



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

**CARRERA DE INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN
(SISTEMAS E INFORMÁTICA)**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO
DE INGENIERO EN SISTEMAS E INFORMÁTICA**

**TEMA: ESTUDIO COMPARATIVO DE LIBRERÍAS DE COMPONENTES
JSF PARA LA IMPLEMENTACIÓN DE UN SISTEMA WEB UTILIZANDO
ARQUITECTURA JEE7, CASO DE ESTUDIO ALBERGUES DE
DAMNIFICADOS DEL TERREMOTO DE MANABÍ**

AUTOR: ARROBO FIGUEROA, ASDRÚBAL PATRICIO

DIRECTOR: ING. CAMPAÑA ORTEGA, EDUARDO MAURICIO. MSc

SANGOLQUÍ

2019

CERTIFICADO

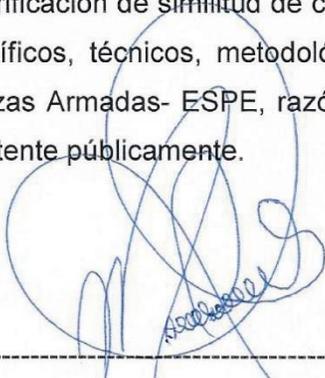


DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
CARRERA DE INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN
(SISTEMAS E INFORMÁTICA)

CERTIFICACIÓN

Certifico que el trabajo de titulación , ***“ESTUDIO COMPARATIVO DE LIBRERÍAS DE COMPONENTES JSF PARA LA IMPLEMENTACIÓN DE UN SISTEMA WEB UTILIZANDO ARQUITECTURA JEE7, CASO DE ESTUDIO ALBERGUES DE DAMNIFICADOS DEL TERREMOTO DE MANABÍ”*** fue realizado por el señor ***Arrobo Figueroa, Asdrúbal Patricio*** el mismo que ha sido revisado en su totalidad, analizado por la herramienta de verificación de similitud de contenido; por lo tanto cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de las Fuerzas Armadas- ESPE, razón por la cual me permito acreditar y autorizar para que lo sustente públicamente.

Sangolquí, 29 de Enero del 2019



Ing. Campaña Ortega, Eduardo Mauricio. MsC

CC: 1708856701

AUTORÍA DE RESPONSABILIDAD



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
CARRERA DE INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN
(SISTEMAS E INFORMÁTICA)

AUTORÍA DE RESPONSABILIDAD

Yo, **Arrobo Figueroa, Asdrúbal Patricio**, declaro que el contenido, desarrollo, ideas y criterios del trabajo de titulación : ***“Estudio comparativo de librerías de componentes jsf para la implementación de un sistema web utilizando arquitectura jee7, Caso de Estudio albergues de damnificados del terremoto de Manabí”*** es de mi autoría y responsabilidad, cumpliendo con los requisitos técnicos, metodológicos y legales establecidos por la Universidad de las Fuerzas Armadas – ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Consecuentemente el contenido de la investigación mencionada es veraz.

Sangolquí, 29 de Enero del 2019

Arrobo Figueroa, Asdrúbal Patricio

C.C.: 1712814647

AUTORIZACIÓN



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
CARRERA DE INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN
(SISTEMAS E INFORMÁTICA)

AUTORIZACIÓN

Yo, **Arrobo Figueroa, Asdrúbal Patricio**, autorizo a la Universidad de las Fuerzas Armadas – ESPE publicar el trabajo de titulación: “***Estudio comparativo de librerías de componentes jsf para la implementación de un sistema web utilizando arquitectura jee7, Caso de Estudio albergues de damnificados del terremoto de Manabí***” en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

Sangolquí, 29 de Enero del 2019

Arrobo Figueroa, Asdrúbal Patricio

CC: 1712814647

DEDICATORIA

Este trabajo va dedicado a mi querida madre Rosa América Figueroa Jiménez, quien a lo largo de mi vida se ha preocupado por mi bienestar y educación, con su apoyo, amor y paciencia ha sido un apoyo para cumplir mis objetivos, su lucha constante por sacarnos adelante a mí y a mis hermanos y son mi ejemplo a seguir, a ella le debo la vida y todo lo que logrado hasta hoy.

