULTADOS

3.1. Gestión del Proyecto

SCRUM considera parte esencial el monitorear los avances del proyecto, pudiendo apreciar la velocidad del proyecto y analizando la eficiencia aplicada; para ello se utiliza la herramienta BurnDown Chart, en el grafico expuesto se contrasta un desarrollo ideal planificado y a su vez un desarrollo hiperrealista denotando las horas usadas en el desarrollo de las tareas. Los tiempos reales y tiempos estimados lo denotamos con líneas de diferentes colores para demostrar la línea de tiempo ideal (línea azul) y la línea de tiempo estimada (línea tomate).

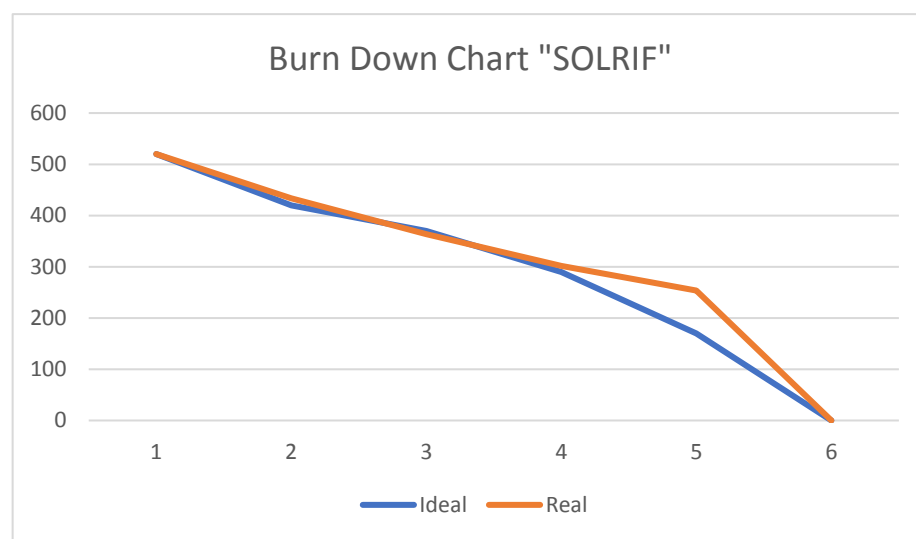


Gráfico 1-3: BurnDown Chart

Realizado por: Tupac Tisalema, 2019

Al analizar el gráfico se observa que los 4 primero Sprint se cumplieron a la par con la planificación presentada, pero, a partir de Sprint 4 hasta el 6 la línea de tiempo se extendió teniendo que llevar a horarios fuera de los tiempos estimados, esto se debe a las implementación y ejecución de pruebas unitarias durante el desarrollo de los métodos.

3.2. Evaluación del tiempo de respuesta

Dado que se planteado como uno de los objetivos a cumplir, el evaluar el tiempo involucrado en el registro de lectura de medidor por el consumo de agua con el sistema informático SOLRIF con relación al proceso manual, para lo cual se realizan los análisis pertinentes.

3.2.6. Mejora de procesos

Proceso registro lecturas de medidor

Para determinar si existe una diferencia de tiempo en el registro de lecturas de medidor por el consumo de agua con respecto al proceso manual, se parte de obtener una muestra, los mismos que serán analizados estadísticamente para refutar o aceptar lo planteado.

3.2.6.1. Tamaño de muestra

Dado que lleva demasiado tiempo tomar las lecturas de medidor de todos los usuarios o clientes; para calcular la muestra, se trabaja con una población infinita. Se utiliza la siguiente ecuación:

$$n = \frac{N * Z^2 * P * Q}{e^2}$$

Donde:

- P= 0.5 probabilidad de incidencia (porcentaje de población que tiene el atributo deseado)
- Q = 0.5 (1-P) porcentaje de población que no tiene el atributo deseado.
- Z = 1.65 (correspondiente a un nivel de confianza de 90%)
- e = 0.15 -->15% (error de estimación máximo aceptado)

Reemplazando se obtiene: 412.0116/4.4272

$$n = \frac{(429) * (1.96)^2 * 0.5 * 0.5}{0.15^2} \qquad n = 30.25$$

Según el resultado obtenido es necesario evaluar el proceso con 30 usuarios; se emplea un cronómetro para medir el tiempo involucrado tanto en el proceso manual y automatizado. El tiempo manual se considera el tiempo que emplea en registrar la lectura, más el tiempo que tarda en transcribir esos registro al documento Excel.

3.2.6.2. Recolección de datos

Para determinar si el uso de la aplicación móvil optimiza tiempo en el proceso de registrar lecturas de medidor, es necesario recolectar datos para realizar el análisis correspondiente. Utilizando un cronómetro se toma tiempos en segundos, del proceso manual y del proceso con la aplicación, realizado por el plomero. A continuación en la **Tabla 1-3** se presenta los valores obtenidos.

Tabla 1-3: Medición de tiempo en el registro de lecturas de medidor

No.	Cliente	No. Medidor	Tiempo SOLRIF	Tiempo manual		
			Segundos	Hoja Registro	Transcripción a Excel	Total
				Segundos		
1	POMAQUIZA SEGUNDO	001	150	135	160	295
2	CHALUIS MARIA	002	170	133	167	300
3	POAQUIZA EDUARDO	003	160	130	164	294
4	MASABAND HAIDA	004	155	130	170	300
5	MASABAND EUSEVIO	005	175	140	168	308
6	XISALEMA MARIA	006	175	135	165	300
7	TISALEMA LUCIANO	007	160	140	162	302
8	CHAULUIS MARIA	008	174	150	180	330
9	AGUALON SEGUNDO	009	165	165	177	342
10	MALLQUI SEGUNDO	010	174	179	193	372
11	CHALUIS SEGUNDO	011	168	165	225	390
12	TOCALEMA JOSE	012	158	151	143	294
13	AGUALON MARIA	013	175	138	162	300
14	AGUALONG ELENA	014	155	160	170	330
15	CHAULUIS SEGUNDO	015	156	143	145	288
16	CAPUZ MARIA	016	145	170	190	360
17	TISALEMA MANUEL	017	165	140	160	300
18	TOCALEMA MARIA	018	152	165	187	352
19	POMAQUIZA SEGUNDO	019	164	135	150	285
20	MASABAND TOMAS	020	170	133	167	300
21	TISALEMA MARIA	021	160	175	215	390
22	GUAPISACA SEGUNDO	022	145	136	164	300
23	BALTASAR JOSE	023	153	165	185	350
24	AGUALONG SIMON	024	170	158	172	330
25	POMAQUIZA SEGUNDO	025	175	143	189	332
26	CAPUZ MARIA	026	165	130	168	298
27	YANSAHUAN MARIA	027	168	137	168	305
28	TOCALEMA MARCOS	028	157	155	175	330

29	TISALEMA MARIA	029	150	165	180	345
30	CAPUZ MARIA	030	160	175	215	390

Realizado por: Tupac Tisalema. 2019

3.2.6.3. *Shapiro – Wilk Contraste de normalidad*

Para continuar, es necesario comprobar si los datos obtenidos proceden de una población con distribución normal, para realizar el análisis mediante la estadística paramétrica o no paramétrica, para este propósito se utiliza la prueba de contraste de normalidad shapiro-wilk, que se puede aplicar siempre y cuando el tamaño de la población se a menor a 50.

Aplicando contraste de normalidad, se obtiene el estadístico de contraste W igual a 0. 94573 y la probabilidad de contraste p igual a 0. 1298, siendo superior al nivel de significancia 0.05 se asume que la distribución de la muestra es normal; se puede aplicar la estadística paramétrica. **Anexo_J**

3.2.6.4. *Análisis de datos – T-student*

Es importante responder, ¿Existe una diferencia significativa entre los tiempos manuales y el sistema SOLRIF en el registro del consumo de agua?, se utiliza T-Student, con un nivel de significancia del 1%, para ello se plantea las siguientes hipótesis:

- **H₀** = No hay disminución en los tiempos de registro de la lectura del consumo de agua al usar el sistema.
- **H₁** = Hay disminución en los tiempos de registro de la lectura del consumo de agua al usar el sistema.

Se aplica la prueba de dos colas.

$$H_0 = \mu_D = 0$$

$$H_1 = \mu_D \neq 0$$

$$\alpha = 1\%$$

Tamaño de muestra = **n** = 30

$$\alpha/2 = 0.01/2 = 0.005$$

$$\text{Grados de libertad} = \mathbf{v} = 30 - 1 = 29$$

$$t_{\alpha/2} = t_{0.005} = 2.7564;$$

Región crítica

- Área de aceptación de la hipótesis nula = entre los valores [-2.7564; 2.7564]
- Área de rechazo de la hipótesis nula = a la izquierda de -2.7564 y a la derecha de 2.7564

Cálculo de diferencia tiempo SOLRIF – manual

Se calcula la media y la desviación estándar de las diferencias calculadas de la **Tabla 1-3**, como se muestra en la **Tabla 2-3**.

Tabla 2-3: Diferencia tiempo SOLRIF - manual

Tiempo SOLRIF	Tiempo manual	Diferencia
Segundos		
150	295	145
170	300	130
160	294	134
155	300	145
175	308	133
175	300	125
160	302	142
174	330	156
165	342	177
174	365	191
168	345	177
158	294	136
175	300	125
155	330	175
156	288	132
145	360	215
165	300	135
152	352	200
164	285	121
170	300	130
160	360	200
145	300	155
153	350	197
170	330	160
175	332	157
165	298	133

168	305	137
157	330	173
150	345	195
160	355	195
Media		157,5333333
Desviación estándar		28,18502089

Realizado por: Tupac Tisalema. 2019

Estadístico de contraste prueba t-Student:

$$t = \frac{\bar{x} - \mu_0}{s/\sqrt{n}}$$

Donde:

\bar{x} = media muestral

S = desviación estándar muestral.

n: = tamaño de la muestra.

$\mu_0 = 0$

Reemplazando valores:

$$t = \frac{157,53 - 0}{28,18/\sqrt{30}} = \frac{157,53}{5,15} = 30,61$$

El valor t calculado es 30.61, siendo este mayor al valor crítico t de la tabla que es 2.7564 por lo cual se acepta la hipótesis alterna dado que se determina que cae en la región de rechazo. Una vez que se determina que si existe una diferencia significativa entre los tiempos, se procede a determinar el porcentaje de disminución del tiempo, como se muestra en el **Gráfico 2-3**.

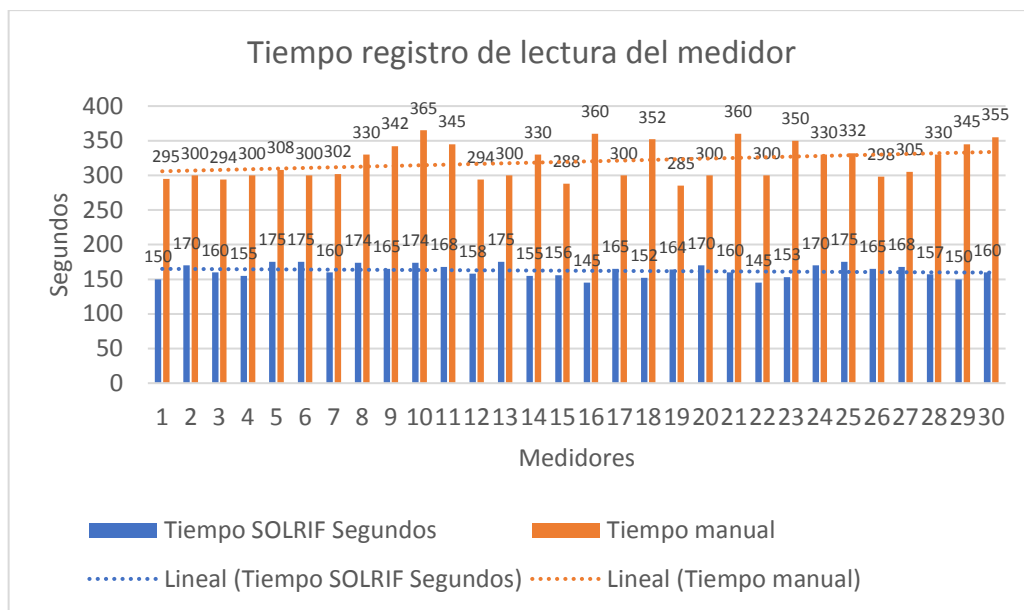


Gráfico 2-3: Tiempo sistema vs tiempo manual

Realizado por: Tupac Tisalema, 2019

La línea de color celeste indica el tiempo en segundos que toma en registrar las lecturas de medidor con el sistema desarrollado y la de color naranja el tiempo involucrado con el proceso manual, para cuantificar se obtiene el promedio del tiempo del sistema y manual, siendo estos valores 162.3 s. y 319.83 s. respectivamente. Se aplica una regla de tres y como resultado se obtiene que con el sistema solo se requiere un 50.74% del tiempo original que es el 100% y reduciendo así en un 49.25% el tiempo utilizado para registrar las lecturas de medidor.

3.3. Evaluación de características de Calidad - Seguridad

Como se expone en el Capítulo I, en el apartado de la **Norma de evaluación ISO/IEC 25000**, Actualización ISO/IEC 25000:2014 (Rojo, 2015), el modelo de calidad del producto software que se planteaba anteriormente solo constaba de 6 métricas de evaluación, hasta ese entonces la seguridad era considerado solo como uno de los parámetros dentro de la funcionalidad.

Con la actualización de la ISO 25000:2014 se agregaron dos métricas más al modelo anterior, las cuales son: Compatibilidad y Seguridad. Por lo tanto teniendo en cuenta estas consideraciones para la presente evaluación se considera a la seguridad como una métrica más, y no como un parámetro o subcaracterística de la funcionalidad.

El propósito de este apartado es evaluar la aplicación web/móvil implementada en la Junta Administradora de Agua Potable “Angahuana Alto” ya en funcionamiento, aplicando la característica de Seguridad, siguiendo los lineamientos de sus parámetros o subcaracterísticas.

3.3.6. Modelo de Evaluación de característica de Calidad - Seguridad

En el presente apartado se define un conjunto de preguntas basadas en el enfoque GQM (*Goal, Question, Metric*) traducido al español meta, pregunta, métrica; método orientado a determinar una métrica que mida cierto objetivo de una manera determinada (Calabrese *et al.*, 2017, p. 662), que luego, mediante conectivos lógicos, indicarán la satisfacción de los objetivos planteados.

A continuación se va estudiar la característica Seguridad, que abarca las subcaracterísticas: Confidencialidad, Integridad, No-Repudio, Responsabilidad y Autenticidad.

