

UNIVERSIDAD MAYOR DE SAN ANDRÉS

FACULTAD DE CIENCIAS PURAS Y NATURALES

CARRERA DE INFORMÁTICA



PROYECTO DE GRADO

**SISTEMA WEB DE CONTROL CENTRALIZADO,
INTEROPERABILIDAD Y FIRMA DIGITAL DE BENEFICIARIOS
DE PROGRAMAS SOCIALES ESTATALES DE VIVIENDAS
CASO: VICEMINISTERIO DE VIVIENDA Y URBANISMO**

**PARA OPTAR AL TÍTULO DE LICENCIATURA EN INFORMÁTICA
MENCIÓN: INGENIERÍA DE SISTEMAS INFORMÁTICOS**

**POSTULANTE: PEDRO ARIEL FERNANDEZ ALI
TUTOR METODOLÓGICO: M.SC. ALDO VALDEZ ALVARADO
ASESOR: LIC. VÍCTOR POZO DÍAZ**

LA PAZ – BOLIVIA

2020



**UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA**



LA CARRERA DE INFORMÁTICA DE LA FACULTAD DE CIENCIAS PURAS Y NATURALES PERTENECIENTE A LA UNIVERSIDAD MAYOR DE SAN ANDRÉS AUTORIZA EL USO DE LA INFORMACIÓN CONTENIDA EN ESTE DOCUMENTO SI LOS PROPÓSITOS SON ESTRICTAMENTE ACADÉMICOS.

LICENCIA DE USO

El usuario está autorizado a:

- a) visualizar el documento mediante el uso de un ordenador o dispositivo móvil.
- b) copiar, almacenar o imprimir si ha de ser de uso exclusivamente personal y privado.
- c) copiar textualmente parte(s) de su contenido mencionando la fuente y/o haciendo la referencia correspondiente respetando normas de redacción e investigación.

El usuario no puede publicar, distribuir o realizar emisión o exhibición alguna de este material, sin la autorización correspondiente.

TODOS LOS DERECHOS RESERVADOS. EL USO NO AUTORIZADO DE LOS CONTENIDOS PUBLICADOS EN ESTE SITIO DERIVARA EN EL INICIO DE ACCIONES LEGALES CONTEMPLADOS EN LA LEY DE DERECHOS DE AUTOR.

Dedicatoria

*A mis padres, Vladimir Fernández y Josefa Alí
son mi orgullo, los principales promotores de mis sueños
y el apoyo incondicional de mi vida.*

AGRADECIMIENTOS

Agradezco a Dios por otorgarme la vida y una familia maravillosa.

A mis padres Vladimir y Josefa, que son mis guías en el camino de la vida, por su apoyo y cariño incondicional, por haberme forjado e inculcado tantos valores y principalmente por esas lecciones tan difíciles de ejecutar, después de tanto tiempo... recién comprendo y agradezco su actuar.

A mis hermanos Alejandra y Cesar por crecer conmigo, por cuidar de mí, y por una infinidad de momentos compartidos e inolvidables.

A mi madrina Guísela Fernández, por escucharme y guiarme con sus consejos.

Al Ing. Edwin Yujra, por darme la oportunidad de desarrollar este proyecto, por la paciencia para instruirme y responder a todas mis interrogantes.

Al Ing. Fabianny Vega, por la paciencia que tuvo conmigo y la ayuda que me brindó durante el desarrollo del proyecto.

Al M. Sc. Aldo Valdez por su enseñanza y buen humor en las clases que imparte, por la paciencia, el tiempo y esfuerzo para llevar a cabo correctamente este proyecto.

Al Lic. Víctor Pozo por tomarse el tiempo para analizar mi proyecto, por guiarme y aclarar mis dudas con toda paciencia y buen humor.

A mis mentores deportivos Miguel Burgos y Richard Acarapi, quienes tuvieron un gran impacto en mi vida con sus enseñanzas.

A los que me ayudaron de forma moral y académica en el avance de esta carrera, con quienes compartí momentos memorables, gracias por su amistad y por estar ahí cuando uno lo necesita.

A la vida, por mi ahijado, mis sobrinos y un sinnúmero de experiencias.

RESUMEN

Debido al constante desarrollo de las Tecnologías de Información y las Comunicaciones es necesario que todo sistema tenga acceso rápido y veraz a la información requerida. En la administración pública debido a la gran cantidad de procesos que se maneja, las organizaciones cumplen determinadas funciones para el manejo correcto de la información.

Entre las funciones que tiene el Viceministerio de Vivienda y Urbanismo está el brindar beneficios sociales a la población mediante diferentes instituciones descentralizadas, el presente proyecto tiene como principal objetivo desarrollar un sistema web centralizado e interoperable que maneje e intercambie la información correspondiente a los beneficiarios.

Para el desarrollo de este proyecto se empleó la metodología de desarrollo ágil SCRUM, que tiene un enfoque iterativo e incremental, donde la principal característica es obtener prototipos cuanto antes adaptándose a los cambios en los requisitos del producto.

Se utilizó la metodología de modelado UWE que cuenta con diversos esquemas para la representación de los procesos, para evaluar y medir la calidad del sistema, se empleó Web-Site QEM que se basa en las normas de la ISO 9126.

Finalmente, los objetivos fueron alcanzados satisfactoriamente, por lo que puede decirse que se realizó un producto de calidad, el cual cumple con las necesidades del cliente.

Palabras clave: Interoperabilidad, centralización, firma digital.

ABSTRACT

Due to the constant development of Information Technology and Communications, it is necessary that every system has quick and truthful access to the required information. In public administration due to the large number of processes that are handled, organizations fulfill certain functions for the correct handling of information.

Among the functions of the Vice Ministry of Housing and Urban Planning is to provide social benefits to the population through different decentralized institutions, the main objective of this project is to develop a centralized and interoperable web system that manages and exchanges the corresponding information to the beneficiaries.

For the development of this project the agile development methodology SCRUM was used, which has an iterative and incremental approach, where the main characteristic is to obtain prototypes as soon as possible adapting to the changes in the requirements of the product.

The UWE modeling methodology was used, which has various schemes for the representation of the processes, to evaluate and measure the quality of the system, Web-Site QEM was used based on the ISO 9126 standards.

Finally, the objectives were satisfactorily achieved, so it can be said that a quality product was made, which meets the needs of the client.

Keywords: Interoperability, centralization, digital signature.

ÍNDICE DE CONTENIDOS

CAPÍTULO I MARCO INTRODUCTORIO.....	1
1.1 INTRODUCCIÓN	2
1.2 ANTECEDENTES	3
1.2.1 ANTECEDENTES INSTITUCIONALES	3
1.2.1.1 AEVIVIENDA	3
1.2.1.2 TITULACIÓN	4
1.2.1.3 PROREVI	4
1.2.1.4 PVS	4
1.2.1.5 PROESHA	5
1.2.1.6 UEVE	5
1.2.1.7 3X1000	5
1.2.1.8 ODESUR	6
1.2.2 VISIÓN DE LA INSTITUCIÓN	6
1.2.3 MISIÓN DE LA INSTITUCIÓN	6
1.3 ANTECEDENTES DE PROYECTOS SIMILARES	6
1.4 PLANTEAMIENTO DEL PROBLEMA	7
1.4.1 PROBLEMA CENTRAL	8
1.4.2 PROBLEMAS SECUNDARIOS	8
1.5 DEFINICIÓN DE OBJETIVOS	8
1.5.1 OBJETIVO GENERAL	8
1.5.2 OBJETIVOS ESPECÍFICOS	9
1.6 JUSTIFICACIÓN	9
1.6.1 JUSTIFICACIÓN ECONÓMICA	9
1.6.2 JUSTIFICACIÓN SOCIAL	10
1.6.3 JUSTIFICACIÓN TECNOLÓGICA.....	10
1.7 ALCANCES Y LÍMITES	10
1.7.1 ALCANCES.....	10
1.7.2 LÍMITES	11
1.8 APORTES.....	12

1.8.1 PRÁCTICO	12
1.8.2 TEÓRICO	12
1.9 METODOLOGÍA	12
CAPÍTULO II MARCO TEÓRICO	14
2.1 INGENIERÍA DEL SOFTWARE	15
2.2 METODOLOGÍAS ÁGILES	16
2.3 METODOLOGÍA SCRUM	17
2.3.1 PROCESO DE DESARROLLO DE SCRUM.....	18
2.3.2 FASES DE SCRUM.....	19
2.3.2.1 PRE-GAME.....	19
2.3.2.2 GAME.....	19
2.3.2.3 POST-GAME.....	20
2.3.3 EVENTOS DE SCRUM.....	20
2.3.3.1 SPRINT.....	21
2.3.3.2 REUNIÓN DE PLANIFICACIÓN DEL SPRINT	21
2.3.3.3 SCRUM DIARIO.....	22
2.3.3.4 REVISIÓN SPRINT	22
2.3.3.5 RETROSPECTIVA DEL SPRINT.....	22
2.3.3.6 REFINAMIENTO	23
2.3.4 ROLES DEL SCRUM	23
2.3.4.1 PRODUCT OWNER	23
2.3.4.2 SCRUM MASTER.....	24
2.3.4.3 SCRUM TEAM.....	25
2.3.5 ARTEFACTOS DEL SCRUM.....	25
2.3.5.1 PRODUCT BACKLOG.....	25
2.3.5.2 SPRINT BACKLOG	26
2.3.5.3 INCREMENTO.....	27
2.4 INGENIERÍA WEB	27
2.4.1 INGENIERÍA WEB BASADA EN UML	28
2.4.1.1 ANÁLISIS DE REQUERIMIENTOS	28
2.4.1.2 MODELO CONCEPTUAL	29

2.4.1.3 MODELO NAVEGACIONAL	30
2.4.1.4 MODELO DE PRESENTACIÓN	31
2.5 INTEROPERABILIDAD	32
2.5.1 INTEROPERABILIDAD SEMÁNTICA.....	33
2.5.2 INTEROPERABILIDAD ORGANIZACIONAL	33
2.5.3 INTEROPERABILIDAD TÉCNICA.....	33
2.5.4 GOBERNANZA DE INTEROPERABILIDAD	33
2.6 SERVICIO WEB.....	34
2.6.1 COMPONENTES DE LOS SERVICIOS WEB	34
2.6.1.1 XML.....	35
2.6.1.2 SOAP.....	35
2.6.1.3 WSDL.....	36
2.6.1.4 UDDI.....	36
2.6.1.5 JSON.....	36
2.6.2 SERVICIO WEB SOAP.....	36
2.6.3 SERVICIO WEB REST	37
2.7 FIRMA DIGITAL.....	38
2.7.1 CARACTERÍSTICAS	38
2.7.1.1 CLAVE ASIMÉTRICA	39
2.7.1.2 HASH	39
2.8 RECURSOS PARA DESARROLLO WEB	40
2.8.1 BOOTSTRAP.....	40
2.8.2 LARAVEL	41
2.8.3 POSTGRESQL	41
CAPÍTULO III MARCO APLICATIVO	42
3.1 INTRODUCCIÓN	43
3.2 DESARROLLO DE LOS EVENTOS DE SCRUM.....	43
3.2.1 PREGAME.....	43
3.2.1.1 REUNIONES	43
3.2.1.2 PRODUCT BACKLOG.....	44
3.2.2 GAME.....	45

3.2.2.1 PLANIFICACIÓN DEL SPRINT: SPRINT 1	45
3.2.2.2 PLANIFICACIÓN DEL SPRINT: SPRINT 2	49
3.2.2.3 PLANIFICACIÓN DEL SPRINT: SPRINT 3	53
3.2.2.4 PLANIFICACIÓN DEL SPRINT: SPRINT 4	58
3.2.2.5 PLANIFICACIÓN DEL SPRINT: SPRINT 5	62
3.2.3 POSTGAME	67
3.2.3.1 PRUEBA DE STRESS	67
CAPÍTULO IV CALIDAD Y SEGURIDAD.....	69
4.1 INTRODUCCIÓN	70
4.2 FACTORES DE CALIDAD.....	70
4.3 WEBSITE QEM.....	70
4.3.1 CARÁCTERISTICAS QEM.....	71
4.3.1.1 USABILIDAD	71
4.3.1.2 FUNCIONALIDAD.....	74
4.3.1.3 CONFIABILIDAD.....	76
4.3.1.4 EFICIENCIA.....	77
4.3.2 RESULTADOS.....	78
4.4 SEGURIDAD DE SISTEMA.....	79
4.4.1 AUTENTIFICACIÓN.....	79
4.4.2 SOLICITUDES DE FALSIFICACIÓN.....	79
4.4.3 INYECCIÓN SQL	80
4.4.4 ATAQUE XSS	81
CAPÍTULO V ANÁLISIS COSTO BENEFICIO.....	82
5.1 INTRODUCCIÓN	83
5.2 COCOMO II.....	83
5.2.1 COSTOS DE DESARROLLO DE SOFTWARE.....	83
5.2.2 COSTOS DE ELABORACIÓN DEL SISTEMA.....	86
5.2.3 COSTOS DE IMPLEMENTACIÓN DEL SISTEMA.....	86
5.2.4 COSTO TOTAL DEL SISTEMA	86
5.3 CALCULO BENEFICIO.....	86
5.3.1 VALOR ACTUAL NETO.....	87

5.3.2 TASA INTERNA DE RETORNO.....	88
5.4 COSTO / BENEFICIO.....	89
CAPÍTULO VI CONCLUSIONES Y RECOMENDACIONES	90
6.1 CONCLUSIONES.....	91
6.2 RECOMENDACIONES.....	92
BIBLIOGRAFÍA.....	93
ANEXOS	
ANEXO A - ÁRBOL DE PROBLEMAS	
ANEXO B - ÁRBOL DE OBJETIVOS	
ANEXO C – MARCO LÓGICO	

ÍNDICE DE FIGURAS

FIGURA 2.1 CAPAS DE LA INGENIERÍA DEL SOFTWARE.....	16
FIGURA 2.2 CICLO DE PROCESO DE DESARROLLO DE SCRUM	18
FIGURA 2.3 FASES DE SCRUM.....	19
FIGURA 2.4 PILA DE PRODUCTO	26
FIGURA 2.5 PILA DE SPRINT.....	27
FIGURA 2.6 MODELO DE CASOS DE USO.....	29
FIGURA 2.7 MODELO CONCEPTUAL	29
FIGURA 2.8 MODELO DE NAVEGACIÓN	30
FIGURA 2.9 MODELO DE PRESENTACIÓN.....	31
FIGURA 2.10 EJEMPLO DE INTEROPERABILIDAD	32
FIGURA 2.11 SERVICIO WEB	35
FIGURA 2.12 SERVICIO WEB SOAP.....	37
FIGURA 2.13 SERVICIO WEB REST	38
FIGURA 2.14 COMPROBACIÓN DE FIRMA DIGITAL	39
FIGURA 3.1 METODOLOGÍA SCRUM Y UWE.....	43
FIGURA 3.2 SPRINT 1 ANÁLISIS DE REQUERIMIENTOS	45
FIGURA 3.3 SPRINT 1 MODELO CONCEPTUAL	46
FIGURA 3.4 SPRINT 1 MODELO NAVEGACIONAL	46
FIGURA 3.5 SPRINT 1 MODELO DE PRESENTACIÓN	47
FIGURA 3.6 SPRINT 1 INICIO DE SESIÓN.....	47
FIGURA 3.7 SPRINT 1 PANEL DE ADMINISTRACIÓN.....	48
FIGURA 3.8 SPRINT 2 ANÁLISIS DE REQUERIMIENTOS	49
FIGURA 3.9 SPRINT 2 MODELO CONCEPTUAL	50
FIGURA 3.10 SPRINT 2 MODELO NAVEGACIONAL	50
FIGURA 3.11 SPRINT 2 MODELO DE PRESENTACIÓN.....	51
FIGURA 3.12 SPRINT 2 CREAR PERFIL.....	52
FIGURA 3.13 SPRINT 2 EDICIÓN DE PERFIL.....	52
FIGURA 3.14 SPRINT 2 ELIMINAR PERFIL	53
FIGURA 3.15 SPRINT 3 ANÁLISIS DE REQUERIMIENTOS.....	54

FIGURA 3.16 SPRINT 3 MODELO CONCEPTUAL.....	55
FIGURA 3.17 SPRINT 3 MODELO NAVEGACIONAL.....	55
FIGURA 3.18 SPRINT 3 MODELO DE PRESENTACIÓN.....	56
FIGURA 3.19 SPRINT 3 FORMULARIO DE BÚSQUEDA.....	56
FIGURA 3.20 SPRINT 3 RESULTADOS DE BÚSQUEDA.....	57
FIGURA 3.21 SPRINT 3 BÚSQUEDA POR TIPO DE DATO.....	57
FIGURA 3.22 SPRINT 4 ANÁLISIS DE REQUERIMIENTOS.....	59
FIGURA 3.23 SPRINT 4 MODELO CONCEPTUAL.....	59
FIGURA 3.24 SPRINT 4 MODELO NAVEGACIONAL.....	60
FIGURA 3.25 SPRINT 4 MODELO DE PRESENTACIÓN.....	60
FIGURA 3.26 SPRINT 4 REPORTE POR DEPARTAMENTO.....	61
FIGURA 3.27 SPRINT 4 REPORTE POR GESTIÓN.....	61
FIGURA 3.28 SPRINT 5 ANÁLISIS DE REQUERIMIENTOS.....	63
FIGURA 3.29 SPRINT 5 MODELO CONCEPTUAL.....	63
FIGURA 3.30 SPRINT 5 MODELO NAVEGACIONAL.....	64
FIGURA 3.31 SPRINT 5 MODELO DE PRESENTACIÓN.....	64
FIGURA 3.32 SPRINT 5 CERTIFICACIÓN VÁLIDA.....	65
FIGURA 3.33 SPRINT 5 CERTIFICACIÓN INVÁLIDA.....	65
FIGURA 3.34 BASE DE DATOS.....	67
FIGURA 3.35 PRUEBA DE STRESS.....	68
FIGURA 4.1 REGISTRO DE USUARIO.....	79
FIGURA 4.2 IMPLEMENTACIÓN DE TOKEN CSRF.....	80
FIGURA 4.3 BINDEO DE PARÁMETROS.....	80
FIGURA 4.4 IMPLEMENTACIÓN DE SINTAXIS.....	81
FIGURA 5.1 USC-COCOMO II 200.4.....	84
FIGURA 5.2 ASPECTOS DE LOS PUNTOS FUSIÓN.....	84
FIGURA 5.3 RESULTADOS OBTENIDOS – COCOMO II.....	85

ÍNDICE DE TABLAS

TABLA 3.1 PRODUCT BACKLOG.....	44
TABLA 3.2 SPRINT 1 BACKLOG.....	45
TABLA 3.3 SPRINT 1 PRUEBA UNITARIA.....	48
TABLA 3.4 SPRINT 2 BACKLOG.....	49
TABLA 3.5 SPRINT 2 PRUEBA UNITARIA.....	53
TABLA 3.6 SPRINT 3 BACKLOG.....	54
TABLA 3.7 SPRINT 3 PRUEBA UNITARIA.....	58
TABLA 3.8 SPRINT 4 BACKLOG.....	58
TABLA 3.9 SPRINT 4 PRUEBA UNITARIA.....	62
TABLA 3.10 SPRINT 5 BACKLOG.....	62
TABLA 3.11 SPRINT 5 PRUEBA UNITARIA.....	66
TABLA 4.1 EVALUACIÓN DE COMPRESIBILIDAD	71
TABLA 4.2 EVALUACIÓN DE MECANISMOS Y RETROALIMENTACIÓN.....	72
TABLA 4.3 EVALUACIÓN DE ASPECTOS DE INTERFAZ Y ESTÉTICOS.....	73
TABLA 4.4 EVALUACIÓN DE MISCELÁNEAS.....	73
TABLA 4.5 EVALUACION TOTAL DE USABILIDAD	73
TABLA 4.6 EVALUACION DE BÚSQUEDA Y RECUPERACIÓN.....	74
TABLA 4.7 EVALUACION DE NAVEGACIÓN Y EXPLORACIÓN	75
TABLA 4.8 EVALUACION DE DOMINIO ORIENTADO AL USUARIO	75
TABLA 4.9 EVALUACION TOTAL DE FUNCIONALIDAD.....	76
TABLA 4.10 EVALUACION DE CONFIABILIDAD.....	76
TABLA 4.11 EVALUACIÓN DE DESEMPEÑO	77
TABLA 4.12 EVALUACIÓN DE ACCESIBILIDAD	77
TABLA 4.13 EVALUACIÓN TOTAL DE EFICIENCIA.....	78
TABLA 4.14 EVALUACIÓN TOTAL DE CALIDAD.....	78
TABLA 5.1 DATOS INTRODUCIDOS EN COCOMO.....	85
TABLA 5.2 COSTOS DE ELABORACION DEL SISTEMA	86
TABLA 5.3 COSTO TOTAL DEL SOFTWARE	86
TABLA 5.4 CANTIDAD NOMINAL POR AÑO.....	87

CAPÍTULO I

MARCO INTRODUCTORIO

1.1 INTRODUCCIÓN

Cualquier negocio a lo largo de su trayectoria, genera una ingente cantidad de información. Si tales cantidades de datos no se gestionan de manera adecuada, surgen importantes contratiempos. Uno de los principales problemas de las empresas a la hora de controlar sus datos es la falta de organización, utilizando diferentes sistemas para almacenar y gestionar la información. Es más que frecuente incurrir en el error humano cuando la información no está dispuesta de una manera ordenada y coherente. La accesibilidad a los datos también se dificulta, y como consecuencia, la inversión de tiempo y recursos tan solo en la gestión de la información, puede llegar a ser exagerada. Por esta razón lo más recomendable es centralizar toda la información de una empresa en una misma base de datos, las bases de datos centralizadas brindan la posibilidad de generar reglas automatizadas para acciones o tareas teniendo en cuenta los datos previamente ingresados en ellas. (Horus, 2018)

Un sistema centralizado tiene una enorme cantidad de beneficios como la integridad de datos, una de las reglas principales en el diseño y creación de una base de datos es que no se permite la redundancia, es decir, ninguna pieza de los datos debe repetirse dentro de la base de datos. Como beneficio de un sistema centralizado es que la curva de aprendizaje para otros usuarios se reduce considerablemente, si todos los procesos están en una misma base de datos los usuarios solo tienen que aprender un sistema, no múltiples sistemas. (Yunbit, 2016)

El Ministerio de Obras Públicas, Servicios y Vivienda (MOPSV) es una institución moderna, sólida y transparente que tiene como objeto promover y gestionar el acceso universal y equitativo de la población boliviana a obras y servicios de calidad, en telecomunicaciones, transportes y vivienda, a través de políticas, normas, programas y proyectos que contribuyan al desarrollo socioeconómico. (OOPP, 2016)

Tiene 3 viceministerios bajo tuición: Viceministerio de Transportes, Viceministerio de Telecomunicaciones y Viceministerio de Vivienda y Urbanismo en el cual existen programas sociales que contienen una gran cantidad de información, y en estos programas la información no está organizada de forma eficiente.

Cada uno de estos programas sociales posee sus propios sistemas de información y no es posible acceder de una manera directa a sus datos, por todo lo mencionado anteriormente se formula el

presente proyecto de grado cuyo fin es la implementación de un sistema centralizado que tenga acceso a toda la información relacionada a los beneficiarios de estos programas sociales.

1.2 ANTECEDENTES

Un programa social es una iniciativa destinada a mejorar las condiciones de vida de la población. Se entiende que un programa de este tipo está orientado a la totalidad de la sociedad o, al menos, a un sector importante que tiene ciertas necesidades aún no satisfechas.

La mayoría de los programas sociales están desarrollados por el Estado, que tiene la responsabilidad de atender las necesidades de todas las personas. Un gobierno, de este modo, puede colocar en marcha planes que busquen garantizar el acceso a la educación, campañas de prevención para cuidar la salud o iniciativas para combatir la desnutrición infantil.

En este caso hay muchos y variados programas sociales que, en estos momentos, están vigentes en las ciudades de todo el mundo. Así, por ejemplo, se han impulsado desde los relativos a la vivienda, para que cualquier ciudadano pueda tener una vivienda digna, hasta los que se refieren a la educación. (Pérez & Merino, 2013)

1.2.1 ANTECEDENTES INSTITUCIONALES

El Viceministerio de Vivienda y Urbanismo dependiente del Ministerio de Obras Públicas, Servicios y Vivienda, tiene varios programas sociales que se encuentran en ejecución, como ser la unidad ejecutora de titulación (UET), el programa de regularización de derecho propietario sobre vivienda (PROREVI), la agencia estatal de vivienda (AEV), además de programas que ya fueron cerrados pero que se conserva la información como ser el Programa de vivienda social y solidaria (PVS), Programa especial de soluciones habitacionales para personas con discapacidad (PROESHA), Unidad ejecutora 3x1000, Unidad ejecutora para viviendas de emergencia (UEVE) y la Organización deportiva suramericana. (VIVIENDA, 2015)

1.2.1.1 AEVIVIENDA

La Agencia Estatal de Vivienda (AEVIVIENDA) es una institución pública descentralizada y especializada en reducir el déficit habitacional, facilitando el derecho al acceso a una vivienda

adecuada y asequible a los hogares bolivianos, ejecutando programas integrales concurrentes que construyen equidad social y calidad de vida.

Al 2020 es una institución pública de excelencia socialmente reconocida que formula, lidera, coordina y ejecuta la política boliviana de vivienda social, hábitat y territorio, en el marco de una convivencia comunitaria en armonía con la Madre Tierra. (AEVIVIENDA, 2017)

1.2.1.2 TITULACIÓN

La Unidad Ejecutora de Titulación (UET) es una entidad que con calidad y transparencia ofrece servicios de regularización de derecho propietario de soluciones habitacionales a familias de escasos recursos, recupera cartera en mora, aportes devengados y concluye tareas pendientes del ex FONVIS (Fondo Nacional de Vivienda Social). Tiene como objetivo concluir todas las tareas pendientes del ex FONVIS en liquidación, para lo cual procesará la documentación e información que permita identificar los procesos judiciales, cartera a recuperar, saneamientos pendientes, minutaciones y otros que pudieren existir. (TITULACION, 2011)

1.2.1.3 PROREVI

El Programa de Regularización del Derecho Propietario Sobre Bienes Inmuebles Urbanos Destinados a Vivienda (PROREVI) fue creada bajo tuición del Ministerio de Obras Públicas, Servicios y Vivienda con el objetivo de facilitar la regularización del derecho propietario de personas naturales que se encuentren en posesión continua, publica, pacífica y de buena fe, de un bien inmueble destinado a vivienda, ubicada dentro del radio urbano o área urbana.

Además de dar cumplimiento a lo establecido en la Constitución Política del Estado, donde toda persona tiene derecho a una vivienda digna y a la propiedad privada. (PROREVI, 2017)

1.2.1.4 PVS

El Programa de Vivienda Social y Solidaria (PVS) es un programa cerrado, que estuvo vigente en el periodo 2006 al 2014 con la misión de atender las necesidades habitacionales requeridas por la población de menores ingresos y a la población que fue afectada por fenómenos naturales adversos, asegurando equidad, transparencia y eficiencia en la administración de los aportes para la vivienda y los recursos públicos.

Se convirtió en una entidad institucionalizada, altamente especializada en proporcionar soluciones habitacionales a la población de menores ingresos, que logro disminuir el déficit cuantitativo y cualitativo de viviendas y logro reconstruir y refaccionar las viviendas afectadas por fenómenos naturales adversos. (VIVIENDA, 2015)

1.2.1.5 PROESHA

El Programa Especial de Soluciones Habitacionales para Personas con Discapacidad (PROESHA) vigente en la gestión 2010 tuvo como objetivo mejorar la calidad de vida de las personas con discapacidad mediante el mejoramiento, ampliación y reconstrucción de vivienda eliminando las barreras físicas y arquitectónicas en su vivienda.

Además de identificar y clasificar las características de la demanda de vivienda del grupo de personas con discapacidad y permitir el acceso a la mejora, ampliación y reconstrucción de la vivienda en áreas rurales del país, a hogares de personas con discapacidad en los casos de extrema pobreza y alto grado de vulnerabilidad. (VIVIENDA, 2015)

1.2.1.6 UEVE

La Unidad Ejecutora para Viviendas de Emergencia (UEVE) es una entidad cerrada, que estuvo vigente en el periodo 2011 al 2014 con el objetivo de viabilizar la dotación de soluciones habitacionales, hábitat y equipamiento a la población afectada por desastres naturales ocasionados por el fenómeno de La Niña 2010-2011 a nivel nacional.

Cabe recalcar que muchas de las soluciones habitacionales fueron construidas en sitios que por primera vez incursionó al Estado con alguna obra. (VIVIENDA, 2015)

1.2.1.7 3X1000

La Unidad Ejecutora 3x1000, con dependencia funcional del Viceministerio de Vivienda y Urbanismo, tuvo por misión establecer condiciones, requisitos y procedimientos para la elaboración de la lista de beneficiarios y la compra de lotes de terreno para la devolución de los aportes efectuados por los Cooperativistas Mineros, Locatarios, Veneros y Lameros en el periodo comprendido entre 1970 y 1991, previo cumplimiento de los requisitos establecidos en Reglamento. (VIVIENDA, 2015)

1.2.1.8 ODESUR

La Organización Deportiva Suramericana (ODESUR) creada en la gestión 2015 fue una unidad con la misión de llevar adelante la ejecución del Estadio para la ciudad de Cochabamba, ejecutando las acciones técnicas, administrativas y legales que fueran necesarias para la consecución del proyecto.

Además de velar en todo momento por la calidad técnica de la obra y su ejecución según plazo y presupuesto, aplicando las técnicas más modernas, herramientas y personal calificado para el control, seguimiento y monitoreo de proyectos. (VIVIENDA, 2015)

1.2.2 VISIÓN DE LA INSTITUCIÓN

En una corta trayectoria de tiempo se convirtió en una entidad institucionalizada, altamente especializada en proporcionar soluciones habitacionales a la población de menores ingresos, que ha logrado disminuir el déficit cuantitativo y cualitativo de viviendas y logró reconstruir y refaccionar las viviendas afectadas por fenómenos naturales adversos. (VIVIENDA, 2015)

1.2.3 MISIÓN DE LA INSTITUCIÓN

Atender las necesidades habitacionales requeridas por los sectores de la población de menores ingresos económicos y a la población que ha sido afectada por fenómenos naturales adversos, asegurando equidad, transparencia y eficiencia en la administración de los aportes para la vivienda y los recursos públicos. (VIVIENDA, 2015)

1.3 ANTECEDENTES DE PROYECTOS SIMILARES

En la carrera de informática de la Universidad Mayor de San Andrés (UMSA) se encontró algunos proyectos similares relacionados al control y centralización de información, dichos proyectos están orientados a la automatización de procesos manuales.

Se encontró los siguientes proyectos de grado:

- Centro de Documentación Virtual Caso: Facultad de Agronomía de la UMSA, realizado por Rudy Ruiz Cadena, en el año 2014, tiene como objetivo crear medios que permitan al usuario tener acceso a la documentación generada en la Facultad de Agronomía además de brindar toda la accesibilidad posible a la base de datos para su respectiva consulta y generar

respaldos cada cierto tiempo para evitar la pérdida de los datos administrados por el Centro de Documentación, para el desarrollo utiliza la metodología XP y para su diseño conceptual WebML, como referencia se utilizó los estándares de calidad ISO 9126 el cual es un estándar internacional para la evaluación de la calidad del software.