A mis hermanos Johnny y Elizabeth quienes han sido unos grandes compañeros de vida con quien he compartido los momentos de alegría y tristeza que nos da la vida.

Y finalmente a mis amigos que han estado conmigo en muchos momentos de mi vida y me supieron guiar y apoyar para ser una persona de bien.

AGRADECIMIENTO

Agradezco en primer lugar a Dios y a la Virgen, por haberme dado la vida, la capacidad y la fuerza intelectual para poder culminar mis estudios.

A la Universidad de las Fuerzas Armadas- ESPE, por darme la oportunidad de estudiar y ser profesional.

Gracias a mi madre por haberme apoyado moral y económicamente, gracias por la confianza que deposito en mí, por inculcarme buenos valores que me hacen ser una persona de bien.

Gracias a mis hermanos Johnny y Elizabeth, que siempre me han brindado su amor y compañía.

A mi director de proyecto de investigación, Ingeniero Mauricio Campaña, por brindarme su apoyo y tiempo incondicional en todo momento, Al Ing. Henry Coral por ser parte de este proceso en la evaluación para cumplir esta meta, los cuales me brindaron sus conocimientos para guiarme y realizar un buen trabajo.

A todos los docentes que desempeñan sus funciones en la Facultad de Ingeniería de Sistemas e Informática los cuales me brindaron sus conocimientos durante la etapa estudiantil y que me ha servido para ponerlos en práctica durante el desarrollo de este proyecto.

Y a todas las personas que en estos años de estudio me brindaron gratos momentos, y que me animaron a cumplir mis anhelos de llegar a ser profesional.

ÍNDICE DE CONTENIDOS

CERTIFICADO	i
AUTORÍA DE RESPONSABILIDAD	ii
AUTORIZACIÓN	iii
DEDICATORIA.....	iv
AGRADECIMIENTO	v
ÍNDICE DE CONTENIDOS.....	vi
ÍNDICE DE TABLAS	x
ÍNDICE DE FIGURAS	xiv
ÍNDICE DE ANEXOS	xvi
RESUMEN.....	xvii
ABSTRACT	xviii
GLOSARIO DE TÉRMINOS.....	xix
CAPÍTULO I	1
ASPECTOS GENERALES.....	1
1.1 INTRODUCCIÓN	1
1.2 ANTECEDENTES.....	2
1.3 PLANTEAMIENTO DEL PROBLEMA.....	3
1.4 JUSTIFICACIÓN.....	5
1.5 OBJETIVOS.....	5
1.5.1 OBJETIVO GENERAL.....	5
1.5.2 OBJETIVOS ESPECÍFICOS.....	6
CAPÍTULO II	7
ESTADO DEL ARTE.....	7
2.1 Herramientas de desarrollo.....	7
2.2 Normas ISO	8
2.2.1 Requerimientos y Evaluación de Calidad del Producto de Software.....	11
2.2.2 Proceso de transición entre ISO/IEC 9126, ISO/IEC 14598.....	11
2.3 Java Server Faces (JSF)	13

2.3.1 Ciclo de vida de Java Server Faces.....	16
2.3.2 Ventajas	18
2.3.3 Desventajas:	18
2.3.4 Lenguaje de Programación.....	20
2.3.5 Entorno de Desarrollo Integrado (IDE).....	21
2.3.6 Servidor de Aplicaciones.....	23
2.3.7 Motor de Base de Datos	24
2.3.8 Reportes.	25
2.3.9 Arquitectura de Software.....	26
2.3.10 Frameworks Java para Seguridad	27
2.4 Metodología de Investigación	29
2.4.1 Programación Extrema (XP).	30
2.4.2 Roles de Programación Extrema (XP).	32
2.4.3 Fases de Programación Extrema (XP).....	32
CAPÍTULO III	38
ANÁLISIS COMPARATIVO DE LIBRERÍAS DE COMPONENTES.....	38
3.1 Librería de componentes JSF	38
3.1.1 OpenFaces.....	38
3.1.2 RichFaces.....	40
3.1.3 PrimeFaces	43
3.1.4 ButterFaces	44
3.1.5 OmniFaces	45
3.1.6 BootsFaces.....	46
3.2 Estudio general de las librerías de componentes.....	47
3.3 Descripción de Requisitos.....	48
3.4 Comparación.....	49
3.4.1 Documentación Oficial.....	49
3.4.2 Configuración	50
3.4.3 Showcase	50
3.4.4 Soporte	51