Tabla 3-3: Subcaracterísticas de Seguridad

ID	SUBCARACTERÍSTICAS	DESCRIPCIÓN
C	CONFIDENCIALIDAD	Evalúa la capacidad de protección contra el acceso de datos e información no autorizados, ya sea accidental o deliberadamente.
I	INTEGRIDAD	Evalúa la capacidad del sistema o componente para prevenir accesos o modificaciones no autorizados a datos o programas de ordenador.
NR	NO-REPUDIO	Evalúa la capacidad de demostrar las acciones o eventos que han tenido lugar, de manera que dichas acciones o eventos no puedan ser repudiados posteriormente.
R	RESPONSABILIDAD	Evalúa la capacidad de rastrear de forma inequívoca las acciones de una entidad.
A	AUTENTICIDAD	Evalúa la capacidad de demostrar la identidad de un sujeto o un recurso.

Realizado por: Tupac Tisalema, 2019

Fuente: (Julieta Calabrese, 2017)

3.3.6.1. Lineamientos

En función de las subcaracterísticas descritas anteriormente, se definieron 33 cuestionamientos para verificar si cumple o no con las especificaciones descritas a continuación, respondiendo con verdadero/falso. **Tabla 4-3**

Tabla 4-3: Lineamientos para la característica de Seguridad

ID	PREGUNTA
P1	¿Se requiere que la contraseña posea al menos 8 caracteres?
P2	¿Se requiere que la contraseña posea letras mayúsculas y minúsculas?
P3	¿Se requiere que la contraseña posea números y letras?
P4	¿Se requiere que la contraseña posea caracteres especiales?
P5	¿El sistema utiliza conexión segura mediante HTTPS?
P6	¿La base de datos posee los datos encriptados?

P7	¿El sistema permite acceder a funcionalidades en las cuales no se tiene permiso?
P8	¿El sistema permite que cualquier persona tenga acceso a la base de datos?
P9	¿El sistema permite que cualquier persona tenga acceso al código del servidor de la aplicación?
P10	¿Cualquier persona tiene acceso al servidor físico?
P11	¿Cualquier persona tiene acceso al servidor remoto?
P12	¿El sistema posee re direccionamientos hacia sitios no seguros?
P13	¿El sistema solicita una confirmación de registro mediante un mail a la hora de registrarse?
P14	¿El sistema permite que cualquier persona pueda modificar la base de datos?
P15	¿El sistema permite que cualquier persona pueda modificar el código del servidor de la aplicación?
P16	¿El sistema permite inyecciones SQL?
P17	¿El sistema posee un historial de acciones realizadas?
P18	¿El sistema posee algoritmos de cifrado de datos?
P19	¿El sistema posee un mecanismo criptográfico, como firma digital?
P20	¿El sistema realiza eliminación lógica al dar de baja algún registro?
P21	¿El sistema posee una protección con certificados SSL?
P22	¿El sistema da aviso cuando se accede desde una ubicación desconocida?
P23	¿El sistema informa vía mail las operaciones realizadas?
P24	¿El sistema guarda fecha, hora, y usuario que almacena algún registro?
P25	¿El sistema registra el tipo de navegador y sistema de operación utilizado para ingresar al sitio?
P26	¿El sistema registra la dirección IP desde la cual se ingresa al sitio?
P27	¿El sistema realiza una comprobación de identidad mediante un certificado digital?
P28	¿El sistema posee un sistema de verificación en dos pasos?
P29	¿Es requerida una clave de segundo nivel para el ingreso al sistema?
P30	¿El sistema realiza una comprobación de identidad mediante datos biométricos?
P31	¿El sistema realiza una comprobación de identidad mediante tarjeta de coordenadas?
P32	¿El sistema realiza una comprobación de identidad mediante credenciales?
P33	¿El sistema realiza una comprobación de identidad mediante una firma electrónica?

Realizado por: Tupac Tisalema. 2019

Fuente: (Julieta Calabrese, 2017)

3.3.6.2. Descripción de criterios de evaluación (CE)

Para alcanzar el objetivo, las respuestas de los lineamientos especificados se combinaron de manera lógica determinando un puntaje a cada uno de los CE.

Tabla 5-3: Descripción de criterios de evaluación (CE)

ID	Nombre	Descripción	Fórmula	Ptos
C-1	Conexiones Seguras	Una conexión se considera segura si se utiliza HTTPS y si no se tienen redireccionamientos hacia sitios no seguros	$P5 \ \& \ \sim P12 = V$	1
C-2	Control de Acceso	Se debe controlar que no se permita acceder a funcionalidades sin autorización, tampoco a la base de datos, al código de la aplicación ni a los servidores, físico ni remoto	$P7 \ \ P8 \ \ P9 \ \ P10 \ \ P11 = F$	1
C-3	Encriptación de datos	Los datos de la base de datos deben estar encriptados	$P6 = V$	1
C-4	Contraseña de bajo nivel	La contraseña se considera de bajo nivel si posee menos de 8 caracteres, no posee letras mayúsculas y minúsculas, no posee letras y números y no posee caracteres especiales	$P1 \ \ P2 \ \ P3 \ \ P4 = F$	0
	Contraseña de medio nivel	La contraseña se considera de medio nivel si posee al menos 8 caracteres o letras mayúsculas y minúsculas o letras y números o	$P1 \ \ P2 \ \ P3 \ \ P4 = V$	0.5
	Contraseña de alto nivel	La contraseña se considera de alto nivel si posee al menos 8 caracteres, letras mayúsculas y minúsculas, letras y números y caracteres especiales	$P1 \ \& \ P2 \ \& \ P3 \ \& \ P4 = V$	1
I-5	Prevención de Accesos	Se debe prevenir que no se permita acceder a funcionalidades sin autorización, tampoco a la base de datos ni al código de la aplicación, y que no se permitan inyecciones SQL	$P7 \ \ P8 \ \ P9 \ \ P16 = F$	1
I-6	Prevención de modificaciones	Se debe prevenir que no se permita modificar datos de la base de datos ni modificar el código de la aplicación sin autorización	$P14 \ \ P15 = F$	1
I-7	Confirmación de datos	Se debe realizar una confirmación de registro por mail	$P13 = V$	1
NR-8	Operaciones Realizadas	Se debe poseer un historial de acciones realizadas o las mismas deben ser enviadas por mail	$P17 \ \ P23 = V$	1
NR-9	Mecanismos de cifrado	Se debe poseer un algoritmo de cifrado de datos o un mecanismo criptográfico, como firma digital, o una protección con certificados SSL	$P18 \ \ P19 \ \ P21 = V$	1

NR-10	Confirmación de acciones	Se debe solicitar una confirmación al realizar una determinada acción	P20 = V	1
NR-11	Registro de Ubicación	Se debe informar si se accedió al sistema desde una ubicación desconocida	P22 = V	1
R-12	Registro de acciones y datos	Se debe poseer un historial de acciones realizadas, o un registro de fecha y hora de ingreso al sistema o de la dirección IP desde la cual se ingresa o del tipo de navegador y sistema de operación utilizado	P17 P24 P25 P26 = V	1
R-13	Control de Ubicación	Se debe dar aviso cuando se accede al sistema desde una ubicación desconocida	P22 = V	1
A-14	Comprobación de identidad	El sistema debe realizar una comprobación de identidad mediante alguno de los siguientes métodos: datos biométricos, tarjeta de coordenadas, credenciales, firma electrónica o certificado digital	P27 P30 P31 P32 P33 = V	1
A-15	Comprobación es adicionales	Se debe poseer un sistema de verificación en dos pasos, o se debe requerir una clave de segundo nivel para el ingreso al sistema o una confirmación de registro mediante un mail	P28 P29 P13 = V	1

Realizado por: Tupac Tisalema, 2019

Fuente: (Julieta Calabrese, 2017)

3.3.6.3. Métricas para cada subcaracterística

Se combinaron los CE para determinar las métricas que satisfacen los objetivos de las subcaracterísticas. Para cada una se definió un nombre, un propósito, un método de aplicación, valores de entradas y formula aplicada. Para mejor comprensión se puede observar la **Tabla 6-3**, que se expone a continuación. Revisar **Anexo_K**.

Tabla 6-3: Métricas para cada subcaracterística

Métrica	Propósito	Método de aplicación	Entradas	Fórmula	Observaciones
Confidencialidad	¿Cuán eficiente es el sistema a la hora de proteger el acceso de datos e información no autorizados, ya sea accidental o deliberadamente?	Contestar las preguntas de los CE correspondientes a la subcaracterística "Confidencialidad" y calcular la puntuación obtenida, sumando los puntajes de los CE referenciados que cumplan con la meta esperada. "Puntaje total" hace referencia al máximo puntaje que se puede obtener.	A = Puntaje obtenido B = Puntaje total	X = A/B	CE a utilizar son: C-1, C-2, C-3 y C-4.
Integridad	¿Cuán capaz es el sistema a la hora de prevenir accesos o modificaciones no autorizados a datos o programas de ordenador?	Contestar las preguntas de los CE correspondientes a la subcaracterística "Integridad" y calcular la puntuación obtenida, sumando los puntajes de los CE referenciados que cumplan con la meta esperada. "Puntaje total" hace referencia al máximo puntaje que se puede obtener.			CE a utilizar son: I-5, I-6 e I-7.
No-Repudio	¿Cuán capaz es el sistema de demostrar las acciones o eventos que han tenido lugar, de manera que dichas acciones o eventos no puedan ser repudiados posteriormente?	Contestar las preguntas de los CE correspondientes a la subcaracterística "No-Repudio" y calcular la puntuación obtenida, sumando los puntajes de los CE referenciados que cumplan con la meta esperada. "Puntaje total" hace referencia al máximo puntaje que se puede obtener.			CE a utilizar son: NR-8, NR-9, NR-10 y NR-11.
Responsabilidad	¿Cuán capaz es el sistema de rastrear de forma inequívoca las acciones de una entidad?	Contestar las preguntas de los CE correspondientes a la subcaracterística "Responsabilidad" y calcular la puntuación obtenida, sumando los puntajes de los CE referenciados que cumplan con la meta esperada. "Puntaje total" hace referencia al máximo puntaje que se puede obtener.			CE a utilizar son: R-12 y R-13.
Autenticidad	¿Cuán capaz es el sistema de demostrar la identidad de un sujeto o un recurso?	Contestar las preguntas de los CE correspondientes a la subcaracterística "Autenticidad" y calcular la puntuación obtenida, sumando los puntajes de los CE referenciados que cumplan con la meta esperada. "Puntaje total" hace referencia al máximo puntaje que se puede obtener.			CE a utilizar son: A14 y A15.

Realizado por: Tupac Tisalema, 2019

Fuente: (Julieta Calabrese, 2017)

Las fórmulas aplicadas para obtener el valor con el que se cumple cada subcaracterística de Seguridad se presentan en la **Tabla 7-3**.

Tabla 7-3: Fórmula para cada subcaracterística

ID	MÉTRICA	FÓRMULA
C	CONFIDENCIALIDAD	$(C1+C2+C3+C4)/4$
I	INTEGRIDAD	$(I5+I6+I7)/3$
NR	NO REPUDIO	$(NR8+NR9+NR10+NR11)/4$
R	RESPONSABILIDAD	$(R12+R13)/2$
A	AUTENTICIDAD	$(A14+A15)/2$

Realizado por: Tupac Tisalema. 2019

Fuente: (Julieta Calabrese, 2017)

Para obtener el valor con el que se cumple o no se cumple la característica Seguridad se define la fórmula que se exponen en la **Tabla 8-3**.

Tabla 8-3: Fórmula para característica seguridad

ID	CARACTERISTICA	FÓRMULA
S	SEGURIDAD	$(C+I+NR+R+A)/5$

Realizado por: Tupac Tisalema. 2019

3.3.7. Evaluación - Seguridad

Con intención de evaluar la característica en seguridad en la aplicación web/móvil implementada, se realizó el proceso de verificación siguiendo los lineamientos especificados en la ISO/IEC 2540 (Julieta Calabrese, 2017).

3.3.7.1. Establecer los requisitos de la evaluación

El objetivo de la evaluación es medir, la seguridad del sistema web/móvil implementado, considerando distintos aspectos para su análisis. El sistema a evaluar se encuentra en su versión final en ambiente de pruebas, siendo utilizado por los diferentes usuarios de la Junta de Agua Potable.

3.3.7.2. Especificar la evaluación

En base a las definiciones realizadas en el apartado 3.3.6.3 sobre las métricas para las subcaracterísticas, la escala de aceptación se define de la siguiente manera:

Tabla 9-3: Escala de Aceptación

Criterio Aceptación	Rango
Inaceptable	$0 \leq X < 40$
Mínimamente aceptable	$40 \leq X < 60$
Rango objetivo	$60 \leq X < 90$
Excede los requerimientos	$90 \leq X \leq 100$

Realizado por: Tupac Tisalema. 2019

Fuente: (Julieta Calabrese, 2017)

Se acepta el propósito, si todas las subcaracterísticas se hallan entre los rangos mínimamente aceptables y excede los requerimientos (Julieta Calabrese, 2017).

3.3.7.3. Diseñar la evaluación

Para proceder con la evaluación se solicitó la colaboración de la persona encargada de la administración del sistema, con quién se verifico los lineamientos definidas en el apartado 3.3.6.1, registrando las respuestas a las interrogantes con Verdadero (V) o Falso (F).

Además, se diseñó una plantilla de Excel dispuesta para el ingreso de las respuestas obtenidas, y en la misma se obtuvieron de manera automática los resultados A y B de las métricas.

3.3.7.4. Ejecutar la evaluación

Se realizó la verificación con la colaboración del Administrador de Sistemas de la Junta de Agua Potable, y en base a las respuestas según lo lineamientos establecidos (**Anexo_L**), para cada CE se obtuvieron los siguientes puntajes que se presenta en la **Tabla 10-3**.