- Sistema Centralizado de Información Teatral Caso: Taller de Artes Escénicas TAE-UMSA, realizado por Estanislao Guido Paco Ramos, en el año 2017, tiene como objetivo implementar un sistema de información de las actividades que realiza el taller de Artes Escénicas y detallar la información de los encargados de elaborar cada una de las obras, y del control de inventario y presupuesto del taller, para el desarrollo utiliza la metodología OpenUp, para el análisis y diseño utilizo una herramienta que proporciona una serie de diagramas que facilitan el desarrollo del sistema, denominado UWE.
- Sistema Web de Control de Inventarios, Ventas de Productos y Mantenimiento de Maquinaria Basado en la Arquitectura SCM Caso: Convertidora de Fibras Grisel COFIGRIL Ltda. realizado por Telassim Ginnola Gómez Jiménez, en el año 2018, tiene como objetivo desarrollar un sistema web de tal manera que tenga un control adecuado de todo el movimiento dentro de la empresa mejorando el contacto con los clientes y ofreciéndoles un servicio más eficiente, para el desarrollo utiliza la metodología XP, complementando con la metodología de modelado WebML.

1.4 PLANTEAMIENTO DEL PROBLEMA

Actualmente en algunas organizaciones se tiene los datos repartidos en diferentes sistemas de información, esto conlleva una serie de problemas derivados de la falta de trabajar bajo un mismo mando central, este tipo de organizaciones tiene muchos empleados que realizan muchas tareas y deben esperar a que otros acaben un proceso para ellos continuar con el mismo, esto genera que los procesos demoren más de la cuenta, existe falta de comunicación y coordinación entre distintas áreas. (Datatec, 2017)

Según observaciones realizadas en el Viceministerio de Vivienda y Urbanismo existen varios programas sociales con sus respectivos sistemas de información, y para la coordinación y generación de reportes se atraviesa un proceso largo de solicitudes los cuales no son atendidos en su momento haciendo que el proceso demore más de lo estimado.

1.4.1 PROBLEMA CENTRAL

¿Cómo controlar, centralizar y certificar la información de los beneficios otorgados por los programas sociales estatales de viviendas en el Viceministerio de Vivienda y Urbanismo?

1.4.2 PROBLEMAS SECUNDARIOS

- La información básica de algunos beneficiarios se encuentra incompleta generando problemas a la hora de sacar reportes.
- Debido a que se tienen diferentes programas sociales cada uno tiene su propia base de datos, por tanto, la información de algunos beneficiarios que se postulan a todos los programas se encuentra duplicada.
- Al hacer la solicitud de los servicios web a las diferentes unidades, debido a la carga de actividades que tiene el personal, no se brinda la prioridad que requiere el proceso perjudicando su avance.
- La información de los beneficiarios que se obtiene de los servicios web no están en la estructura que se solicita y para que se pueda hacer una entrega de este formato se tiene que hacer un trabajo adicional para cumplir con la solicitud.
- No se tiene la manera de garantizar que una persona tenga solo un beneficio, dándose el caso de que pueda acceder o posea más de un beneficio, lo cual no está permitido.
- El trámite que hace la población en general para poder acceder a un beneficio, pasa por un papeleo en los diferentes programas para su posterior comprobación y certificación, lo que lleva mucho tiempo para realizar esta consulta.
- A pesar de tener mucha información de los beneficiarios, no se cuenta con detalles generales, como ser la cantidad de beneficiarios por gestión, por departamento y por programa.

1.5 DEFINICIÓN DE OBJETIVOS

1.5.1 OBJETIVO GENERAL

Desarrollar un sistema web centralizado interoperable que controle y certifique la información de los beneficios que otorgan los programas sociales estatales de viviendas dependientes del Viceministerio de Vivienda y Urbanismo.

1.5.2 OBJETIVOS ESPECÍFICOS

- Crear mecanismos que, al momento de encontrar campos vacíos de información, genere una respuesta alternativa por defecto.
- Implementar consultas que encuentren y reporten sobre casos en los que se encuentre duplicidad de información.
- Proporcionar permisos de usuario al personal respectivo que lo solicite para que puedan realizar sus procesos pendientes.
- Diseñar un esquema de solicitudes que obtenga la información de los servicios web y las convierta a un formato estandarizado.
- Generar múltiples campos de verificación sobre la información de un beneficiario o posible beneficiario y reporte sobre casos en los que exista más de un beneficio.
- Proporcionar un formulario de consulta de beneficiarios, en el que la población ya no tenga que acudir a las oficinas del Viceministerio de Viviendas y Urbanismo y tener que hacer todo el trámite de manera presencial, sino que pueda saber directamente vía web si tiene algún beneficio.
- Generar reportes que muestren información sobre la cantidad de beneficiarios por gestión, por departamento y por programa.

1.6 JUSTIFICACIÓN

1.6.1 JUSTIFICACIÓN ECONÓMICA

Debido a la necesidad de tener una garantía referente a que solo debe existir un beneficio por persona directa o indirectamente y que, por el contrario, al existir más de un beneficio por persona se estaría ocasionando daño económico al estado, con el desarrollo del presente proyecto se tendrá un método de verificación veloz y confiable, que encuentre a personas que tengan más de un beneficio y a la vez que prevenga de posibles postulantes que quieran acceder a un beneficio, ahorrando inversiones innecesarias al estado.

El desarrollo del proyecto no tendrá costo alguno, debido a que se emplearán herramientas de software libre y código abierto, aprovechando en gran manera los recursos para brindar al Viceministerio de Viviendas y Urbanismo un producto de calidad que cumpla con todas sus exigencias y requerimientos.

1.6.2 JUSTIFICACIÓN SOCIAL

El proyecto a implementarse beneficiará de gran manera al Viceministerio de Vivienda y Urbanismo y a todo el personal encargado del manejo de los programas sociales, debido a que, existirá una mejora en las solicitudes entre programas, el proceso de comprobación se mecanizará de manera que ya no se tenga que hacer las solicitudes de servicios de forma verbal o escrita, sino que sea automática, evitando el desgaste humano, ayudando al personal a aumentar su nivel productivo.

Al tener un control ordenado y preciso sobre los beneficiarios que se tienen en los programas sociales y con la ayuda del formulario de consulta web que se encargará de brindar información inmediata, se podrá evitar procesos que agotan a la población, como ser el venir a las oficinas de la institución, hacer filas para la atención, esperar por el resultado de si puede acceder o no a los beneficios.

1.6.3 JUSTIFICACIÓN TECNOLÓGICA

Actualmente el Viceministerio de Vivienda y Urbanismo cuenta con las herramientas tecnológicas necesarias para la implementación del sistema web propuesto, lo que permitirá dar mejor uso a los equipos de computación con los que se cuenta. El sistema ayudará con nuevos servicios que hasta ahora no se encuentran automatizados y se encargará de hacer una verificación exhaustiva de la información de los beneficiarios. Para el desarrollo se empleará el lenguaje de programación PHP junto al *framework* Laravel, el gestor de base de datos PostgreSQL y para la interfaz gráfica se hará uso de Bootstrap con HTML5.

1.7 ALCANCES Y LÍMITES

1.7.1 ALCANCES

Basado en las necesidades de la institución, el sistema contendrá los siguientes alcances:

- Módulo de Autenticación y Roles, en el cual dependiendo del tipo de rol que tenga el usuario, podrá tener acceso a diferentes áreas del sistema.
- Módulo de Administración, donde se encontrarán las funcionalidades referentes a la creación, edición, asignación y eliminación de perfiles, roles, y datos personales de los usuarios.

- Módulo de Búsqueda, donde se podrá verificar y filtrar la información de los beneficiarios por diferentes atributos y también en este módulo se encontrarán beneficiarios que tengan algún tipo de observación.
- Módulo de Servicios, que hace referencia a la conexión de los servicios web que se usará con los distintos programas sociales, tanto en el caso de consumir servicios como de brindarlos y se dará información específica sobre los beneficiarios.
- Módulo de Estadísticas, donde se ofrecerá información general de forma gráfica, referente a los beneficiarios por programa, por gestión, por departamento, etc.
- Módulo de Auditoría y Seguridad, en el cual se guardará en la base de datos todo movimiento que se realice en la parte de administración, además de proteger de posibles vulnerabilidades y ataques al sistema.
- Módulo de Reportes, para la impresión de los resultados que se requiera de los beneficiarios, con los debidos códigos de verificación.

1.7.2 LÍMITES

Los límites para el desarrollo del proyecto son:

- El servicio proporcionado por este sistema será exclusivamente del Viceministerio de Viviendas y Urbanismo y no así de otros Viceministerios que son dependientes del Ministerio de Obras Públicas, Servicios y Vivienda.
- Únicamente usuarios asignados por el administrador del sistema tendrán un acceso a la información general de los beneficiarios, que será temporal y dependiendo de los permisos que se le asignen.
- El sistema no almacena imágenes referentes al lugar o algún documento escaneado del beneficiario, se limita a guardar información relevante (de forma literal) de los mismos.
- El sistema se encargará de manejar solo la información correspondiente a los beneficiarios, y no de las demás partes que puedan existir en los programas sociales.
- La interacción con el sistema se limitará al idioma español.
- El formulario de consulta web será solamente para mostrar información respecto a si una persona es beneficiaria o no, y el documento que emite, no tiene validez alguna, solo es de comprobación.

1.8 APORTES

1.8.1 PRÁCTICO

El sistema web contribuirá de manera significativa a la institución, debido a que permite ofrecer un servicio eficaz a los usuarios al realizar las consultas respectivas de los beneficiarios, contando con información correcta y precisa al momento de solicitar servicios, también cambiara la forma en que se ha operado hasta el momento, el sistema contribuirá con una base de datos centralizada de manera que se pueda compartir información requerida entre otras organizaciones pertenecientes al Viceministerio de Vivienda y Urbanismo.

1.8.2 TEÓRICO

El proyecto a ser diseñado se espera sea una referencia futura para sistemas orientados a la centralización de datos, esto debido a que para el desarrollo del mismo se hará uso de la metodología SCRUM que es una metodología ágil y flexible adaptada para gestionar soluciones digitales, principalmente el desarrollo de software.

Paralelamente se tomarán en cuenta ciertas herramientas de ingeniería web basada en UML o UWE (UML - Based Web Engineering) que provee especificaciones gráficas formales para un proceso de diseño completo que puede ser asistido por herramientas de diseño visuales.

1.9 METODOLOGÍA

La metodología a utilizar será la del Método Científico, el cual es un método de estudio sistemático de la naturaleza que incluye las técnicas de observación, reglas para el razonamiento y la predicción, ideas sobre la experimentación planificada y los modos de comunicar los resultados experimentales y teóricos, es también afectado naturalmente por los prejuicios cognitivos ya que los efectos asociativos de nuestra mente son los que permiten, al mismo tiempo, lanzar el mayor número de hipótesis . Sin embargo, el método, si es bien ejecutado en sus últimos y más importantes pasos, permite desecharlas, y no solo debe ser el hecho probado por la experimentación directa, sino que debe ser posible repetirlo.

Los aspectos de la investigación que se tomaran en cuenta son la investigación exploratoria y la investigación descriptiva; a la exploratoria se la considera el primer acercamiento científico a un problema. Se utiliza cuando este aún no ha sido abordado o no ha sido suficientemente estudiado

y las condiciones existentes no son aun determinantes, mientras que la descriptiva se efectúa cuando se desea describir, en todos sus componentes principales, una realidad. (Tamayo, 2003)

Para el desarrollo del sistema se hará uso de la metodología ágil SCRUM debido a que se requerirá de un proceso de revisión y modificación constante. Para el modelado y análisis del sistema se utilizará ingeniería web basada en UML que sirve para modelar aplicaciones web, prestando una especial atención a la sistematización y personalización. Este sistema provee de diferentes modelos para el diseño del software entre los que se pueden mencionar:

- Modelo de casos de uso para capturar los requisitos del sistema.
- Modelo conceptual para el contenido.
- Modelo de usuario o modelo de navegación que incluye modelos estáticos y dinámicos.
- Modelo de estructura de presentación o modelo de flujo de presentación.

Para la implementación del sistema web se utilizarán las siguientes herramientas tecnológicas:

- Plataforma: Windows
- Lenguaje de programación: PHP
- Sistema gestor de Base de Datos: PostgreSQL
- Framework backend: Laravel
- Plantilla para maquetación: Bootstrap

CAPÍTULO II

MARCO TEÓRICO

2.1 INGENIERÍA DEL SOFTWARE

Según Sommerville (2005) la ingeniería del software es una disciplina de la ingeniería que comprende todos los aspectos de la producción de software desde las etapas iniciales de la especificación del sistema, hasta el mantenimiento de este después de que se utiliza, cuya meta es el desarrollo costeable de sistemas de software.

La ingeniería de software, también, incorpora el análisis precedente de la situación, el bosquejo del proyecto, el desarrollo del software, el ensayo necesario para comprobar su funcionamiento correcto y poner en funcionamiento el sistema. El desarrollo del software va unido a lo que se conoce como “ciclo de vida del software” que consiste en cuatro etapas que se conocen como: concepción, elaboración, construcción y transición.

La concepción determina la repercusión del proyecto y diseña el modelo de negocio; la elaboración precisa la planificación del proyecto, especificando las características y apoya la arquitectura; la construcción es la elaboración del producto; y la transición es la entrega del producto terminado a los usuarios. Al culminar este ciclo, comienza el mantenimiento del software, el cual consiste en una etapa en la que el software ofrece soluciones a errores que son denunciados por los usuarios, principalmente y se incorporan actualizaciones para hacer frente a los nuevos requisitos (Cevallos, 2017).

La ingeniería de software es una tecnología de varias capas, cualquier enfoque de ingeniería incluyendo software de ingeniería debe basarse en un compromiso organizacional de calidad. La administración total de la calidad alimenta la cultura de mejora continua, y es esta la que lleva en última instancia al desarrollo de enfoques cada vez más eficaces de la ingeniería de software. El fundamento en el que se apoya la ingeniería de software es el compromiso de calidad.

Otro fundamento para la ingeniería de software es la capa proceso. El proceso de ingeniería de software es el aglutinante que une las capas de tecnología y permite el desarrollo racional y oportuno del software de computo, también define una estructura que debe establecerse para la obtención eficaz de tecnología de Ingeniería de Software. Los métodos de la Ingeniería de Software proporcionan la experiencia técnica para elaborar software. Incluyen un conjunto amplio de tareas, como comunicación, análisis de los requerimientos, modelación del diseño, construcción del programa, pruebas y apoyo. (Sommerville, 2005)

Las herramientas de la Ingeniería de Software proporcionan un apoyo para el proceso y los métodos. Cuando se integran las herramientas de modo que la información creada por una pueda ser utilizada por otra, queda establecido un sistema llamado ingeniería de software asistido por computadora que apoya al desarrollo de software. Ver figura 2.1



Figura 2.1 Capas de la Ingeniería de Software

Fuente: Pressman, 2010

Los cambios en la tecnología de ingeniería de software son de hecho rápidos, pero al mismo tiempo, el progreso con frecuencia es muy lento. Para cuando se toma una decisión a fin de adoptar un nuevo proceso, método o herramienta; realizar la capacitación necesaria para comprender su aplicación e introducir la tecnología en la cultura de desarrollo de software, llega algo más nuevo (e incluso mejor) y el proceso comienza de nuevo (Pressman, 2010).

2.2 METODOLOGÍAS ÁGILES

En los años 80 y principios de los 90, existía una opinión general de que la mejor forma de obtener un mejor software era a través de una planificación cuidadosa del proyecto, una garantía de calidad formalizada, la utilización de métodos de análisis y procesos de desarrollo de software controlados y rigurosos. Sin embargo, con este enfoque se pasaba más tiempo pensando en cómo se debía desarrollar el sistema que en programar el desarrollo y las pruebas. Cuando cambiaban los requerimientos, se hacía esencial rehacer el trabajo, y al menos en principio, la especificación y el diseño tenían que cambiar con el programa.

El descontento con estos enfoques pesados condujo a varios desarrolladores de software en los años 90 a proponer nuevos métodos de desarrollo ágil. Estos permitieron a los equipos de desarrollo centrarse en el software mismo en vez de su diseño y documentación. Los métodos ágiles

universalmente dependen de un enfoque iterativo para la especificación, desarrollo y entrega del software, y principalmente fueron diseñados para apoyar al desarrollo de aplicaciones de negocio donde los requerimientos del sistema normalmente cambiaban rápidamente durante el proceso de desarrollo. Están pensados para entregar software funcional de forma rápida a los clientes, quienes pueden entonces proponer que se incluyan en iteraciones posteriores del sistema nuevos requerimientos o cambios en los mismos (Sommerville, 2005).

Según Muradas (2018) las metodologías ágiles se definen como un conjunto de tareas y procedimientos dirigidos a la gestión de proyectos. Son aquellos métodos en los cuales tanto las necesidades como las soluciones a estas evolucionan con el pasar del tiempo, a través del trabajo en equipo de grupos multidisciplinarios que se caracterizan por tener las siguientes cualidades:

- Desarrollo evolutivo y flexible.
- Autonomía de los equipos.
- Planificación.
- Comunicación.

2.3 METODOLOGÍA SCRUM

Según Palacio (2015) Scrum es un modelo de desarrollo ágil que se puede adoptar de forma técnica, aplicando reglas definidas, caracterizado por:

- Adoptar una estrategia de desarrollo incremental, en lugar de la planificación y ejecución completa del producto.
- Basar la calidad del resultado más en el conocimiento tácito de las personas en equipos auto-organizados, que en la calidad de los procesos empleados.
- Solapamiento de las diferentes fases del desarrollo, en lugar de realizarlas una tras otra en un ciclo secuencial o de cascada.

Scrum es un método para trabajar en equipo a partir de iteraciones o Sprints, por lo que su objetivo será controlar proyectos con un gran volumen de cambios de última hora, se centra en ajustar sus resultados y responder a las exigencias reales y exactas del cliente. De ahí, que se vaya revisando cada entregable, ya que los requerimientos van variando a corto plazo. El tiempo mínimo para un Sprint es de una semana y el máximo es de cuatro semanas.

La base fundamental de esta metodología radica en la división del trabajo completo en distintos apartados o bloques que pueden ser abordados en periodos cortos de tiempo. Esta organización del proceso de creación de software permite potenciar la agilidad, simplicidad, flexibilidad y la colaboración en la realización del proyecto. (Palacio, 2015)

2.3.1 PROCESO DE DESARROLLO DE SCRUM

El trabajo a ser realizado en un proyecto es un listado en el Product Backlog, que es una lista de todos los cambios requeridos sobre un producto.

Los proyectos se realizan durante una serie de iteración de un mes de duración llamadas Sprints. Al comienzo de cada Sprint tiene lugar un Sprint Planning Meeting durante el cual el Product Owner prioriza el Product Backlog y el Scrum Team selecciona las tareas que serán completadas durante el Sprint que va a comenzar. Estas tareas son removidas del Product Backlog para ser llevadas al Sprint Backlog. (Albaladejo, 2015)

Durante el Sprint el equipo se mantiene en contacto a través de los Daily Meetings para informar sobre los avances, problemas y cambios que se hizo respecto al día anterior. Y al final del Sprint debe mostrar la funcionalidad completa en el Sprint Review Meeting.

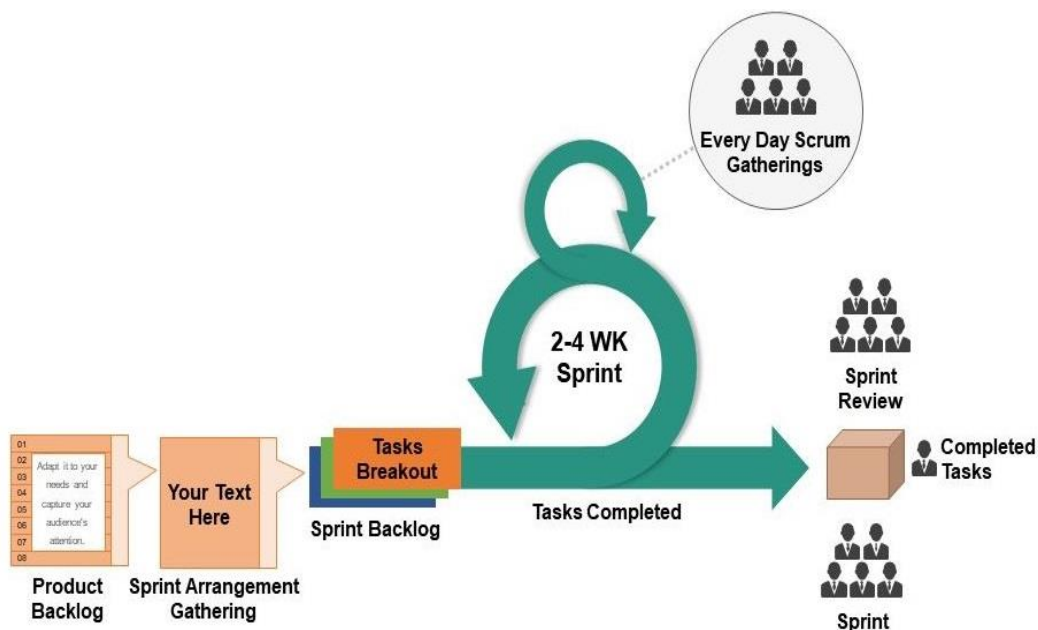


Figura 2.2 Ciclo de Proceso de Desarrollo de Scrum

Fuente: Slide Team, 2019

2.3.2 FASES DE SCRUM

De acuerdo con Scrum Manager (2013), la metodología Scrum comprende las siguientes fases:

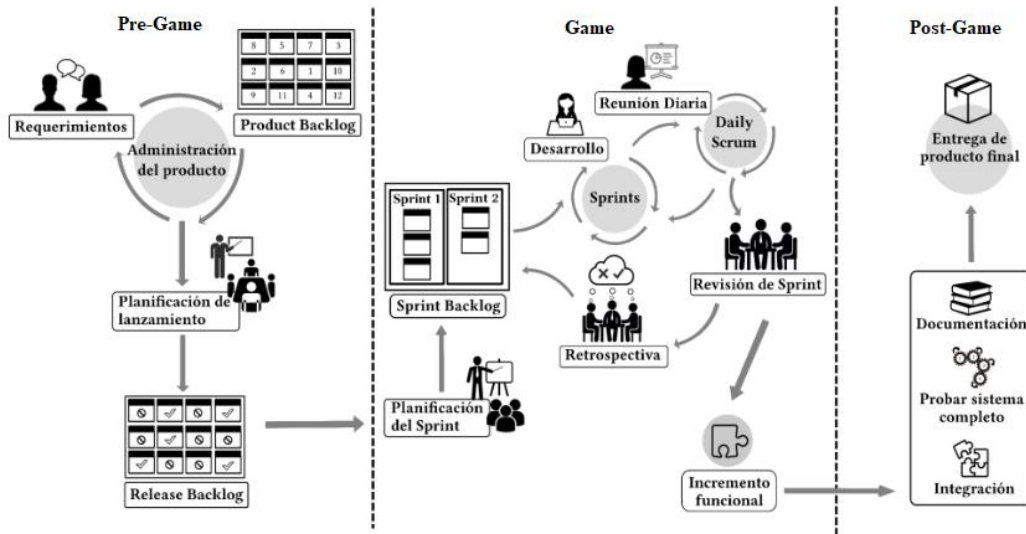


Figura 2.3 Fases de Scrum

Fuente: Delgadillo & Figueroa, 2019

2.3.2.1 PRE-GAME

Según Albaladejo (2009) en esta fase se prepara el escenario para determinar el esfuerzo de desarrollo iterativo – incremental, se le conoce también como Sprint 0, debido a que representa la fase inicial en la que se pretende entender el negocio para el que se desarrollara el proyecto con el fin de tomar decisiones que sirvan para agregar valor al producto. Es normal que en esta fase existan inexactitudes respecto de las estimaciones. En general lo que debe hacerse en esta etapa es establecer la visión, fijar las expectativas y asegurar el financiamiento.

Es en esta etapa “Preparación” contiene algunas tareas que son candidatas en la que se desarrolla el Product Backlog, considerándose todos los objetivos que se pretende sean alcanzados, en esta misma etapa, además, se diseñan las historias de usuario correspondientes al producto.

2.3.2.2 GAME

Desarrollo iterativo-incremental del sistema, donde cada iteración (Sprint) se realiza en un periodo de tiempo y ofrece un incremento operacional que satisface un subconjunto de la Pila de Producto (Product Backlog List). En cada iteración del Game se realizan las siguientes tareas:

- **Planeación del Sprint**, antes de comenzar cada sprint o iteración, se lleva a cabo dos reuniones consecutivas, en la primera se refina y se prioriza nuevamente el backlog del producto, además de elegir las metas de la iteración. En la segunda reunión se deben considerar cómo alcanzar los requerimientos y crear el backlog del sprint.
- **Desarrollo del Sprint**, el trabajo generalmente se organiza en iteraciones de 30 días (Sprints). El Sprint es el desarrollo de la nueva funcionalidad para el producto. Esta fase provee la siguiente documentación: Backlog del Sprint con las actividades realizadas, los responsables y la duración de cada actividad.
- **Revisión del Sprint**, al final de cada iteración se lleva a cabo una reunión, en donde se presenta la nueva funcionalidad del producto, las metas incluyendo la información de las funciones, diseño, ventajas, inconvenientes y esfuerzos del equipo. (Albaladejo, 2009)

2.3.2.3 POST-GAME

Según Albaladejo (2009) se centra en la integración de los incrementos producidos y liberando el sistema en el entorno de usuario. Las tareas que se llevan a cabo en esta etapa son:

- Integración de los incrementos producidos durante los sprints
- Pruebas de todo el sistema
- Preparación de la documentación del usuario
- Preparación de la formación y el material de comercialización
- Capacitar a los usuarios y operadores del sistema
- Sistema de conversión / empaquetado
- Las pruebas de aceptación

2.3.3 EVENTOS DE SCRUM

Según Schwaber y Sutherland (2013) en Scrum existen eventos predefinidos con el fin de crear regularidad y minimizar la necesidad de reuniones no definidas. Todos los eventos son bloques de tiempo, de tal modo que todos tienen una duración máxima. Una vez que comienza un Sprint, su duración es fija y no puede acortarse o alargarse. Los demás eventos pueden terminar siempre que se alcance el objetivo del evento, asegurando que se emplee una cantidad apropiada de tiempo sin permitir desperdicio en el proceso. Además del propio Sprint, que es un contenedor del resto de

eventos, cada uno de los eventos de Scrum constituye una oportunidad formal para la inspección y adaptación de algún aspecto.

2.3.3.1 SPRINT

El corazón de Scrum es el Sprint, es un bloque de tiempo de un mes o menos durante el cual se crea un incremento de producto “Terminado”, utilizable y potencialmente desplegable. Es más conveniente si la duración de los Sprints es consistente a lo largo del esfuerzo de desarrollo.

En los eventos de Scrum, cada nuevo Sprint comienza inmediatamente después de la finalización del Sprint anterior. Los Sprints contienen y consisten de la Reunión de Planificación del Sprint “*Sprint Planning Meeting*”, los Scrums Diarios “*Daily Scrums*”, el trabajo de desarrollo, la Revisión del Sprint “*Sprint Review*”, y la Retrospectiva del Sprint “*Sprint Retrospective*”. (Schwaber y Sutherland, 2013)

Durante el Sprint:

- No se realizan cambios que puedan afectar al Objetivo del Sprint.
- Los objetivos de calidad no disminuyen.
- El alcance puede ser clarificado y renegociado entre el Dueño de Producto y el Equipo de Desarrollo a medida que se va aprendiendo más.

2.3.3.2 REUNIÓN DE PLANIFICACIÓN DEL SPRINT

Según Schwaber y Sutherland (2013) el trabajo a realizar durante el Sprint se planifica en la Reunión de Planificación de Sprint. Este plan se crea mediante el trabajo colaborativo del Equipo Scrum completo. La Reunión de Planificación de Sprint tiene un máximo de duración de ocho horas para un Sprint de un mes. Para Sprints más cortos, el evento es usualmente más corto.

El Scrum Master se asegura de que el evento se lleve a cabo y que los asistentes entiendan su propósito. El Scrum Master enseña al Equipo Scrum a mantenerse dentro del bloque de tiempo. La Reunión de Planificación de Sprint responde a las siguientes preguntas:

- ¿Qué puede entregarse en el Incremento resultante del Sprint que comienza?
- ¿Cómo se conseguirá hacer el trabajo necesario para entregar el Incremento?

2.3.3.3 SCRUM DIARIO

El Scrum Diario es una reunión con un bloque de tiempo de 15 minutos para que el Equipo de Desarrollo sincronice sus actividades y cree un plan para las siguientes 24 horas. Esto se lleva a cabo inspeccionando el trabajo avanzado desde el último Scrum Diario y haciendo una proyección acerca del trabajo que podría completarse antes del siguiente.

El Scrum Diario se realiza a la misma hora y en el mismo lugar todos los días para reducir la complejidad. Durante la reunión, cada miembro del Equipo de Desarrollo explica:

- ¿Qué se hizo ayer para ayudar al Equipo de Desarrollo a lograr el Objetivo del Sprint?
- ¿Qué se hará hoy para ayudar al Equipo de Desarrollo a lograr el Objetivo del Sprint?
- ¿Se visualiza algún impedimento que evite que el Equipo de Desarrollo logre el Objetivo del Sprint?

Los Scrum Diarios mejoran la comunicación, eliminan la necesidad de mantener otras reuniones, identifican y eliminan impedimentos relativos al desarrollo, resaltan y promueven la toma de decisiones rápida, y mejoran el nivel de conocimiento del Equipo de Desarrollo. El Scrum Diario constituye una reunión clave de inspección y adaptación. (Schwaber y Sutherland, 2013)

2.3.3.4 REVISIÓN SPRINT

Al final del Sprint se lleva a cabo una Revisión de Sprint para inspeccionar el Incremento y adaptar la Lista de Producto si fuese necesario. Durante la Revisión de Sprint, el Equipo Scrum y los interesados colaboran acerca de lo que se hizo durante el Sprint. Basándose en esto, y en cualquier cambio a la Lista de Producto durante el Sprint, los asistentes colaboran para determinar las siguientes cosas que podrían hacerse para optimizar el valor.

Se trata de una reunión informal, no una reunión de seguimiento, y la presentación del Incremento tiene como objetivo facilitar la retroalimentación de información y fomentar la colaboración. (Schwaber y Sutherland, 2013)

2.3.3.5 RETROSPECTIVA DEL SPRINT

Según Schwaber y Sutherland (2013) la Retrospectiva de Sprint es una oportunidad para el Equipo Scrum de inspeccionarse a sí mismo y crear un plan de mejoras que sean abordadas durante el

siguiente Sprint. La Retrospectiva de Sprint tiene lugar después de la Revisión de Sprint y antes de la siguiente Reunión de Planificación de Sprint. Se trata de una reunión restringida a un bloque de tiempo de tres horas para Sprints de un mes. Para Sprints más cortos se reserva un tiempo proporcionalmente menor. El Scrum Master se asegura de que el evento se lleve a cabo y que los asistentes entiendan su propósito. El Scrum Master enseña a todos a mantener el evento dentro del bloque de tiempo fijado. El Scrum Master participa en la reunión como un miembro del equipo ya que la responsabilidad del proceso Scrum recae sobre él. El propósito de la Retrospectiva de Sprint es:

- Inspeccionar cómo fue el último Sprint en cuanto a personas, relaciones, procesos y herramientas.
- Identificar y ordenar los elementos más importantes que salieron bien y las posibles mejoras.
- Crear un plan para implementar las mejoras a la forma en la que el Equipo Scrum desempeña su trabajo.

2.3.3.6 REFINAMIENTO

El refinamiento del *Product Backlog* es una práctica recomendada para asegurar que éste siempre esté preparado. Esta ceremonia sigue un patrón similar al resto y tiene una agenda fija específica en cada Sprint. Se estima su duración en 2 horas máximo por semana del Sprint. Es responsabilidad del *product owner* agendar, gestionar y dirigir esta reunión.