3.4.5 Popularidad	51
3.4.6 Criterio: Infraestructura	60
3.5 Construcción del Modelo de Calidad	62
3.5.1 Criterios de decisión para la evaluación.....	62
3.5.2 Matriz de Evaluación de calidad.....	63
3.5.3 Evaluación de los Frameworks	64
3.5.4 Aplicación de los criterios de decisión para las métricas	65
3.6 Resumen Comparativo	85
3.6.1 Análisis de Parámetros	85
3.6.2 Selección del Framework de presentación	87
3.6.3 Selección del framework en base a puntajes promedio	87
3.6.4 Resultados Totales de los frameworks	88
3.6.5 Calificación del framework	89
CAPÍTULO IV.....	90
DISEÑO, DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA DE ALBERGUES ..	90
4.1 Planificación.....	90
4.1.1 Especificación de Requerimientos	90
4.2 Iteraciones	102
4.2.1 Plan de Entrega	103
4.2.2 Reuniones	104
4.3 Diseño.....	104
4.4 Procesos	105
4.4.1 Proceso de Administración de Albergues	105
4.4.2 Proceso de Administración de Damnificados	105
4.4.3 Proceso de Administración de Personal Administrativo	106
4.4.4 Proceso de Administración de Parte Diario	107
4.4.5 Diagrama E – R (Esquema Conceptual)	108
4.5 Desarrollo.....	109
4.5.1 Integrantes y Roles de usuarios.	109
4.5.2 Prototipos	112

4.5.3	Prototipo para Inicio de Sesión.....	112
4.5.4	Prototipo para Creación de Nuevas Cuentas	113
4.5.5	Prototipo de la página de inicio Administrador General.....	114
4.5.6	Prototipo de la página de inicio Administrador - Albergue	114
4.6	Interfaces de Usuario	115
4.6.1	Página de inicio.....	115
4.6.2	Plantilla principal	116
4.6.3	Página de entrada de datos	118
4.7	Codificación	119
4.8	Pruebas.....	119
4.8.1	Pruebas Unitarias	119
4.8.2	Pruebas de Aceptación	119
CAPÍTULO V	146
CONCLUSIONES Y RECOMENDACIONES	146
5.1	CONCLUSIONES	146
5.2	RECOMENDACIONES	147
5.3	TRABAJO FUTURO.....	147
BIBLIOGRAFÍA	148
ANEXOS	153

ÍNDICE DE TABLAS

Tabla 1	<i>Componentes de la tecnología Java Server Faces</i>	14
Tabla 2	<i>Ciclo de Vida JSF</i>	17
Tabla 3	<i>Valoración por Categorías - Escala de Likert</i>	19
Tabla 4	<i>Comparativo entre Lenguajes de Programación</i>	20
Tabla 5	<i>Comparación de Entornos de Desarrollo Integrado (IDE's)</i>	22
Tabla 6	<i>Cuadro Comparativo entre Servidores de Aplicaciones</i>	23
Tabla 7	<i>Cuadro Comparativo de Motores de Base de Datos</i>	24
Tabla 8	<i>Cuadro Comparativo de Generadores de Reportes</i>	25
Tabla 9	<i>Comparativo de Arquitecturas de Software</i>	26
Tabla 10	<i>Frameworks Java para Seguridad</i>	28
Tabla 11	<i>Comparativo de las Metodologías Tradicionales vs Agiles</i>	30
Tabla 12	<i>Valores de la Programación Extrema XP</i>	31
Tabla 13	<i>Roles de Programación Extrema XP</i>	32
Tabla 14	<i>Fases de la Programación Extrema (XP)</i>	33
Tabla 15	<i>Características de la Etapa de Planificación</i>	34
Tabla 16	<i>Características de la Etapa de Diseño</i>	35
Tabla 17	<i>Características de la Etapa de Codificación</i>	36
Tabla 18	<i>Características de la Etapa de Pruebas</i>	37
Tabla 19	<i>Criterios Comparativos de Librerías de Componentes JSF</i>	48
Tabla 20	<i>Documentación</i>	48
Tabla 21	<i>Infraestructura</i>	49
Tabla 22	<i>Plataformas Móviles</i>	49
Tabla 23	<i>Análisis comparativo de la Documentación Oficial</i>	50
Tabla 24	<i>Estimación de Visitas OpenFaces</i>	52
Tabla 25	<i>Datos Obtenidos de visitas RichFaces</i>	54