Tabla 10-3: Puntajes obtenidos

ID	Nombre	Descripción	Fórmula	Ptos. Obtenidos
C-1	Conexiones Seguras	Una conexión se considera segura si se utiliza HTTPS y si no se tienen redireccionamientos hacia sitios no seguros	$P5 \ \& \ \sim P12 = V$	0/1
C-2	Control de Acceso	Se debe controlar que no se permita acceder a funcionalidades sin autorización, tampoco a la base de datos, al código de la aplicación ni a los servidores, físico ni remoto	$P7 \ \ P8 \ \ P9 \ \ P10 \ \ P11 = F$	1/1

C-3	Encriptación de datos	Los datos de la base de datos deben estar encriptados	P6 = V	1/1
C-4	Contraseña de bajo nivel	La contraseña se considera de bajo nivel si posee menos de 8 caracteres, no posee letras mayúsculas y minúsculas, no posee letras y números y no posee caracteres especiales	P1 P2 P3 P4 = F	0/1
	Contraseña de medio nivel	La contraseña se considera de medio nivel si posee al menos 8 caracteres o letras mayúsculas y minúsculas o letras y números o	P1 P2 P3 P4 = V	
	Contraseña de alto nivel	La contraseña se considera de alto nivel si posee al menos 8 caracteres, letras mayúsculas y minúsculas, letras y números y caracteres especiales	P1 & P2 & P3 & P4 = V	
I-5	Prevención de accesos	Se debe prevenir que no se permita acceder a funcionalidades sin autorización, tampoco a la base de datos ni al código de la aplicación, y que no se permitan inyecciones SQL	P7 P8 P9 P16 = F	1/1
I-6	Prevención de modificaciones	Se debe prevenir que no se permita modificar datos de la base de datos ni modificar el código de la aplicación sin autorización	P14 P15 = F	1/1
I-7	Confirmación de datos	Se debe realizar una confirmación de registro por mail	P13 = V	0/1
NR-8	Operaciones realizadas	Se debe poseer un historial de acciones realizadas o las mismas deben ser enviadas por mail	P17 P23 = V	1/1
NR-9	Mecanismos de cifrado	Se debe poseer un algoritmo de cifrado de datos o un mecanismo criptográfico, como firma digital, o una protección con certificados SSL	P18 P19 P21 = V	0/1
NR-10	Confirmación de acciones	Se debe solicitar una confirmación al realizar una determinada acción	P20 = V	1/1
NR-11	Registro de ubicación	Se debe informar si se accedió al sistema desde una ubicación desconocida	P22 = V	0/1
R-12	Registro de acciones y datos	Se debe poseer un historial de acciones realizadas, o un registro de fecha y hora de ingreso al sistema o de la dirección IP desde la cual se ingresa o del tipo de navegador y sistema de operación utilizado	P17 P24 P25 P26 = V	1/1

R-13	Control de ubicación	Se debe dar aviso cuando se accede al sistema desde una ubicación desconocida	P22 = V	0/1
A-14	Comprobación de identidad	El sistema debe realizar una comprobación de identidad mediante alguno de los siguientes métodos: datos biométricos, tarjeta de coordenadas, credenciales, firma electrónica o certificado digital	P27 P30 P31 P32 P33 = V	1/1
A-15	Comprobaciones adicionales	Se debe poseer un sistema de verificación en dos pasos, o se debe requerir una clave de segundo nivel para el ingreso al sistema o una confirmación de registro mediante un mail	P28 P29 P13 = V	0/1

Realizado por: Tupac Tisalema, 2019

Con los puntajes obtenidos se aplica las fórmulas plateadas anteriormente para cada subcaracterística y se realizar los cálculos que se describen a continuación. Los resultados obtenidos se validan y se verifican mediante la plantilla de Excel, que sirve como herramienta de apoyo para automatizar los cálculos. Revisar **Anexo_M**.

C1=1; C2=1; C3=1; C4=0;

$$C = \frac{C1 + C2 + C3 + C4}{4}$$

$$C = \frac{0 + 1 + 1 + 0}{4}$$

$$C = \frac{2}{4}$$

$$C = 0.5$$

I5=1; I6=1; I7=0;

$$I = \frac{I5 + I6 + I7}{3}$$

$$I = \frac{1 + 1 + 0}{3}$$

$$I = \frac{2}{3}$$

$$I = 0.6667$$

NR8=1; NR9=0; NR10=1; NR11=0;

$$NR = \frac{NR8 + NR9 + NR10 + NR11}{4}$$

$$NR = \frac{1 + 0 + 1 + 0}{4}$$

$$NR = \frac{2}{4}$$

$$NR = 0.5$$

R12=1; R13=0;

$$R = \frac{R12 + R13}{2}$$

$$R = \frac{1 + 0}{2}$$

$$R = \frac{1}{2}$$

$$R = 0.5$$

A14=1; A15=0;

$$A = \frac{A14 + A15}{2}$$

$$A = \frac{1 + 0}{2}$$

$$A = \frac{1}{2}$$

$$A = 0.5$$

Al realizar los cálculos necesarios se lograron obtener los siguientes resultados que presentan en la **Tabla 11-3**.

Tabla 11-3: Resultados subcaracterísticas de seguridad

ID	MÉTRICA	RESULTADO	RESULTADO (%)
C	CONFIDENCIALIDAD	0.5	50%
I	INTEGRIDAD	0.6667	66.67%
NR	NO REPUDIO	0.5	50%
R	RESPONSABILIDAD	0.5	50%
A	AUTENTICIDAD	0.5	50%

Realizado por: Tupac Tisalema. 2019

A continuación se describe los cálculos realizados para obtener el valor de la característica seguridad.

$$S = \frac{C + I + NR + R + A}{5}$$

$$S = \frac{0.5 + 0.6667 + 0.5 + 0.5 + 0.5}{5}$$

$$S = \frac{2.6667}{5}$$

$$S = 0.5333$$

Para finalizar, se aplica la fórmula de la característica seguridad y se obtiene el siguiente valor que se muestra en la **Tabla 12-3**.

Tabla 12-3: Resultados característica seguridad

ID	CARACTERÍSTICA	RESULTADO	RESULTADO (%)
S	SEGURIDAD	0.5333	53.33%

Realizado por: Tupac Tisalema. 2019

Para mayores detalles se puede revisar el **Anexo_N**.

3.3.7.5. Finalización de la evaluación

El sistema evaluado tiene las subcaracterísticas Integridad en el rango objetivo, Confidencialidad, No-Repudio, Responsabilidad y Autenticidad en el rango mínimamente aceptable. Para mejor apreciación se expone la **Tabla 10-3**.

Tabla 13-3: Resultados de aceptación subcaracterísticas de seguridad

ID	MÉTRICA	RESULTADO (%)	ACEPTACIÓN
C	CONFIDENCIALIDAD	50%	Mínimamente aceptable
I	INTEGRIDAD	66,67%	Rango objetivo
NR	NO REPUDIO	50%	Mínimamente aceptable
R	RESPONSABILIDAD	50%	Mínimamente aceptable
A	AUTENTICIDAD	50%	Mínimamente aceptable

Realizado por: Tupac Tisalema. 2019

Según estos resultados, se aprecia que la subcaracterística Integridad, satisface considerablemente, mientras que las demás cumplen solo con lo mínimo. En base a estos se obtiene el valor de cumplimiento de la característica Seguridad, que a continuación se puede observar en la **Tabla 14-3**.

Tabla 14-3: Resultados de aceptación característica seguridad

ID	CARACTERÍSTICA	RESULTADO (%)	ACEPTACIÓN
S	SEGURIDAD	53.33%	Mínimamente aceptable

Realizado por: Tupac Tisalema. 2019

3.3.8. Análisis de la característica – Seguridad

Anteriormente, en el punto 3.3.7.2 al especificar la evaluación, se define una escala de aceptación en la cual se determina que la evaluación es aceptable, si todas las subcaracterísticas se encuentran entre los rangos mínimamente aceptables y excede los requerimientos.

Por lo tanto, al finalizar la evaluación y con los resultados obtenidos que se exponen de mejor manera en el **Gráfico 4-3**, a pesar de que las subcaracterísticas Confidencialidad, No-Repudio, Responsabilidad y Autenticidad caen en la escala de aceptación Mínimamente aceptable, se considera que el sistema implementado está dentro del rango de aceptación, determinando de esta manera que si satisface con la característica de seguridad con un 53.33%, en cuanto a la calidad del producto software.

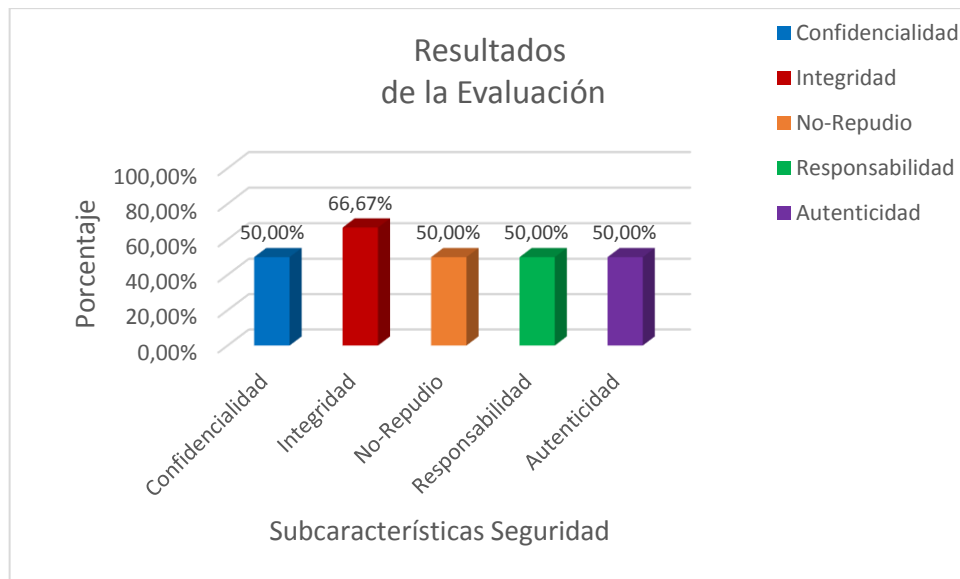


Gráfico 3-3: Subcaracterísticas de seguridad del sistema evaluado

Realizado por: Tupac Tisalema.2018

A pesar de estar dentro del rango de aceptación, el sistema implementado tiene falencias con respecto a la Seguridad, esto se debe a que inicialmente la aplicación fue pensada específicamente para cumplir con los procesos descritos por el cliente; tales como registro de lecturas de medidor y el registro de los respectivos pagos.

CONCLUSIONES

- Se recolectó la información pertinente para conocer las necesidades existentes en relación al proceso de registro de lecturas de medidor y recaudaciones en la Junta Administradora de Agua Potable Angahuana Alto, obteniendo 35 requerimientos que fueron implementadas entre 25 historias de usuario y 9 historias técnicas.
- En el desarrollo del proyecto se estudió las herramientas utilizadas, profundizando en Android y AndroidStudio; prioritariamente se realizó una indagación sobre la aplicación del TDD y la ISO/IEC 25000, las cuales se aplicaron durante la fase de desarrollo y evaluación respectivamente.
- Para el desarrollo del sistema web/móvil se ejecutó la metodología ágil SCRUM, la cual permitió al equipo de trabajo coordinar y organizar adecuadamente las actividades planteadas, llevándolas hacia una correcta implementación de los requerimientos.
- En el desarrollo del sistema SOLRIF, se realizó pruebas sólo de los métodos de mayor relevancia, impacto y alto riesgo en el funcionamiento del sistema, en su gran mayoría métodos de acceso a datos; esto porque debido a la poca experiencia de trabajo con TDD, el tiempo de implementación y ejecución de las pruebas empezaron a exceder considerablemente el tiempo estimado.
- Se analizó el proceso de registro de lectura de medidor mediante la aplicación móvil y se determinó que existe una reducción del 49.25% en relación al tiempo que se emplea en realizar el mismo proceso manualmente, aumentando así la productividad en la institución.
- Según la evaluación de la seguridad realizada con el estándar ISO/IEC 25000, el sistema SOLRIF se encuentra dentro del rango de aceptación, obteniéndose un valor del 53.33%, en cuanto a la calidad el producto referente al criterio de seguridad.
- El sistema SOLRIF, a nivel general se implementó con la arquitectura en 3 capas; datos, negocio y presentación; y también se utilizó el patrón arquitectónico MVC tanto en la aplicación web como en la móvil. Para el desarrollo del servicio web se utilizó la arquitectura REST (Representational State Transfer) la cual permite una correcta interacción con la aplicación móvil implementada en Android.

- En las pruebas unitarias de los métodos de acceso se manejó la consistencia y la persistencia en la base de datos; en la clase test generada con JUnit, en el método con la notación `@Before` se definió los datos a emplearse especificando el tipo de dato correspondiente, controlando así la consistencia; cuando la prueba pasó satisfactoriamente se ejecutó el método con la notación `@After` donde se definió el proceso para eliminar los datos insertados, logrando así la persistencia; de esta manera se consiguió mantener la integridad de los datos durante la ejecución de las pruebas unitarias.

RECOMENDACIONES

Se recomienda:

- Que para el correcto funcionamiento de la aplicación móvil, su instalación y uso sea en Smarth Phones Android igual o superior a la versión 5.1, considerando que no se realizó pruebas con versiones anteriores.
- Que las imágenes a cargar no superen los 2 megas, para evitar que la base de datos se desborde por el uso de memoria en exceso.
- Para emplear TDD de manera adecuada en proyectos de relevancia, es necesario tener algo de experiencia con esta técnica, conocer y dominar los pasos a ejecutar; porque aplicarlo sin conocimiento previo puede generar inconvenientes al momento de la programación, provocando que el desarrollo exceda del tiempo planificado.
- Realizar el mantenimiento adecuado de la base de datos para evitar inconvenientes en su funcionamiento.

BIBLIOGRAFÍA

ACADEMIA ANDROID. *Android Studio v1.0: características y comparativa con Eclipse.* [en línea]. 2014. [Consulta: 20 noviembre 2017]. Disponible en: <http://academiaandroid.com/android-studio-v1-caracteristicas-comparativa-eclipse/>.

AGENCIA DE REGULACIÓN Y CONTROL DE AGUA. *REGULACIÓN Nro. DIR-ARCA-RG-002-2016* [en línea]. 4 abril 2016. Quito, Ecuador: Agencia de Regulación y Control de Agua. [Consulta: 25 noviembre 2017]. Disponible en: http://www.regulacionagua.gob.ec/wp-content/uploads/downloads/2016/04/reg_02.pdf.

AGENCIA DE REGULACIÓN Y CONTROL DEL AGUA. *La Institución.* [en línea]. 2017. [Consulta: 17 abril 2018]. Disponible en: <http://www.regulacionagua.gob.ec/la-institucion/>.

ARAÚJO, A. *Test Driven Development Fortalezas y Debilidades.* En: U.F.– INCO. (ed.) [en línea]. 2007. Montevideo, Uruguay: [Consulta: 24 noviembre 2017]. Disponible en: <http://www-pao.ksc.nasa.gov/history/mercury/mercury-overview.htm>.

BLE, C. *Capítulo 2. ¿Qué es el Desarrollo Dirigido por Tests? (TDD).* *Creative Commons Atribución SinDerivadas 3.0 Unported* [en línea]. 2013. [Consulta: 20 julio 2017]. Disponible en: http://librosweb.es/libro/tdd/capitulo_2.html.

CALABRESE, J., MUÑOZ, R., PASINI, A., ESPONDA, S., BORACCHIA, M. y PESADO, P. *Asistente para la evaluación de características de calidad de producto de software propuestas por ISO/IEC 25010 basado en métricas definidas usando el enfoque GQM.* [en línea]. 2017. La Plata: [Consulta: 31 marzo 2018]. Disponible en: http://sedici.unlp.edu.ar/bitstream/handle/10915/63778/Documento_completo.pdf-PDFA.pdf?sequence=1.

CLARÍS, P. *Metodología SCRUM para desarrollo de software a medida.* *SOFTENG* [en línea]. 2018. [Consulta: 22 noviembre 2017]. Disponible en: <https://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum.html>.