Los participantes de esta reunión son todo el equipo Scrum, así como cualquier recurso adicional que considere necesario el PO y que pueda contribuir a aclarar el requerimiento. (Roche, 2017)

2.3.4 ROLES DEL SCRUM

Scrum ayuda a una permanente relación entre destinatarios, a estos se les asigna determinados roles y responsabilidades, para el desarrollo del proyecto. (Gallardo, 2014)

2.3.4.1 PRODUCT OWNER (DUEÑO DEL PRODUCTO)

A veces es el mismo cliente. En otros casos, especialmente cuando se trata de proyectos complejos, actúa como su representante directo, es necesario que este rol recaiga en una única persona. Es el

único con la potestad para decidir las funcionalidades y características del producto; según las circunstancias del proyecto es posible incluso que delegue en el equipo de desarrollo, o en alguien de su confianza, pero la responsabilidad siempre es suya.

Para ejercer este rol es necesario:

- Conocer perfectamente el entorno de negocio del cliente, las necesidades y el objetivo que se persigue con el sistema que se está construyendo.
- Tener la visión del producto, así como las necesidades concretas del proyecto, para poder priorizar eficientemente el trabajo.
- Disponer de atribuciones y conocimiento del plan del producto suficiente para tomar las decisiones necesarias durante el proyecto, incluidas para cubrir las expectativas previstas de retorno de la Inversión del proyecto.
- Recibir y analizar de forma continua retroinformación del entorno de negocio (evolución del mercado, competencia, alternativas) y del proyecto (sugerencias de equipo, alternativas técnicas, pruebas y evaluación de cada incremento).

Tiene un dialogo directo y permanente con el Scrum Master, que es su nexo directo con quienes ejecutan las labores. Solo entra en contacto con el Scrum Team al final de cada una de las iteraciones para evaluar las entregas parciales. (Palacio, 2015)

2.3.4.2 SCRUM MASTER (DIRECTOR DEL PROYECTO)

Es el encargado de garantizar que el proceso cumplirá con las directrices del modelo Scrum. Muchos lo denominan líder del proyecto, pero en realidad es mucho más que eso. Es el encargado de mantener una visión global del mismo y de emplearse a fondo en cualquier circunstancia, sea la que sea. Además, fluctúa entre el plano práctico y el plano directivo proporciona:

- Asesoría y formación al equipo para trabajar de forma auto organizada y con responsabilidad de equipo.
- Revisión y validación de la pila de producto.
- Moderación de las reuniones.
- Gestión de las dinámicas de grupo en el equipo.
- Resolución de impedimentos que en el sprint pueden entorpecer la ejecución de las tareas.

- Configuración, diseño y mejora continua de las prácticas de Scrum en la organización. Respeto de la organización y los implicados, con las pautas de tiempos y formas de Scrum.

Al crecer la fluidez de la organización y evolucionar hacia un marco de Scrum más pragmático, puede eliminarse el rol de Scrum Master, cuando estas responsabilidades ya estén institucionalizadas en la organización. (Palacio, 2015)

2.3.4.3 SCRUM TEAM (EQUIPO DE TRABAJO)

Según Palacio (2015) el Scrum Team hace referencia al grupo de personas que ejecuta las tareas propuestas. Aquí entran tanto los arquitectos, ingenieros, programadores, diseñadores y demás profesionales como las personas que realizan labores administrativas.

Trabaja de forma cohesionada y auto organizada. No hay un gestor para delimitar, asignar y coordinar las tareas. Son los propios miembros los que lo realizan. En el equipo:

- Todos conocen y comprenden la visión del propietario del producto.
- Aportan y colaboran con el propietario del producto en el desarrollo de la pila del producto.
- Comparten de forma conjunta el objetivo de cada sprint y la responsabilidad del logro.
- Todos los miembros participan en la toma de decisiones.
- Se respetan las opiniones y aportes de todos.
- Todos conocen el modelo de trabajo con Scrum.

2.3.5 ARTEFACTOS DEL SCRUM

Según Roche (2017) se denomina artefacto a aquellos elementos físicos que se producen como resultado de la aplicación de Scrum. Los tres principales artefactos o herramientas de Scrum son: el Product Backlog, Sprint Backlog, y el incremento.

2.3.5.1 PRODUCT BACKLOG (PILA DE PRODUCTO)

Es la principal fuente de información sobre el producto en Scrum, una lista, en cualquier formato que contiene todos los requerimientos que necesitamos implementar en el producto. Esta lista es el resultado del trabajo del Product Owner con el cliente y los distintos comités, también refleja el estado real del trabajo pendiente de implementar con el producto, así como el ya realizado.

	Item #	Description	Est	By
Very High				
	1	Finish database versioning	16	KH
	2	Get rid of unneeded shared Java in database	8	KH
		- Add licensing	-	-
	3	Concurrent user licensing	16	TG
	4	Demo / Eval licensing	16	TG
		- Analysis Manager	-	-
	5	File formats we support are out of date	160	TG
	6	Round-trip Analyses	250	MC
High				
		- Enforce unique names	-	-
	7	In main application	24	KH
	8	In import	24	AM
		- Admin Program	-	-
	9	Delete users	4	JM
		- Analysis Manager	-	-
	10	When items are removed from an analysis, they should show up again in the pick list in lower 1/2 of the analysis tab	8	TG
		- Query	-	-
	11	Support for wildcards when searching	16	T&A
	12	Sorting of number attributes to handle negative numbers	16	T&A
	13	Horizontal scrolling	12	T&A
		- Population Genetics	-	-
	14	Frequency Manager	400	T&M
	15	Query Tool	400	T&M
	16	Additional Editors (which ones)	240	T&M
	17	Study Variable Manager	240	T&M
	18	Haplotypes	320	T&M
	19	Add icons for v1.1 or 2.0	-	-
		- Pedigree Manager	-	-
	20	Validate Derived kindred	4	KH
Medium				
		- Explorer	-	-
	21	Launch tab synchronization (only show queries/analyses for logged in users)	8	T&A
	22	Delete settings (?)	4	T&A

Figura 2.4 Pila de Producto

Fuente: Abad, 2014

El Product Backlog debe ser gestionado en exclusiva por el Product Owner, siendo su principal función la de priorizar aquellos elementos que tienen más valor en cada etapa y detallarlos para que el equipo de desarrollo sea capaz de valorarlos y ejecutarlos.

Al comenzar a utilizar Scrum, no es necesario una lista completa y exhaustiva de todos los requerimientos. Se toma como un inventario que contiene cualquier tipo de trabajo que deba realizarse en el producto: funcionalidades, bugs, historias de usuario, y tareas técnicas.

Los elementos del Product Backlog tienen como atributos la descripción, el orden y el valor, es recomendable empezar con los dos o tres requerimientos más urgentes arriba e ir añadiendo elementos conforme se va descubriendo más necesidades del producto. (Roche, 2017)

2.3.5.2 SPRINT BACKLOG (PILA DE SPRINT)

Se trata de una lista de elementos en los que trabajar durante la etapa de Sprint. Estos elementos normalmente se componen de tareas técnicas más pequeñas que permiten conseguir un incremento de software terminado.

Todo el trabajo que el equipo de desarrollo haya seleccionado para hacer el siguiente Sprint pasa al Sprint Backlog.

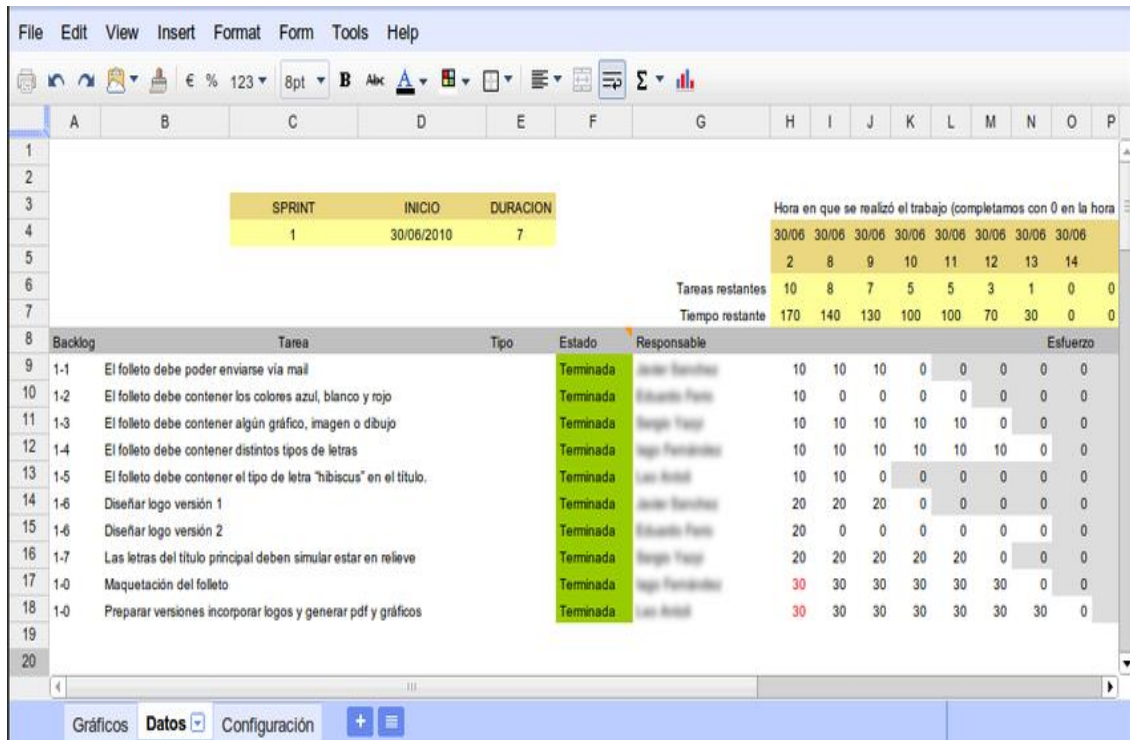


Figura 2.5 Pila de Sprint

Fuente: Yazzy, 2011

Este artefacto es un elemento para visualizar el trabajo a realizar durante cada Sprint y está gestionado por el equipo de desarrollo. Su propósito es mantener la transparencia dentro del desarrollo, actualizándolo durante la iteración especialmente a través de los Daily Scrums. (Siqueira, 2017)

2.3.5.3 INCREMENTO

Para Roche (2017) si Scrum tuviera que ser reducido a una sola cosa, sería a entregar una pieza de software terminado en cada Sprint. Un incremento es el resultado del Sprint, es la suma de todas las tareas, casos de uso, historias de usuario y cualquier elemento que se desarrolló durante el Sprint y que será puesto a disposición del usuario final en forma de software, aportando un valor de negocio al producto que se está desarrollando.

2.4 INGENIERÍA WEB

La ingeniería web es el uso de métodos sistemáticos, disciplinados y cuantificables al desempeño eficaz, operatividad y crecimiento de programas de muy buena calidad en la World Wide Web. Los

expertos en la web necesitan utilizar herramientas y técnicas basadas en la ingeniería de software, para poder garantizar el buen funcionamiento y administración de los sitios web.

La ingeniería web toma prestado muchos de los conceptos y principios básicos de la ingeniería de software, dando importancia a las mismas actividades técnicas y de gestión, para poder garantizar el buen funcionamiento y administración de los sitios web.

Existen diferencias sutiles en la forma en que se llevan a cabo estas actividades, pero la filosofía primordial es idéntica dado que dicta un enfoque disciplinado para el desarrollo de un sistema basado en computadora (Pressman, 2002).

2.4.1 INGENIERÍA WEB BASADA EN UML

Ingeniería Web basada en UML (UWE) es un enfoque de ingeniería de software para el desarrollo de aplicaciones Web. La metodología UWE, presentada por Koch y sus colegas, soporta el desarrollo de aplicaciones Web con especial énfasis en sistematización, personalización y generación de código semi-automático.

UWE proporciona guías para la construcción de modelos de forma sistemática enfocadas en el estudio de casos de uso. Las actividades del modelado son:

- Análisis de Requerimientos.
- Modelo Conceptual.
- Modelo Navegacional.
- Modelo de Presentación.

Es fundamentada en el análisis y diseño orientado a objetos, con procesos iterativos e incrementales definidas en el Proceso Unificado (UP). En su notación se utilización perfiles ligeros de UML (UML profile), un perfil en UWE presenta estereotipos utilizados en el modelado de aspectos de navegación y presentación de las aplicaciones Web. (Salas, 2017)

2.4.1.1 ANÁLISIS DE REQUERIMIENTOS

El análisis de requerimientos trata de capturar y describir detalladamente los requerimientos de funcionalidad y de calidad de servicio del producto que desarrolla. Al definir el análisis de requerimientos, debe mencionarse un tipo de modelo denominado “Modelo de Casos de Uso” el

cual se utiliza para describir los requisitos funcionales del sistema en términos de casos de uso. Un caso de uso en UML se define como una unidad coherente de la funcionalidad proporcionada por la aplicación que obra recíprocamente con uno o más actores de aplicación. (Salas, 2017)

En otras palabras, un modelo de casos de uso representa una parte del comportamiento del sistema sin revelar la estructura interna. Ver figura 2.6

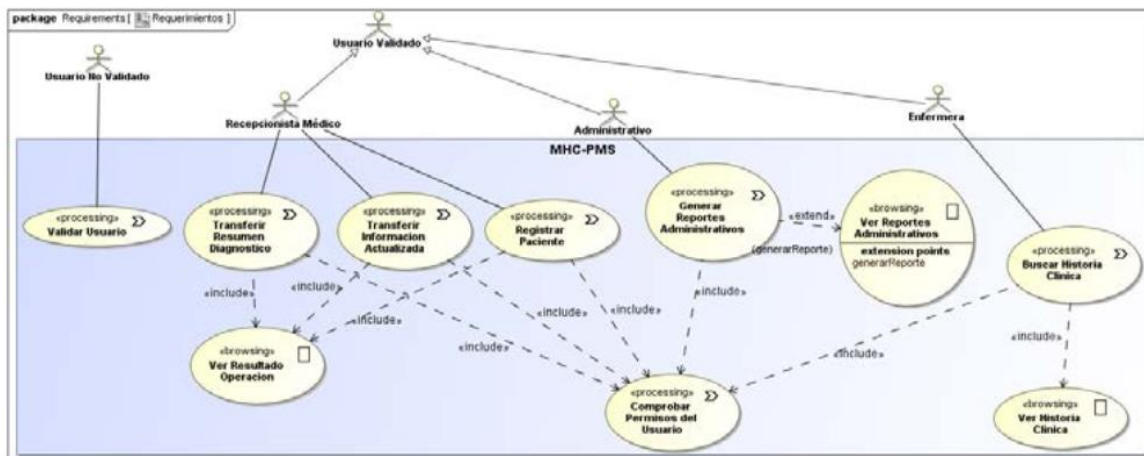


Figura 2.6 Modelo de Casos de Uso

Fuente: Benavides, 2016

2.4.1.2 MODELO CONCEPTUAL

El diseño del modelo lógico-conceptual debe realizarse según los modelos de casos de uso que ya se hayan establecido y que se definen en la especificación de requerimientos. Ver figura 2.7

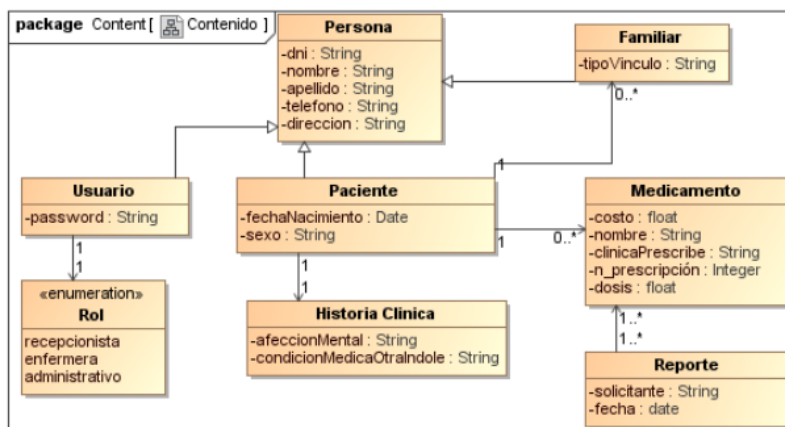


Figura 2.7 Modelo Conceptual

Fuente: Benavides, 2016

El modelo conceptual debe de incluir aquellos objetos que estén implicados en las tareas y actividades comunes que los usuarios finales realizaran en la aplicación web, esto significa diferenciar los elementos de mayor importancia para la realización de una tarea o, que resultan como el efecto de otra. La siguiente figura presenta un ejemplo claro del modelo conceptual de UWE. (Salas, 2017)

2.4.1.3 MODELO NAVEGACIONAL

Para Guerrero (2014) este modelo permite una visión amplia de los procesos de la página web que se representan en los diagramas de navegación; pueden interpretarse también con las interfaces del sistema web, para el caso se tiene estereotipos o iconos que ayudan al diseño de los diagramas de presentación. El diagrama de presentación de la metodología UWE, permite al usuario comprender y analizar, sobre el área de trabajo al que se someterá con la implantación del sistema.

En la siguiente figura, se muestra la aplicación de los iconos que pertenecen a los diagramas de presentación. Ver figura 2.8

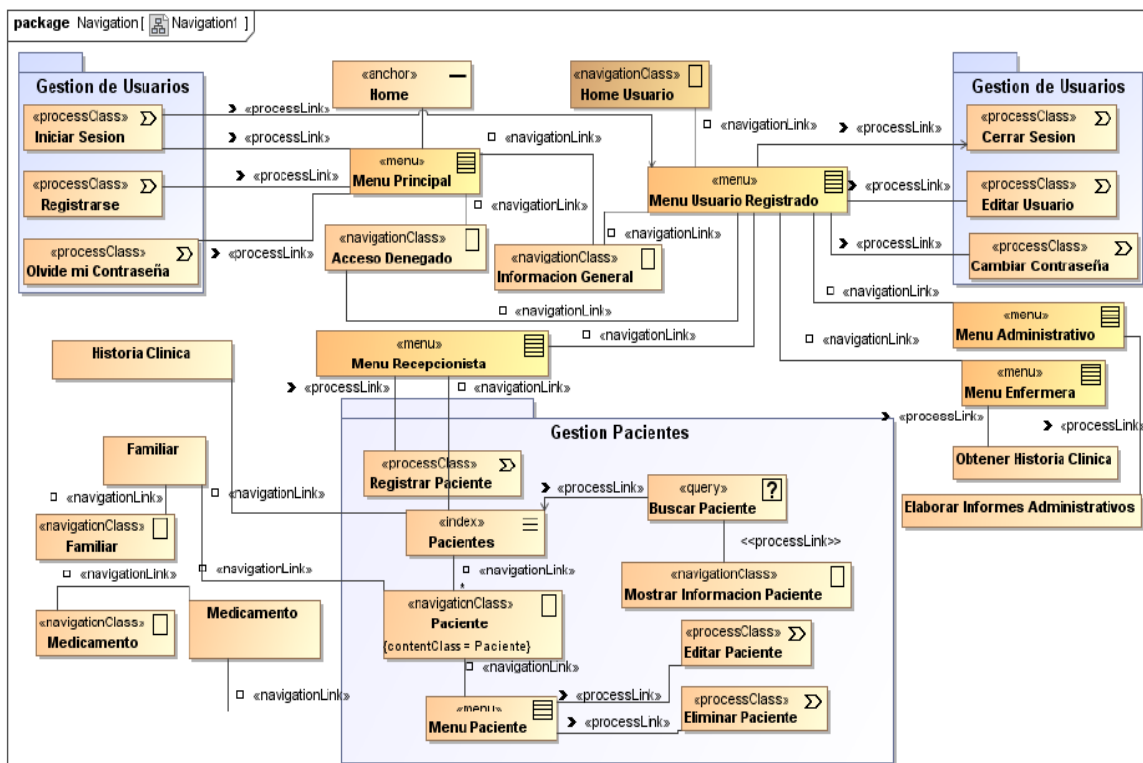


Figura 2.8 Modelo de Navegación
Fuente: Benavides, 2016

Las clases de navegación («navigationClass») representan nodos navegables de la estructura de hipertexto; los enlaces de navegación («navigationLink») muestran vínculos directos entre las clases de navegación; las rutas alternativas de navegación son manejadas por menú («menu»).

Los accesos se utilizan para llegar a múltiples instancias de una clase de navegación («index» o «guidedTour») o para seleccionar los elementos («query»). Las clases de procesos («processClass») forman los puntos de entrada y salida de los procesos de negocio en este modelado y la vinculación entre sí y a las clases de navegación se modela por enlaces de procesos («processLink»). (Guerrero, 2014)

2.4.1.4 MODELO DE PRESENTACIÓN

Para Guerrero (2014) este modelo permite una visión amplia de los procesos de la página web que se representan en los diagramas de navegación; pueden interpretarse también con las interfaces del sistema web, para el caso se tiene estereotipos o iconos que ayudan al diseño de los diagramas de presentación.

El diagrama de presentación de la metodología UWE, permite al usuario comprender y analizar, sobre el área de trabajo al que se someterá con la implantación del sistema. Ver figura 2.9

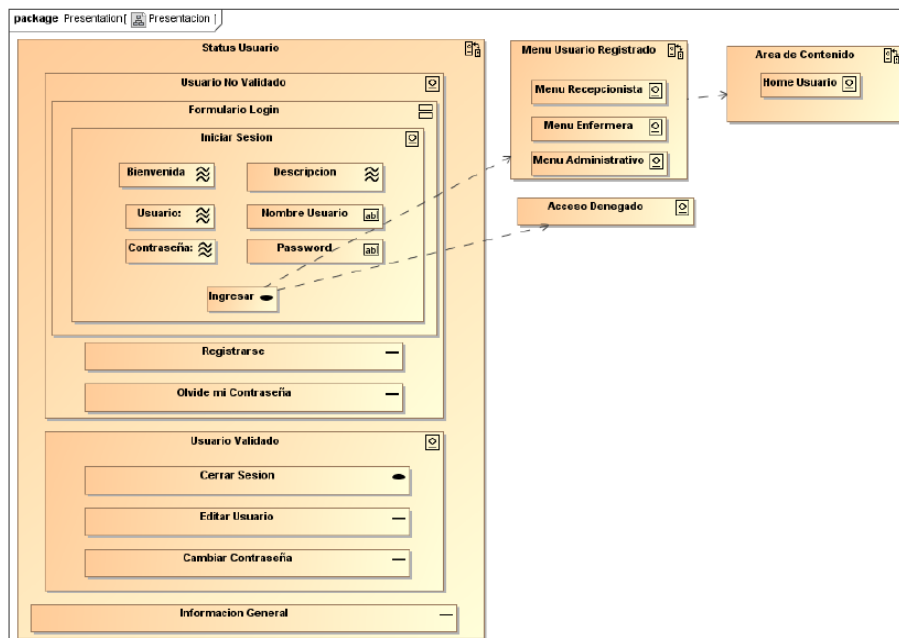


Figura 2.9 Modelo de Presentación

Fuente: Benavides, 2016

2.5 INTEROPERABILIDAD

Según Gonzales (2009) el mejoramiento de las condiciones de intercambio de información debe ser un aspecto trascendental para toda organización, para evitar en consecuencia que se sigan desarrollando soluciones informáticas ignorando las lecciones del pasado. Basta recordar cómo, cada agencia o empresa incorporó tecnologías de información y las comunicaciones considerando únicamente sus necesidades particulares, dando lugar a lo que hoy se conoce como islas informáticas que se caracterizan por un manejo ineficiente y descoordinado de la información, que prácticamente imposibilita la interacción entre ellas e impide por ejemplo que los tramites del estado los pueda realizar un ciudadano en un solo sitio. Ver figura 2.10

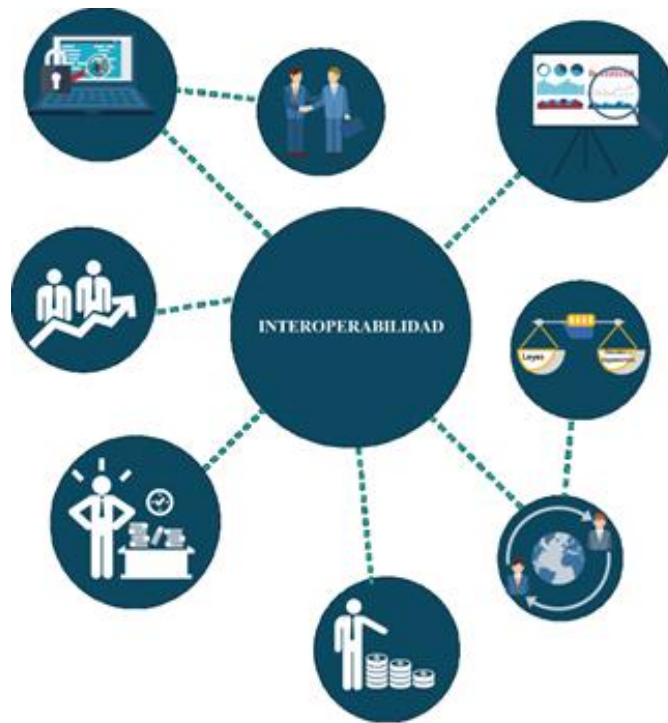


Figura 2.10 Ejemplo de Inteoperabilidad

Fuente: QA Lovers, 2016

Para lograr la integración y el trabajo coordinado de los sistemas de información al interior de una organización, entre las diferentes empresas que hacen uso de las Tecnologías de la Información (TI), deben solucionarse problemas del tipo semántico como las incongruencias debidas a estructuras de información incomprensibles de una agencia a otra, problemas organizacionales causados por procesos administrativos descoordinados que se repiten innecesariamente, problemas

técnicos causados por computadores y sistemas informáticos de diferentes tecnologías incompatibles entre sí y problemas de gobernanza debidos a la falta de normas y de la institucionalidad necesarias para lograr acuerdos de intercambio de información entre las agencias que terminen siendo prácticas homogéneas y estándares aceptados por todas ellas. (Gonzáles, 2009)

El análisis del fenómeno de interoperabilidad se desarrolla con base en una tipología que considera los siguientes cuatro aspectos relevantes: semánticos, organizacionales, técnicos y de gobernanza.

2.5.1 INTEROPERABILIDAD SEMÁNTICA

Se ocupa de asegurar que el significado preciso de la información intercambiada sea entendible sin ambigüedad por todas las aplicaciones que intervengan en una determinada transacción y habilita a los sistemas para combinar información recibida con otros recursos de información y así procesarlos de forma adecuada. (Gonzáles, 2009)

2.5.2 INTEROPERABILIDAD ORGANIZACIONAL

Se ocupa de definir los objetivos de negocios, modelar los procesos y facilitar la colaboración de administraciones que desean intercambiar información y pueden tener diferentes estructuras organizacionales y procesos internos. Además de eso, busca orientar, con base en los requerimientos de la comunidad usuaria, los servicios que deben estar disponibles, fácilmente identificables, accesibles y orientados al usuario. (Gonzáles, 2009)

2.5.3 INTEROPERABILIDAD TÉCNICA

Cubre las cuestiones técnicas (hardware, software, telecomunicaciones), necesarias para interconectar sistemas computacionales y servicios, incluyendo aspectos clave como interfaces abiertas, servicios de interconexión, integración de datos y middleware, presentación e intercambio de datos, accesibilidad y servicios de seguridad. (Gonzáles, 2009)

2.5.4 GOBERNANZA DE INTEROPERABILIDAD

Es considerado a nivel de gobiernos y se refiere a los acuerdos entre estos y actores que participan en proceso de interoperabilidad y a la forma de alcanzarlos. También se refiere a la definición de los espacios de dialogo donde se definan los acuerdos. Con la gobernanza, se busca que las autoridades públicas cuenten con la institucionalidad necesaria para establecer estándares de

interoperabilidad, asegurar su adopción, y dotar a las agencias de capacidad organizacional y técnica necesarias para ponerlos en práctica (CEPAL, Unión Europea, 2007).

2.6 SERVICIO WEB

Un servicio web es una vía de intercomunicación e interoperabilidad entre máquinas conectadas en Red. En el mundo de Internet se han popularizado enormemente, ya se trate de servicios web públicos o privados.

Generalmente, la interacción se basa en el envío de solicitudes y respuestas entre un cliente y un servidor, que incluyen datos. El cliente solicita información, enviando a veces datos al servidor para que pueda procesar su solicitud.

El servidor genera una respuesta que envía de vuelta al cliente, adjuntando otra serie de datos que forman parte de esa respuesta. Por tanto, podemos entender un servicio web como tráfico de mensajes entre dos máquinas (García, 2015).

2.6.1 COMPONENTES DE LOS SERVICIOS WEB

El término Servicio Web describe una forma estandarizada de integrar aplicaciones Web mediante el uso de XML, SOAP, WSDL y UDDI sobre los protocolos de la internet. XML es usado para describir los datos, SOAP se ocupa de la transferencia de los datos, WSDL se emplea para describir los servicios disponibles y UDDI se ocupa para conocer cuáles son los servicios disponibles. Uno de los usos principales es permitir la comunicación entre las empresas y entre las empresas y sus clientes.

Los Servicios Web permiten a las organizaciones intercambiar datos sin necesidad de conocer los detalles de sus respectivos Sistemas de Información. A diferencia de los modelos Cliente/Servidor, tales como un servidor de páginas Web, los servicios web no proveen al usuario una interfaz gráfica. En vez de ello, los servicios web comparten la lógica de negocio, los datos y los procesos, por medio de una interfaz de programas a través de la red.

Es decir, conectan programas, por tanto, son programas que no interactúan directamente con los usuarios y permiten a distintas aplicaciones, de diferentes orígenes, comunicarse entre ellos sin necesidad de escribir programas costosos. (Saffirio, 2006)

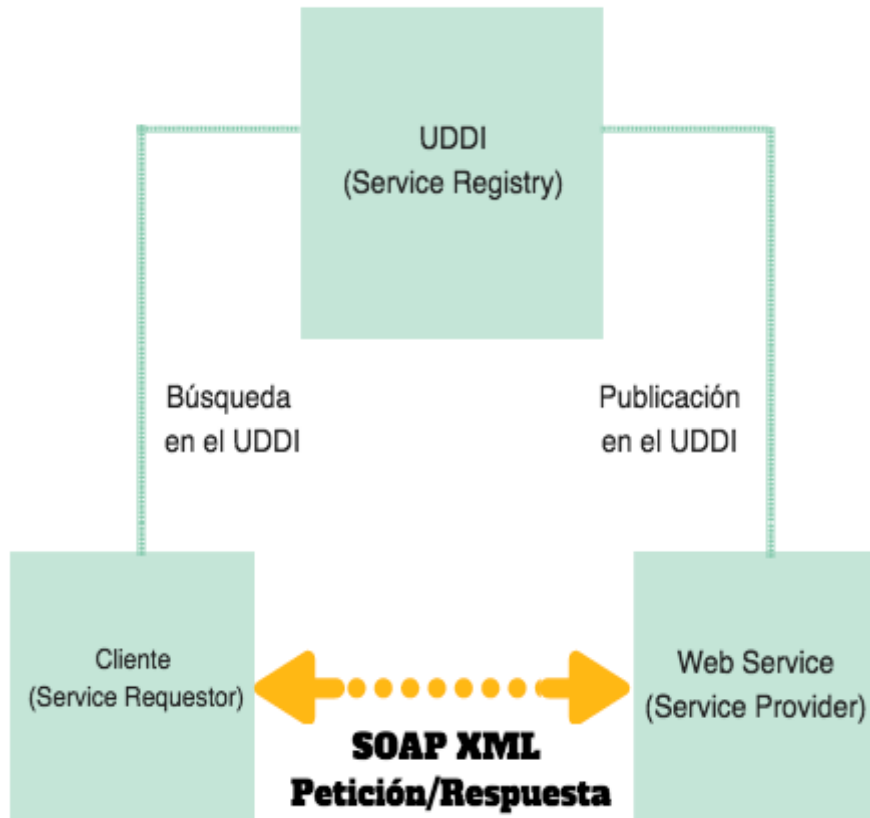


Figura 2.11 Servicio Web
Fuente: QA Lovers, 2016

Según Lázaro, (2018) Un servicio web es un término genérico para una función de software interoperable de máquina a máquina que se aloja en una ubicación direccionable de red, están estandarizados y funcionan con los siguientes componentes:

2.6.1.1 XML

Abreviación de Extensible Markup Lenguaje. El XML es una especificación desarrollada por la World Wide Web. Permite a los desarrolladores crear sus propios tags (comando que va insertado en un documento), que les permiten habilitar definiciones, transmisiones, validaciones, e interpretación de los datos entre aplicaciones y entre organizaciones. (Lázaro, 2018)

2.6.1.2 SOAP

Es un protocolo escrito en XML para el intercambio de información entre aplicaciones. Es un formato para enviar mensajes, diseñado especialmente para servir de comunicación en Internet,

pudiendo extender los HTTP headers. Es una forma de definir que información se envía y como mediante XML. Basicamente es un protocolo para acceder a un servicio Web. (Lázaro, 2018)

2.6.1.3 WSDL

WSDL es un lenguaje basado en XML para describir los servicios web y como acceder a ellos. Es el formato estándar para describir un servicio web, y fue diseñado por Microsoft e IBM. WSDL es una parte integral del estándar UDDI, y es el lenguaje que este utiliza. (Lázaro, 2018)

2.6.1.4 UDDI

UDDI es un estándar XML para describir, publicar y encontrar servicios web. Es un directorio donde las compañías pueden registrar y buscar servicios web. Es un directorio de interfaces de servicios web descritos en WSDL que se comunican mediante SOAP. (Lázaro, 2018)

2.6.1.5 JSON

JSON (JavaScript Object Notation) Notación de Objetos de JavaScript es un formato ligero de intercambio de datos. Leerlo y escribirlo es simple para humanos, mientras que para las maquinas es simple interpretarlo y generarlo.