Tabla 26	<i>Datos Obtenidos de visitas a PrimeFaces</i>	55
Tabla 27	<i>Datos Obtenidos de visitas ButterFaces</i>	56
Tabla 28	<i>Datos Obtenidos de visitas OmniFaces</i>	58
Tabla 29	<i>Datos Obtenidos de visitas BootsFaces</i>	59
Tabla 30	<i>Navegadores soportados por frameworks JSF</i>	60
Tabla 31	<i>Servidores soportados Frameworks JSF</i>	60
Tabla 32	<i>Soporte Ajax en Frameworks JSF</i>	60
Tabla 33	<i>Soporte de Frameworks JSF a plataformas móviles</i>	61
Tabla 34	<i>Criterios de Decisión para la evaluación de Calidad</i>	62
Tabla 35	<i>Ponderación asignada para la calidad interna y externa</i>	63
Tabla 36	<i>Aplicación de Criterios de Decisión en base a una matriz de calidad</i>	65
Tabla 37	<i>Análisis de Parámetros</i>	85
Tabla 38	<i>Ponderación de los Criterios del Parámetro Facilidad para el desarrollo</i>	86
Tabla 39	<i>Selección del framework de presentación</i>	87
Tabla 40	<i>Selección del framework de acuerdo al mejor puntaje</i>	88
Tabla 41	<i>Resultados Totales de los frameworks</i>	88
Tabla 42	<i>Calificación del framework</i>	89
Tabla 43	<i>Historia 1- iteración 1</i>	91
Tabla 44	<i>Historia 2- iteración 1</i>	92
Tabla 45	<i>Historia 3- iteración 1</i>	92
Tabla 46	<i>Historia 4- iteración 1</i>	93
Tabla 47	<i>Historia 5- iteración 1</i>	93
Tabla 48	<i>Historia 6- iteración 1</i>	94
Tabla 49	<i>Historia 7- iteración 1</i>	94
Tabla 50	<i>Historia 8- iteración 1</i>	95
Tabla 51	<i>Historia 9- iteración 1</i>	95

Tabla 52 <i>Historia 10- iteración 2</i>	96
Tabla 53 <i>Historia 11- iteración 2</i>	96
Tabla 54 <i>Historia 12- iteración 2</i>	97
Tabla 55 <i>Historia 13- iteración 2</i>	97
Tabla 56 <i>Historia 14- iteración 3</i>	98
Tabla 57 <i>Historia 15- iteración 3</i>	98
Tabla 58 <i>Historia 16- iteración 3</i>	99
Tabla 59 <i>Historia 17- iteración 3</i>	99
Tabla 60 <i>Historia 18- iteración 3</i>	100
Tabla 61 <i>Requerimientos No Funcionales 1 - Tiempo de Respuesta</i>	100
Tabla 62 <i>Requerimientos No Funcionales 2- Utilización de colores</i>	101
Tabla 63 <i>Requerimientos No Funcionales 3- Iconos de Opciones</i>	101
Tabla 64 <i>Requerimientos No Funcionales 4- Métodos de Acceso</i>	101
Tabla 65 <i>Requerimientos No Funcionales 5 - Plataforma Open Source</i>	101
Tabla 66 <i>Requerimientos No Funcionales 6 - Accesibilidad</i>	102
Tabla 67 <i>Plan de Entrega- Iteración 1</i>	103
Tabla 68 <i>Plan de Entrega- Iteración 2</i>	104
Tabla 69 <i>Plan de Entrega - Iteración 3</i>	104
Tabla 70 <i>Integrantes del Sistema y Roles de Usuario</i>	109
Tabla 71 <i>Prueba de aceptación Iniciar Sesión</i>	120
Tabla 72 <i>Prueba de aceptación para Gestionar Procesos</i>	121
Tabla 73 <i>Prueba de aceptación para Gestionar Albergues</i>	122
Tabla 74 <i>Prueba de Aceptación Gestionar Administradores de Albergue</i>	124
Tabla 75 <i>Prueba de Aceptación Gestionar Cabecera de Servicios</i>	125
Tabla 76 <i>Prueba de Aceptación Gestionar Servicios Generales</i>	126
Tabla 77 <i>Prueba de Aceptación Gestionar Damnificados</i>	128