- CUENCA, L.F., VEINTIMILLA, A., RUIZ, J.A. y MARIO BERNABE RON.** *ESTUDIO DE LA TÉCNICA TEST DRIVEN DEVELOPMENT (TDD) Y DESARROLLO DEL SISTEMA PARA LA ADMINISTRACIÓN DE CONSULTORIOS MÉDICOS.* [en línea]. 2014. Sangolquí, Ecuador: [Consulta: 31 agosto 2018]. Disponible en: <https://repositorio.espe.edu.ec/bitstream/21000/9667/1/AC-SI-ESPE-048066.pdf>.
- DEPT. CIENCIA DE LA COMPUTACIÓN E IA.** *Introducción a JavaServer Faces. Experto Java Universidad de Alicante* [en línea]. 2014. [Consulta: 15 julio 2015]. Disponible en: <http://www.jtech.ua.es/j2ee/publico/jsf-2012-13/sesion01-apuntes.html>.
- DIMES, T.** *Conceptos Básicos De Scrum: Desarrollo De Software Agile Y Manejo De Proyectos Agile* [en línea]. 2015. S.l.: Babelcube Incorporated. [Consulta: 17 julio 2015]. ISBN 1507102739. Disponible en: <https://books.google.com/books?id=ETuXBgAAQBAJ&pgis=1>.
- FERNÁNDEZ, Y. y DÍAZ, Y.** *Patrón Modelo-Vista-Controlador* [en línea]. 2012. La Habana: Revista Telem@tica. [Consulta: 31 octubre 2017]. Disponible en: <http://revistatelematica.cujae.edu.cu/index.php/tele/article/view/15/10>.
- GONZALEZ, A.N.** *¿Qué es Android?* [en línea]. 2011. [Consulta: 20 noviembre 2017]. Disponible en: <https://www.xatakandroid.com/sistema-operativo/que-es-android>.
- GONZÁLEZ, F.J.** *Maqueta DCC PostgreSQL - Aula de Software Libre. Aula de Software Libre de la UCO. (CC-BY-SA)* [en línea]. 2014. [Consulta: 5 noviembre 2018]. Disponible en: <https://www.uco.es/aulasoftwarelibre/577-maqueta-dcc-postgresql/>.
- GONZÁLEZ, M.** *Herramienta de Desarrollo Netbeans.* [en línea]. 2015. S.l.: [Consulta: 31 enero 2018]. Disponible en: http://www.consultorjava.com/wp/wp-content/uploads/2015/09/herramienta_desarrollo_netbeans.pdf.
- GUERRERO, R.M.** *PostgreSQL-ES / Emc2Net. Emc2Net* [en línea]. 2016. [Consulta: 5 mayo 2018]. Disponible en: <https://e-mc2.net/es/postgresql-es>.
- HERNANDEZ, U.** *MVC (Model, View, Controller) explicado. codigofacilito* [en línea]. 2015. [Consulta: 31 enero 2018]. Disponible en: <https://codigofacilito.com/articulos/mvc-model-view-controller-explicado>.

- HERRANZ, J.I.** *TDD como metodología de diseño de software - Paradigma*. [en línea]. 2011. [Consulta: 20 noviembre 2017]. Disponible en: <https://www.paradigmadigital.com/dev/tdd-como-metodologia-de-diseno-de-software/>.
- ISO 25000.** *ISO 25010: Seguridad*. [en línea]. 2018a. [Consulta: 5 noviembre 2017]. Disponible en: <http://iso25000.com/index.php/normas-iso-25000/iso-25010?limit=3&start=6>.
- ISO 25000.** *La familia de normas ISO/IEC 25000*. [en línea]. 2018b. [Consulta: 5 noviembre 2018]. Disponible en: <http://iso25000.com/index.php/normas-iso-25000?limit=4&limitstart=0>.
- JURADO, C.B., PLAZA, J.G., PERDOMO, F.R. y MENA, G.** *Diseño Ágil con TDD*. 2010. S.l.: [Lulu]. ISBN 1445264714.
- MARTIN, S.A.** *PostgreSQL* [en línea]. 2011. S.l.: SIA OmniScriptum Publishing. [Consulta: 17 julio 2015]. Disponible en: <https://www.eae-publishing.com/catalog/details/store/es/book/978-3-8465-7527-7/postgresql>.
- MARTÍNEZ, E.** *Cómo utilizar la metodología Scrum para acometer proyectos complejos. Innovation & Entrepreneurship Business School* [en línea]. 2013. [Consulta: 11 diciembre 2017]. Disponible en: <https://www.iebschool.com/blog/metodologia-scrum-agile-scrum/>.
- PECH, F., GOMEZ, M., CRUZ, L. de la y LARA, S.** *Desarrollo de Aplicaciones web con JPA, EJB, JSF y PrimeFaces*. [en línea]. 2011. Tabasco, México: [Consulta: 20 noviembre 2017]. Disponible en: <https://www.tamps.cinvestav.mx/~fpech/sd/files/paper001.pdf>.
- PEÑARROCHA, M.** *Una experiencia aplicando Test-Driven Development (TDD) usando una herramienta JUnit*. [en línea]. 2016. Valencia, España: [Consulta: 24 noviembre 2017]. Disponible en: [https://riunet.upv.es/bitstream/handle/10251/76346/PEÑARROCHA - Una experiencia aplicando Test-Driven Development %28TDD%29 usando una herramienta JUnit.pdf?sequence=2&isAllowed=y](https://riunet.upv.es/bitstream/handle/10251/76346/PEÑARROCHA%20-%20Una%20experiencia%20aplicando%20Test-Driven%20Development%20%28TDD%29%20usando%20una%20herramienta%20JUnit.pdf?sequence=2&isAllowed=y).
- PEREZ, L.** *MODELADO DE SISTEMAS DE INFORMACION: Familia iso 25000*. Blogger [en línea]. 2015. [Consulta: 11 diciembre 2017]. Disponible en: <http://modelado-luchoperez.blogspot.com/2015/09/familia-iso-25000.html>.

PMOINFORMATICA. *Test Driven Development (TDD): Ventajas y desventajas - La Oficina de Proyectos de Informática.* Blogger [en línea]. 2012. [Consulta: 20 noviembre 2017]. Disponible en: <http://www.pmoinformatica.com/2012/12/test-driven-development-ventajas-y.html>.

REYES, A. *¿Que es NetBeans? explique 3 de sus características.* Blogger [en línea]. 2017. [Consulta: 14 mayo 2017]. Disponible en: <http://investigacion1javaep.bligoo.ec/que-es-netbeans-explique-3-de-sus-caracteristicas#.V4fEXjUXcg5>.

RIOS, S. *JSF 2 + Hibernate 4 + Spring 4: PrimeFaces 5 with JAX-WS y EJB'S* [en línea]. 2015. S.l.: Sergio Rios. [Consulta: 17 julio 2015]. Disponible en: <https://books.google.com/books?id=N2AvBwAAQBAJ&pgis=1>.

ROJO, M.I. *Ensalada de Bits: ISO 25000 y el software seguro.* *blogspot* [en línea]. 2015. [Consulta: 31 octubre 2018]. Disponible en: <http://ensaladadebits.blogspot.com/2015/01/iso-25000-y-el-software-seguro.html>.

SECRETARÍA NACIONAL DE PLANIFICACIÓN Y DESARROLLO - SENPLADES. *Plan Nacional de Desarrollo 2017-2021-Toda una Vida* [en línea]. 22 septiembre 2017. Quito, Ecuador: REPÚBLICA DEL ECUADOR CONSEJO NACIONAL DE PLANIFICACIÓN (CNP). [Consulta: 19 abril 2018]. Disponible en: http://www.planificacion.gob.ec/wp-content/uploads/downloads/2017/10/PNBV-26-OCT-FINAL_0K.compressed1.pdf.

TORRES, O.I. *Aplicación y evaluación de la metodología desarrollo orientado por pruebas (TDD), cas de estudio: spot.co.* [en línea]. 2017. Duitama: [Consulta: 18 octubre 2018]. Disponible en: [https://reunir.unir.net/bitstream/handle/123456789/6562/TORRES CORREDOR%20OSCAR.pdf?sequence=1&isAllowed=y](https://reunir.unir.net/bitstream/handle/123456789/6562/TORRES%20CORREDOR%20OSCAR.pdf?sequence=1&isAllowed=y).

TRIBUNAL CONSTITUCIONAL DE LA REPÚBLICA DE ECUADOR. *LEY ORGÁNICA DE RECURSOS HÍDRICOS, USOS Y APROVECHAMIENTO DEL AGUA* [en línea]. 6 agosto 2014. Quito, Ecuador: LEXIS. [Consulta: 28 octubre 2017]. Disponible en: <https://www.agua.gob.ec/wp-content/uploads/2012/10/LEYD-E-RECURSOS-HIDRICOS-II-SUPLEMENTO-RO-305-6-08-204.pdf>.

VACA, P.A., MALDONADO, C., INCHAURRONDO, C., PERETTI, J., ROMERO, M.S.
y **BUENO, M.** *Test-Driven Development-Una aproximación para entender su utilidad en el proceso de desarrollo de Software.* [en línea]. 2015. S.l.: [Consulta: 24 noviembre 2017].
Disponible en: <http://conaiisi.frc.utn.edu.ar/PDFsParaPublicar/1/schedConfs/7/158-524-1-DR.pdf>.

ANEXOS

Anexo A. Diagrama de actividades

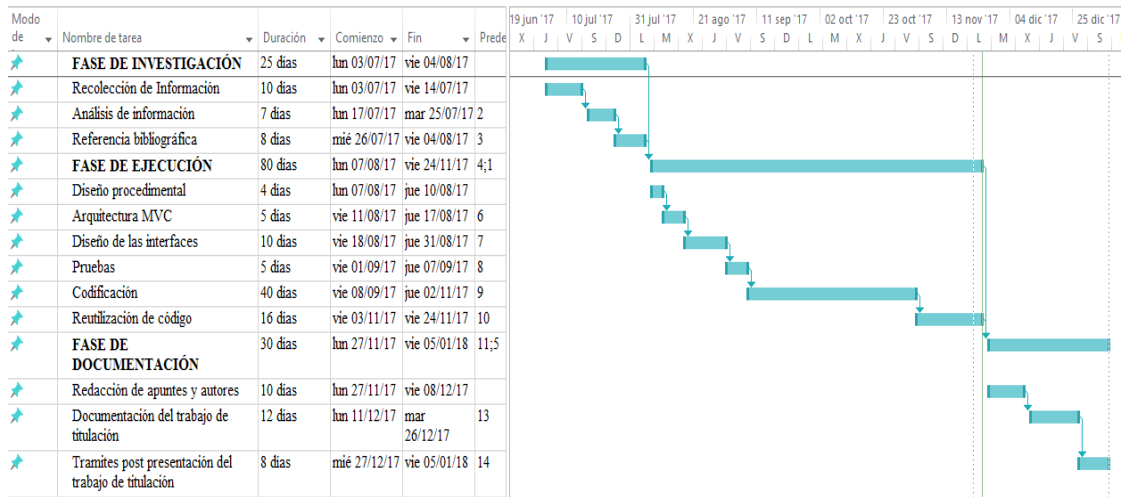


Figura a-1. Actividades Diagrama Gantt

Realizado por: Tupac Tisalema, 2019

Anexo B. Estimación de costos y esfuerzos

TABLA DE PUNTOS DE FUNCIÓN

A continuación, se muestra las tablas a1, a2 y a3 pretenden medir el tamaño del software, de acuerdo a su complejidad y el número de las siguientes funcionalidades:

- Pantallas donde el usuario ingresa datos (**EI**).
- Informes, gráficos, listado de datos (Datos calculados) (**EO**).
- Buscar, recuperar y mostrar datos al usuario (**EQ**).
- Tablas en la base de datos (**ILF**).
- Datos obtenidos de otro sistema (**EIF**).

Tabla b1. Tabla puntos de función de acuerdo a los valores estándar (IFPUG) International

Tipo/Complejidad	Baja	Media	Alta
Entrada Externa (EI)	3 PF	4 PF	6 PF
Salida Externa (EO)	4 PF	5 PF	7 PF
Consulta Externa (EQ)	3 PF	4 PF	6 PF
Archivo lógico interno (ILF)	7 PF	10 PF	15 PF
Archivo de interfaz externo (EIF)	5 PF	7 PF	10 PF

Realizado por: Tupac Tisalema. 2019

Function Point Users Group.

A continuación, se determinará los puntos de función para cada requerimiento se considera que la mayoría de las funcionalidades tienen un complejidad intermedia y alta.

Tabla b2. Tabla puntos de función de cada historia de usuario.

ID Requerimiento	Tipo	Complejidad PF
HU-01	EI	3
HU-02	EIF	7
HU-03	EI	4
HU-04	EIF	5
HU-05	EI	3
HU-06	EIF	5
HU-07	EIF	5
HU-08	EI	3
HU-09	EI	4
HU-10	EI	3
HU-11	EIF	5
HU-12	EIF	5
HU-13	EIF	5
HU-14	EIF	5
HU-15	EQ	3
HU-16	EQ	4
HU-17	EQ	3
HU-18	EQ	4
HU-19	EO	4
HU-20	EO	5
HU-21	EO	4
HU-22	EO	5
HT-01	ILF	7
HT-02	EO	4
HT-03	EO	4
HT-04	EIF	5
HT-05	EIF	5
	PFSA	119

Realizado por: Tupac Tisalema. 2019

Puntos de función sin ajustar (PFSA): 130

Tabla b3. Tabla puntos de función proyecto laboratorios virtuales.

Tipo/Complejidad	Baja	Media	Alta	Total
Entrada Externa (EI)	4*3 PF	2*4 PF	0*6 PF	20
Salida Externa (EO)	4*4 PF	2*5 PF	0*7 PF	26
Consulta Externa (EQ)	2*3 PF	2*4 PF	0*6 PF	14
Archivo lógico interno (ILF)	1*7 PF	0*10 PF	0*15 PF	7
Archivo de interfaz externo (EIF)	9*5 PF	1*7 PF	0*10 PF	52
			PFSA	119

Realizado por: Tupac Tisalema. 2019

Cálculo de puntos de función ajustados

Tabla b4. Puntos de función ajustados.

Factor de ajuste	Puntaje
Comunicación de datos	4
Procesamiento distribuido	4
Objetivos de rendimiento	1
Configuración del equipamiento	1
Tasa de transacciones	3
Entrada de datos en línea	5
Interfaces con el usuario	2
Actualizaciones en línea	3
Procesamiento complejo	1
Re-usabilidad código	1
Facilidad de implementación	-
Facilidad de operación	1
Instalaciones múltiples	2
Facilidad de cambios	4
Factor de ajuste	32

Realizado por: Tupac Tisalema. 2019

Puntos de función ajustado (PFA) = Puntos de función sin ajustar (PFSA) * [0.65+(0.01*factor ajuste)]

$$\text{PFA} = 119 * [0.65 + (0.01 * 32)]$$

$$\text{PFA} = 115.4 \implies 115$$

Estimación Esfuerzo

Determina la cantidad de horas/hombre de acuerdo al lenguaje de programación que se use para el desarrollo del proyecto, para este proyecto el lenguaje de programación Java, que pertenece al lenguaje de programación de 4ta generación.