Está constituido por dos estructuras:

- Una colección de pares de nombre/valor. En varios lenguajes esto es conocido como un objeto, registro, estructura, diccionario, lista de claves o un arreglo asociativo.
- Una lista ordenada de valores. En la mayoría de los lenguajes, esto se implementa como arreglos, vectores, listas o secuencias.

Estas son estructuras universales; virtualmente todos los lenguajes de programación las soportan de una forma u otra. Es razonable que un formato de intercambio de datos que es independiente del lenguaje de programación se base en estas estructuras (Crockford, 2006).

2.6.2 SERVICIO WEB SOAP

Los servicios SOAP o mejor conocimos simplemente como Web Services, son servicios que basan su comunicación bajo el protocolo SOAP (Simple Object Access Protocol) el cual este definido por Wikipedia como “protocolo estándar que define cómo dos objetos en diferentes procesos

pueden comunicarse por medio de intercambio de datos XML”. SOAP es un formato más pesado, tanto en tamaño como en procesamiento, pues los XML tiene que ser analizados y procesados a un árbol DOM, resolver espacios de nombre “*namespaces*” antes de poder empezar a procesar el documento.

Los XML además tienen métodos de validación muy potentes y ampliamente utilizados y su formato es sumamente estricto, por lo que lleva más tiempo procesarlo. (Blancarte, 2017)

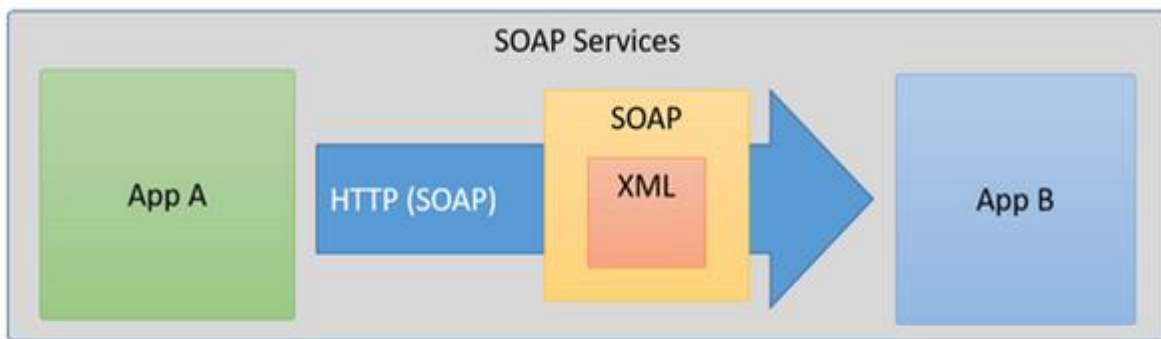


Figura 2.12 Servicio Web Soap

Fuente: Blancarte, 2017

2.6.3 SERVICIO WEB REST

REST (Representational State Transfer) es una tecnología flexible a diferencia de SOAP, transporta datos por medio del protocolo HTTP, pero este permite utilizar diversos métodos que proporciona HTTP para comunicarse, como lo son GET, POST, PUT, DELETE, PATCH y a la vez, utiliza códigos de respuesta nativos de HTTP (404, 200, 204, 409).

REST permite transmitir prácticamente cualquier tipo de datos, ya que el tipo de datos está definido por el Header Content Type, lo que nos permite mandar, XML, JSON, Binarios (imágenes, documentos), etc. Que contrasta con SOAP que solo permite la transmisión de datos en formato XML.

A pesar de la gran variedad de tipos de datos que se pueden mandar con REST, una gran parte transmite en JSON por un motivo muy importante, JSON es interpretado de forma natural por JavaScript, lo que hace que algunos frameworks se aprovechen al máximo, pues pueden enviar peticiones directas al servidor y obtener los datos de una forma nativa (Blancarte, 2017).

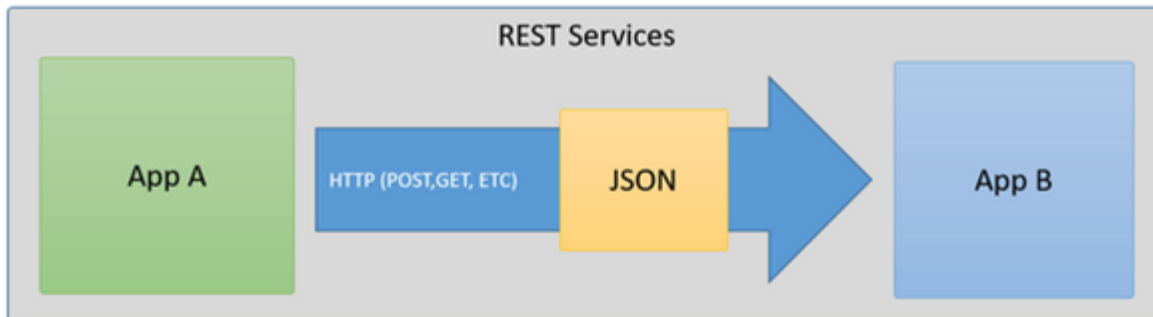


Figura 2.13 Servicio Web Rest

Fuente: Blancarte, 2017

2.7 FIRMA DIGITAL

Según Ponce (2019), la firma digital es una solución tecnológica que permite añadir a documentos digitales y mensajes de correo electrónico una huella o marca única, a través de ciertas operaciones matemáticas. Se utiliza principalmente para firmar documentos PDF y correos electrónicos, pero también permite firmar documentos de texto, plantillas, imágenes y virtualmente cualquier tipo de documento.

La firma digital facilita el reemplazo de documentación en papel por su equivalente en formato digital, su tecnología está incorporada en transacciones electrónicas, formularios web y navegación en páginas seguras, permite al receptor del mensaje o documento:

- Asegurar que el contenido no pudo ser modificado luego de la firma sin dejar evidencia de la alteración (Integridad).
- Tener garantías de que la firma se realizó bajo el control absoluto del firmante (Exclusividad).
- Identificar al firmante de forma fehaciente (Autenticación).
- Demostrar el origen de la firma y la integridad del mensaje ante terceros, de modo que el firmante no pueda negar o repudiar su existencia o autoría (No Repudio).

2.7.1 CARACTERÍSTICAS

La tecnología de firma digital se sostiene de dos pilares: un método que hace imposible la alteración de la firma y una infraestructura que permite certificar la identidad del firmante.

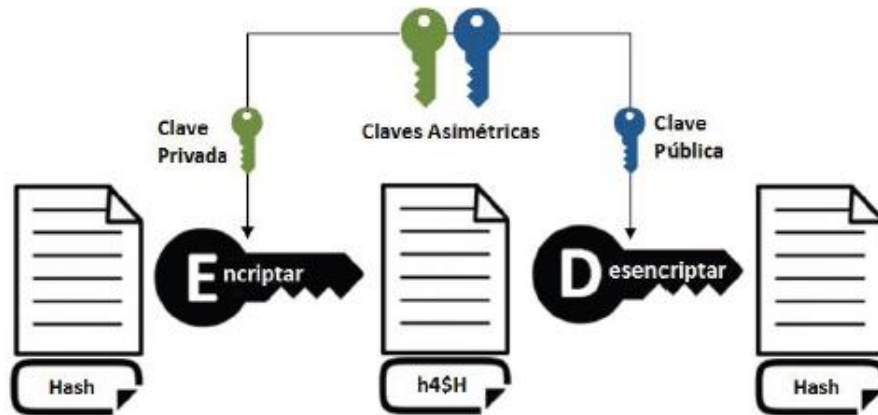


Figura 2.14 Comprobación de Firma Digital

Fuente: Manual SAS, 2019

2.7.1.1 CLAVE ASIMÉTRICA

La clave Asimétrica es un método de criptografía o codificación, en el que se generan dos números de gran longitud mediante una fórmula matemática compleja. Estos números, llamados "claves", son distintos, pero están relacionados de modo tal que lo que se cifra o encripta con una clave solo puede descifrarse con la otra. A este par de claves se los conoce como Clave Pública y Clave Privada. La clave pública se distribuye y la clave privada la conserva el propietario, protegida por una o varias contraseñas que solo él conoce. El par de claves funciona siempre en conjunto, no es posible cifrar y descifrar un documento con una misma clave.

Cuando se aplica la clave privada sobre un documento digital en su totalidad, este queda cifrado o encriptado. Es decir, se vuelve ilegible para cualquiera que no posea la clave pública con que descifrarlo. En firma digital, ya que no se busca encriptar el mensaje sino darle una marca de autenticación, la clave asimétrica se utiliza de forma indirecta, no sobre el documento, sino sobre un resumen del mismo, denominado hash. (MFD, 2019)

2.7.1.2 HASH

El hash (también conocido como digesto o huella digital), es un resumen único que identifica a un documento digital. Se puede aplicar a cualquier documento, incluso a una cadena de texto. Se obtiene al aplicar una fórmula matemática llamada "función unidireccional de resumen", o función hash. El resultado suele expresarse en números y letras minúsculas de la "a" a la "f" (sistema hexadecimal).

- Es un resumen, porque sin importar el tamaño del documento, la función devuelve un hash de la misma longitud.
- Es unidireccional, porque no es posible convertir el hash nuevamente en el documento original, ni conocer el contenido del documento a partir del hash.
- Al ser una función matemática, aplicarla sobre un mismo documento o mensaje devuelve siempre el mismo hash.
- Es estadísticamente imposible encontrar dos documentos distintos que posean el mismo hash.
- Dos documentos pueden parecer a simple vista idénticos, pero poseer distinto hash. Aunque parezcan idénticos, si el hash difiere, no pueden considerarse el mismo documento digital. (MFD, 2019)

2.8 RECURSOS PARA DESARROLLO WEB

Las fases del desarrollo de una aplicación web, así como los lenguajes de programación usados, son extensas y variadas, por ello es necesario herramientas específicas para cada una de ellas, como el gestor de base de datos a ocupar, la maquetación HTML/CSS, programación Cliente/Servidor, etc.

2.8.1 BOOTSTRAP

Según Fontela (2015), Bootstrap es un *framework* CSS desarrollado inicialmente (en el año 2011) por Twitter que permite dar forma a un sitio web mediante librerías CSS que incluyen tipografías, botones, cuadros, menús y otros elementos que pueden ser utilizados en cualquier sitio web. Aunque el desarrollo del *framework* Bootstrap fue iniciado por Twitter, fue liberado bajo licencia MIT en el año 2011 y su desarrollo continúa en un repositorio de GitHub.

Es una excelente herramienta para crear interfaces de usuario limpias y totalmente adaptables a todo tipo de dispositivos y pantallas, sea cual sea su tamaño. Además, Bootstrap ofrece las herramientas necesarias para crear cualquier tipo de sitio web utilizando los estilos y elementos de sus librerías.

Este *framework* es compatible con desarrollo web responsive, entre otras características se han reforzado las siguientes:

- Soporte bastante bueno (casi completo) con HTML5 y CSS3, permitiendo ser usado de forma muy flexible para desarrollo web con unos excelentes resultados.
- Se añadió un sistema GRID que permite diseñar usando un GRID de 12 columnas donde se debe plasmar el contenido, con esto podemos desarrollar responsive de forma mucho más fácil e intuitiva.
- Establece Media Queries para 4 tamaños de dispositivos diferentes variando dependiendo del tamaño de su pantalla, estas Media Queries permiten desarrollar para dispositivos móviles y tablets de forma mucho más fácil.
- Permite insertar imágenes responsive, es decir, con solo insertar la imagen se adaptarán al tamaño de pantalla. (Fontela, 2015)

2.8.2 LARAVEL

Laravel es un *framework* con un enfoque fresco y moderno, está hecho para arquitectura MVC (Modelo Vista Controlador), cuenta con un código modular y extensible por medio de un administrador de paquetes y un soporte robusto para manejo de bases de datos.

Cuenta con su respectiva documentación clara y completa, además ofrece las funcionalidades necesarias para desarrollar aplicaciones modernas de manera fácil y segura. Está equipado con una gran cantidad de características interesantes, incluyendo enrutamiento RESTful, PHP nativo, cuenta con un motor ligero, etc. (Pims, 2018)

2.8.3 POSTGRESQL

Según Paredes (2019) PostgreSQL es una de las opciones más valoradas en lo que se refiere a bases de datos relacionales *open-source*. Michael Stonebraker inició el proyecto bajo el nombre de “Post Ingres” a mediados de los 80 con la idea de solucionar problemas existentes en las bases de datos en esa época. Como muchos otros proyectos de código abierto, el desarrollo de PostgreSQL no es manejado por una empresa o persona, sino que es dirigido por una comunidad de desarrolladores que trabajan de forma desinteresada, altruista, libre o apoyados por organizaciones comerciales. Dicha comunidad es denominada el PGDG (PostgreSQL Global Development Group).

CAPÍTULO III

MARCO APLICATIVO

3.1 INTRODUCCIÓN

En este capítulo se describe el proceso detalladamente de la aplicación de la metodología Scrum, estos representaran los procesos que comprende las distintas partes del sistema web en desarrollo. Al momento de definir los Sprint, se tomó en cuenta cada módulo existente en el sistema web y aplicando los principios de modelado web UWE, cada Sprint consta de:

- Análisis de Requerimientos
- Modelo Conceptual.
- Modelo Navegacional.
- Modelo de Presentación.

En la figura 3.1 se puede observar claramente el modelo adaptado entre las dos metodologías.



Figura 3.1 Metodología SCRUM y UWE

Fuente: Elaboración Propia

3.2 DESARROLLO DE LOS EVENTOS DE SCRUM

3.2.1 PREGAME

En esta fase de planificación del proyecto se tomaron en cuenta elementos relevantes.

3.2.1.1 REUNIONES

Para la correcta realización del sistema, se llevaron a cabo reuniones en las cuales se expuso los problemas que tenía la organización al momento de trabajar con la información de los beneficiarios y se llegó a posibles soluciones, además de expresar otros requerimientos para el nuevo sistema,

estas reuniones no duraron más de 4 horas y se llevaron a cabo en la oficina principal del área de sistemas del Programa de Mejora de la Gestión Municipal (PMGM).

3.2.1.2 PRODUCT BACKLOG

A partir del conjunto de especificaciones y requerimientos que se obtuvieron de las reuniones llevadas a cabo con el cliente, se pudo construir una lista de labores a cumplir.

ID	TAREA	MÓDULO	SPRINT	PRIORIDAD
R1 - 1	Ingreso al sistema	Autenticación	1	Alta
R1 - 2	Verificación de roles de usuario	Autenticación	1	Alta
R1 - 3	Salida del sistema	Autenticación	1	Alta
R2 - 1	Registro y edición de perfiles	Administración	2	Alta
R2 - 2	Registro y edición de roles	Administración	2	Alta
R2 - 3	Registro y edición de personas	Administración	2	Alta
R2 - 4	Registro, asignación de permisos, e información personal a usuarios	Administración	2	Alta
R3 - 1	Conexión con los servicios web	Servicios y Consulta	3	Alta
R3 - 2	Consulta de beneficiarios con y sin autenticación mediante CI	Servicios y Consulta	3	Alta
R3 - 3	Migración de información de servicios web a base de datos	Servicios y Consulta	3	Alta
R3 - 4	Búsqueda específica de beneficiarios por diferentes atributos	Servicios y Consulta	3	Media
R4 - 1	Beneficiarios por departamento	Estadísticas	4	Alta
R4 - 2	Beneficiarios por gestión	Estadísticas	4	Alta
R4 - 3	Beneficiarios por programa	Estadísticas	4	Alta
R4 - 4	Tipos de grafica de beneficiarios	Estadísticas	4	Media
R5 - 1	Reporte válido de certificado de existencia de beneficio	Reportes	5	Alta
R5 - 2	Reporte inválido de existencia de beneficio	Reportes	5	Media

Tabla 3.1 Product Backlog
Fuente: Elaboración Propia

Estas labores a cumplir están organizadas por un identificador, la tarea a realizar, el modulo al cual corresponde, el número de sprint y la prioridad.

3.2.2 GAME

En esta fase, se desarrolló todos los requerimientos a partir del Product Backlog, donde se ha determinado que el sistema software se desarrollara en 5 Sprints

3.2.2.1 PLANIFICACIÓN DEL SPRINT: SPRINT 1

El módulo de autenticación se toma en cuenta para la planificación del primer sprint, basándose en la lista de requerimientos del Product Backlog (ver Tabla 3.2)

ID	TAREA	MÓDULO	SPRINT	PRIORIDAD
R1 - 1	Ingreso al sistema	Autenticación	1	Alta
R1 - 2	Verificación de roles de usuario	Autenticación	1	Alta
R1 - 3	Salida del sistema	Autenticación	1	Alta

Tabla 3.2 Sprint 1 Backlog
Fuente: Elaboración Propia

a) DESARROLLO DEL SPRINT: SPRINT 1

En esta fase se aplicó diseños simples siguiendo los principios de la metodología UWE para el desarrollo del primer sprint.

b) ANÁLISIS DE REQUERIMIENTOS

Se elaboró un modelo de casos de uso que representa al primer sprint. (Ver figura 3.2)

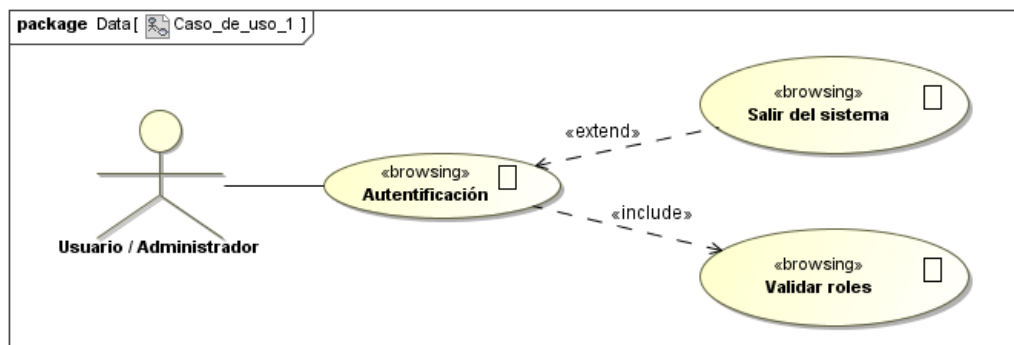


Figura 3.2 Sprint 1 Análisis de Requerimientos
Fuente: Elaboración Propia

c) MODELO CONCEPTUAL

Se tiene un modelo conceptual que representa al primer módulo del sistema. (Ver figura 3.3)

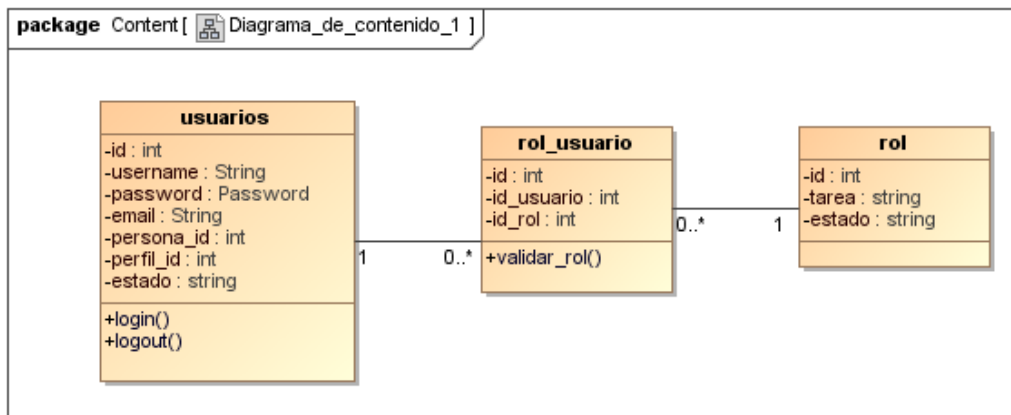


Figura 3.3 Sprint 1 Modelo Conceptual

Fuente: Elaboración Propia

d) MODELO NAVEGACIONAL

Se representa el modelo navegacional que fue diseñado para el primer módulo del sistema. (Ver figura 3.4)

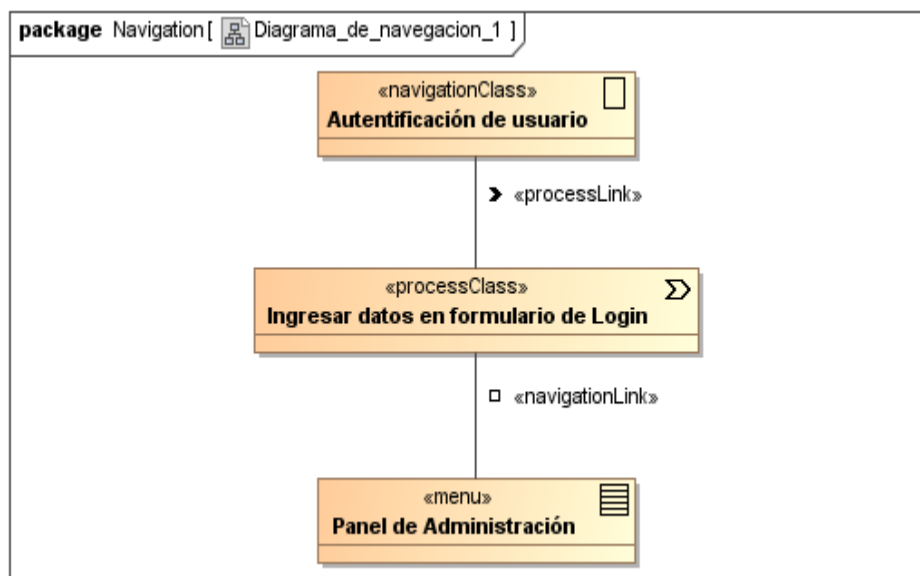


Figura 3.4 Sprint 1 Modelo Navegacional

Fuente: Elaboración Propia

e) **MODELO DE PRESENTACIÓN**

Se representa el modelo de presentación para el primer módulo del sistema. (Ver figura 3.5)

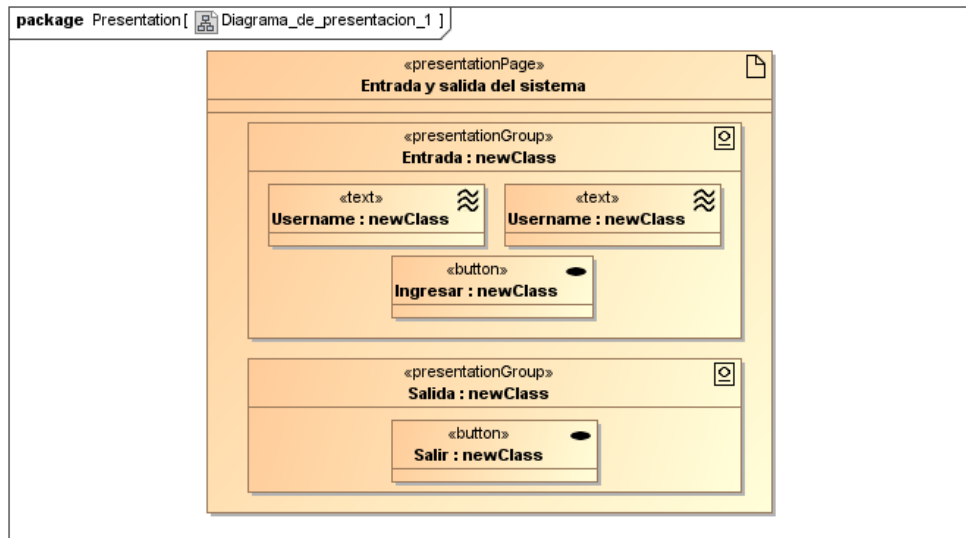


Figura 3.5 Sprint 1 Modelo de Presentación

Fuente: Elaboración Propia

f) **REVISIÓN DEL SPRINT: SPRINT 1**

Se realizó la correspondiente inspección del progreso del primer sprint, y a continuación se brinda una demostración del producto en el entorno del cliente o usuario.

En la captura de pantalla que se hizo del producto, se muestra el módulo de inicio de sesión por parte de los usuarios. (Ver figura 3.6)

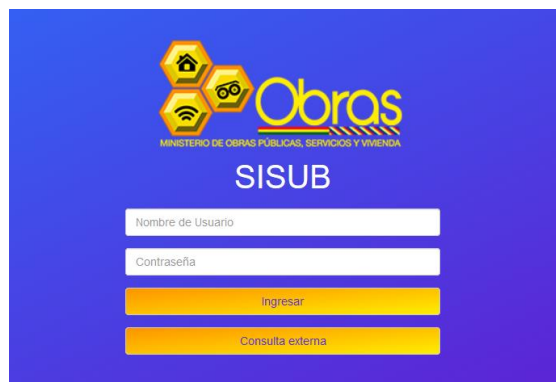


Figura 3.6 Sprint 1 Inicio de Sesión

Fuente: Elaboración Propia

Una vez reconocido un usuario, pasará al panel de administración. (Ver figura 3.7)

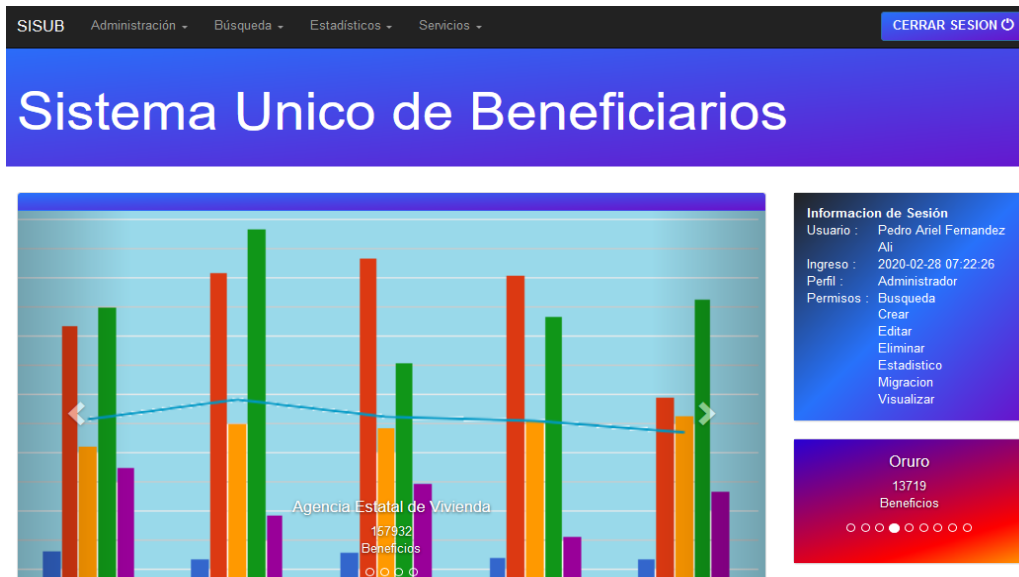


Figura 3.7 Sprint 1 Panel de Administración

Fuente: Elaboración Propia

g) PRUEBA UNITARIA

En esta fase se verificó que los objetivos del sprint 1 están implementados de manera apropiada y correcta. (ver Tabla 3.3)

PRUEBA UNITARIA SPRINT 1	
Descripción	Ingreso y salida al sistema por parte del usuario, además de control de roles en el panel de administración.
Objetivos	1. Ingreso al sistema. 2. Verificación interna de roles y tareas. 3. Salida del sistema.
Condiciones	Servidor ejecutándose, ingresar al sistema con los datos correctos.
Resultado Esperado	Ejecución satisfactoria de las tareas descritas en el Sprint 1, por parte del usuario.
Resultado Obtenido	El usuario puede ingresar y salir al sistema, también le da acceso a determinadas tareas dependiendo de los roles que tiene asignado.