Tabla 78 <i>Prueba de Aceptación Gestionar Personal Administrativo</i>	129
Tabla 79 <i>Prueba de Aceptación Gestionar Mantenimiento</i>	131
Tabla 80 <i>Prueba de aceptación Gestionar Reportes</i>	132
Tabla 81 <i>Prueba de aceptación Gestionar Parte Diario</i>	134
Tabla 82 <i>Prueba de aceptación Gestionar parte diario militar</i>	135
Tabla 83 <i>Prueba de aceptación Gestionar parte diario de personas</i>	137
Tabla 84 <i>Prueba de aceptación Gestionar Usuarios</i>	139
Tabla 85 <i>Prueba de aceptación Crear Usuarios</i>	140
Tabla 86 <i>Prueba de aceptación Crear Roles de Usuario</i>	141
Tabla 87 <i>Prueba de aceptación Crear Roles</i>	143
Tabla 88 <i>Prueba de aceptación Cambiar Clave</i>	144

ÍNDICE DE FIGURAS

Figura 1. Arquitectura JSF	15
Figura 2. Ciclo de vida de Java Server Faces	16
Figura 3. Fases de la Programación Extrema (XP)	33
Figura 4. Logo Open Faces	39
Figura 5. Logo RichFaces.....	40
Figura 6. Arquitectura RichFaces	42
Figura 7. Logo PrimeFaces.....	43
Figura 8. Logo ButterFaces	44
Figura 9. Logo OmniFaces	45
Figura 10. Logo BootsFaces.....	46
Figura 11. Formulario Alexa para calcular visitantes mensuales	51
Figura 12. Visitas acumuladas al sitio OpenFaces	52
Figura 13. Datos Obtenidos de los visitantes a OpenFaces	53
Figura 14. Visitas acumuladas al sitio RichFaces.....	53
Figura 15. Datos Obtenidos de los visitantes a RichFaces.....	54
Figura 16. Visitas acumuladas al sitio PrimeFaces.....	55
Figura 17. Datos Obtenidos de los visitantes a PrimeFaces.....	56
Figura 18. Visitas al sitio ButterFaces.....	56
Figura 19. Datos Obtenidos de los visitantes a ButterFaces	57
Figura 20. Cantidad de Visitas al sitio OmniFaces	57
Figura 21. Datos Obtenidos de los visitantes a ButterFaces	58

Figura 22. Cantidad de Visitas al sitio BootsFaces.....	59
Figura 23. Datos Obtenidos de los visitantes a BootsFaces.....	59
Figura 24. Proceso de Administración de Albergues	105
Figura 25. Proceso de Administración de Damnificados	106
Figura 26. Proceso de Administración de Personal Administrativo.....	106
Figura 27. Proceso de Administración de Parte Diario	107
Figura 28. Diagrama E-R (Esquema Conceptual)	108
Figura 29. Roles del Administrador General	110
Figura 30. Roles del Administrador de Albergue	111
Figura 31. Prototipo de la plantilla general para Inicio de Sesión	113
Figura 32. Prototipo de la plantilla general para la Creación de Nuevas Cuentas.....	113
Figura 33. Prototipo de la plantilla Página de inicio del Administrador General.....	114
Figura 34. Prototipo de la plantilla Página de inicio del Administrador-Albergue	115
Figura 35. Página de inicio para la validación y Registro de Inicio de Sesión	116
Figura 36. Plantilla principal, utilizada para la estructura y apariencia del sistema.....	117
Figura 37. Contenido de la plantilla principal	118
Figura 38. Página de entrada de datos de Usuarios del Sistema de Albergues	118