Los puntos de función en cierto modo son una medida subjetiva, por lo cual la estimación dependerá en gran medida de la experiencia del equipo de desarrollo.

Tabla b5. Estimación Esfuerzo

Lenguaje	Horas PF promedio	Líneas de código por PF
Ensamblador	25	300
COBOL	15	100
Lenguajes de 4ta generación	8	20

Realizado por: Tupac Tisalema, 2019

$$\text{H/H} = \text{PFA} * \text{Horas PF promedio}$$

$$\text{H/H} = 115 * 8$$

$$\text{H/H} = \mathbf{920} \text{ horas hombre}$$

CASO 1

6 horas diarias de trabajo

1 mes = 20 días laborable

1 desarrollador

$$\mathbf{920} \text{ horas por hombre} / \mathbf{6} \text{ horas por día} = \mathbf{153.3} \text{ días de trabajo por hombre}$$

$\mathbf{153} / \mathbf{20}$ días = 7.6 meses de trabajo para desarrollar el software de lunes a viernes 6 horas diarias con 1 desarrollador.

Estimación del pre-supuesto del proyecto

Sueldo mensual desarrolladores: \$300.00 _____ Otros costos del proyecto: 650\$

COSTO ESTIMADO

Costo = (Desarrolladores * duración meses * sueldos) + otros costos

Costo = (1 * 7.6 * 550) + 650;

Costo = 2,930.00\$

Conclusión

En conclusión, el tiempo estimado con un solo desarrollador para el finalizar el proyecto es de 7 meses y tendrá un costo estimado de \$2,930.00 dólares americanos.

Anexo C. Factibilidad Técnica

Tabla c1. Presupuesto Designado Para la Ejecución del Proyecto

ÍTEMS	CANTIDAD	VALOR UNITARIO	TOTAL
SOFTWARE			
WINDOWS SERVER	2012	250.00	250.00
EQUIPO HARDWARE			
UPC REGULADOR DE ENERGÍA ELÉCTRICA	1	70.00	70.00
Impresora EPSON L380	1	300.00	300.00
Servidor HPE ML30 Gen9 E3-1220v5-MCA Svr/S/Buy	1	1100.00	1100.00
PC de escritorio	2	500.00	1000.00
OTROS			
Internet	500	0.50	250.00
TOTAL			2970.00

Realizado por: Tupac Tisalema, 2019

Anexo D. Módulo de Autenticación

El Módulo de autenticación y gestión de proyectos nos permite un ingreso seguro al sistema contando con usuarios registrados para concederles permisos de gestión de datos, si un usuario no está registrado en la base de datos local del sistema no podrá acceder por ningún medio cuidado

la integridad y seguridad de datos alojados. Si la autenticación es exitosa dentro del sistema web podrá ejercer diferentes tareas asignadas por el administrador dependiendo los permisos que se otorgue a cada usuario del sistema.

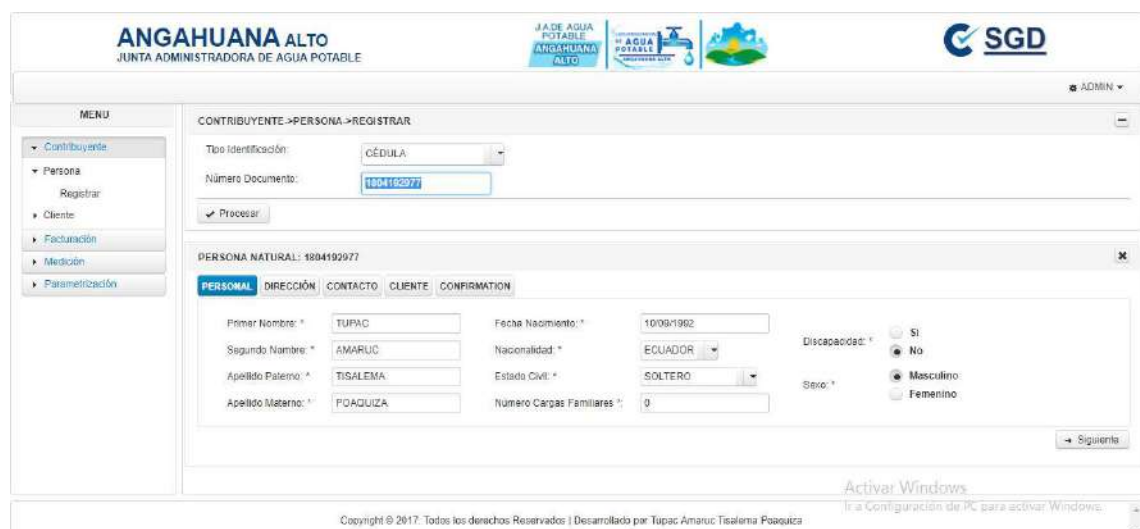


The screenshot shows the login interface for the water utility system. At the top, there are three logos: 'J.A. DE AGUA POTABLE ANGAHUANA ALTO', 'JUNTA ADMINISTRADORA DE AGUA POTABLE ANGAHUANA ALTO', and a landscape logo. Below the logos is a blue header with the text 'INICIAR SESIÓN'. The main content area features a circular icon of a water tap with a single drop. Below this are two input fields: 'USUARIO' and 'CONTRASEÑA', each with a corresponding icon (a person and a lock). There is a 'Remember me' checkbox and a 'Need help?' link. At the bottom is a large blue button labeled 'Ingresar'.

Figura d-1. Pantalla Principal de Login
Realizado por: Tupac Tisalema, 2019

Módulo Contribuyente

Registrar: Permite registrar datos de la persona que posterior de pueden convertir en clientes de la junta o usuarios del sistema.



The screenshot displays the 'Registrar persona' form within the water utility system. The header includes the 'ANGAHUANA ALTO' logo and the 'SGD' logo. The breadcrumb trail is 'CONTRIBUYENTE->PERSONA->REGISTRAR'. The form contains the following fields and options:

- Tipo Identificación:** A dropdown menu set to 'CÉDULA'.
- Número Documento:** A text input field containing '1804162977'.
- Procesar:** A button with a checkmark icon.
- PERSONA NATURAL: 1804162977:** A window title for the personal data section.
- Tabs:** 'PERSONAL', 'DIRECCIÓN', 'CONTACTO', 'CLIENTE', and 'CONFIRMATION'. The 'PERSONAL' tab is active.
- Personal Data Fields:**
 - Primer Nombre:** TUPAC
 - Segundo Nombre:** AMARUC
 - Apellido Paterno:** TISALEMA
 - Apellido Materno:** POAQUIZA
 - Fecha Nacimiento:** 10/08/1982
 - Nacionalidad:** ECUADOR
 - Estado Civil:** SOLTERO
 - Numero Cargas Familiares:** 0
 - Discapacidad:** Radio buttons for 'Si' and 'No'.
 - Sexo:** Radio buttons for 'Masculino' and 'Femenino'.
- Siguiente:** A button at the bottom right of the form.

At the bottom of the page, there is a copyright notice: 'Copyright © 2017. Todos los derechos Reservados | Desarrollado por Tupac Amaruc Tisalema Poaquiza' and a Windows activation watermark.

Figura d-2. Pantalla registrar persona
Realizado por: Tupac Tisalema, 2019

Contratar: Brinda la posibilidad de que una persona al convertirse en cliente de la junta contrate los servicios disponibles.

ANGAHUANA ALTO
JUNTA ADMINISTRADORA DE AGUA POTABLE

Logo: JADE AGUA POTABLE ANGAHUANA ALTO, ANGAHUANA ALTO AGUA POTABLE, SGD

ADMIN

MENU

- Contribuyente
- Persona
 - Registrar
- Clients
 - Servicio
 - Contratados
 - Contratados por cliente
 - Contratar
 - Gestionar
- Facturación
- Medición
- Parametrización

CONTRATAR SERVICIO

Buscar por: Num. Cliente Tipo Identificación Num. Cliente: 1 Proceso

CLIENTE: EDWIN ROBERTO TISALEMA CUYANQUILLO

INFORMACIÓN CONTRATO CONFIRMATION

C.I./RUC: 1805068440 Número: 1

Cliente: EDWIN ROBERTO TISALEMA CUYANQUILLO

Oficina: MATRIZ

Sector: ZONA 1

Categoría: RESIDENCIAL

Vía: CALLE DE ASFALTO

Servicios Contratados: 3

SERVICIOS DISPONIBLES

N° MEDIDOR	SERVICIO	SECTOR	CATEGORIA	DOMICILIO	TIENE MEDIDOR	COSTO	
NN	ALCANTARILLADO	ZONA 1	RESIDENCIAL		NO	\$200.0	<input type="checkbox"/>
NN	MANTENIMIENTO	ZONA 1	RESIDENCIAL		NO	\$0.0	<input type="checkbox"/>
NN	AGUA POTABLE	ZONA 1	RESIDENCIAL		SI	\$200.0	<input type="checkbox"/>

Activar Windows
Ir a Configuración de PC para activar Windows.

Copyright © 2017. Todos los derechos Reservados | Desarrollado por Tupac Amaru Tisalema Pooquizo

Figura d-3. Pantalla contratar servicio

Realizado por: Tupac Tisalema, 2019

Registrar cuentas por cobrar: Esto a ayuda que al momento de migrar datos se pueda registrar las cuentas por cobrar por adquisición o consumo, pendientes por cobrar a los clientes.

REGISTRO POR COBRAR

Cliente: 1805068440 TISALEMA CUYANQUILLO EDWIN ROBERTO

Tipo: Seleccione

Servicio: Seleccione

Descripción: DESCRIPCIÓN

Valor: 0.0

Guardar Cancelar

RUBROS POR COBRAR

TIPO	SERVICIO	DESCRIPCIÓN	VALOR	ESTADO	ACCIONES
ADQUISICION	AGUA POTABLE	CONSUMO DE ENERO A MARZO.	150.0	POR COBRAR	<input type="checkbox"/>

Figura d-4. Pantalla contratar servicio

Realizado por: Tupac Tisalema, 2019

Módulo Medición

Lectura medidor – Web: Aquí se podrá registrar las lecturas de medidor realizadas por el plomero.

The screenshot shows a web application interface for recording meter readings. At the top, there are logos for 'ANGAHUANA ALTO JUNTA ADMINISTRADORA DE AGUA POTABLE', 'J.A. DE AGUA POTABLE ANGAHUANA ALTO', 'AGUA POTABLE', and 'SGD'. The main area is titled 'MEDICIÓN > LECTURA > REGISTRAR'. It features a search bar with 'Num. Medidor' selected and a value of '001'. Below this is a 'Procesar' button. A section titled 'LECTURA DE MEDIDOR' contains two tabs: 'INFORMACIÓN' (selected) and 'CONFIRMACIÓN'. The 'INFORMACIÓN' tab displays the following data: C.I./RUC: 1805069440, Número: 1, Num. Medidor: 001, Fecha Corte: 01/02/2018, Cliente: EDWIN ROBERTO TISALEMA CUYANQUILLO, Lectura Anterior (m3): 0, Observación: MEDIDOR EN BUEN ESTADO, Dirección: ZONA 1 HG, Lectura Actual (m3): 150, Servicio: AGUA POTABLE, Período Actual: ENERO - 2018, and Categoría: RESIDENCIAL. A sidebar menu on the left lists various options under 'Medición', including 'Registrar' and 'Gestionar'. At the bottom, there is a copyright notice for 2017 and a Windows activation watermark.

Figura d-5. Pantalla lectura medidor - web

Realizado por: Tupac Tisalema, 2019

Lectura medidor – Móvil: Brinda al plomero la posibilidad de registrar directamente las lecturas de medidor.

The screenshot shows a mobile application interface for recording meter readings. The app is titled 'SolRif' and has two main tabs: 'LECTURA' (selected) and 'INFORMACIÓN'. The 'LECTURA' screen shows a search bar with 'Medición' and 'ATISALEMA' next to it. Below this is a 'Buscar Cliente' button. The search criteria are: 'Buscar por: Número Medidor', 'Tipo identificación: CÉDULA', and 'Número Medidor: 001'. A large green play button is visible. Below the search criteria, there is a list of details: CI/RUC: 1802527067, Número: 10, Cliente: SEGUNDO ENRIQUE TISALEMA TISALEMA, Dirección: ZONA 4 001-TERRAZA DE LADRILLO, Servicio: AGUA POTABLE, Categoría: RESIDENCIAL, and Número Medidor: 001. The 'INFORMACIÓN' screen shows: Categoría: RESIDENCIAL, Número Medidor: 001, Lectura anterior (m3): 100, Lectura actual (m3): 0, Período Actual: DICIEMBRE - 2017, Fecha Corte: (empty), and Observación: (empty). At the bottom, there is a section titled 'Imagen Medidor' with a photo of a meter and two icons for camera and gallery.

Figura d-6. Pantalla lectura medidor - móvil

Realizado por: Tupac Tisalema, 2019

Módulo facturación

Generación factura: Procesa los comprobantes para cobrar mientras exista una adquisición o este registrada la lectura de medidor correspondiente al periodo vigente.

ANGAHUANA ALTO
JUNTA ADMINISTRADORA DE AGUA POTABLE

FACTURACIÓN->COBROS

Buscar por: Num. Cliente Tipo Identificación

Num. Cliente:

CLIENTE: EDWIN ROBERTO TISALEMA CUYANQUILLO

NÚMERO	FECHA GENERACIÓN PERIODO	CLIENTE	ESTADO	ACCIONES
2	2018-04-16 20:25:59.69	EDWIN ROBERTO TISALEMA CUYANQUILLO	GENERADO	<input type="button" value="p"/>
3	2018-04-16 20:42:07.649	EDWIN ROBERTO TISALEMA CUYANQUILLO	GENERADO	<input type="button" value="p"/>

Activar Windows
Ir a Configuración de PC para activar Windows.

Copyright © 2017. Todos los derechos Reservados | Desarrollado por Tupac Amaru Tisalema Poaquiza

Figura d-7. Pantalla generación factura

Realizado por: Tupac Tisalema, 2019

Cobro: Permite realizar los cobros de los comprobantes generados.

J.A. DE AGUA POTABLE ANGAHUANA ALTO

JUNTA ADMINISTRADORA DE AGUA POTABLE ANGAHUANA ALTO

Dirección: Comunidad Angahuana Alto - Zona 4
R.U.C.: 1603219284001
AUTO. SRI: 1121827300
OBLIGADO A LLEVAR CONTABILIDAD

COMPROBANTE
N°. 000002

Nombre: EDWIN ROBERTO TISALEMA CUYANQUILLO Dirección: ZONA 1 KKJ
RUC / C.I.: 1905068440 Fecha Emisión: 16/04/2018

CODIGO	CANTIDAD	UNIDAD MEDIDA	DESCRIPCIÓN	P. UNIT.	DESCT.	P. TOTAL	IVA UNIT.	IVA TOTAL
AGP	1	-	ADQUISICION AGUA POTABLE RESIDENCIAL EN ZONA 1 ASA BLANCA	\$ 250.0	\$ 0.0	\$ 250.0	\$ 0.0	\$ 0.0
ALC	1	-	ADQUISICION ALCANTARILLADO RESIDENCIAL EN ZONA 1 CASA BLANCA	\$ 250.0	\$ 0.0	\$ 250.0	\$ 0.0	\$ 0.0
M	1	-	ADQUISICION MANTENIMIENTO RESIDENCIAL EN ZONA 1 ASA BLANCA	\$ 0.0	\$ 0.0	\$ 0.0	\$ 0.0	\$ 0.0
Sub Total:						\$ 500.0	Total:	\$ 0.0
Descuento:						\$ 0.0		
Iva:						\$ 0.0		
Valor Total:						\$ 500.0		

Figura d-8. Pantalla cobro

Realizado por: Tupac Tisalema, 2019

Módulo Parametrización

Gestión usuarios: Facilita la creación de usuarios, modificación de sus datos y si es el caso la inactivación de los mismos.