Tabla 3.3 Sprint 1 Prueba Unitaria

Fuente: Elaboración Propia

3.2.2.2 PLANIFICACIÓN DEL SPRINT: SPRINT 2

El módulo de administración se toma en cuenta para la planificación del segundo sprint, basándose en la lista de requerimientos del Product Backlog (ver Tabla 3.4)

ID	TAREA	MÓDULO	SPRINT	PRIORIDAD
R2 - 1	Registro y edición de perfiles	Administración	2	Alta
R2 - 2	Registro y edición de roles	Administración	2	Alta
R2 - 3	Registro y edición de personas	Administración	2	Alta
R2 - 4	Registro, asignacion de permisos, e información personal a usuarios	Administración	2	Alta

Tabla 3.4 Sprint 2 Backlog
Fuente: Elaboración Propia

a) DESARROLLO DEL SPRINT: SPRINT 2

En esta fase se aplicó diseños simples siguiendo los principios de la metodología UWE para el desarrollo del segundo sprint.

b) ANÁLISIS DE REQUERIMIENTOS

Se elaboró un modelo de casos de uso que representa al segundo sprint adjuntando los avances ya hechos en el Sprint anterior. (Ver figura 3.8)

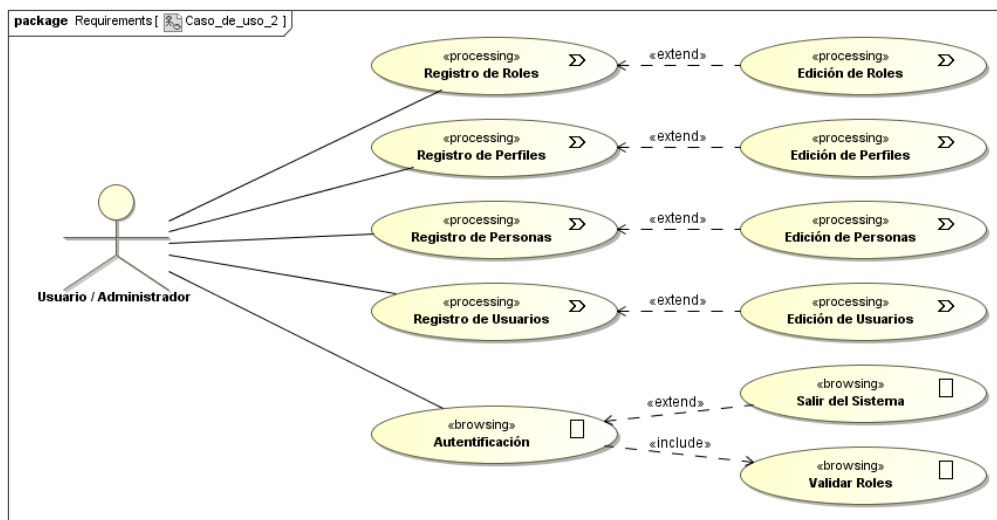


Figura 3.8 Sprint 2 Análisis de Requerimientos
Fuente: Elaboración Propia

c) MODELO CONCEPTUAL

Se tiene un modelo conceptual que representa al segundo módulo. (Ver figura 3.9)

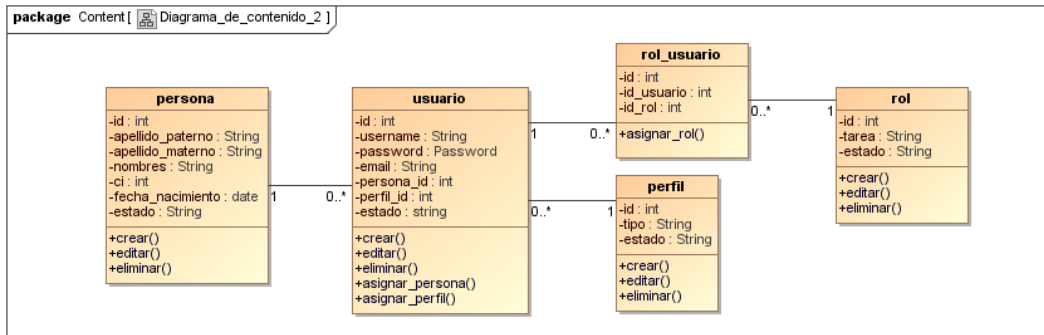


Figura 3.9 Sprint 2 Modelo Conceptual

Fuente: Elaboración Propia

d) MODELO NAVEGACIONAL

Se representa el modelo navegacional que fue diseñado para el segundo módulo del sistema. (Ver figura 3.10)

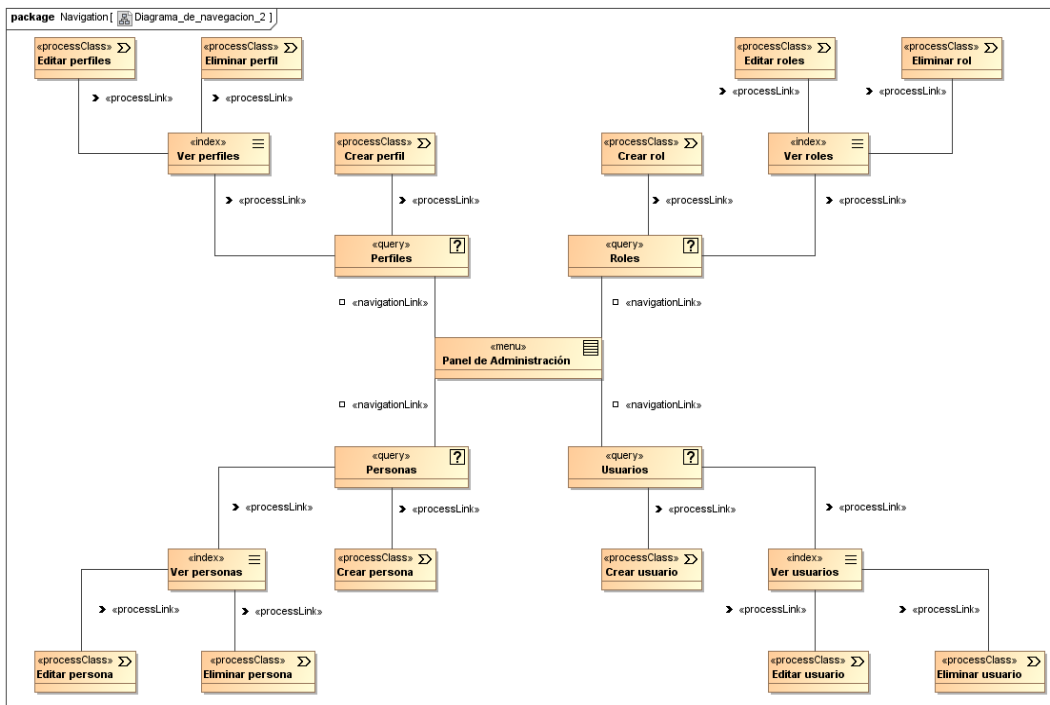


Figura 3.10 Sprint 2 Modelo Navegacional

Fuente: Elaboración Propia

e) MODELO DE PRESENTACIÓN

Aquí se representa el modelo de presentación que fue diseñado para el segundo módulo del sistema.
(Ver figura 3.11)

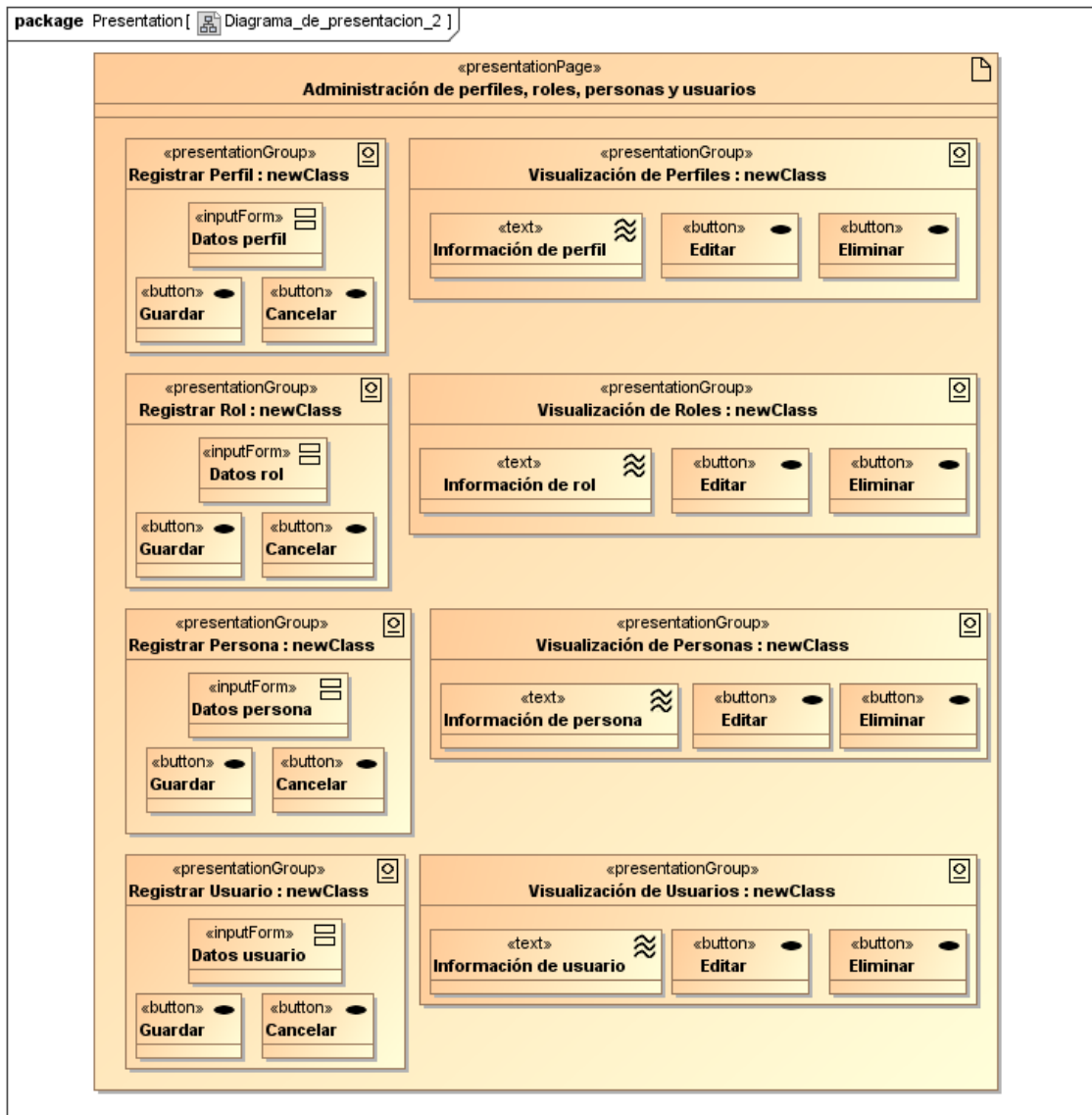


Figura 3.11 Sprint 2 Modelo de Presentación

Fuente: Elaboración Propia

f) REVISIÓN DEL SPRINT: SPRINT 2

Se realizó la correspondiente inspección del progreso del segundo sprint, y a continuación se brinda una demostración del producto en el entorno del cliente o usuario.

En la captura de pantalla que se hizo del producto, se muestra la parte de creación de perfiles. (Ver figura 3.12)

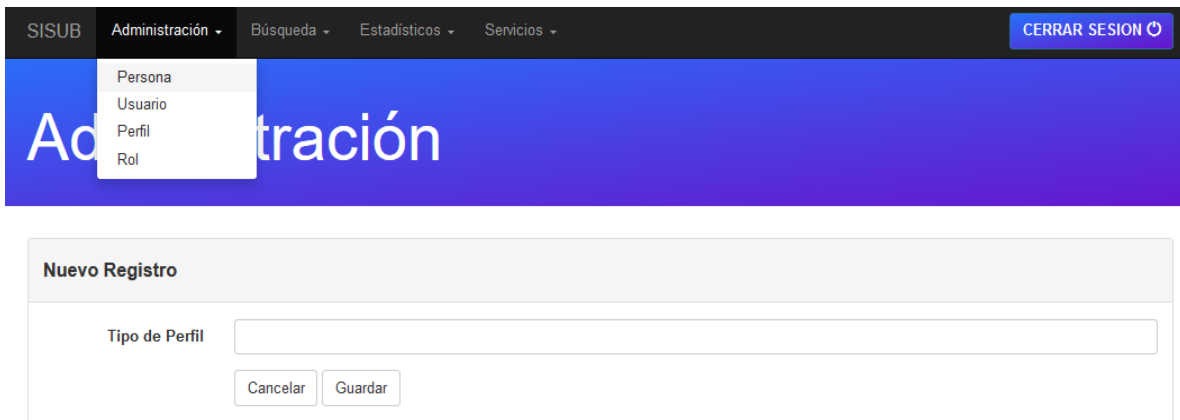


Figura 3.12 Sprint 2 Crear Perfil

Fuente: Elaboración Propia

La captura de pantalla que se hizo del producto, muestra la pantalla de edición de un perfil, que muestra sus registros iniciales. (Ver figura 3.13)

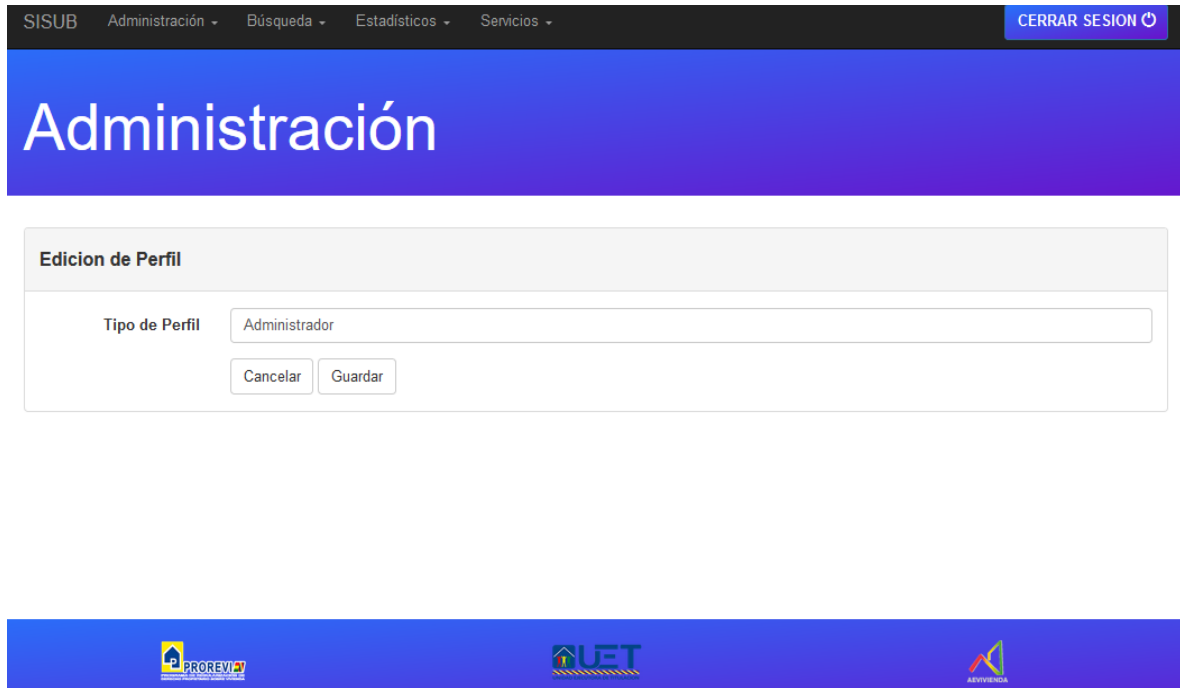


Figura 3.13 Sprint 2 Edición de Perfil

Fuente: Elaboración Propia

La captura de pantalla que se hizo del producto, muestra la pantalla donde se puede eliminar (botón rojo) un perfil de la lista de perfiles. (Ver figura 3.14)

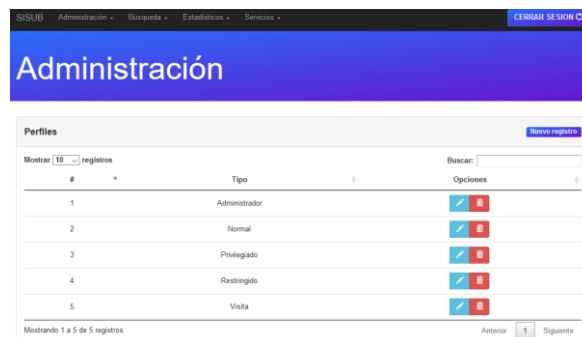


Figura 3.14 Sprint 2 Eliminar Perfil
Fuente: Elaboración Propia

g) PRUEBA UNITARIA

En esta fase se verificó que los objetivos del sprint 2 están implementados de manera apropiada y correcta. (ver Tabla 3.5)

PRUEBA UNITARIA SPRINT 2	
Descripción	Creación, edición y eliminación de nuevos perfiles, roles, personas y usuarios.
Objetivos	<ol style="list-style-type: none"> 1. Crear, editar, eliminar perfil. 2. Crear, editar, eliminar rol. 3. Crear, editar, eliminar persona. 4. Crear, editar, eliminar usuario.
Condiciones	Servidor ejecutándose, ingresar al sistema con los datos correctos y permisos disponibles.
Resultado Esperado	Ejecución satisfactoria de las tareas descritas en el Sprint 2, por parte del usuario.
Resultado Obtenido	El usuario puede crear, editar, eliminar perfiles, roles, personas y usuarios de manera satisfactoria.

Tabla 3.5 Sprint 2 Prueba Unitaria
Fuente: Elaboración Propia

3.2.2.3 PLANIFICACIÓN DEL SPRINT: SPRINT 3

El módulo de servicios y consulta se toma en cuenta para la planificación del tercer sprint, basándose en la lista de requerimientos del Product Backlog (ver Tabla 3.6)

ID	TAREA	MÓDULO	SPRINT	PRIORIDAD
R3 - 1	Conexión con los servicios web	Servicios y Consulta	3	Alta
R3 - 2	Consulta de beneficiarios con y sin autenticación mediante CI	Servicios y Consulta	3	Alta
R3 - 3	Migración de información de servicios web a base de datos	Servicios y Consulta	3	Alta
R3 - 4	Búsqueda específica de beneficiarios por diferentes atributos	Servicios y Consulta	3	Media

Tabla 3.6 Sprint 3 Backlog
Fuente: Elaboración Propia

a) DESARROLLO DEL SPRINT: SPRINT 3

En esta fase se aplicó diseños simples siguiendo los principios de la metodología UWE para el desarrollo del tercer sprint.

b) ANÁLISIS DE REQUERIMIENTOS

Se elaboró un modelo de casos de uso que representa al tercer sprint adjuntando los avances ya hechos en Sprints anteriores. (Ver figura 3.15)

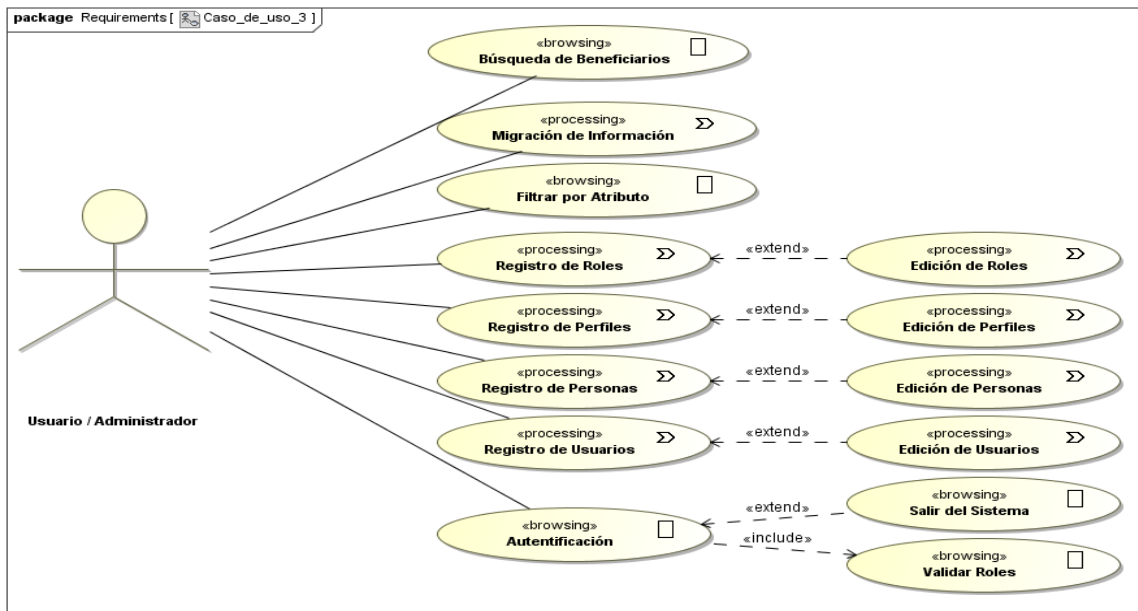


Figura 3.15 Sprint 3 Análisis de Requerimientos
Fuente: Elaboración Propia

c) MODELO CONCEPTUAL

Se tiene un modelo conceptual que representa al tercer módulo. (Ver figura 3.16)

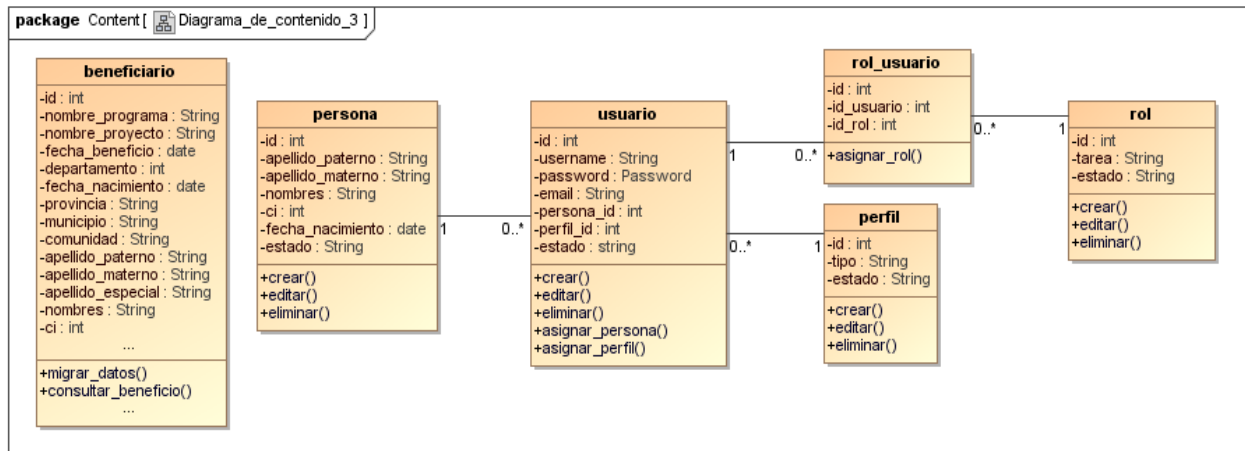


Figura 3.16 Sprint 3 Modelo Conceptual

Fuente: Elaboración Propia

d) MODELO NAVEGACIONAL

Se representa el modelo navegacional que fue diseñado para el tercer módulo del sistema. (Ver figura 3.17)

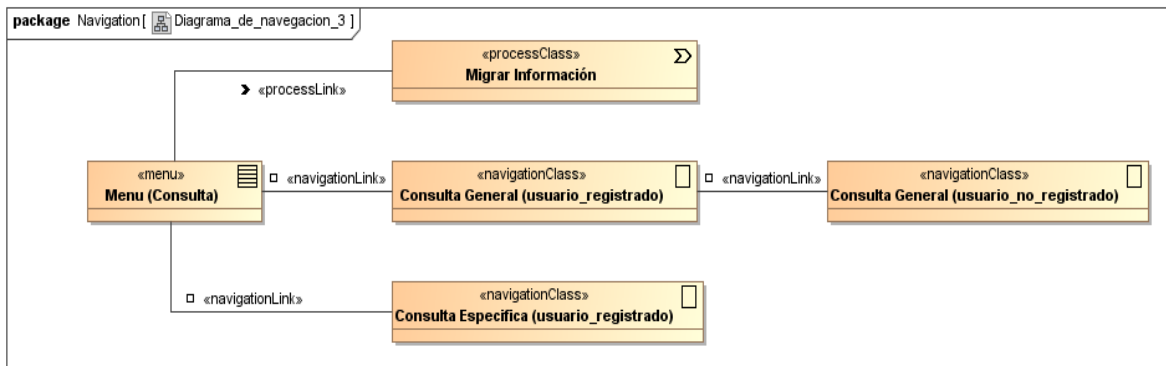


Figura 3.17 Sprint 3 Modelo Navegacional

Fuente: Elaboración Propia

e) MODELO DE PRESENTACIÓN

Aquí se representa el modelo de presentación que fue diseñado para el tercer módulo del sistema. (Ver figura 3.18)

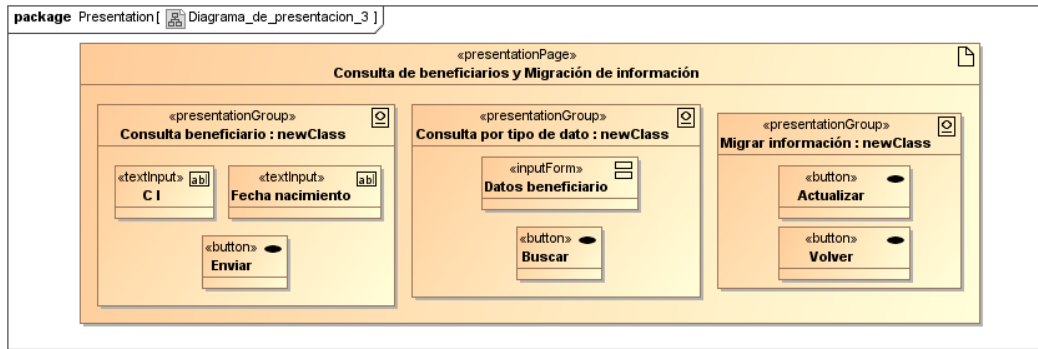


Figura 3.18 Sprint 3 Modelo de Presentación
Fuente: Elaboración Propia

f) REVISIÓN DEL SPRINT: SPRINT 3

Se realizó la correspondiente inspección del progreso del tercer sprint, y a continuación se brinda una demostración del producto en el entorno del cliente o usuario. En la captura de pantalla que se hizo del producto, se muestra la consulta externa por parte de cualquier persona. (Ver figura 3.19)

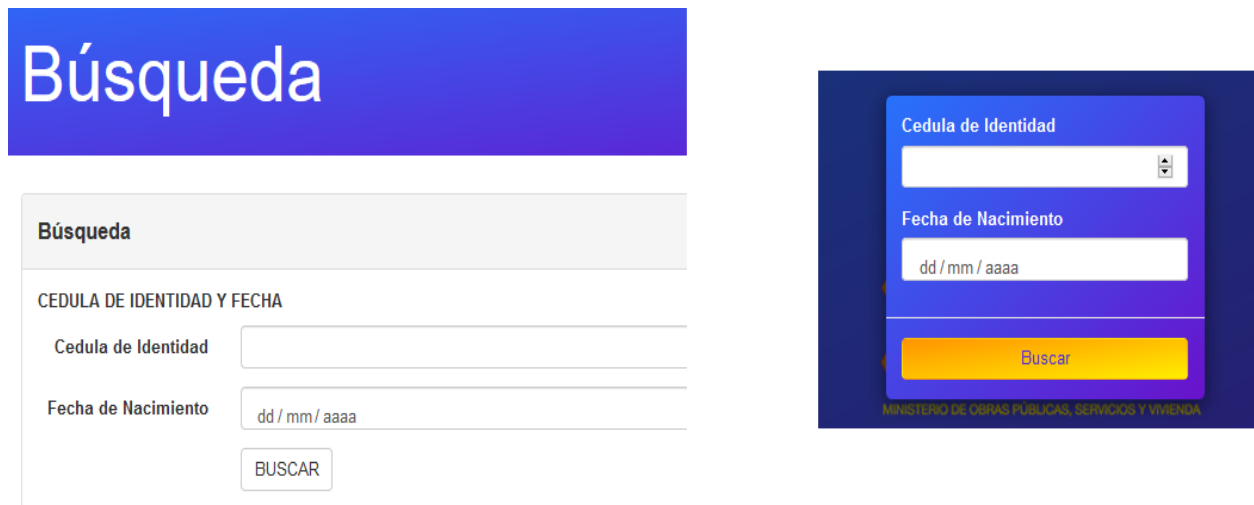


Figura 3.19 Sprint 3 Formulario de búsqueda
Fuente: Elaboración Propia

La captura de pantalla que se hizo del producto, muestra los resultados de una búsqueda que se efectuó previamente. (Ver figura 3.20)

Resultado en SEQIP			
Apellido Paterno	ZALLES	Apellido Materno	PANIAGUA
Nombres	JUAN FULGENCIO	Ci	4035888
Nacimiento	17/08/1977	País	BOLIVIA
Departamento	ORURO	Provincia	CERCADO

Resultado de búsqueda en PROREVI
No se encontro ningún registro

Resultado de búsqueda en AEVIENDA	
Registro 1	
Nombre de programa	AGENCIA ESTATAL DE VIVIENDA
Nombre de proyecto	CONSTRUCCIÓN DE 50 VIVIENDAS EN EL MUNICIPIO DE BELÉN DE ANDAMARCA - ORURO
Fecha de beneficio	2014-07-04
Departamento	ORURO
Provincia	Sud Carangas
Municipio	Belén de Andamarca
Comunidad	1. AYLLU ARANSAYA (9 u/h) 2. AYLLU UNRINSAYA (8 u/h) 3. CANTON CALAMA (5 u/h) 4. CRUZ DE HUAYLLAMARCA (11 u/h) 5. REAL MACHACAMARCA (17 u/h)
Apellido paterno	ZALLES
Apellido materno	PANIAGUA
Apellido especial	
Nombres	JUAN FULGENCIO

Figura 3.20 Sprint 3 Resultados de búsqueda

Fuente: Elaboración Propia

La captura de pantalla que se hizo del producto, muestra la lista total de beneficiarios con las opciones de filtro respectivas. (Ver figura 3.21)

Búsqueda		
DETALLE GENERAL DE TODOS LOS BENEFICIARIOS		
Mostrar <input type="text" value="10"/> registros	Buscar: <input type="text"/>	
Nombre de Programa	Nombre de Proyecto	Fecha de Beneficio
AEVIENDA	MEJORAMIENTO, AMPLIACIÓN Y/O RENOVACIÓN DE 109 VIVIENDAS EN EL MUNICIPIO DE ICLA - CHUQUISACA	2016-01-07
AEVIENDA	MEJORAMIENTO, AMPLIACIÓN Y/O RENOVACIÓN DE 109 VIVIENDAS EN EL MUNICIPIO DE ICLA - CHUQUISACA	2016-01-07
AEVIENDA	MEJORAMIENTO, AMPLIACIÓN Y/O RENOVACIÓN DE 109 VIVIENDAS EN EL MUNICIPIO DE ICLA - CHUQUISACA	2016-01-07
AEVIENDA	MEJORAMIENTO, AMPLIACIÓN Y/O RENOVACIÓN DE 109 VIVIENDAS EN EL MUNICIPIO DE ICLA - CHUQUISACA	2016-01-07
AEVIENDA	MEJORAMIENTO, AMPLIACIÓN Y/O RENOVACIÓN DE 109 VIVIENDAS EN EL MUNICIPIO DE ICLA - CHUQUISACA	2016-01-07
AEVIENDA	MEJORAMIENTO, AMPLIACIÓN Y/O RENOVACIÓN DE 109 VIVIENDAS EN EL MUNICIPIO DE ICLA - CHUQUISACA	2016-01-07
AEVIENDA	MEJORAMIENTO, AMPLIACION Y/O RENOVACION DE 100 VIVIENDAS EN EL MUNICIPIO DE BUENA VISTA - SANTA CRUZ	2015-12-17
AEVIENDA	MEJORAMIENTO, AMPLIACION Y/O RENOVACION DE 100 VIVIENDAS EN EL MUNICIPIO DE BUENA VISTA - SANTA CRUZ	2015-12-17
AEVIENDA	MEJORAMIENTO, AMPLIACIÓN Y/O RENOVACIÓN DE 109 VIVIENDAS EN EL MUNICIPIO DE ICLA - CHUQUISACA	2016-01-07
AEVIENDA	MEJORAMIENTO, AMPLIACIÓN Y/O RENOVACIÓN DE 83 VIVIENDAS EN EL MUNICIPIO DE MIZQUE - COCHABAMBA	2016-09-20
<input type="text" value="aev"/> <input type="button" value="Search Nombre de Proye"/> <input type="button" value="Search Fecha de E"/>		
Mostrando 61 a 70 de 157,932 registros (filtered from 223,098 total entries)		
Anterior 1 ... 6 <input type="text" value="7"/> 8 ... 15794 Siguiente		

Figura 3.21 Sprint 3 Búsqueda por tipo de dato

Fuente: Elaboración Propia

g) PRUEBA UNITARIA

En esta fase se verificó que los objetivos del sprint 3 están implementados de manera apropiada y correcta. (ver Tabla 3.7)

PRUEBA UNITARIA SPRINT 3	
Descripción	Búsqueda, migración y filtro de los beneficiarios.
Objetivos	1. Buscar beneficiarios. 2. Migrar información de beneficiarios. 3. Aplicar filtros a la información.
Condiciones	Servidor ejecutándose, ingresar al sistema con los datos correctos y permisos disponibles.
Resultado Esperado	Ejecución satisfactoria de las tareas descritas en el Sprint 3, por parte del usuario.
Resultado Obtenido	Una persona ajena a la institución puede consultar sobre los beneficios, también los usuarios, además de migrar y aplicar filtros.