ÍNDICE DE ANEXOS

ANEXO A: MANUAL DE USUARIO	153
ANEXO B: MANUAL TÉCNICO	153

RESUMEN

El presente proyecto parte de la necesidad de la Universidad de las Fuerzas Armadas - ESPE de automatizar los procesos de administración de Albergues debido a que al no implementar un sistema de control provoca desventajas en a llevar a cabo el control físico como es la infraestructura de cada albergue, tipo de construcción, área en la que se encuentra, si dispone servicios básicos como son agua, luz, alimentación , internet, servicios de salud para ayuda de los damnificados, en la actualidad en los albergues se encuentra trabajando personal de las Fuerzas Armadas que son los administradores de estos albergues y personal de diferentes instituciones públicas como son del (Ministerio de Inclusión Económica y Social (MIES) , Ministerio de Salud y de otros Ministerios todo este grupo de personas FFAA, personal de servidores públicos de los diferentes Ministerios son la parte Administrativa por lo tanto se desea llevar un control administrativo del personal que trabaja para los albergues mediante el uso de tecnologías y herramientas de código abierto que actualmente se encuentran a la vanguardia de las nuevas tecnologías en el campo profesional del desarrollo de software, permitiendo a las organizaciones contar con una herramienta eficiente que contribuya en el manejo e integración de la información.

PALABRAS CLAVE:

- **ADMINISTRACIÓN DE ALBERGUES**
- **LIBRERÍAS JSF**
- **DESARROLLO DE SOFTWARE**
- **SISTEMA WEB**
- **ARQUITECTURA JEE**

ABSTRACT

The present project is based on the need of the University of the Armed Forces - ESPE to automate the processes of hostel management, because by not implementing a control system it causes disadvantages in carrying physical control as is the infrastructure of each shelter, type of construction, area in which it is located, if it has basic services such as water, electricity, food, internet, health services to help the victims, currently in the shelters are working personnel of the Armed Forces who are the administrators of these shelters and staff of different public institutions such as the Ministry of Economic and Social Inclusion (MIES), Ministry of Health and other Ministries this whole group of people FFAA , staff of public servants of the different Ministries are the Administrative part, therefore, wants to take administrative control of the staff working for the shelters by using open source technologies and tools that are currently at the forefront of new technologies in the professional field of software development, allowing organizations to have a tool efficient that contributes to the management and integration of information.

KEYWORDS:

- **HOSTEL ADMINISTRATION**
- **JSF LIBRARIES**
- **SOFTWARE DEVELOPMENT**
- **WEB SYSTEM**
- **JEE ARCHITECTURE**

GLOSARIO DE TÉRMINOS

- **AJAX: Asynchronous JavaScript And XML:** Es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (Rich Internet Applications). Permite realizar cambios sobre las páginas sin necesidad de recargarlas, mejorando la interactividad, velocidad y usabilidad en las aplicaciones.
- **Aplicación Web:** Es una aplicación que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador.
- **Arquitectura Cliente – Servidor:** La arquitectura cliente-servidor consiste básicamente en un cliente que realiza peticiones a otro programa (el servidor) que le da respuesta.
- **Arquitectura Web:** Una arquitectura simplificada del web, es una típica arquitectura cliente/servidor, en el cual de un lado se encuentra el cliente que está compuesto de browsers web, capaces de mostrar y solicitar documentos sobre una red.
- **Base de Datos MySQL (MariaDB):** Es un sistema de gestión de bases de datos relacional, multihilo y multiusuario.
- **Facelets:** herramienta web de código abierto para crear plantillas de Java Server Faces (JSF)
- **Hipermedia:** como una forma especial de multimedia interactiva que emplea estructuras de navegación más complejas que aumentan el control del usuario sobre el flujo de la información.