The screenshot shows the 'LISTA DE USUARIOS' (User List) interface. At the top, there are logos for 'ANGAHUANA ALTO JUNTA ADMINISTRADORA DE AGUA POTABLE', 'J.A.D.E. AGUA POTABLE ANGAHUANA ALTO', and 'SGD'. A navigation menu on the left includes 'Contribuyente', 'Facturación', 'Medición', 'Parametrización', 'Personas', 'Por Definir', 'Servicio', and 'Seguridad'. The 'Parametrización' menu is expanded, showing 'Funcionalidades', 'Menu', 'Modulos', 'Perfiles', 'Usuarios', and 'Roles'. The main area displays a table of users with columns for 'NOMBRE', 'APELLIDO', 'USERNAME', 'FECHA REGISTRO', 'ROL', 'OFICINA', 'ESTADO', and 'ACCIONES'. Two users are listed: 'SUPER ADMINISTRADOR ADMIN' and 'JOSE TISALEMA JTISALEMA'. The interface also includes a 'Nuevo' button and a 'Activar Windows' watermark.

NOMBRE	APELLIDO	USERNAME	FECHA REGISTRO	ROL	OFICINA	ESTADO	ACCIONES
SUPER	ADMINISTRADOR	ADMIN	2018-03-02 08:48:15.0	ADMINISTRADOR DE SISTEMAS	MATRIZ	<input checked="" type="checkbox"/>	 
JOSE	TISALEMA	JTISALEMA	2018-03-14 21:45:25.295	SECRETARIO	MATRIZ	<input checked="" type="checkbox"/>	 

Figura d-9. Pantalla gestión usuario

Realizado por: Tupac Tisalema, 2019

The 'NUEVO USUARIO' form contains the following fields and options:

- Cédula de identidad sin guión *: 1804192977
- Nombre: TUPAC
- Apellido: TISALEMA
- Nombre de usuario: TTISALEMA
- Password: 123Abc
- Email: (empty field)
- Rol: seleccione un rol (dropdown menu)
- Oficina: seleccione una oficina (dropdown menu)
- Perfiles: ADMINISTRADOR, INFORMACIÓN, OPERARIO

Buttons: Guardar, Cancelar

Figura d-10. Pantalla gestión usuario (2)

Realizado por: Tupac Tisalema, 2019

Tarifas: Permite definir las tarifas a cobrar tanto por adquisición y consumo de servicios.

NUEVA TARIFA

Servicio: AGUA POTABLE

Categoría: RESIDENCIAL

Sector: ZONA 1

Vía: CALLE DE ASFALTO

Tipo tarifa: ADQUISICION

Unidad Medida: UNIDAD

Cantidad: 1

Costo por cantidad total: (\$) 350

Iva (%): 0.0

Costo por unidad: **\$ 350.0**

Guardar Cancelar

Figura d-11. Pantalla gestión tarifas
Realizado por: Tupac Tisalema, 2019

ANGAHUANA ALTO
JUNTA ADMINISTRADORA DE AGUA POTABLE

ASAC AGUA POTABLE ANGAHUANA ALTO

ASAC AGUA POTABLE ANGAHUANA ALTO

SGD

ADMIN

MENU

- Configuración
- Facturación
- Medición
- Parametrización
- Presetas
- PIV Control
- Servicio
- Tarifa
- Calidad
- Gestor Servicio
- Seguridad

PARAMETRIZACIÓN > SERVICIO > TARIFA

Nuevo

SERVICIO	CATEGORIA	SECTOR	VIA	TIPO TARIFA	CANTIDAD	MEDIDA	P. UNIT.	P. TOTAL	IVA	ACCIONES
AGUA POTABLE	RESIDENCIAL	ZONA 1	CALLE DE TIERRA	ADQUISICION	1	UNIDAD	130.0	130.0	0.0 %	[B] [E] [D]
ALCANTARILLADO	RESIDENCIAL	ZONA 1	CALLE DE ASFALTO	ADQUISICION	1	UNIDAD	250.0	250.0	0.0 %	[B] [E] [D]
ALCANTARILLADO	RESIDENCIAL	ZONA 1	CALLE DE TIERRA	ADQUISICION	1	UNIDAD	250.0	250.0	0.0 %	[B] [E] [D]
MANTEENIMIENTO	RESIDENCIAL	ZONA 1	CALLE DE ASFALTO	ADQUISICION	1	UNIDAD	9.0	9.0	0.0 %	[B] [E] [D]
MANTEENIMIENTO	RESIDENCIAL	ZONA 1	CALLE DE TIERRA	ADQUISICION	1	UNIDAD	9.0	9.0	0.0 %	[B] [E] [D]
MANTEENIMIENTO	RESIDENCIAL	ZONA 1	CALLE DE ASFALTO	CUOTA	1	UNIDAD	1.0	1.0	0.0 %	[B] [E] [D]

Activar Windows
Vaya a Configuración de PC para activar Windows.

Copyright © 2017. Todos los derechos Reservados | Desarrollado por: Tupac Amaru Tisalema Pomaqueza

Figura d-12. Pantalla gestión tarifa (2)
Realizado por: Tupac Tisalema, 2019

Anexo E. Diseño de la base de datos

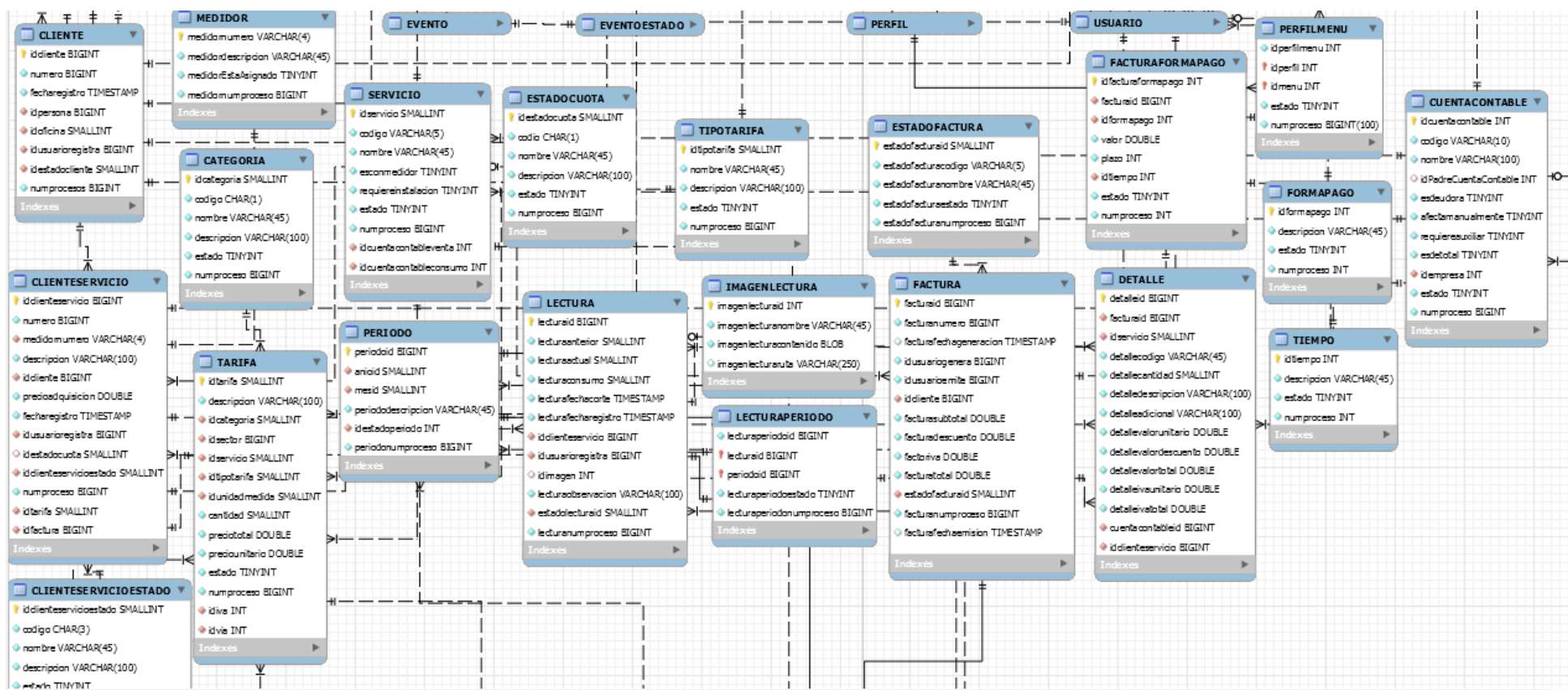


Figura e-1. Diseño base de datos

Realizado por: Tupac Tisalema, 2019

Anexo F. Historias de Usuario

Tabla f1. Historia de Usuario 01

Historia de Usuario	
Número: HU-01	Como administrador necesito una interfaz de autenticación para ingresar las credenciales al identificarme en el sistema.
Modificación de historia de usuario:	
Usuario: Administrador	Iteración Asignada: 1
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 8
Riesgo en el Desarrollo: Bajo (Alto / Medio / Bajo)	Puntos Reales: 8
Descripción: Como <i>administrador</i> necesito una <i>interfaz de autenticación</i> , para ingresar las credenciales al momento de identificarme en el sistema.	
Observaciones: Sólo se permite letras mayúsculas en el formulario para nombre de usuario, en el formulario de contraseña los caracteres deben ser ocultos.	

Realizado por: Tupac Tisalema, 2019

Tabla f2. Historia de Usuario 02

Historia de Usuario	
Número: HU-02	Como secretario necesito registrar los valores por cobrar que mantiene los usuarios con las Junta.
Modificación de historia de usuario:	
Usuario: Secretario	Iteración Asignada: 1
Prioridad en el Negocio: Medio (Alta / Media / Baja)	Puntos Estimados: 32
Riesgo en el Desarrollo: Medio (Alto / Medio / Bajo)	Puntos Reales: 34
Descripción: Como <i>secretario</i> necesito <i>registrar los valores por cobrar</i> que mantienen los usuarios con la Junta, para poder cobrar los valores pendientes a cada usuario.	
Observaciones:	

Realizado por: Tupac Tisalema, 2019

Tabla f3. Historia de Usuario 03

Historia de Usuario	
Número: HU-03	Como administrador necesito autenticarme en el sistema para poder acceder.
Modificación de historia de usuario:	
Usuario: Administrador	Iteración Asignada: 2
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 40
Riesgo en el Desarrollo: Medio (Alto / Medio / Bajo)	Puntos Reales: 40
Descripción: Como <i>administrador</i> necesito <i>autenticarme en el sistema</i> para poder acceder y realizar las tareas pertinentes.	
Observaciones: El usuario debe estar registrado en el sistema.	

Realizado por: Tupac Tisalema, 2019

Tabla f4. Historia de Usuario 04

Historia de Usuario	
Número: HU-04	Como administrador deseo visualizar una interfaz de bienvenida al acceder al sistema.
Modificación de historia de usuario:	
Usuario: Administrador	Iteración Asignada: 2
Prioridad en el Negocio: Baja (Alta / Media / Baja)	Puntos Estimados: 8
Riesgo en el Desarrollo: Baja (Alto / Medio / Bajo)	Puntos Reales: 7
Descripción: Como <i>administrador</i> necesito <i>visualizar una interfaz de bienvenida</i> al acceder al sistema, para una mejor presentación.	
Observaciones:	

Realizado por: Tupac Tisalema, 2019

Tabla f5. Historia de Usuario 05

Historia de Usuario	
Número: HU-05	Como secretario necesito registrar los datos de los usuarios o clientes de la Junta.
Modificación de historia de usuario:	
Usuario: Secretario	Iteración Asignada: 2
Prioridad en el Negocio: Alto (Alta / Media / Baja)	Puntos Estimados: 32
Riesgo en el Desarrollo: Medio (Alto / Medio / Bajo)	Puntos Reales: 33
Descripción: Como <i>secretario</i> necesito <i>registrar los datos de los usuarios o clientes de la Junta</i> , para utilizarlos en los procesos pertinentes.	
Observaciones: Se registrar sólo cuando el documento de identificación (Cédula, Ruc, etc.) del cliente sea válido	

Realizado por: Tupac Tisalema, 2019

Tabla f6. Historia de Usuario 06

Historia de Usuario	
Número: HU-06	Como secretario necesito realizar el proceso de adquisición del servicio de agua potable.
Modificación de historia de usuario:	
Usuario: Secretario	Iteración Asignada: 3
Prioridad en el Negocio: Alta (Alta / Media / Baja)	Puntos Estimados: 40
Riesgo en el Desarrollo: Alta (Alto / Medio / Bajo)	Puntos Reales: 45
Descripción: Como <i>secretario</i> necesito <i>realizar el proceso de adquisición del servicio de agua potable</i> , para tener registro del servicio prestado al cliente.	
Observaciones:	

Realizado por: Tupac Tisalema, 2019

Anexo G. Plan de Pruebas

TestLink Community [configure \$tlCfg->document_generator->company_name]



Informe de Diseño del Plan de Pruebas

Proyecto de Pruebas: Plan de pruebas del sistema SOLRIF

Plan de Pruebas: SOLRIF

Imprimido por TestLink el 20/12/2018

2012 © TestLink Community

Tabla de Contenidos

1.1. Módulo contribuyente

1.1.1. Registrar

SOL-1: PA_05

Proyecto de Pruebas: Plan de pruebas del sistema SOLRIF

Plan de pruebas del sistema SOLRIF.

Plan de Pruebas: SOLRIF

Plan para pruebas de funcionalidad.

1.1. Suite de Pruebas: Módulo contribuyente

En este módulo se permitirá registrar/actualizar datos de personas, adquirir servicios y registrar las cuentas por cobrar de cada cliente

1.1.1. Suite de Pruebas: Registrar

Registrar datos de persona que se convertirá en cliente de la junta y/o usuario del sistema.