Tabla 3.7 Sprint 3 Prueba Unitaria

Fuente: Elaboración Propia

3.2.2.4 PLANIFICACIÓN DEL SPRINT: SPRINT 4

El módulo de estadísticas es donde se hará el conteo de los beneficiarios que se encuentran en los programas sociales y se desarrollara el cuarto sprint, (ver Tabla 3.8)

ID	TAREA	MÓDULO	SPRINT	PRIORIDAD
R4 - 1	Beneficiarios por departamento	Estadísticas	4	Alta
R4 - 2	Beneficiarios por gestión	Estadísticas	4	Alta
R4 - 3	Beneficiarios por programa	Estadísticas	4	Alta
R4 - 4	Tipos de grafica de beneficiarios	Estadísticas	4	Media

Tabla 3.8 Sprint 4 Backlog

Fuente: Elaboración Propia

a) DESARROLLO DEL SPRINT: SPRINT 4

En esta fase se aplicó diseños simples siguiendo los principios de la metodología UWE para el desarrollo del cuarto sprint.

b) ANÁLISIS DE REQUERIMIENTOS

Se tiene un modelo de casos de uso del cuarto sprint y los avances anteriores. (Ver figura 3.22)

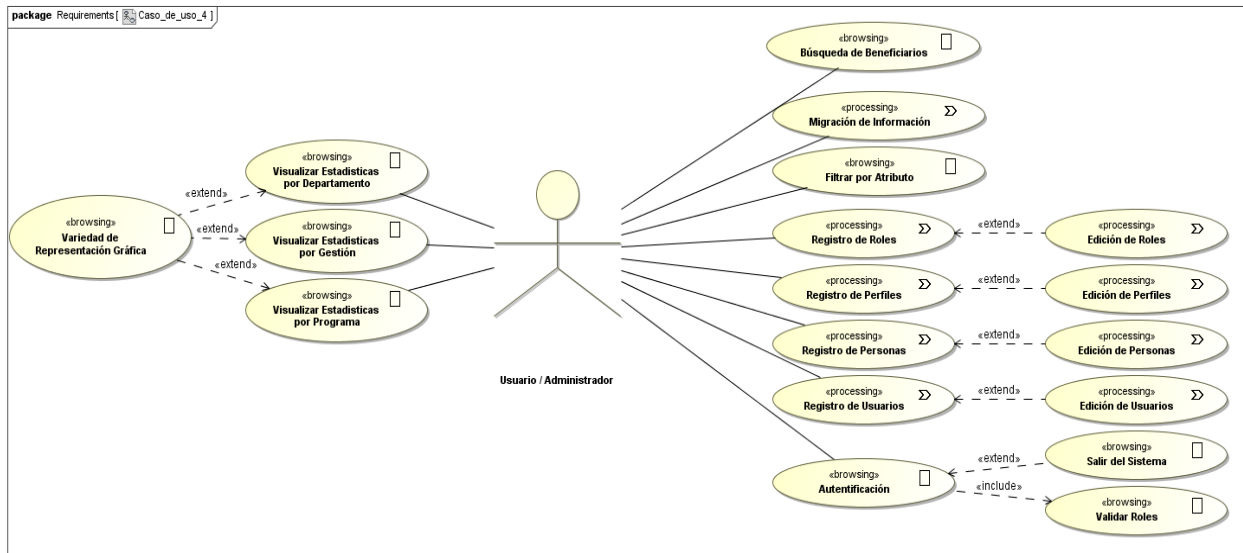


Figura 3.22 Sprint 4 Análisis de Requerimientos

Fuente: Elaboración Propia

c) MODELO CONCEPTUAL

Se tiene un modelo conceptual que representa al cuarto módulo. (Ver figura 3.23)

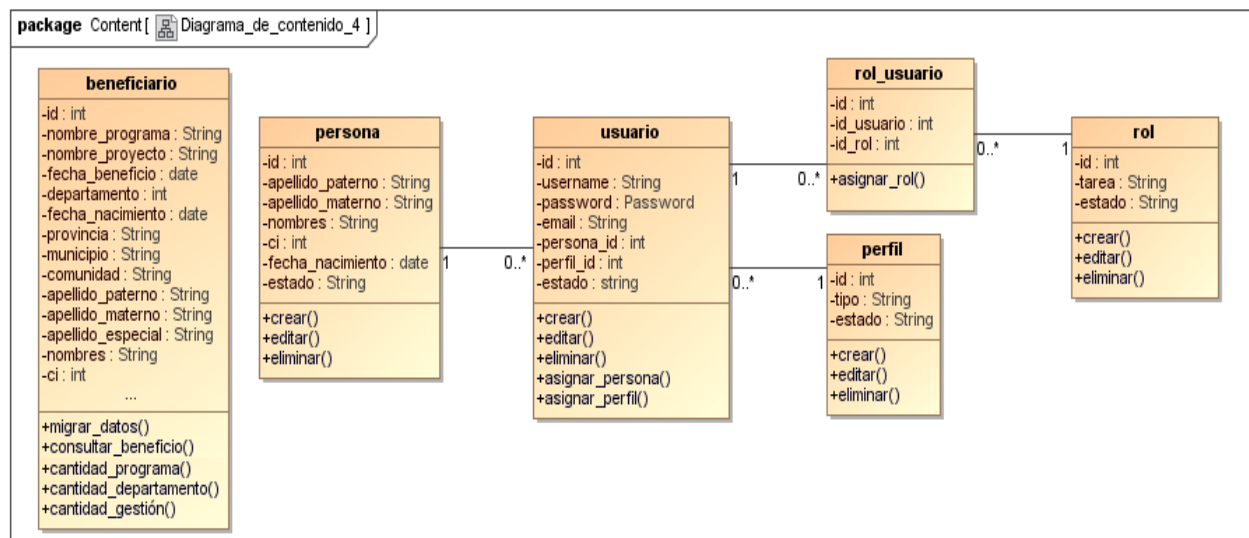


Figura 3.23 Sprint 4 Modelo Conceptual

Fuente: Elaboración Propia

d) MODELO NAVEGACIONAL

Se representa el modelo navegacional que fue diseñado para el cuarto módulo del sistema. (Ver figura 3.24)

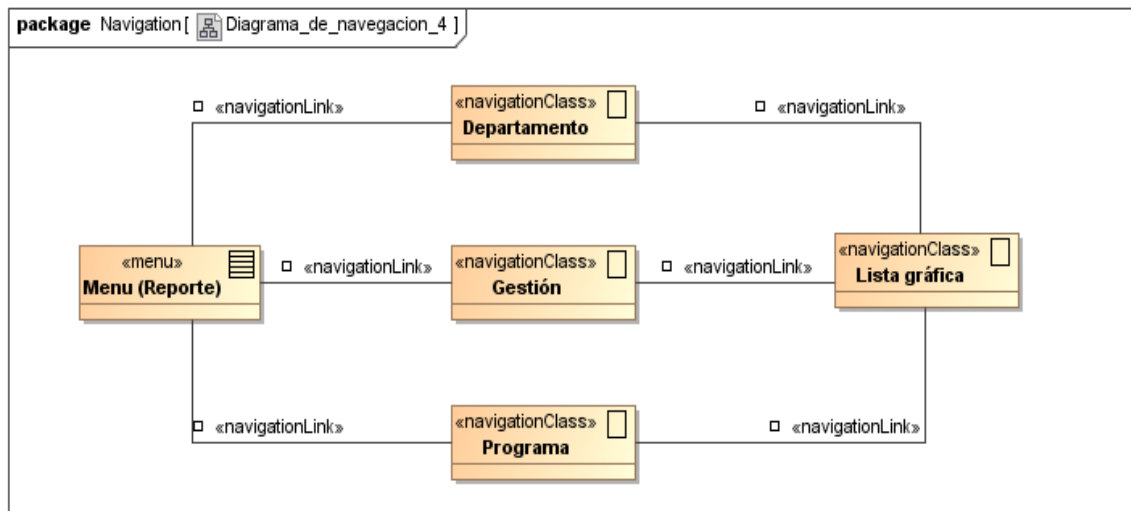


Figura 3.24 Sprint 4 Modelo Navegacional

Fuente: Elaboración Propia

e) MODELO DE PRESENTACIÓN

Aquí se representa el modelo de presentación que fue diseñado para el cuarto módulo del sistema. (Ver figura 3.25)

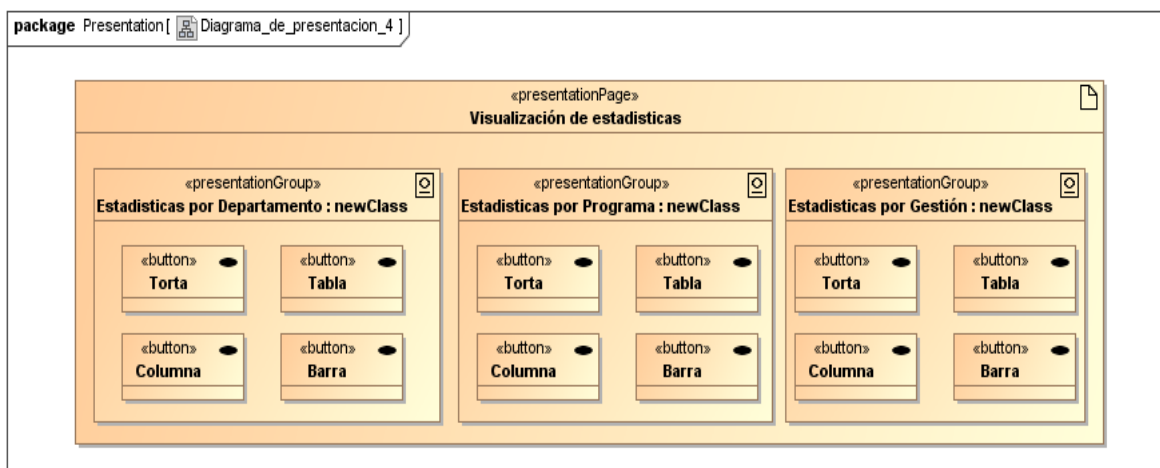


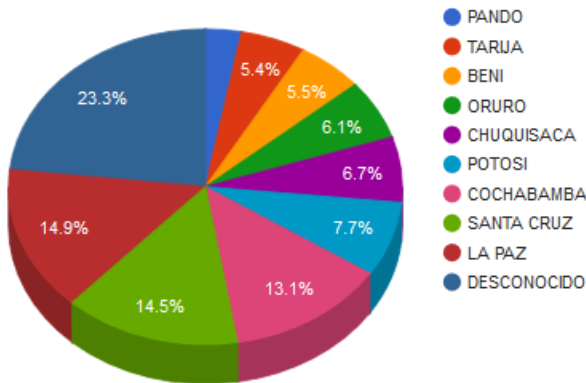
Figura 3.25 Sprint 4 Modelo de Presentación

Fuente: Elaboración Propia

f) REVISIÓN DEL SPRINT: SPRINT 4

Se realizó la correspondiente inspección del progreso del cuarto sprint, y a continuación, se muestra los gráficos estadísticos de los beneficiarios. (Ver figura 3.26)

Beneficios del Programa AEVIVIENDA



Aevivienda	
Departamento	Beneficios
PANDO	6058
TARJA	11019
BENI	11269
ORURO	12614
CHUQUISACA	13789
POTOSI	15763
COCHABAMBA	27037
SANTA CRUZ	29750
LA PAZ	30633
DESCONOCIDO	47898

Figura 3.26 Sprint 4 Reporte por departamento

Fuente: Elaboración Propia

La captura de pantalla que se hizo del producto, muestra gráficos estadísticos por gestión. (Ver figura 3.27)

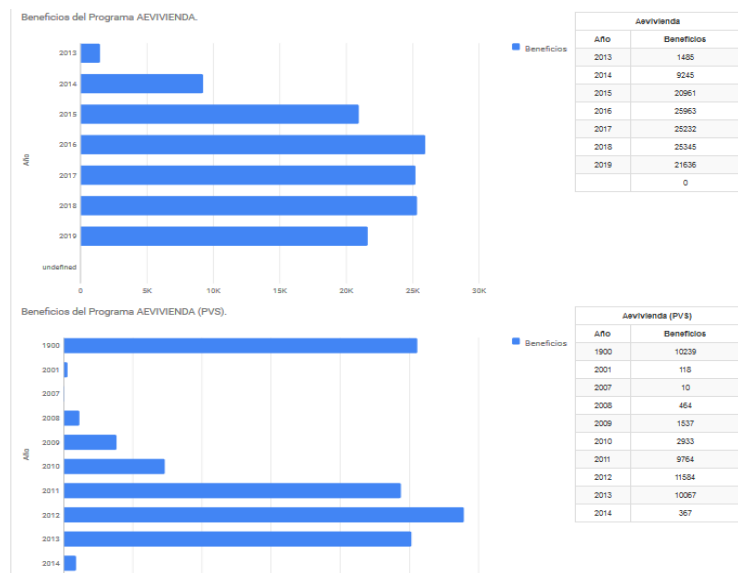


Figura 3.27 Sprint 4 Reporte por Gestión

Fuente: Elaboración Propia

g) PRUEBA UNITARIA

En esta fase se verificó que los objetivos del sprint 4 están implementados de manera apropiada y correcta. (ver Tabla 3.9)

PRUEBA UNITARIA SPRINT 4	
Descripción	Visualización de datos estadísticos con una variedad de representaciones gráficas.
Objetivos	1. Visualizar por programa. 2. Visualizar por gestión. 3. Visualizar por departamento.
Condiciones	Servidor ejecutándose, ingresar al sistema con los datos correctos y permisos disponibles.
Resultado Esperado	Ejecución satisfactoria de las tareas descritas en el Sprint 4, por parte del usuario.
Resultado Obtenido	El usuario puede visualizar correctamente los datos estadísticos y también representarlos gráficamente de varias maneras.

Tabla 3.9 Sprint 4 Prueba Unitaria

Fuente: Elaboración Propia

3.2.2.5 PLANIFICACIÓN DEL SPRINT: SPRINT 5

El módulo de reportes se toma en cuenta para la planificación de quinto sprint, basándose en la lista de requerimientos del Product Backlog (ver Tabla 3.10)

ID	TAREA	MÓDULO	SPRINT	PRIORIDAD
R5 - 1	Reporte válido de certificado de existencia de beneficio	Reportes	5	Alta
R5 - 2	Reporte inválido de existencia de beneficio	Reportes	5	Media

Tabla 3.10 Sprint 5 Backlog

Fuente: Elaboración Propia

a) DESARROLLO DEL SPRINT: SPRINT 5

Se aplicó diseños simples con los principios de UWE para el desarrollo del quinto sprint.

b) ANÁLISIS DE REQUERIMIENTOS

Se elaboró un modelo de casos de uso que representa al quinto sprint. (Ver figura 3.28)

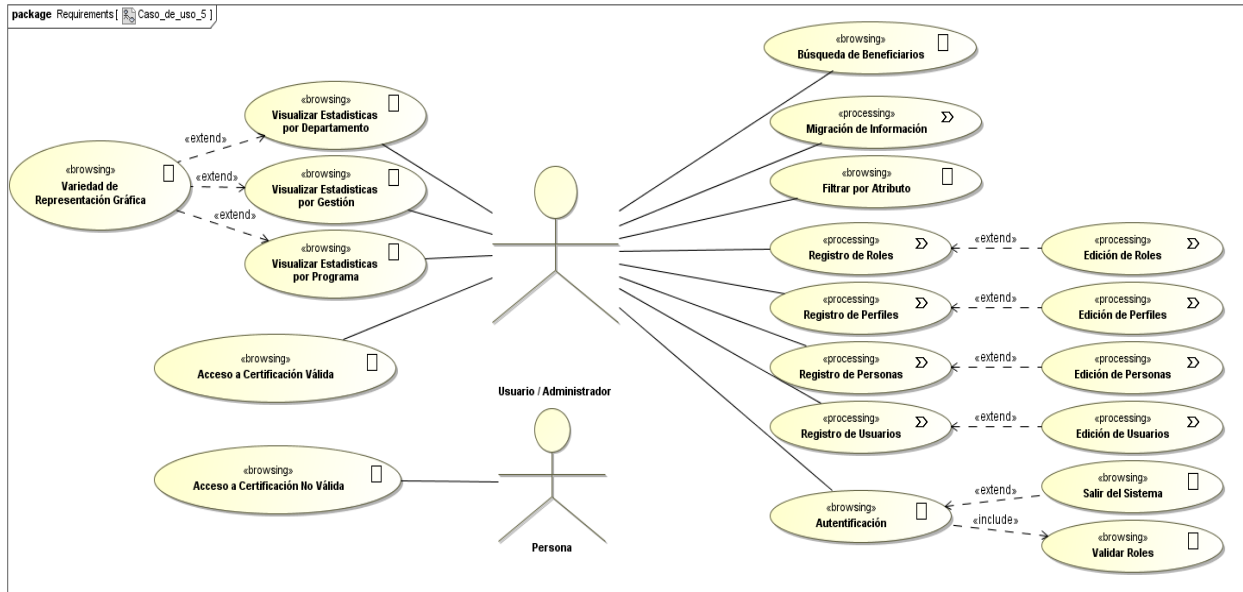


Figura 3.28 Sprint 5 Análisis de Requerimientos
Fuente: Elaboración Propia

c) MODELO CONCEPTUAL

Se tiene un modelo conceptual que representa al quinto módulo. (Ver figura 3.29)

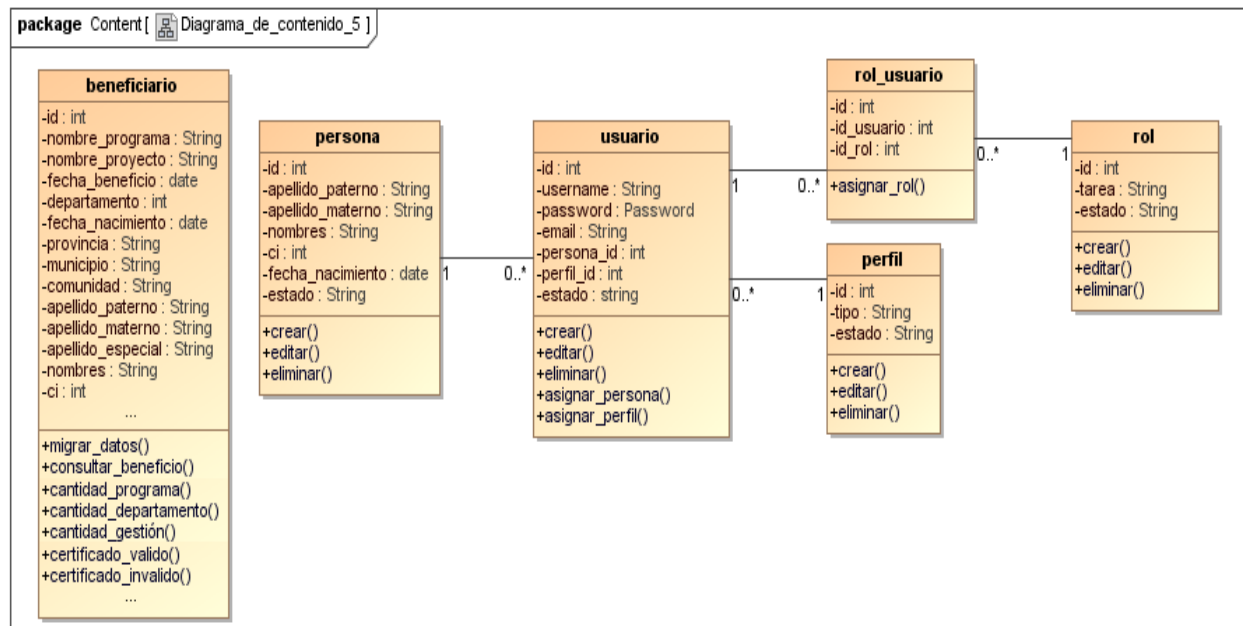


Figura 3.29 Sprint 5 Modelo Conceptual
Fuente: Elaboración Propia

d) MODELO NAVEGACIONAL

Se representa el modelo navegacional que fue diseñado para el quinto módulo del sistema. (Ver figura 3.30)

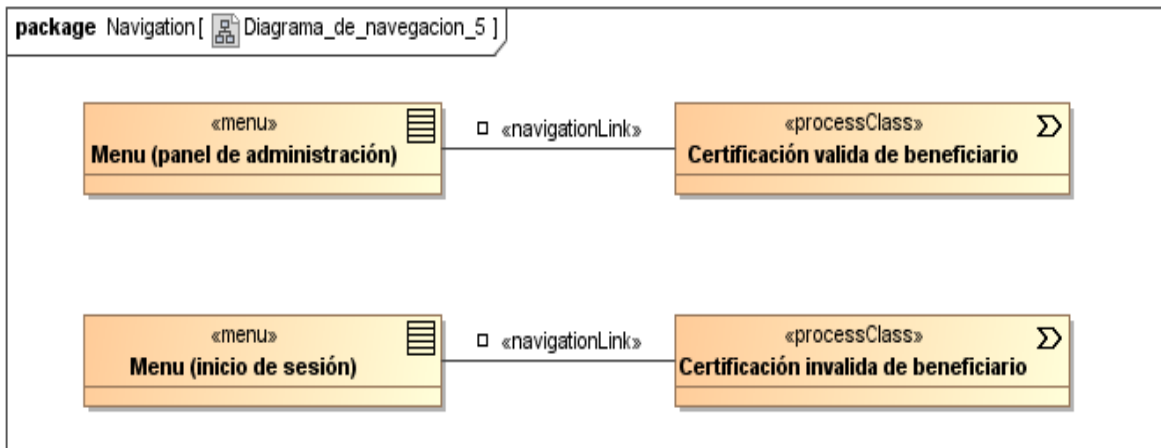


Figura 3.30 Sprint 5 Modelo Navegacional

Fuente: Elaboración Propia

e) MODELO DE PRESENTACIÓN

Aquí se representa el modelo de presentación que fue diseñado para el quinto módulo del sistema. (Ver figura 3.31)

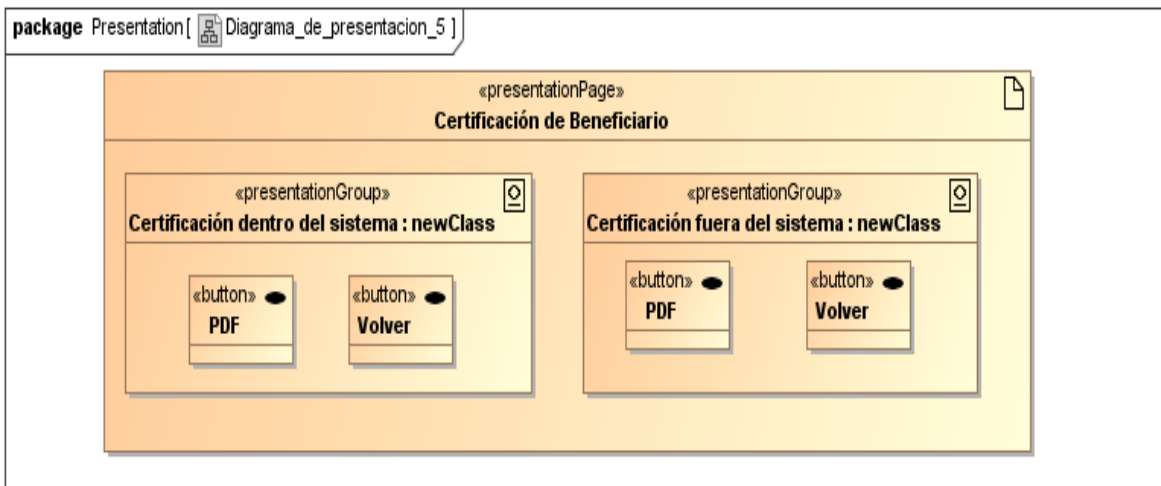


Figura 3.31 Sprint 5 Modelo de Presentación

Fuente: Elaboración Propia

f) **REVISIÓN DEL SPRINT: SPRINT 5**

Se realizó la correspondiente inspección del progreso del quinto sprint, y a continuación, se muestra el reporte cuando se hace un usuario autenticado y con permisos. (Ver figura 3.32)

ESTADO PLURINACIONAL DE BOLIVIA
MINISTERIO DE OBRAS PUBLICAS SERVICIOS Y VIVIENDA
VICEMINISTERIO DE VIVIENDA Y URBANISMO
CERTIFICADO DE PROGRAMAS DE VIVIENDAS ESTATALES

DATOS DEL SOLICITANTE

Apellido Paterno	FERNANDEZ	Apellido Materno	ALI
Nombre	PEDRO ARIEL	Cédula de Identidad	7991176

RESULTADOS EN PROYECTOS : PROREVI
El/la solicitante NO ha sido beneficiado(a) anteriormente.

RESULTADOS EN PROYECTOS : TITULACION
El/la solicitante NO ha sido beneficiado(a) anteriormente.

RESULTADOS EN PROYECTOS : AEVIENDA
El/la solicitante NO ha sido beneficiado(a) anteriormente.

RESULTADOS EN PROYECTOS : IN1000
El/la solicitante NO ha sido beneficiado(a) anteriormente.

RESULTADOS EN PROYECTOS : RECONSTRUCCION
El/la solicitante NO ha sido beneficiado(a) anteriormente.

Nota: El presente reporte muestra el estado de datos de beneficiarios con los distintos programas de Vivienda del Viceministerio de Vivienda y Urbanismo.

Figura 3.32 Sprint 5 Certificación Válida

Fuente: Elaboración Propia

La captura de pantalla que se hizo del producto, muestra la certificación inválida que da el sistema cuando una persona no está autenticada. (Ver figura 3.33)

ESTADO PLURINACIONAL DE BOLIVIA
MINISTERIO DE OBRAS PUBLICAS SERVICIOS Y VIVIENDA
VICEMINISTERIO DE VIVIENDA Y URBANISMO
CERTIFICADO DE PROGRAMAS DE VIVIENDAS ESTATALES

DATOS DEL SOLICITANTE

Apellido Paterno	FERNANDEZ	Apellido Materno	ALI
Nombre	PEDRO ARIEL	Cédula de Identidad	7991176

RESULTADOS EN PROYECTOS : PROREVI
El/la solicitante NO ha sido beneficiado(a) anteriormente.

RESULTADOS EN PROYECTOS : TITULACION
El/la solicitante NO ha sido beneficiado(a) anteriormente.

RESULTADOS EN PROYECTOS : AEVIENDA
El/la solicitante NO ha sido beneficiado(a) anteriormente.

RESULTADOS EN PROYECTOS : IN1000
El/la solicitante NO ha sido beneficiado(a) anteriormente.

RESULTADOS EN PROYECTOS : RECONSTRUCCION
El/la solicitante NO ha sido beneficiado(a) anteriormente.

Nota: El presente reporte muestra el estado de datos de beneficiarios con los distintos programas de Vivienda del Viceministerio de Vivienda y Urbanismo.

Figura 3.33 Sprint 5 Certificación Inválida

Fuente: Elaboración Propia

g) PRUEBA UNITARIA

En esta fase se verificó que los objetivos del sprint 5, referente a los reportes que brindará el sistema están implementados de manera apropiada y correcta. (ver Tabla 3.11)

PRUEBA UNITARIA SPRINT 5	
Descripción	Mostrar una certificación válida o inválida dependiendo la persona que este manejando.
Objetivos	1. Certificado Válido. 2. Certificado Inválido.
Condiciones	Servidor ejecutándose, ingresar al sistema con los datos correctos y permisos disponibles.
Resultado Esperado	Ejecución satisfactoria de las tareas descritas en el Sprint 5, por parte del usuario.
Resultado Obtenido	Si fuera un usuario con los permisos correspondientes, accede a la certificación de manera satisfactoria, si no es un usuario autenticado brinda un certificado inválido.

Tabla 3.11 Sprint 5 Prueba Unitaria

Fuente: Elaboración Propia

Finalmente, se hará un sondeo general de las tablas que se tomaron en cuenta por cada Sprint, para la posterior representación de la estructura de la base de datos.

- Sprint 1 Autenticación. usuarios, rol_usuarios, roles, logs.
- Sprint 2 Administración. usuarios, rol_usuarios, roles, perfiles, logs, activity_logs.
- Sprint 3 Servicios y Consulta. usuarios, rol_usuarios, roles, activity_logs.
- Sprint 4 Estadísticas. usuarios, rol_usuarios, roles, beneficiarios.
- Sprint 5 Reportes. usuarios, rol_usuarios, roles, firmas, certificados.

Al finalizar todos los Sprints y como se vio en los casos de uso y también en los diagramas de contenido, desde el primer Sprint hasta el último, se fue construyendo paso a paso un esquema general donde se puede apreciar las funcionalidades que requiere el sistema.

Es así que, al cumplir con los requerimientos establecidos, y al no existir posibles cambios en la base de datos, se procederá a representar la estructura. (Ver figura 3.34)

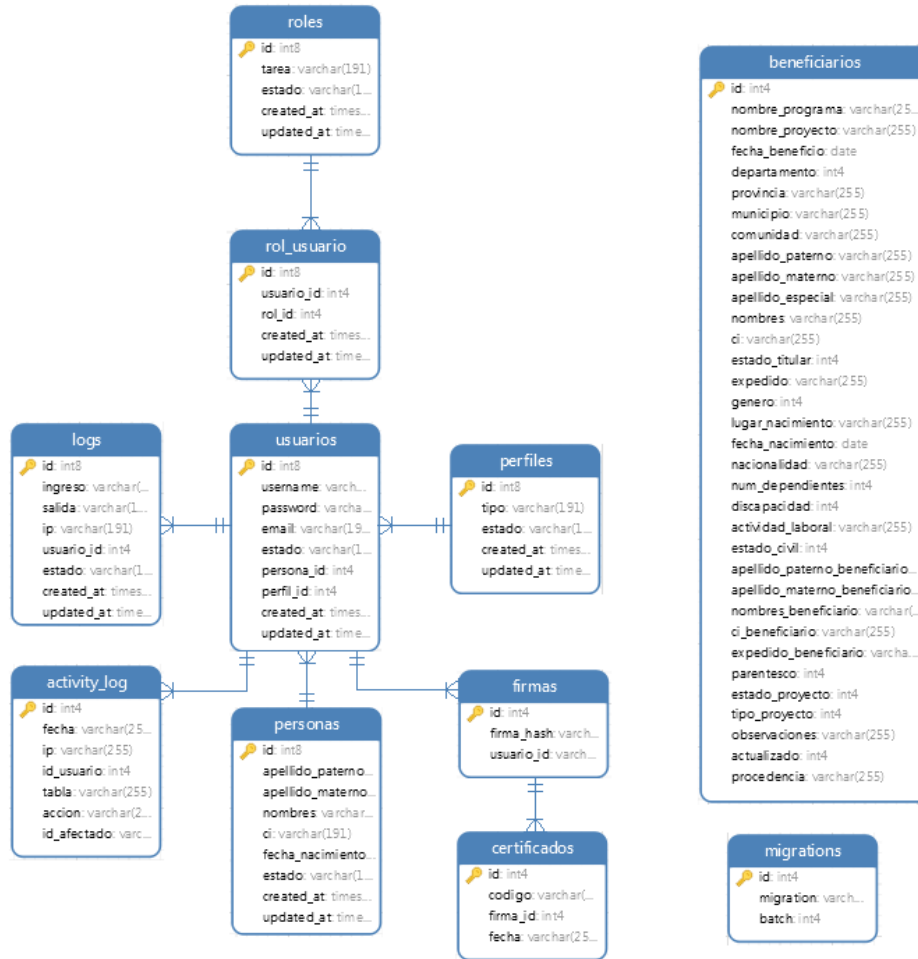


Figura 3.34 Base de Datos
Fuente: Elaboración Propia

3.2.3 POSTGAME

Una vez verificados que los objetivos descritos en el Product Backlog fueran cumplidos, en esta fase se procedió a someter al sistema a varias pruebas de stress

3.2.3.1 PRUEBA DE STRESS

Esta prueba es aplicada para poner a prueba la fortaleza y fiabilidad del sistema sometiéndolo a condiciones de uso extremas. Un buen plan de pruebas de stress debe contemplar el desarrollo de no uno, sino varios casos de stress. Cada caso diferirá en el volumen del estímulo a aplicar sobre la aplicación (cantidad de usuarios, cantidad de peticiones, etc.), el tiempo que durará cada estímulo y la duración total del experimento, entre otras variables.