Caso de Prueba SOL-1: PA_05	
<u>Autor:</u>	Admin
<u>Resumen:</u> Paso 1: Ubicarse en el menú contribuyente. Paso 2: Desglosar el menú contribuyente. Paso 3: Seleccionar el sub-ítem Persona. Paso 4: Seleccionar el ítem registrar. Paso 5: Seleccionar el tipo de identificación. Paso 6: Ingresar el número del documento de identificación. Paso 7: Pulsar el botón Procesar Paso 8: Ingresar los datos solicitados. Paso 9: Pulsar el botón guardar.	
<u>Precondiciones:</u> El documento de identificación (Cédula o Ruc) debe ser válida.	
<u>Tipo de ejecución:</u>	Manual
<u>Duración estimada de la ejec. (min):</u>	10.00

<u>Prioridad:</u>	Media
<u>Requisitos</u>	Ninguno
<u>Keywords:</u>	Ninguno

Métricas de tiempos de ejecución

Tiempo estimado para ejecutar 1 casos de prueba (min):10

Tiempo usado para ejecutar 1 casos de prueba (min):12

Anexo H. Pruebas Unitarias

Clase: mLectura

Método: insertar

Escribir las especificaciones

Verificar que el método insertar de la clase mLectura retorne true, cuando se haya insertado correctamente los datos.

Agregar prueba

```
@Before
public void setUp() {
    lectura = new cLectura();
    lectura.setLecturaAnterior((short) 0);
    lectura.setLecturaActual((short) 30);
    lectura.setLecturaConsumo((short) (lectura.getLecturaActual() - lectura.getLecturaAnterior()));
    lectura.setLecturaFechaRegistro(new Timestamp((new Date()).getTime()));
    lectura.setLecturaFechaCorte(new Date());
    lectura.setIdClienteServicio(297); //Servicio Contratado Agua Potable
    lectura.setIdUsuarioRegistra(1);
    lectura.setObjEstadoLectura(new cEstadoLectura((short) 1));
    periodo = 1;
}

@After
public void tearDown() {
    mLectura.eliminar(lectura.getLecturaId());
}

@Test
public void testLecturaInsertar() throws Exception {
    System.out.println("lecturaInsertar");
    cLectura objLectura = lectura;
    long periodoid = periodo;
    boolean expResult = true;
    boolean result = mLectura.insertar(objLectura, periodoid);
    assertEquals(expResult, result);
    // TODO review the generated test code and remove the default call to fail.
    //fail("The test case is a prototype.");
}
```

Figura h-1. Prueba unitario

Realizado por: Tupac Tisalema, 2019

Ejecutar prueba y verificar que falle

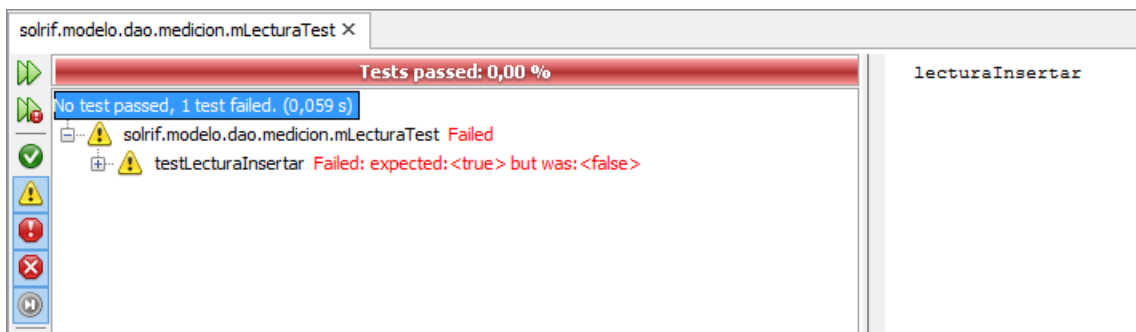


Figura h-2. Test Failed

Realizado por: Tupac Tisalema, 2019

Implementar el código que hace funcionar el ejemplo

Escribir código mínimo y suficiente para hacer que la prueba pase

```
public static boolean insertar(cLectura objLectura, long periodoid) throws Exception {
    boolean resp = false;
    String sql;

    sql = "SELECT *FROM medicion.fn_insert_lectura(?,?,?,?,?,?,?,?,?,?)";
    ArrayList<Parametro> lstParam = new ArrayList<>();
    lstParam.add(new Parametro(1, objLectura.getLecturaAnterior()));
    lstParam.add(new Parametro(2, objLectura.getLecturaActual()));
    lstParam.add(new Parametro(3, objLectura.getLecturaConsumo()));
    lstParam.add(new Parametro(4, new Timestamp(objLectura.getLecturaFechaCorte().getTime())));
    lstParam.add(new Parametro(5, objLectura.getLecturaFechaRegistro()));
    lstParam.add(new Parametro(6, objLectura.getIdClienteServicio()));
    lstParam.add(new Parametro(7, objLectura.getIdUsuarioRegistra()));
    lstParam.add(new Parametro(8, objLectura.getIdImagen()));
    lstParam.add(new Parametro(9, objLectura.getLecturaObservacion()));
    lstParam.add(new Parametro(10, objLectura.getObjEstadoLectura().getEstadoLecturaId()));
    lstParam.add(new Parametro(11, periodoid));
    ConjuntoResultado rs = AccesoDatos.ejecutarQuery(sql, lstParam);
    while (rs.next()) {
        resp = rs.getBoolean(0);
    }

    return resp;
}
```

Figura h-3. Método insertar

Realizado por: Tupac Tisalema, 2019

Ejecutar prueba y verificar que pase

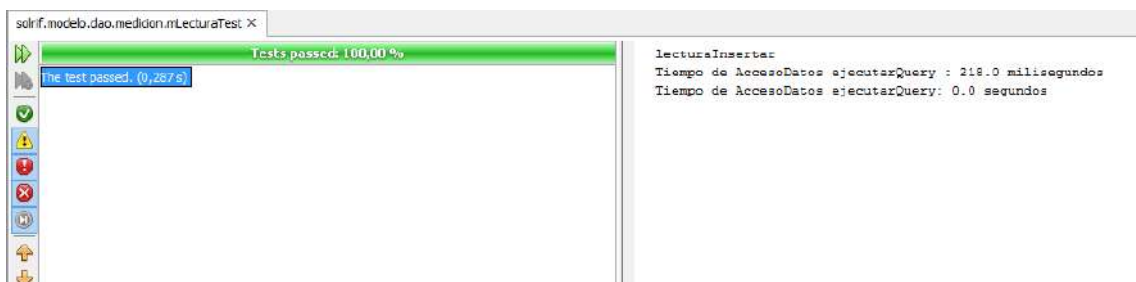


Figura h-4. Test passed

Realizado por: Tupac Tisalema, 2019

Refactorizar

Eliminar duplicidad y hacer mejoras

```
public static boolean insertar(cLectura objLectura, long periodoid) throws Exception {
    boolean resp = false;
    String sql;
    try {
        sql = "SELECT *FROM medicion.fn_insert_lectura(?, ?, ?, ?, ?, ?, ?, ?, ?, ?)";
        ArrayList<Parametro> lstParam = new ArrayList<>();
        lstParam.add(new Parametro(1, objLectura.getLecturaAnterior()));
        lstParam.add(new Parametro(2, objLectura.getLecturaActual()));
        lstParam.add(new Parametro(3, objLectura.getLecturaConsumo()));
        lstParam.add(new Parametro(4, new Timestamp(objLectura.getLecturaFechaCorte().getTime())));
        lstParam.add(new Parametro(5, objLectura.getLecturaFechaRegistro()));
        lstParam.add(new Parametro(6, objLectura.getIdClienteServicio()));
        lstParam.add(new Parametro(7, objLectura.getIdUsuarioRegistra()));
        lstParam.add(new Parametro(8, objLectura.getIdImagen()));
        lstParam.add(new Parametro(9, objLectura.getLecturaObservacion()));
        lstParam.add(new Parametro(10, objLectura.getObjEstadoLectura().getEstadoLecturaId()));
        lstParam.add(new Parametro(11, periodoid));
        ConjuntoResultado rs = AccesoDatos.ejecutarQuery(sql, lstParam);
        while (rs.next()) {
            resp = rs.getBoolean(0);
        }
    } catch (Exception e) {
        throw e;
    }
    return resp;
}
```

Figura h-5: Refactorización

Realizado por: Tupac Tisalema, 2019

Ejecutar prueba y verificar que pase

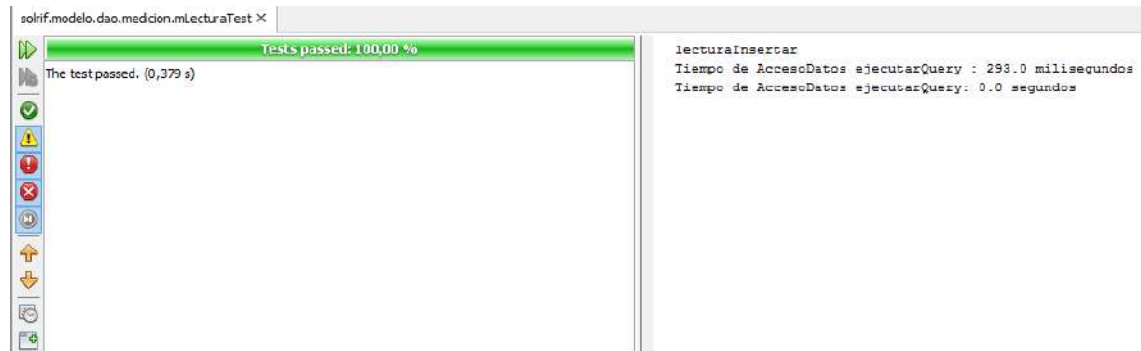


Figura h-1. Fin prueba unitaria

Realizado por: Tupac Tisalema, 2019

Anexo I. Pruebas de Aceptación

Tabla i1. Prueba de Aceptación 01

Prueba de Aceptación	
Código: PA_01	Historia de Usuario: HU-01. Como administrador necesito una interfaz de autenticación para ingresar las credenciales al identificarme en el sistema.
Nombre: Verificar que exista una interfaz de autenticación para ingresar las credenciales.	
Responsable: Tupac Tisalema	Fecha: 25/09/2017
Descripción: Verificar que exista una interfaz de autenticación para ingresar las credenciales de identificación.	
Condiciones de ejecución: <ul style="list-style-type: none">• Disponible la conexión al servidor web.	
Pasos de ejecución: <ol style="list-style-type: none">1. Ejecutar la aplicación web.	
Resultado esperado: Se visualiza la interfaz de autenticación para ingresar las credenciales de identificación.	
Evaluación de la prueba: Exitoso.	

Realizado por: Tupac Tisalema, 2019

Tabla i2. Prueba de Aceptación 02

Prueba de Aceptación	
Código: PA_02	Historia de Usuario: HU-02. Como secretario necesito registrar los valores por cobrar que mantiene los usuarios con las Junta.
Nombre: Verificar que se registre correctamente los valores por cobrar.	
Responsable: Tupac Tisalema	Fecha: 29/09/2017
Descripción: Verificar que cuando se registre correctamente los valores por cobrar de un cliente, se emita un mensaje informativo indicando que los datos se guardaron correctamente.	
Condiciones de ejecución: <ul style="list-style-type: none">• Cliente debe mantener deudas con la Junta.• Usuario debe tener acceso a la funcionalidad de gestionar cliente.	
Pasos de ejecución: <ol style="list-style-type: none">1. Ubicarse en el menú.2. Seleccionar la opción Contribuyente.3. Seleccionar el subítem Gestionar.4. Introducir datos solicitados.5. Seleccionar cliente.6. Pulsar el botón Registro por Cobrar.7. Ingresar los datos solicitados.8. Pulsar el botón Guardar.	
Resultado esperado: Se emite un mensaje informativo indicando que los datos se guardaron correctamente.	
Evaluación de la prueba: Exitoso.	

Realizado por: Tupac Tisalema, 2019

Tabla i3. Prueba de Aceptación 03

Prueba de Aceptación	
Código: PA_03	Historia de Usuario: HU-03. Como administrador necesito autenticarme en el sistema para poder acceder.
Nombre: Verificar que al ingresar las credenciales correctas se permita acceder al sistema.	
Responsable: Tupac Tisalema	Fecha: 06/10/2017
Descripción: Verificar que cuando se ingrese las credenciales correctas de autenticación, se permita el acceder al sistema.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • Usuario debe estar registrado en el sistema. • Usuario debe estar activo. 	
Pasos de ejecución:	
<ol style="list-style-type: none"> 1. Ejecutar aplicación web. 2. Ingresar las credenciales correctas. 3. Pulsar botón “Ingresar”. 	
Resultado esperado: Se accede al sistema.	
Evaluación de la prueba: Exitoso.	

Realizado por: Tupac Tisalema, 2019

Tabla i4. Prueba de Aceptación 04

Prueba de Aceptación	
Código: PA_04	Historia de Usuario: HU-04. Como administrador deseo visualizar una interfaz de bienvenida al acceder al sistema.
Nombre: Verificar que se visualice una interfaz de bienvenida.	
Responsable: Tupac Tisalema	Fecha: 09/10/2017
Descripción: Verificar que se visualice una interfaz de bienvenida al acceder al sistema.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • SN 	
Pasos de ejecución:	
<ol style="list-style-type: none"> 1. Iniciar sesión en el sistema. 	
Resultado esperado: Se visualiza el mensaje de bienvenida al acceder al sistema.	
Evaluación de la prueba: Exitoso.	

Realizado por: Tupac Tisalema, 2019

Tabla i5. Prueba de Aceptación 05

Prueba de Aceptación	
Código: PA_05	Historia de Usuario: HU-05. Como secretario necesito registrar los datos de los usuarios o clientes de la Junta.
Nombre: Verificar que se registre correctamente los datos del cliente.	
Responsable: Tupac Tisalema	Fecha: 13/10/2017
Descripción: Verificar que se registre correctamente los datos del cliente y se emita un mensaje informando que se han guardado correctamente.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> El documento de identificación (Cédula o Ruc) debe ser válida. 	
Pasos de ejecución:	
<ol style="list-style-type: none"> Ubicarse en el menú contribuyente. Desglosar el menú contribuyente. Seleccionar el sub-ítem Persona. Seleccionar el ítem registrar. Seleccionar el tipo de identificación. 	<ol style="list-style-type: none"> Ingresar el número del documento de identificación. Pulsar el botón Procesar Ingresar los datos solicitados. Pulsar el botón guardar.
Resultado esperado: Datos guardados correctamente y emisión de mensaje informativo.	
Evaluación de la prueba: Exitoso.	

Realizado por: Tupac Tisalema, 2019

Tabla i6. Prueba de Aceptación 06

Prueba de Aceptación	
Código: PA_06	Historia de Usuario: HU-06. Como secretario necesito realizar el proceso de adquisición del servicio de agua potable.
Nombre: Verificar que se realice correctamente el proceso de adquisición del servicio.	
Responsable: Tupac Tisalema	Fecha: 20/10/2017
Descripción: Verificar que se realice correctamente el proceso de adquisición del servicio de agua potable y se emita el mensaje indicando que los datos se guardaron correctamente.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> El cliente debe estar registrado en el sistema. Debe existir al menos un servicio registrado en el sistema. Debe existir al menos un sector, categoría y vía registrado en el sistema. 	
Pasos de ejecución:	
<ol style="list-style-type: none"> Ubicarse en el menú contribuyente. Seleccionar el sub-ítem Cliente. Seleccionar el ítem Adquisición. 	<ol style="list-style-type: none"> Buscar cliente por número/identificación. Ingresar los datos solicitados Pulsar el botón Guardar.
Resultado esperado: Se guarda correctamente los datos de adquisición y se emite el mensaje informativo.	
Evaluación de la prueba: Exitoso.	

Realizado por: Tupac Tisalema, 2019

Anexo J. Shapiro-Wilks - Contraste de normalidad

Se analiza el conjunto de datos que se muestran a continuación mediante shaphiro-wilk para determinar si proceden de una población con una distribución normal, utilizando el lenguaje de programación de estadística R como se muestra en el gráfico de este anexo.

150; 180; 210; 240; 270; 180; 180; 210; 180; 180; 198; 198; 198; 207; 212; 214; 217; 221; 225; 229; 150; 180; 180; 200; 215; 180; 180; 260; 150; 150

Para ello se plante las siguientes hipótesis:

H_0 : La muestra procede de una población normal.

H_1 : La muestra no procede de una población normal.



Figura j-1. Contraste de normalidad

Realizado por: Tupac Tisalema, 2019

Como se aprecia en el gráfico anterior se aprecia que el estadístico de contraste (W) tiene un valor a 0.94573, para tamaño de muestra igual 30 y una probabilidad de contraste (p) igual a 0.1298 siendo superior al nivel de significancia 0.05, por lo que se concluye no hay evidencias para rechazar la hipótesis nula y por lo cual se acepta que la muestra procede de una población normal.

Anexo K. Métricas de Seguridad

Métricas para la característica de calidad Seguridad								
Subcaracterística	Métrica	Fase del ciclo de vida de calidad del producto	Propósito de la métrica de calidad	Método de aplicación	Fórmula	Valor deseado	Tipo de medida	Recursos utilizados
Confidencialidad	Capacidad de control de acceso	Interna/Externa	¿Qué tan controlables son los accesos al sistema?	Contar el número de diferentes tipos de operaciones ilegales detectados y el número de tipos de operaciones ilegales en la especificación	$X = A / B$ A = Número de diferentes tipos de operaciones ilegales detectados B = Número de tipos de operaciones ilegales en la especificación Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1, es el mejor	X= Contable/ Contable A= Contable B= Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester
	Encriptación de datos	Interna/Externa	¿Qué tan correctamente es la implementación de encriptación / desencriptación de datos de acuerdo a la especificación de requerimientos	Contar el número de elementos de datos encriptados/ desencriptados correctamente y el número de elementos de datos que requiere el encriptación/	$X = A / B$ A = Número de elementos de datos encriptados/ desencriptados correctamente B = Número de elementos de datos que requiere el	$0 \leq X \leq 1$ El más cercano a 1, es el mejor	X= Contable/ Contable A= Contable B= Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester
			?	desencriptación	encriptación/ desencriptación Dónde: $B > 0$			
Integridad	Prevención de corrupción de datos	Interna/Externa	¿Hasta qué punto se puede prevenir la corrupción de datos?	Contar el número de casos de corrupción de datos ocurridos en la actualidad y el número de accesos donde se espera que ocurran daños de datos	$X = A / B$ A = Número de casos de corrupción de datos ocurridos en la actualidad B = Número de accesos donde se espera que ocurran daños de datos Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 0, es el mejor	X= Contable/ Contable A= Contable B= Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester
No repudio	Utilización de firma digital	Interna/Externa	¿Qué proporción de eventos que requieran no - repudio se procesan utilizando la firma digital?	Contar el número de eventos procesados usando firma digital y el número de eventos que requieran la propiedad de no - repudio	$X = A / B$ A = Número de eventos procesados usando firma digital B = Número de eventos que requieran la propiedad de no - repudio Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1, es el mejor	X= Contable/ Contable A= Contable B= Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester
					$B > 0$			
Responsabilidad	Capacidad de auditoría de acceso	Interna/Externa	¿Qué tan completa es la pista de auditoría en relación al acceso de los usuarios al sistema y a los datos?	Contar el número de accesos al sistema y los datos registrados en el log del sistema y el número de accesos ocurridos en la realidad	$X = A / B$ A = Número de accesos ocurridos en la realidad B = Número de accesos al sistema y los datos registrados en el log del sistema Dónde: $B > 0$	$0 \leq X \leq 1$ El más cercano a 1, es el mejor	X= Contable/ Contable A= Contable B= Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester
Autenticidad	Métodos de autenticación	Interna/Externa	¿Qué tan bien el sistema autentica la identidad de un sujeto o recurso?	Contar el número de métodos de autenticación previstos	$X = A$ A = Número de métodos de autenticación previstos	$X \geq 0$ Donde X es mayor a 0, siendo X el mejor igual o mayor a 2	X= Contable	Especificación de requerimientos, Código fuente, Desarrollador, Tester

Fuente: ISO/IEC 25010

Anexo L. Lineamientos aplicados



**ESCUELA SUPERIOR
POLITECNICA DE
CHIMORAZO**



Encuesta dirigida al administrador de sistemas de la Junta administradora de agua potable "Angahuana Alto" con respecto a la aplicación web/móvil implementada denominada "SOLRIF".

Responda con Verdadero (V) o Falso (F) las siguientes preguntas:

ID	PREGUNTA	RESPUESTA
P1	¿Se requiere que la contraseña posea al menos 8 caracteres?	F
P2	¿Se requiere que la contraseña posea letras mayúsculas y minúsculas?	F
P3	¿Se requiere que la contraseña posea números y letras?	F
P4	¿Se requiere que la contraseña posea caracteres especiales?	F
P5	¿El sistema utiliza conexión segura mediante HTTPS?	F
P6	¿La base de datos posee los datos encriptados?	V
P7	¿El sistema permite acceder a funcionalidades en las cuales no se tiene permiso?	F
P8	¿El sistema permite que cualquier persona tenga acceso a la base de datos?	F
P9	¿El sistema permite que cualquier persona tenga acceso al código del servidor de la aplicación?	F
P10	¿Cualquier persona tiene acceso al servidor físico?	F
P11	¿Cualquier persona tiene acceso al servidor remoto?	F
P12	¿El sistema posee re direccionamientos hacia sitios no seguros?	F
P13	¿El sistema solicita una confirmación de registro mediante un mail a la hora de registrarse?	F
P14	¿El sistema permite que cualquier persona pueda modificar la base de datos?	F
P15	¿El sistema permite que cualquier persona pueda modificar el código del servidor de la aplicación?	F
P16	¿El sistema permite inyecciones SQL?	F
P17	¿El sistema posee un historial de acciones realizadas?	F
P18	¿El sistema posee algoritmos de cifrado de datos?	F
P19	¿El sistema posee un mecanismo criptográfico, como firma digital?	F
P20	¿El sistema realiza eliminación lógica al dar de baja algún registro?	V
P21	¿El sistema posee una protección con certificados SSL?	F

P22	¿El sistema da aviso cuando se accede desde una ubicación desconocida?	F
P23	¿El sistema informa vía mail las operaciones realizadas?	F
P24	¿El sistema guarda fecha, hora, y usuario que almacena algún registro?	V
P25	¿El sistema registra el tipo de navegador y sistema de operación utilizado para ingresar al sitio?	F
P26	¿El sistema registra la dirección IP desde la cual se ingresa al sitio?	F
P27	¿El sistema realiza una comprobación de identidad mediante un certificado digital?	F
P28	¿El sistema posee un sistema de verificación en dos pasos?	F
P29	¿Es requerida una clave de segundo nivel para el ingreso al sistema?	F
P30	¿El sistema realiza una comprobación de identidad mediante datos biométricos?	F
P31	¿El sistema realiza una comprobación de identidad mediante tarjeta de coordenadas?	F
P32	¿El sistema realiza una comprobación de identidad mediante credenciales?	V
P33	¿El sistema realiza una comprobación de identidad mediante una firma electrónica?	F

Firma:  _____
 Administrador de Sistemas



Figura I-1. Lineamientos verificados

Realizado por: Tupac Tisalema, 2019

Anexo M. Obtención de puntaje

Tabla m-1. Puntos obtenidos

ID	Nombre	Descripción	Fórmula	Ptos	Ptos. Obtenidos
C-1	Conexiones Seguras	Una conexión se considera segura si se utiliza HTTPS y si no se tienen redireccionamientos hacia sitios no seguros	$P5 \ \& \ \sim P12 = V$	1	0
C-2	Control de Acceso	Se debe controlar que no se permita acceder a funcionalidades sin autorización, tampoco a la base de datos, al código de la aplicación ni a los servidores, físico ni remoto	$P7 \ \ P8 \ \ P9 \ \ P10 \ \ P11 = F$	1	1
C-3	Encriptación de datos	Los datos de la base de datos deben estar encriptados	$P6 = V$	1	1
C-4	Contraseña de bajo nivel	La contraseña se considera de bajo nivel si posee menos de 8 caracteres, no posee letras mayúsculas y minúsculas, no posee letras y números y no posee caracteres especiales	$P1 \ \ P2 \ \ P3 \ \ P4 = F$	0	0
	Contraseña de medio nivel	La contraseña se considera de medio nivel si posee al menos 8 caracteres o letras mayúsculas y minúsculas o letras y números o	$P1 \ \ P2 \ \ P3 \ \ P4 = V$	0.5	
	Contraseña de alto nivel	La contraseña se considera de alto nivel si posee al menos 8 caracteres, letras mayúsculas y minúsculas, letras y números y caracteres especiales	$P1 \ \& \ P2 \ \& \ P3 \ \& \ P4 = V$	1	
TOTAL				4	2
I-5	Prevención de accesos	Se debe prevenir que no se permita acceder a funcionalidades sin autorización, tampoco a la base de datos ni al código de la aplicación, y que no se permitan inyecciones SQL	$P7 \ \ P8 \ \ P9 \ \ P16 = F$	1	1
I-6	Prevención de modificaciones	Se debe prevenir que no se permita modificar datos de la base de datos ni modificar el código de la aplicación sin autorización	$P14 \ \ P15 = F$	1	1
I-7	Confirmación de datos	Se debe realizar una confirmación de registro por mail	$P13 = V$	1	0
TOTAL				3	2
NR-8	Operaciones realizadas	Se debe poseer un historial de acciones realizadas o las mismas deben ser enviadas por mail	$P17 \ \ P23 = V$	1	1
NR-9	Mecanismos de cifrado	Se debe poseer un algoritmo de cifrado de datos o un mecanismo criptográfico, como firma digital, o una protección con certificados SSL	$P18 \ \ P19 \ \ P21 = V$	1	0
NR-10	Confirmación de acciones	Se debe solicitar una confirmación al realizar una determinada acción	$P20 = V$	1	1

NR-11	Registro de ubicación	Se debe informar si se accedió al sistema desde una ubicación desconocida	P22 = V	1	0
TOTAL				4	2
R-12	Registro de acciones y datos	Se debe poseer un historial de acciones realizadas, o un registro de fecha y hora de ingreso al sistema o de la dirección IP desde la cual se ingresa o del tipo de navegador y sistema de operación utilizado	P17 P24 P25 P26 = V	1	1
R-13	Control de ubicación	Se debe dar aviso cuando se accede al sistema desde una ubicación desconocida	P22 = V	1	0
TOTAL				2	1
A-14	Comprobación de identidad	El sistema debe realizar una comprobación de identidad mediante alguno de los siguientes métodos: datos biométricos, tarjeta de coordenadas, credenciales, firma electrónica o certificado digital	P27 P30 P31 P32 P33 = V	1	1
A-15	Comprobaciones adicionales	Se debe poseer un sistema de verificación en dos pasos, o se debe requerir una clave de segundo nivel para el ingreso al sistema o una confirmación de registro mediante un mail	P28 P29 P13 = V	1	0
TOTAL				2	1

Realizado por: Tupac Tisalema, 2019

Anexo N. Resultados Evaluación

Métrica	Propósito	Método de aplicación	Entradas		Fórmula	Observaciones	Resultado (%)	Aceptación
			A = Puntaje obtenido	B = Puntaje total				
Confidencialidad	¿Cuán eficiente es el sistema a la hora de proteger el acceso de datos e información no autorizados, ya sea accidental o deliberadamente?	Contestar las preguntas de los CE correspondientes a la subcaracterística "Confidencialidad" y calcular la puntuación obtenida, sumando los puntajes de los CE referenciados que cumplan con la meta esperada. "Puntaje total" hace referencia al máximo puntaje que se puede obtener.	2	4	X = A/B	CE a utilizar son: C-1, C-2, C-3 y C-4.	50,00%	Minimamente aceptable
Integridad	¿Cuán capaz es el sistema a la hora de prevenir accesos o modificaciones no autorizados a datos o programas de ordenador?	Contestar las preguntas de los CE correspondientes a la subcaracterística "Integridad" y calcular la puntuación obtenida, sumando los puntajes de los CE referenciados que cumplan con la meta esperada. "Puntaje total" hace referencia al máximo puntaje que se puede obtener.	2	3		CE a utilizar son: I-5, I-6 e I-7.	66,67%	Rango objetivo
No-Repudio	¿Cuán capaz es el sistema de demostrar las acciones o eventos que han tenido lugar, de manera que dichas acciones o eventos no puedan ser repudiados posteriormente?	Contestar las preguntas de los CE correspondientes a la subcaracterística "No-Repudio" y calcular la puntuación obtenida, sumando los puntajes de los CE referenciados que cumplan con la meta esperada. "Puntaje total" hace referencia al máximo puntaje que se puede obtener.	2	4		CE a utilizar son: NR-8, NR-9, NR-10 y NR-11.	50,00%	Minimamente aceptable
Responsabilidad	¿Cuán capaz es el sistema de rastrear de forma inequívoca las acciones de una entidad?	Contestar las preguntas de los CE correspondientes a la subcaracterística "Responsabilidad" y calcular la puntuación obtenida, sumando los puntajes de los CE referenciados que cumplan con la meta esperada. "Puntaje total" hace referencia al máximo puntaje que se puede obtener.	1	2		CE a utilizar son: R-12 y R-13.	50,00%	Minimamente aceptable
Autenticidad	¿Cuán capaz es el sistema de demostrar la identidad de un sujeto o un recurso?	Contestar las preguntas de los CE correspondientes a la subcaracterística "Autenticidad" y calcular la puntuación obtenida, sumando los puntajes de los CE referenciados que cumplan con la meta esperada. "Puntaje total" hace referencia al máximo puntaje que se puede obtener.	1	2		CE a utilizar son: A14 y A15.	50,00%	Minimamente aceptable
Seguridad							53,33%	Minimamente aceptable

Figura n-1. Resultados de la Evaluación

Realizado por: Tupac Tisalema, 2019