A continuación, se muestra los resultados de la prueba de transferencia de datos, memoria del sistema y carga del CPU:

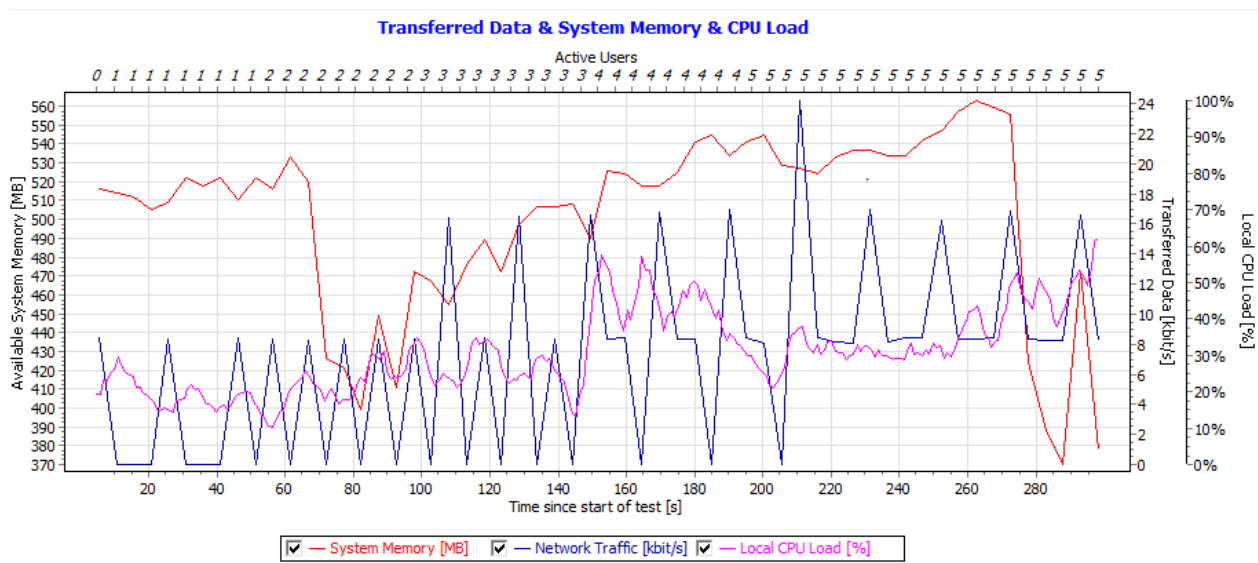


Figura 3.35 Prueba de Stress

Fuente: Elaboración Propia

Inicialmente se puede observar que el uso de memoria estuvo con una proporción alta, luego fue estabilizándose, para finalmente estar con un bajo uso de la memoria del sistema, por otra parte, el tráfico de red empezó con una proporción media para luego bajar y subir de forma frecuente, debido a las constantes solicitudes de servicios y consultas que tiene el sistema, y finalmente el uso local de la CPU se mantuvo con una proporción media.

Esta prueba de stress fue realizada con la aplicación “Web Server Stress Tool 8” donde se tiene puede resaltar los siguientes resultados:

- Memoria del sistema 5%.
- Tráfico de red 35%.
- Carga local de CPU 65%.
- 5 usuarios activos.
- 5 minutos de prueba.

De la prueba de Stress realizada al sistema pudo concluirse que la capacidad de tiempo de respuesta era estable para cada usuario.

CAPÍTULO IV

CALIDAD Y SEGURIDAD

4.1 INTRODUCCIÓN

En el presente capítulo se realizará y comprobará la calidad y seguridad del sistema web que se desarrolló para el Viceministerio de Vivienda y Urbanismo.

4.2 FACTORES DE CALIDAD

Hablar de calidad del software implica la necesidad de contar con parámetros que permitan establecer los niveles mínimos que un producto debe alcanzar para que se considere de calidad.

El problema es que la mayoría de las características que definen al software no se pueden cuantificar fácilmente; generalmente, se establecen de forma cualitativa, lo que dificulta su medición, ya que se requiere establecer métricas que permitan evaluar cuantitativamente cada característica dependiendo del tipo de software que se pretende calificar. En este sentido se han realizado muchos trabajos que establecen propuestas para el establecimiento de los factores cualitativos que afectan la calidad del software (ISO-9126, 2012).

4.3 WEBSITE QEM

La estrategia propuesta, denominada Metodología de Evaluación de Calidad de Sitios Web propone un enfoque sistemático, disciplinado y cuantitativo que se adecue a la evaluación, comparación y análisis de calidad de sistemas de información centrados en la Web, éste mismo está basado en las normas de calidad de la ISO 9126. Estas características de alto nivel son: usabilidad, funcionalidad, confiabilidad, eficiencia, portabilidad, y mantenibilidad. (Olsina, 1999)

- **Usabilidad:** Se define como un conjunto de atributos que otorgan el esfuerzo necesario para su uso, y en la evaluación individual de dicho uso, mediante un conjunto de usuarios declarados implícitos.
- **Funcionalidad:** Se define como un conjunto de atributos que otorgan la existencia de un conjunto de funciones y sus propiedades específicas. Las funciones son aquellas que satisfacen conjuntos de usuarios declarados implícitos.
- **Confiabilidad:** Se define como un conjunto de atributos de habilidad del software para mantener condiciones de establecer su propio nivel de desempeño por un periodo determinado.

- **Eficiencia:** Se define como un conjunto de atributos que otorgan la relación entre nivel de rendimiento del software y la cantidad de recursos usados por el usuario, bajo las condiciones establecidas.
- **Mantenibilidad:** Se define como un conjunto de atributos que otorgan el esfuerzo necesario para hacer modificaciones específicas.
- **Portabilidad:** Se define como un conjunto de atributos que otorgan la habilidad de software para ser transferido de un entorno a otro. (Olsina, 1999)

4.3.1 CARACTERÍSTICAS QEM

Se especifican las características de usabilidad, funcionabilidad, confiabilidad y eficiencia.

4.3.1.1 USABILIDAD

Según Olsina (1999) consiste de un conjunto de atributos que permiten evaluar el esfuerzo necesario que deberá invertir el usuario para utilizar el sistema web. Es una característica de calidad de producto de alto nivel, que se la puede medir mediante cálculo a partir de métricas directas e indirectas, y representa la capacidad del producto para ser utilizado. El criterio de evaluación es un criterio binario, discreto y absoluto: sólo se pregunta si está disponible (1) o si no está disponible (0). Para evaluar la usabilidad se deben considerar las siguientes características:

- **Comprensibilidad global del sitio:** Es una característica que representa a todas aquellas facilidades que permiten a la audiencia, tener una rápida comprensión tanto de la estructura organizativa, como del contenido del sitio Web como un todo.

Característica: Comprensibilidad Global del Sitio			
Nro.	Subcaracterística		Resultado
1	Esquema de Organización Global		0,67
	1.1	Mapa del sitio	1,00
	1.2	Tabla de contenidos	1,00
	1.3	Indice alfabético	0,00
2	Calidad en el sistema etiquetado		1,00
3	Visita guiada orientada al usuario		1,00
4	Mapa de imagen		1,00
TOTAL			0,92

Tabla 4.1 Evaluación de Comprensibilidad

Fuente: Elaboración Propia

El valor obtenido es de **92%** el cual representa el grado de Comprensibilidad del Sitio por parte del usuario al navegar por el sistema.

- **Mecanismos de ayuda y retroalimentación en línea:** Este atributo representa a un conjunto de preguntas (agrupadas y enlazadas) que se realizan con mayor frecuencia, y que están ya publicadas en el sitio con sus respectivas respuestas. A su vez, las respuestas pueden estar enlazadas a otros contenidos. Esto favorece al mecanismo de aprendizaje y/o ayuda, evitando potencialmente la demora cognitiva de los visitantes.

Característica: Mecanismos de Ayuda y Retroalimentación en Línea			
Nro.	Subcaracterística		Resultado
1	Calidad de la ayuda		1,00
	1.1	Ayuda explicada orientada al usuario	1,00
	1.2	Ayuda de la búsqueda	1,00
2	Indicador de última actualización		1,00
	2.1	Global	0,00
	2.2	Restringido	1,00
TOTAL			0,92

Tabla 4.2 Evaluación de Mecanismos y Retroalimentación

Fuente: Elaboración Propia

El valor obtenido es de **92%** el cual representa el grado de Evaluación de Mecanismos de Ayuda y Retroalimentación por parte del usuario al navegar por el sistema.

- **Aspectos de interfaces y estéticos:** Son factores y elementos relativos a la interacción del usuario, enfocados a un entorno o dispositivo concretos cuyo resultado es la generación de una percepción positiva o negativa de dicho servicio, producto o dispositivo.

Característica: Aspectos de Interfaces y Estéticos			
Nro.	Subcaracterística		Resultado
1	Cohesión al agrupar los objetos de control principales		1,00
2	Permanencia y estabilidad de los controles principales		1,00
	2.1	Permanencia de los controles directos	1,00
	2.2	Permanencia de los controles indirectos	1,00
	2.3	Estabilidad	1,00
3	Aspectos de estilo		1,00
	3.1	Uniformidad en el color de enlaces	1,00
	3.2	Uniformidad en el estilo global	1,00
	3.3	Guía del estilo global	1,00

4	Preferencia estética	1,00
TOTAL		1,00

Tabla 4.3 Evaluación de Aspectos de Interfaz y Estéticos

Fuente: Elaboración Propia

El valor obtenido es de **100%** el cual representa el grado de Evaluación de Aspectos de Interfaces y Estéticos por parte del usuario al navegar por el sistema.

- **Misceláneas:** Este atributo modela el número de lenguajes extranjeros soportados por un sitio (sitios de dominios de aplicación de índole académica, museos, comercio electrónico y otros). Además, especifica el nivel de soporte para cada lenguaje: Total (todas las páginas del sitio), parcial (algunos subsitios del sitio), o mínimo (algunas páginas o documentos de algunos subsitios). No se computa obviamente el lenguaje nativo del sitio Web, como lenguaje extranjero.

Característica: Miscelaneas		
Nro.	Subcaracterística	Resultado
1	Soporte lenguaje extranjero	0,00
2	Atributo que es lo nuevo	1,00
3	Indicador de resolución de pantalla	1,00
TOTAL		0.67

Tabla 4.4 Evaluación de Misceláneas

Fuente: Elaboración Propia

El valor obtenido es de **67%** el cual representa el grado de Evaluación de Misceláneas por parte del usuario al navegar por el sistema.

La usabilidad de la aplicación evaluada estará determinada por el promedio de las características mencionadas como se muestra en la siguiente tabla:

Nro	Criterio	Resultado
1	Comprensibilidad global del sitio	0,92
2	Mecanismos de ayuda y retroalimentacion en línea	0,92
3	Aspectos de interfaces y esteticos	1,00
4	Misceláneas	0,67
Evaluación total de Usabilidad		0,87

Tabla 4.5 Evaluación Total de Usabilidad

Fuente: Elaboración Propia

Finalmente, el valor obtenido es de **87%** el cual representa el grado de Evaluación Total de Usabilidad por parte del usuario.

4.3.1.2 FUNCIONALIDAD

Para determinar la calidad de la funcionalidad de la aplicación se debe analizar la búsqueda y exploración de contenidos.

El criterio de evaluación es un criterio binario, discreto y absoluto. Sólo se pregunta si está disponible (1) o si no está disponible (0).

Para evaluar la funcionalidad se deben considerar las siguientes características:

- **Aspectos de búsqueda y recuperación:** Es una característica que modela el mecanismo y que permite tener un modo directo de encontrar información.

Característica: Aspectos de Búsqueda y Recuperación		
Nro.	Subcaracterística	Resultado
1	Búsqueda en el sitio web sin autenticación	1,00
	1.1 Búsqueda de beneficiarios restringida	1,00
2	Búsqueda en el sitio web con autenticación	1,00
	2.1 Búsqueda de beneficiarios sin restricción	1,00
	2.2. Búsqueda de usuarios	1,00
	2.3 Búsqueda de roles	1,00
	2.4 Búsqueda de personas	1,00
	2.5 Búsqueda de perfiles	1,00
3	Recuperación de datos de servicios web	0.67
	3.1 Aevivienda	1,00
	3.2 Prorevi	1,00
	3.3 Titulación	0,00
TOTAL		0.89

Tabla 4.6 Evaluación de Búsqueda y Recuperación

Fuente: Elaboración Propia

El valor obtenido es de **89%** el cual representa el grado de Evaluación de Búsqueda y Recuperación de Datos por parte del usuario.

- **Aspectos de navegación y exploración:** Facilidad con la que un usuario puede desplazarse por todas las páginas que componen un sitio web.

Característica: Aspectos de Navegación y Exploración		
Nro.	Subcaracterística	Resultado
1	Navegabilidad	1,00
	1.1 Orientación	1,00
	1.2 Promedio de enlaces por página	1,00
2	Objetos de control de navegaciones	1,00
	2.1 Permanencia en la presentación de controles contextuales	1,00
	2.2 Estabilidad	1,00
3	Nivel de desplazamiento	1,00
	3.1 Desplazamiento vertical	1,00
	3.2 Desplazamiento horizontal	1,00
4	Predicción navegacional	1,00
	4.1 Enlace con título	1,00
	4.2 Calidad de la fase de enlace	1,00
TOTAL		1,00

Tabla 4.7 Evaluación de Navegación y Exploración

Fuente: Elaboración Propia

El valor obtenido es de **100%** el cual representa el grado de Evaluación de Navegación y Exploración por parte del usuario.

- **Aspectos de dominio orientados al usuario:** Se refieren a la idoneidad enciclopédica de los temas de los artículos, pero no limitan directamente su contenido.

Característica: Aspectos de Dominio Orientado al Usuario		
Nro.	Subcaracterística	Resultado
1	Relevancia de contenido	1,00
2	Servicios on-line	1,00
TOTAL		1,00

Tabla 4.8 Evaluación de Dominio Orientado al Usuario

Fuente: Elaboración Propia

El valor obtenido es de **100%** el cual representa el grado Aspectos de Dominio Orientado al Usuario.

La funcionalidad de la aplicación evaluada estará determinada por el promedio de las características anteriormente mencionadas, tomando los resultados de los tres aspectos como muestra la siguiente tabla:

Nro	Criterio	Resultado
1	Aspectos de búsqueda y recuperación	0,89
2	Aspectos de navegación y exploración	1,00
3	Aspectos del dominio orientados al usuario	1,00
Evaluación total de Funcionalidad		0.97

Tabla 4.9 Evaluación Total de Funcionalidad

Fuente: Elaboración Propia

Finalmente, el valor obtenido es de **97%** el cual representa el grado de Evaluación Total de la Funcionalidad del Sistema.

4.3.1.3 CONFIABILIDAD

La medición de esta característica está definida por el complemento de los casos de deficiencia encontrados en la aplicación. Es un criterio de variable normalizada, continuo y absoluto; en donde si BL = Número de enlaces rotos encontrados, TL = Número total de enlaces del sitio, la fórmula para computar la variable será: $X = 100 - (BL * 100/TL) * 10$; donde, si $X < 0$ entonces $X = 0$.

- **No deficiencia y control de páginas:** Este atributo representa básicamente la ausencia de los enlaces encontrados que conducen a nodos destinos inaccesibles, también se toma la ortografía en las páginas.

Característica: Confiabilidad		
Nro.	Subcaracterística	Resultado
1	No deficiencia	1,00
	1.1 Errores de enlaces rotos	0,00
	1.2 Errores de enlaces inválidos	0,00
	1.3 Errores no implementados	0,00
	1.4 Errores de conexión con servicio web como servidor	0,00
	1.5 Errores de conexión con servicio web como cliente	0,00
2	Páginas	1,00
	2.1 Páginas muertas	0,00
	2.2 Errores de ortografía	0,00
	2.3 Páginas bajo construcción	0,00
TOTAL		1,00

Tabla 4.10 Evaluación Confiabilidad

Fuente: Elaboración Propia

El valor obtenido es de **100%** el cual representa el grado Aspectos de Confiabilidad del Sistema de parte del Usuario.

4.3.1.4 EFICIENCIA

Es una característica de calidad de producto de alto nivel que puede medirse mediante cálculo a partir de métricas directas e indirectas, y principalmente representa a la relación entre el grado de performance del artefacto y la cantidad de recursos (tiempo, espacio, entre otros) usados bajo ciertas condiciones. El criterio de evaluación es un criterio binario, discreto y absoluto: Sólo se pregunta si está disponible (1) o si no está disponible (0). Para evaluar la eficiencia se deben considerar las siguientes características:

- **Desempeño:** Se mide el tamaño de todas las páginas (estáticas) del sitio web. El tamaño de cada página se especifica como una función del tiempo de espera y de la velocidad mínima establecida por una línea de comunicación dada.

Característica: Desempeño		
Nro.	Subcaracterística	Resultado
1	Páginas de acceso rápido	1,00
TOTAL		1,00

Tabla 4.11 Evaluación Desempeño

Fuente: Elaboración Propia

El valor obtenido es de **100%** el cual representa el grado de Aspectos de Desempeño del Sistema, evaluado por el Usuario.

- **Accesibilidad:** Este atributo representa la accesibilidad a la información que está en las páginas. Es de relevancia que el sitio entero sea editado.

Característica: Accesibilidad		
Nro.	Subcaracterística	Resultado
1	Accesibilidad de la información	1,00
	1.1 Soporte a versión solo texto	1,00
	1.1 Legibilidad	1,00
2	Accesibilidad de ventanas	1,00
TOTAL		1,00

Tabla 4.12 Evaluación Accesibilidad

Fuente: Elaboración Propia

El valor obtenido es de **100%** el cual representa el grado de Accesibilidad del Sistema, evaluado por el Usuario.

La eficiencia de la aplicación evaluada estará determinada por el promedio de las características anteriormente mencionadas, que son el desempeño y la accesibilidad, como muestra la siguiente tabla:

Nro	Criterio	Resultado
1	Desempeño	1,00
2	Accesibilidad	1,00
Evaluación total de Eficiencia		1,00

Tabla 4.13 Evaluación Total de Eficiencia

Fuente: Elaboración Propia

Finalmente, el valor obtenido es de **100%** el cual representa el grado de Evaluación Total de la Eficiencia del Sistema.

4.3.2 RESULTADOS

La calidad total de la aplicación web estará determinada por el promedio de las características de usabilidad, funcionalidad, confiabilidad y eficiencia como muestra la siguiente tabla:

Nro	Criterio	Resultado
1	Usabilidad	0.87
2	Funcionalidad	0.97
3	Confiabilidad	1,00
4	Eficiencia	1,00
Evaluación Total de Calidad		0.96

Tabla 4.14 Evaluación Total de Calidad

Fuente: Elaboración Propia

WebSite QEM indica que el nivel de aceptabilidad insatisfactorio para la calidad total de la aplicación web se encuentra en el rango de 0-40, el nivel de aceptabilidad marginal entre 41-60 y el nivel de aceptabilidad satisfactorio se encuentra entre 61-100 por ciento.

El nivel de calidad total del Sistema implementado, es de un **96%** indicando que la calidad del proyecto es satisfactoria. Con este resultado obtenido de la evaluación, se concluye que: 96 usuarios de 100 consideran al Sistema con un nivel de aceptación satisfactorio.

4.4 SEGURIDAD DEL SISTEMA

Una de las preocupaciones más importantes de en el desarrollo de un sistema web es la seguridad, en esta fase se mostrará la aplicación de métodos para mantener la seguridad del sistema.

4.4.1 AUTENTIFICACIÓN

La autenticación de usuarios es una de las formas más comunes y a la vez obligatorias para la protección y acceso al sistema. En este caso el proceso de autenticación englobó el almacenamiento de datos de usuario y su posterior comprobación mediante un formulario en la parte de inicio de sesión.

Los criterios de seguridad aplicados para la autenticación fueron:

- Nombre de Usuario. Debe ser un valor único, no puede existir otro usuario con el mismo nombre de usuario.
- Contraseña. Se aplicó un algoritmo para la encriptación de las contraseñas.
- Estado. Donde solo los usuarios con determinado valor podrán acceder al sistema.
- Verificación de roles. Brindará acceso total o limitado a las funciones del sistema.

A continuación, se muestra un registro en la base de datos referentes a cómo se maneja

username	password	email	estado	persona_id ▲	perfil_id	created_at	updated_at
ariel.fernandez	\$2y\$10\$M5spd96Jx/CCyN!	ariel@gma	Activo	1	1	(Null)	2019-10-06 20:46

Figura 4.1 Registro de Usuario

Fuente: Elaboración Propia

4.4.2 SOLICITUDES DE FALSIFICACIÓN

Un ataque *CSRF* (*Cross-Site Request Forgery*) fuerza al navegador web validado a enviar una petición a una aplicación web vulnerable, la cual entonces realiza la acción elegida a través de la víctima. Para proteger al sistema de este tipo de ataques, se propone el uso de tokens *CSRF* del *framework* Laravel, estos se utilizan para garantizar que terceros no puedan iniciar tal solicitud, generando un token que debe ser filtrado junto con el contenido del formulario, este token realizará una comprobación con el valor adicionalmente guardado en la sesión del usuario. Si coincide, la solicitud es considerada válida, de lo contrario es considerada inválida. (Auger, 2010)

```

<form method="post" action="{{ url('rol/guardar') }}" class="form-horizontal">
  {{ csrf_field() }}
  <div class="form-body">
    <div class="form-group form-md-line-input">
      <label class="col-md-2 control-label">Tarea</label>
      <div class="col-md-10">
        <input type="text" class="form-control" id="tarea" name="tarea" minlength="3" maxlength="20">
        <div class="form-control-focus"> </div>
      </div>
    </div>
  </div>
  <div class="form-actions">
    <div class="row">
      <div class="col-md-offset-2 col-md-10">
        <a class="btn default" href="{{ url('home') }}">Cancelar</a>
        <button class="btn blue">Guardar</button>
      </div>
    </div>
  </div>
</form>

```

Figura 4.2 Implementación de token CSRF

Fuente: Elaboración Propia

4.4.3 INYECCIÓN SQL

La inyección SQL es una técnica donde un atacante crea o altera comandos SQL existentes para exponer datos ocultos, sobrescribir datos valiosos, o peor aún, ejecutar comandos peligrosos a nivel de sistema del equipo que hospeda la base de datos. Esto se logra a través de la práctica de tomar la entrada del usuario y combinarla con parámetros estáticos para elaborar una consulta SQL. Muchos de los mecanismos de Laravel para acceso a datos ya tienen implícitos sistemas de filtrado de información, para la protección de posibles ataques de inyección SQL se hace uso del “bindeo” de parámetros en las consultas. Al realizar el “*binding*” del parámetro usando el procedimiento, se asegura la protección de la sentencia a ejecutar.

```

public function store(Request $request){
  if(Session::get('login') != true){
    return redirect('/');
  }else{
    $tipo = $request->input('tipo');
    $hora = date("Y-m-d H:i:s");
    $ip = $this->getRealIP();
    $id_usuario = session('usuario_id');
    $id_afectado = DB::table('perfiles')->orderBy('id', 'desc')->value('id')+1;

    DB::insert('insert into activity_log (fecha, ip, id_usuario, tabla, accion, id_afectado) values (?, ?, ?, ?, ?, ?)', [$hora, $ip,
    $id_usuario, 'perfiles', 'crear', $id_afectado]);

    DB::insert('insert into perfiles (tipo, estado, created_at, updated_at) values (?, ?, ?, ?)', [$tipo, 'Activo', $hora, $hora]);

    return redirect('perfil/ver');
  }
}

```

Figura 4.3 Bindeo de Parámetros

Fuente: Elaboración Propia

4.4.4 ATAQUE XSS

Los ataques XSS (*Cross-Site Scripting*) son un tipo de inyección, en la que se inyectan secuencias de comandos maliciosas en sitios web benignos y de confianza. Los ataques XSS se producen cuando un atacante utiliza una aplicación web para enviar un código malicioso, generalmente en forma de una secuencia de comandos del navegador, a un usuario final diferente. Debido a que piensa que el script proviene de una fuente de confianza, el script malicioso puede acceder a cualquier cookie, token de sesión u otra información confidencial que el navegador mantenga y que se utilice con este sitio. (OWASP, 2016)

A continuación, se muestra una captura donde se aplica la protección de ataques XSS:

```
<label class="col-md-2 control-label">Tarea</label>
<div class="col-md-10">
  <input type="text" class="form-control" id="tarea" name="tarea" value="{{ $rol->tarea }}" minlength="3"
  maxlength="20" >
  <div class="form-control-focus"> </div>
</div>
```

Figura 4.4 Implementación de sintaxis

Fuente: Elaboración Propia

En Laravel se hace uso de la sintaxis de doble llave (`{{ }}`) que evitara automáticamente a cualquier entidad HTML transmitida a través de una variable desde la vista. Esto es muy importante, ya que un usuario malintencionado puede pasar cadenas a un comentario o perfil.

CAPÍTULO V
ANÁLISIS COSTO
BENEFICIO

5.1 INTRODUCCIÓN

Según Gómez (2010) el análisis Costo/Beneficio está basado en la razón de los beneficios a los costos asociada con un proyecto en particular. Se considera que un proyecto es atractivo, cuando los beneficios derivados de su implementación exceden sus costos asociados. Por lo tanto, el primer paso en un análisis Costo/Beneficio es determinar cuáles son beneficios y cuales costos.

5.2 COCOMO II

A pesar que la solución informática fue desarrollada por una sola persona en el marco de un Proyecto de Grado, es importante conocer la estimación real del costo del proyecto en condiciones reales, así como tener presente la valoración del tiempo y del esfuerzo necesario para el emprendimiento, traducido en índices monetarios.

Un método para calcular estos parámetros es el COCOMO II (*CO*nstructive *CO*st *MO*del), modelo que, en su segunda versión, está orientado a los Puntos Función. Para estimar el costo total del sistema se tomarán en cuenta los siguientes costos: Costo de la elaboración del proyecto, costo del software desarrollado y costos de la implementación del sistema.

Los Puntos Función, miden la aplicación desde una perspectiva del usuario, dejando de lado todos los detalles de codificación, es totalmente independiente de todas las consideraciones de lenguaje, es una técnica de estimación de software desarrollada por Allan Albrecht en 1979 mientras trabajaba en IBM, quien definió conceptos para medir el software a partir de valoraciones de funcionalidades de software entregadas al usuario y no a partir de aspectos técnicos, con la intención de producir valoraciones independientes de la tecnología y fases del ciclo de vida utilizado.

Una de las principales aplicaciones del método es en la determinación de valoraciones (estimaciones) del producto de software a desarrollar. (Gómez. 2010)

5.2.1 COSTOS DE DESARROLLO DE SOFTWARE

Actualmente existen varias herramientas que ayudan a obtener el análisis de la metodología COCOMO II. En este proyecto, se utilizó la aplicación USC-COCOMO II.2000.0, la cual facilita el trabajo del cálculo:

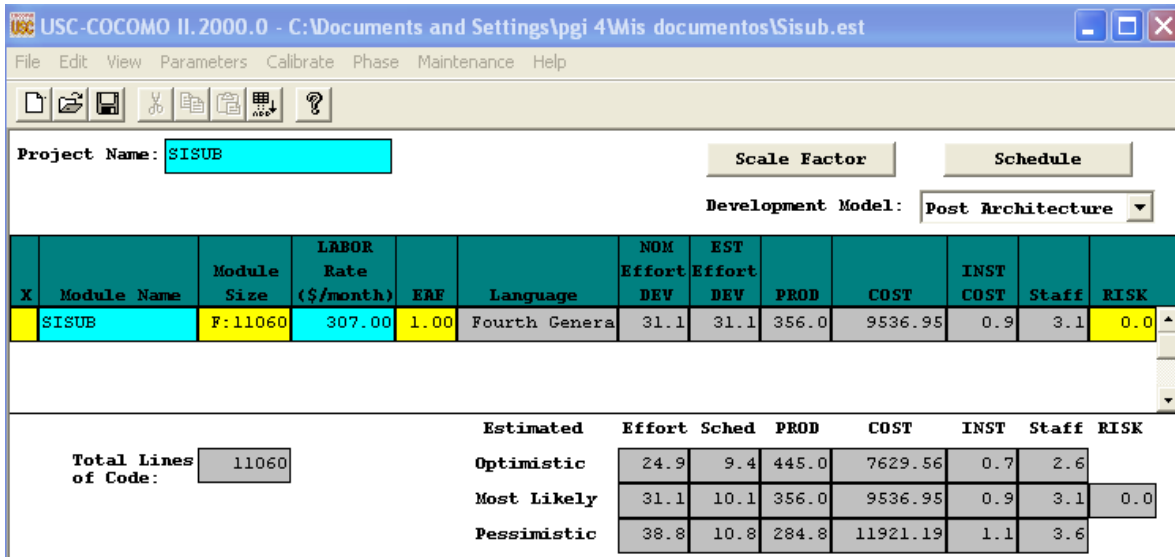


Figura 5.1 USC-COCOMO II.200.4

Fuente: Elaboración Propia

El cálculo se basó en los Puntos Función, explicado previamente en el presente capítulo.

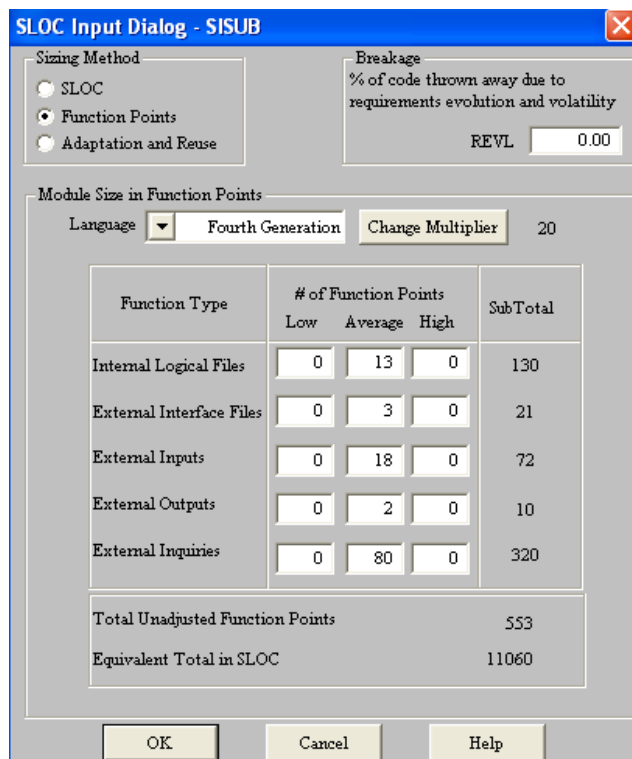


Figura 5.2 Aspectos de los Puntos de Función

Fuente: Elaboración Propia

Datos introducidos:

CAMPO	VALOR
Modelo de desarrollo	Post-Arquitectura
Salario mensual	307
Inputs	13
Outputs	3
Files	18
Interfaces	2
Queries	80

Tabla 5.1 Datos Introducidos en COCOMO

Fuente: Elaboración Propia

Nótese que:

- El lenguaje de desarrollo define un multiplicador de acuerdo a lenguaje, en este caso PHP y su multiplicador es 20.
- El salario mínimo de un programador es de 307\$ que equivalen a Bs. 2122

A continuación, se muestra el resultado obtenido con los datos ingresados descritos anteriormente:

The screenshot shows the USC-COCOMO II software interface. The project name is SISUB, the development model is Post Architecture, and the language is Fourth Genera. The main table displays project parameters and results for the SISUB module.

X	Module Name	Module Size	LABOR Rate (\$/month)	ERF	Language	NOM Effort DEV	EST Effort DEV	PROD	COST	INST COST	Staff	RISK
	SISUB	F:11060	307.00	1.00	Fourth Genera	29.5	29.5	375.3	9047.97	0.8	3.0	0.0

Total Lines of Code:	Estimated	Effort	Sched	PROD	COST	INST	Staff	RISK
11060	Optimistic	23.6	9.2	469.1	7238.38	0.7	2.6	
	Most Likely	29.5	9.8	375.3	9047.97	0.8	3.0	0.0
	Pessimistic	36.8	10.4	300.2	11309.97	1.0	3.5	

Figura 5.3 Resultados Obtenidos – COCOMO II

Fuente: Elaboración Propia

- **Tiempo de desarrollo:** 9.8 semanas
- Número de programadores: 3
- **Costo del proyecto:** \$us 9047.97 que equivalen a Bs. 62583.91

5.2.2 COSTOS DE ELABORACIÓN DEL SISTEMA

Para el costo del proyecto se toman en cuenta detalles como el internet, pasaje por parte de los programadores, se puede apreciar en la siguiente tabla:

DETALLE	COSTO (\$us)
Análisis y diseño	150
Costo de herramientas	
Material de escritorio	5
Internet	80
Llamadas	20
Pasajes	20
Otros	25
TOTAL	300

Tabla 5.2 Costos de Elaboración del Sistema

Fuente: Elaboración Propia

5.2.3 COSTOS DE IMPLEMENTACIÓN DEL SISTEMA

Las herramientas de software que se usaron para el desarrollo del presente proyecto son Laravel, PHP, Bootstrap y PostgreSQL, las cuales son de uso libre además de ser utilizadas por empresas de software libre.

5.2.4 COSTOS TOTAL DEL SOFTWARE

El costo total del software se muestra en la siguiente tabla:

DETALLE	COSTO (\$us)
Costo de desarrollo	9047,97
Costo de elaboración	300,00
TOTAL	9347,97

Tabla 5.3 Costos Total del Software

Fuente: Elaboración Propia

Por lo tanto, el costo total del proyecto es de 9347.97 \$us, que equivale a 64658.98 Bs.

5.3 CALCULO BENEFICIO

Para evaluar el beneficio se calculará con el método del Valor Actual Neto (VAN) y la tasa interna de Retorno (TIR).

5.3.1 VALOR ACTUAL NETO (VAN)

El Valor Actual Neto (VAN), es un método cuya principal aplicación es determinar la rentabilidad de una inversión. Como su nombre indica, trata de determinar el valor que ahora tiene la inversión sobre la base de los importes que se percibirán en plazos determinados. Para complementar, junto a la Tasa Interna de Rentabilidad (TIR), se dará una idea bastante clara de la factibilidad/rentabilidad del proyecto de inversión.

El VAN se calcula por medio de los flujos de inversión, cuyo resultado refleja si la inversión en el proyecto genera beneficios o no. Su fórmula es la siguiente:

$$VAN = \sum \frac{Ganancias}{(1+k)^n} - I_o$$

Donde:

- Ganancias: ingreso del flujo anual
- I_o : es el valor del desembolso inicial de la inversión
- n : es el número de periodos considerados
- k : tasa de descuento o tasa de interés de la inversión

Los valores de ganancia esperados para el presente proyecto se calculan para 4 años, para este caso en particular utilizaremos una tasa de descuento del 11%, ya que es la tasa actual de interés de préstamo en las entidades financieras. Para calcular el valor del VAN se tiene lo siguiente:

$$\text{Inversión} = 9347.97 \text{ \$us}$$

$$\text{TD} = 11\%$$

Los valores de ganancia esperados se detallan en la siguiente tabla:

TIEMPO	1er Año	2do Año	3er Año	4to Año
Flujo de caja neto	4200	4400	4800	5200

Tabla 5.4 Cantidad nominal por año
Fuente: Elaboración Propia

Para hallar el VAN tenemos:

$$VAN = \frac{4200}{(1 + 0.11)^1} + \frac{4400}{(1 + 0.11)^2} + \frac{4800}{(1 + 0.11)^3} + \frac{5200}{(1 + 0.11)^4} - 9347.97 = 4942.07$$

$$VAN = 4975.07 \text{ \$us} = 34412.07 \text{ Bs}$$

Como el valor obtenido del VAN es mayor a cero, se puede afirmar que el proyecto es rentable.

5.3.2 TASA INTERNA DE RETORNO (TIR)

El TIR es una tasa de descuento TD de un proyecto de inversión para que sea rentable. Cuando el VAN toma un valor igual a 0, k pasa a llamarse TIR.

En términos generales:

Las inversiones más interesantes son aquellas que proporcionan mayor TIR.

- Si TIR es inferior a la tasa de descuento de la empresa, la inversión debería ser desestimada.
- Si TIR es superior la tasa de descuento de la empresa la inversión es factible.

$$TIR = -I_0 + \frac{Q_1}{(1 + k)^1} + \frac{Q_2}{(1 + k)^2} + \dots + \frac{Q_n}{(1 + k)^3}$$

Donde:

- I_0 : es el valor del desembolso inicial de la inversión
- n : es el número de periodos considerados
- k : tasa de descuento o tasa de interés de la inversión

Entonces para hallar el TIR se necesita la inversión de 9347.97 \$us

Para hallar el TIR se hace uso de la fórmula del VAN, solo hace que el valor de VAN sea igual a 0, se tiene la siguiente ecuación:

$$0 = \frac{4200}{(1 + TIR)^1} + \frac{4400}{(1 + TIR)^2} + \frac{4800}{(1 + TIR)^3} + \frac{5200}{(1 + TIR)^4} - 9347.97$$

$$TIR = 34\%$$

Se tiene en el cálculo que la tasa interna de retorno es superior a la tasa de descuento; por lo tanto, la inversión es factible.

5.4 COSTO / BENEFICIO

Para hallar el costo beneficio de un proyecto se aplica la siguiente ecuación:

$$\frac{C : \textit{Costo}}{B : \textit{Beneficio}}$$

Reemplazando los valores previamente calculados en la ecuación, tenemos:

$$\frac{C}{B} = \frac{64658.98}{34412.07} = 1.87$$

Por tanto, por cada boliviano invertido, la organización tiene una ganancia de 0.87 ctvs.

CAPÍTULO VI
CONCLUSIONES Y
RECOMENDACIONES

6.1 CONCLUSIONES

Durante el desarrollo del presente proyecto, se pudo evidenciar que la implementación de un sistema web centralizado e interoperabilidad no es una tarea sencilla, pero es imprescindible llevar un control de los beneficios que el estado otorga a la población para una distribución equitativa de los servicios que las organizaciones del estado brindan. Finalmente, este capítulo detalla las conclusiones y recomendaciones a las cuales llegaremos con la culminación del proyecto, así como las recomendaciones pertinentes.

A la culminación del presente proyecto y conforme a las actividades definidas para el análisis e implementación del Sistema Web de Control Centralizado, Interoperabilidad y Firma Digital de Beneficiarios de Programas Sociales Estatales de Viviendas Caso: Viceministerio de Vivienda y Urbanismo. Se concluye que:

- Se logró implementar mecanismos que, al momento de encontrar campos vacíos de información, generan una descripción alternativa.
- Se implementó consultas que encuentran y reportan sobre casos en los que se encuentra duplicidad de información.
- Se logró que todos los usuarios tengan roles iniciales asignados a sus cuentas, y posteriormente estos roles pueden retirados o agregados.
- Se pudo diseñar un esquema de solicitudes que obtiene la información de los servicios web y los convierte a un formato estándar.
- Se implementó consultas que pasan por múltiples campos de verificación acerca de un beneficiario y genera reportes en caso de que tenga más de un beneficio.
- Se logró implementar una página web en la cual toda la población puede consultar y saber si tiene algún beneficio dentro de los programas del Viceministerio de Vivienda y Urbanismo.

Es importante resaltar que para que la población pueda acceder a los beneficios que otorga el estado es necesario tener la información de forma instantánea y acertada, el sistema implementado cumple con las necesidades y requerimientos que la organización solicitó.

Se pudo evidenciar que la implementación de nuevas tecnologías en toda organización ayuda en la automatización de procesos y hace más fiable la información requerida, en este caso, la integración de metodologías fueron SCRUM, UWE y Web-Site QEM, con las cuales se logró desarrollar el proyecto.

6.2 RECOMENDACIONES

Para ampliar el presente proyecto de grado, se hacen las siguientes recomendaciones:

- Se recomienda que los usuarios cambien sus contraseñas semanalmente o mensualmente para dar mayor seguridad al sistema.
- Realizar un mantenimiento constante a la Base de Datos mediante *Backups* periódicos debido a la gran cantidad de información que se recibe diariamente de los servicios web de los diferentes programas del Viceministerio de Vivienda y Urbanismo.
- Se recomienda la actualización y mantenimiento del sistema implantado, esto para un correcto funcionamiento y evitar sorpresivas fallas en el presente y futuro.
- Se recomienda implementar un sistema de control georeferencial, ya que la información que provee este sistema puede complementarse y ser más específico con respecto a los datos de los beneficiarios.
- Se recomienda que se implemente un control de tráfico de información de los servicios web.
- Se recomienda la utilización de herramientas de programación brindadas por PHP, debido a la interfaz amigable para el desarrollador.
- Se recomienda utilizar algún *framework*, ya que este facilita el desarrollo del producto.

BIBLIOGRAFÍA

- Albaladejo, X. (2015). *El Expendedor – Juego de Simulación de Scrum*. Recuperado de <https://proyectosagiles.org/2009/09/13/expendedor-juego-simulacion-scrum/>
- Alegsa, L. (2016). *Definición de aplicación web*. Recuperado de http://www.alegsa.com.ar/Dic/aplicacion_web.php
- Alquicira, C. (2015). *¿Qué son las metodologías ágiles?* Recuperado de http://www.qualtop.com/que_son_las_metodologias_agiles
- Astete, A. y Muñoz, M. (2016). *Tipos de Investigación*. 12 p.
- Baquero, J. (2015). *¿Qué son los web services y qué tecnología usar en su desarrollo?* Recuperado de <https://www.arsys.es/blog/programacion/disenio-web/web-services-desarrollo/>
- Barrios, J. (2017). *Firma digital en documentos .pdf* Recuperado de <https://www.juanbarrios.com/firma-digital-en-documentos-pdf/>
- Blancarte, O. (2017). *Soap Vs Rest ¿Cuál es mejor?* Recuperado de <https://www.oscarblancarteblog.com/2017/03/06/soap-vs-rest-2/>
- Blancarte, O. (2014). *¿Qué es Service-oriented Architecture? (SOA)* Recuperado de <https://www.oscarblancarteblog.com/2014/07/23/que-es-service-oriented-architecture-soa/>
- Cevallos, K. (2017). *El Software y la Ingeniería de Software*. Recuperado de <https://ingsoftwarekarlacevallos.wordpress.com/category/el-software-y-la-ingenieria-de-software/>
- Cevallos, K. (2015). *UML: Casos de Uso*. Recuperado de <https://ingsoftwarekarlacevallos.wordpress.com/2015/06/04/uml-casos-de-uso/>
- Chakray (2018). *Interoperabilidad: Definición e Importancia* Recuperado de <https://www.chakray.com/es/interoperabilidad-definicion-e-importancia/>
- Fontela, A. (2015). *¿Qué es Bootstrap?* Recuperado de <https://raiolanetworks.es/blog/que-es-bootstrap/>
- Gómez, A. (2010). *COCOMO – Un modelo de estimación de proyectos de software*. Recuperado de <https://blogadmi1.files.wordpress.com/2010/11/cocomo01full.pdf> 68p.

- Gómez, T. (2018). *Sistema Web de Control de Inventarios, Ventas de Productos y Mantenimiento de Maquinaria Basado en la Arquitectura SCM Caso: Convertidora de Fibras Grisel COFIGRIL LTDA*. Para optar al grado de Licenciatura en Ingeniería de Sistemas. La Paz, Universidad Mayor de San Andrés, Carrera de Informática. 185 p.
- Gómez, L. (2007). *Interoperabilidad en los Sistemas de Información Documental (SID): La información debe fluir*. Recuperado de <https://core.ac.uk/download/pdf/17036276.pdf>
- Gonzales, H. (2009). *Interoperabilidad entre los sistemas informáticos*. Recuperado de https://www.researchgate.net/publication/283270575_INTEROPERABILIDAD_ENTRE_LOS_SISTEMAS_INFORMATICOS
- Jus.gob.ar (2019). *Manual para SAS de Ciudad Autónoma de Buenos Aires*. Recuperado de http://www.jus.gob.ar/media/3175414/manual_de_firma_digital_actualizado.pdf 24p.
- Lázaro, D. (2018). *Introducción a los Web Services*. Recuperado de <https://diego.com.es/introduccion-a-los-web-services>
- Luján, S. (2002). *Programación de Aplicaciones Web: historia, principios básicos y clientes web*. 354 p.
- Mariño, S. y Alfonzo, P. (2014). *Implementación de SCRUM en el diseño del proyecto del Trabajo Final de Aplicación. Vol. 19*. 6 p.
- Maximilians, L. (2016). *UWE-UML-based Web Engineering*. Recuperado de <http://uwe.pst.ifi.lmu.de/aboutUwe.html>
- Mamani, L. (2014). *Sistema Web de registro y seguimiento de pacientes, elaboración y emisión de análisis efectuados, registros y guía médica, farmacéutica y hospitalaria, caso: Laboratorio clínico "Adolfo Kolping"*. Para optar al grado de licenciatura en Informática. La Paz, Universidad Mayor de San Andrés, Carrera de Informática 130 p.
- Mazario, M. (2015). *Los 11 pasos para implementar metodología SCRUM*. Recuperado de <http://managementplaza.es/blog/los-11-pasos-para-implementar-metodologia-scrum/>
- Morse, J. (2007). *Asuntos críticos en los métodos de investigación cualitativa*. Recuperado de <https://publicaciones.ua.es/es/ver-producto.php>

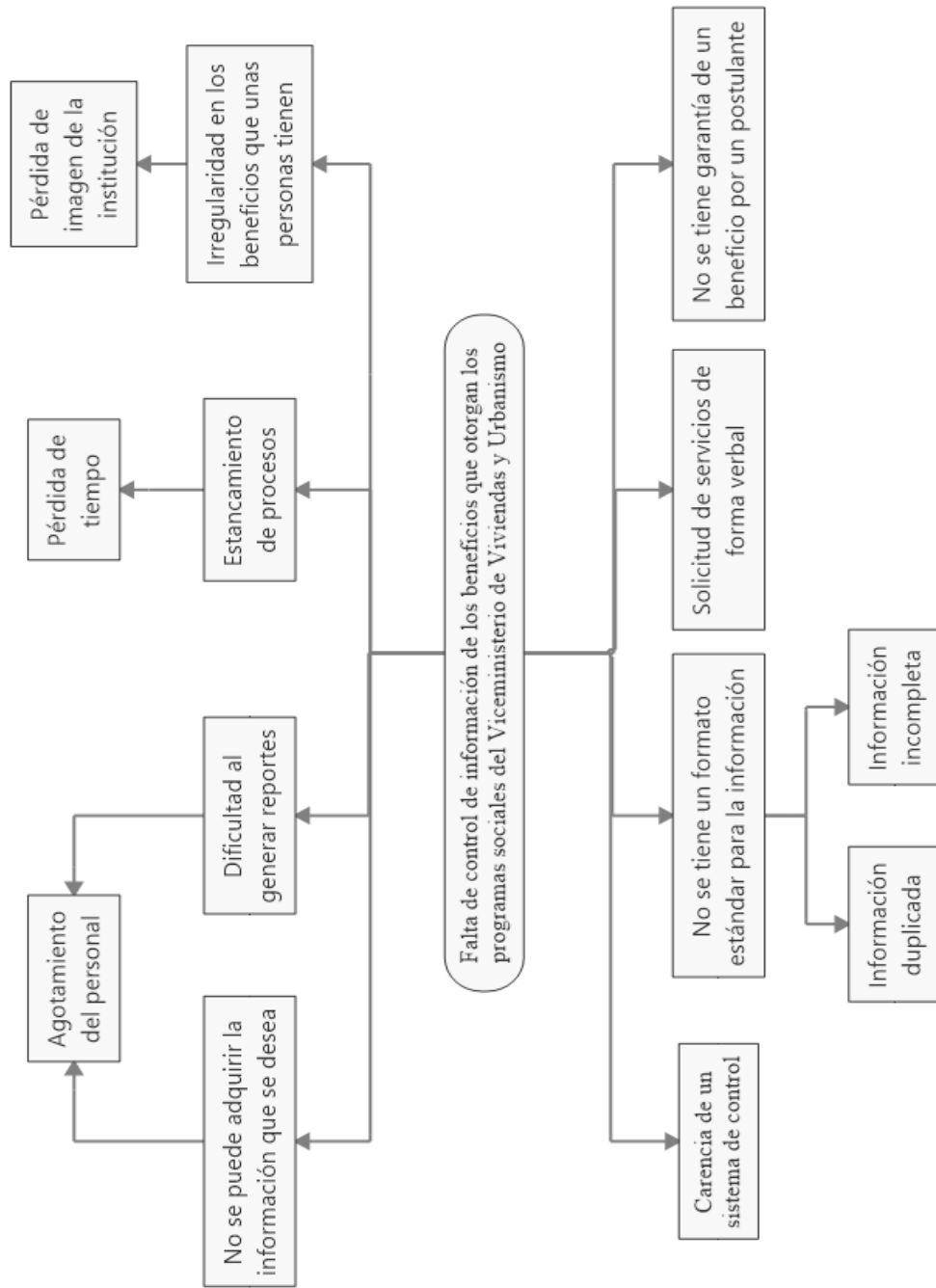
- Muradas, Y. (2028). *Conoce las 3 metodologías ágiles más usadas*. Recuperado de <https://openwebinars.net/blog/conoce-las-3-metodologias-agiles-mas-usadas/>
- OBS Business School (2018). *Tendencias & Innovación (Las 5 etapas en los “Sprints” de un desarrollo Scrum)*. Sede-Colombia, Perú Recuperado de <https://www.obs-edu.com/int/blog-investigacion/project-management/las-5-etapas-en-los-sprints-de-un-desarrollo-scrum>
- Olsina, L. (1999). *Metodología cuantitativa para la evaluación y comparación de calidad de sitios web*. Argentina, Universidad Nacional de La Plata, Facultad de ciencias Exactas. 265 p. Recuperado de https://www.sel.unsl.edu.ar/ApuntesMaes/Anteriores/CursoOlsina/Transparencias/Olsina_Sa nLuis_QEM.pdf
- Paco, E. (2017). *Sistema Centralizado de Información Teatral Caso: Taller de Artes Escénicas TAE-UMSA*. Para optar al grado de Licenciatura en Ingeniería de Sistemas. La Paz, Universidad Mayor de San Andrés, Carrera de Informática. 80 p.
- Palmero, R. (2010). *Los sistemas integrados de gestión de la información*. Recuperado de <https://www.gestiopolis.com/sistemas-integrados-gestion-informacion/>
- Paredes, F. (2019). *POSTGRESQL*. Recuperado de <https://websarrolladores.com/2019/06/12/postgresql/>
- Pérez y Merino. (2015). *Ensayo de Programas Sociales* Recuperado de <https://www.buenastareas.com/ensayos/Ensayo-De-Programas-Sociales/69320536.html>
- Perforce (2018). *¿What Is a Product Backlog in Agile?*. Recuperado de <https://www.perforce.com/resources/hns/agile-product-backlog-basics>
- Pims, T. (2018). *Laravel un gran framework en Php*. Recuperado de <https://arpentechologies.com/es/blog/disenio-web/laravel-un-gran-framework-en-php/>
- Platzi (2018). *¿Cómo funciona la metodología Scrum y sus Fases?* Recuperado de <https://platzi.com/blog/como-funciona-la-metodologia-scrum-y-sus-fase/>
- Ponce, D. (2018). *¿Qué es una Firma Digital?* Recuperado de <https://blog.signaturit.com/es/que-es-una-firma-digital>

- Pressman, R. (2010). *Ingeniería del Software. Un Enfoque Práctico*. 7a ed. México D.F. 805 p.
- QALovers (2016). *Pruebas de Interoperabilidad*. Recuperado de https://www.qalovers.com/2016/02/pruebas-de-interoperabilidad_35.html
- Ramos, C. (2017). *Los Eventos en # Scrum*. Recuperado de <https://cristinaramosvega.com/los-eventos-scrum/>
- Robles, J. (2013). *Seguridad en Ingeniería de Software*. Recuperado de <https://ithjrp.blogspot.com/>
- Roche, J. (2017). *Artefactos Scrum: las 3 herramientas clave de gestión*. Recuperado de <https://www2.deloitte.com/es/es/pages/technology/articles/artefactos-scrum.html>
- Rodriguez, G. et. Al (1996). *Metodología de la Investigación Cualitativa*. 35 p.
- Ruiz, R. (2014). *Centro de Documentación Virtual Caso: Facultad de Agronomía de la UMSA*. Para optar al grado de Licenciatura en Ingeniería de Sistemas. La Paz, Universidad Mayor de San Andrés, Carrera de Informática. 138 p.
- Saffirio, M. (2006). *¿Qué son los Web Services?* Recuperado de <https://msaffirio.wordpress.com/2006/02/05/%C2%BFque-son-los-web-services/>
- Salas, M. (2017). *Ingeniería Web Basada en UML*. Recuperado de <http://marcelosalasvargas.blogspot.com/2017/06/ingenieria-web-basada-en-uml.html>
- Sandoval, J. (2016). *Sprint Planning (Planificación de la Iteración)*. Recuperado de <https://medium.com/@jlsandovaln/sprint-plannings-planificaci%C3%B3n-de-la-iteraci%C3%B3n-ddb78ac60536>
- Sinnaps (2019). *Metodología Scrum*. Recuperado de <https://www.sinnaps.com/blog-gestion-proyectos/metodologia-scrum>
- Scrum.org. (2018). *What is Sprint Planning*. Recuperado de <https://www.scrum.org/resources/what-is-sprint-planning>
- Softeng Portal Builder (2018). *Proceso y Roles de Scrum*. Recuperado de <https://www.softeng.es/es/empresa/metodologias-de-trabajo/metodologia-scrum/proceso-roles-de-scrum.html>

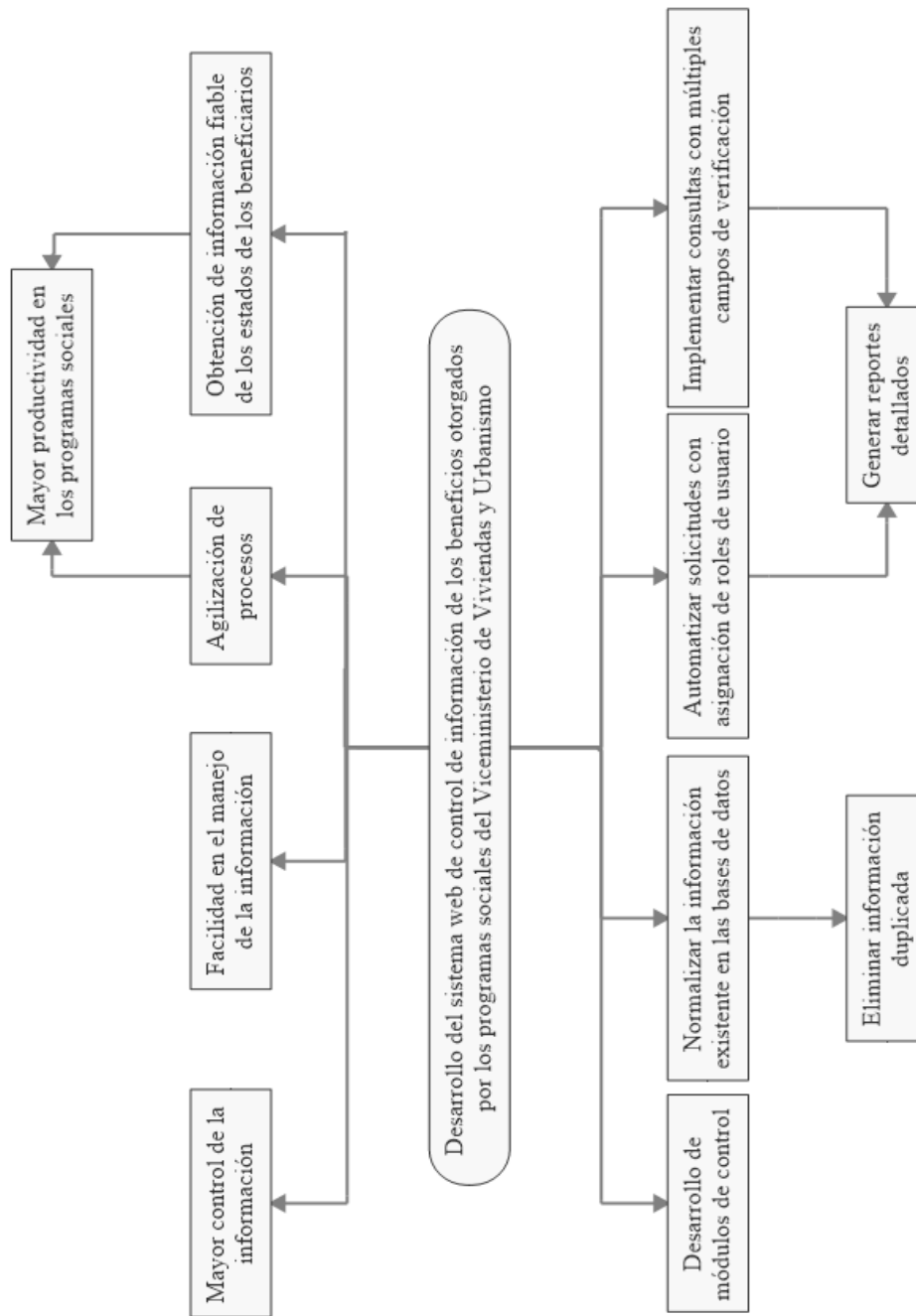
- Sommerville, I. (2005). *Introducción. En: "Ingeniería del software"*. 7ª ed. Madrid. Pearson. Educación S.A. pp. 712
- Tenstep (2018). *SRUM: Cómo escribir historias de usuarios sin morir en el intento*. Recuperado de <https://www.tenstep.ec/portal/articulos-boletin-tenstep/41-scrum/253-scrum-como-escribir-historias-de-usuarios-sin-morir-en-el-intento>
- Trigas, M. (2012). *Metodología Scrum*. 56 p.
- Universitat Politècnica de València. (2012). *¿Qué es una Firma Electrónica?* Recuperado de <https://www.upv.es/contenidos/CD/info/711250normalc.html>
- UWE – UML – Based Web Engineering. (1999). *Tutorial – Requirements Model (Spanish)*. Recuperado de <https://uwe.pst.ifi.lmu.de/teachingTutorialRequirementsSpanish.html>
- Velasco, R. (2017). *OWASP TOP-10 2017*. Recuperado de <https://www.redeszone.net/2017/11/21/top-10-vulnerabilidades-2017-owasp/>
- YunbitSoftware. (2016). *Por Que Tener un Sistema Centralizado de Datos* Recuperado de <https://www.yunbitsoftware.com/blog/2016/06/24/sistema-centralizado-datos/>
- Zuñiga, J. y Rosanz, M. (2004). *Introducción a la Ingeniería Web Basada en UML*. 5 p.

ANEXOS

ANEXO A – ÁRBOL DE PROBLEMAS



ANEXO B – ÁRBOL DE OBJETIVOS



ANEXO C – MARCO LÓGICO

RESUMEN NARRATIVO	INDICADORES VERIFICABLES	MEDIOS DE VERIFICACIÓN	VARIABLES NO CONTROLABLES
<p>FIN</p> <p>Lograr un mayor control y verificación el manejo de la información de los programas sociales, haciendo uso de la tecnología web.</p>	<p>La postulantes tendran una alternativa inmediata de certificación por un costo mínimo. Los funcionarios podran disponer de un software en línea y con toda la información que requieran.</p>	<p>Reportes internos de todo el movimiento que se hace dependiendo lo solicite el administrador.</p>	<p>Cuando lo requiera un funcionario, se brindará permisos temporales de acceso al sistema, para poder ejecutar una solicitud.</p>
<p>OBJETIVO GENERAL</p> <p>Desarrollar un sistema web centralizado interoperable que controle y certifique la información de los beneficios que otorgan los programas estatales sociales de vivienda dependientes del Viceministerio de Viviendas y Urbanismo.</p>	<p>El sistema entrara en funcionamiento desde 1 de diciembre de 2019.</p>	<p>Certificado de conformidad elaborado por el jefe de unidad mencionando que se encuentra conforme con el proyecto realizado. Documentación del proyecto al cabo de 5 meses despues de la iniciación del proyecto.</p>	<p>Existe disponibilidad de los recursos necesarios para la implementación del sistema.</p>
<p>COMPONENTES</p> <ul style="list-style-type: none"> Módulo de administración de permisos, información general y seguimiento de los beneficios otorgados a las personas por los programas sociales. Módulo de consultas y creación de servicios, donde se opere con las distintas bases de datos, y se estandarice la información sacada de estos para su posterior envío a los usuarios que lo requieran. Módulo de reportes de beneficiarios por fecha, territorio, cantidad, inversión y estado en el que se encuentra el proceso. Módulo de consulta inmediata, donde una persona que quiera postularse puede obtener inmediatamente su certificado de no beneficio. 	<p>La presentación de los módulos se terminara a mas tardar, dos semanas antes de la presentacion final.</p>	<p>Manuales de usuario. Código fuente del sistema, con los resultados obtenidos de las pruebas realizadas. Capacitación del personal para los nuevos estandares que se tendran en los programas sociales. Colocar el sistema en producción.</p>	<p>Se cuenta con equipos de desarrollo, bases de datos reales, servidores y toda la infraestructura para el diseño del Sistema Web Centralizado, Interoperabilidad y Firma Digital de Beneficiarios de Programas Estatales de Viviendas.</p>

<p>ACTIVIDADES</p> <ul style="list-style-type: none"> • Exploración. Se tiene diferentes involucrados como ser los usuarios del sistema, entre administrador y un usuario normal, además un servicio de consulta en línea. • Inicialización, el ámbito en el que se desarrollara el proyecto será netamente relacionada a la información de los beneficiarios. • Producción, se diseñara la base de datos centralizada donde se trabajara de manera interoperable con la información de los demás programas sociales. • Estabilización, se verificara que los datos en la nueva base de datos (central) sean fiables, comprobando que no exista redundancia de información o datos incompletos. • Prueba y corrección, se sometera el sistema a prueba por partes, verificando que este la información clara en la base de datos del programa social, y en la base de datos central, así como se evaluara la verificación de beneficios en línea. 	<p>Se necesitara un analista que controle esta base de datos centralizada y tenga conocimiento en los distintos programas que están vigentes en el Viceministerio de Viviendas y Urbanismo.</p>	<p>Informes y entrevistas. Control de avance del diseño del sistema efectuado por el supervisor del área de sistemas. Aprobación de las pruebas de funcionamiento.</p>	<p>Se cuenta con la colaboración de los distintos programas que forman parte del Viceministerio de Viviendas y Urbanismo.</p>
--	---	--	---

DOCUMENTACIÓN