- **IDE:** (Integrated Development Environment) Entorno integrado de desarrollo. Es una aplicación empaquetada que consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica.
- **Indicadores:** Es una comparación de tipos de datos de medida cuantitativa (variables, tablas, cifras, datos estadísticos), o una observación cualitativa, para mostrar nociones o indicios de algo.
- **Interactividad:** Es la acción que se genera a modo de diálogo entre ordenador y usuario.
- **Java Runtime Environment:** Es un conjunto de utilidades que permite la ejecución de programas Java
- **JMX:** Es la tecnología que define una arquitectura de gestión, los patrones de diseño, y los servicios para la monitorización/administración de aplicaciones basadas en Java.
- **Metodología Xp:** Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, promueve el trabajo en equipo y se preocupa por el aprendizaje de los desarrolladores.
- **Metodologías de desarrollo de software:** Es un marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo en sistemas de información.
- **Multimedia:** se utiliza para referirse a cualquier objeto o sistema que utiliza múltiples medios de expresión (físicos o digitales) para presentar o comunicar información.

- **Navegación:** es una aplicación que opera a través de Internet, interpretando la información de archivos y sitios web que se halla alojada en un servidor dentro de la Internet (World Wide Web) o en un servidor local.
- **Plataforma Java:** Sirve para ejecutar aplicaciones desarrolladas usando el lenguaje de programación Java.
- **Rich Internet Applications (RIA's):** Es una aplicación web que tiene la mayoría de las características de las aplicaciones de escritorio tradicionales. Estas aplicaciones utilizan un navegador web estandarizado para ejecutarse y por medio de complementos o mediante una máquina virtual se agregan las características adicionales.
- **Servidor de Aplicación Wildfly 11:** Es un servidor de aplicaciones J2EE de código abierto que puede ser utilizado en cualquier sistema operativo para el que esté disponible Java.
- **Sitio web:** es una colección de páginas web relacionadas y comunes a un dominio de Internet o subdominio en la World Wide Web en Internet.
- **Tecnologías Vista (JSF):** es el framework oficial de Java Enterprise para el desarrollo de interfaces de usuario (UI) avanzadas en aplicaciones web.
- **XHTML:** lenguaje de marcado pensado para sustituir a HTML como estándar para las páginas web.
- **XML:** (Extensible Markup Language) Lenguaje de marcas para almacenar datos, de forma que luego se los pueda leer sin mayores inconvenientes gracias a su estructura y estándar.

CAPÍTULO I

ASPECTOS GENERALES

1.1 INTRODUCCIÓN

El 16 de abril del 2016, el país sufrió una de las peores desgracias de las últimas dos décadas. 663 personas fallecieron en este evento, que afectó principalmente a las provincias de Manabí y Esmeraldas, lo cual obligó a la población mayormente afectada a alejarse de sus viviendas no solo por su destrucción sino por los graves daños en sus estructuras, que inviabilizaban su permanencia en ellas. (Vera, 2017)

Ante esta catástrofe suscitada, en apoyo al estado ecuatoriano la Universidad de las Fuerzas Armadas – ESPE junto con otras entidades ministeriales del estado han llevado el Control y Administración de los Albergues, los mismos que están destinados a ayudar a las personas que por causa del terremoto perdieron su casa, y no tienen suficientes recursos para poder subsistir, por lo que se han visto en la necesidad de solicitar ayuda tanto en organizaciones como en fundaciones de asistencia social. (Vera, 2017)

Actualmente, se tienen dificultades al momento de llevar un control automatizado por lo que los registros de información de los datos obtenidos de los usuarios se realizan de forma manual (Excel, Word, Correo electrónico, llamadas telefónicas), lo cual no presta ninguna clase de ayuda eficiente, lo cual da origen a que se dificulte el ingreso de información de manera que dificulta la toma de decisiones orientadas al manejo y control de los albergues en cuanto a que no se puede gestionar tareas de coordinación con los administradores de otros albergues.

La Universidad de las Fuerzas Armadas-ESPE buscando siempre el mejoramiento en la calidad de servicios para sus actividades, desea realizar un estudio comparativo de librerías de componentes (OpenFaces, RichFaces, PrimeFaces, OmniFaces, ButterFaces, BootsFaces) para seleccionar la mejor herramienta que se adapte al

“Sistema web utilizando arquitectura jee7, Caso de Estudio albergues de damnificados del terremoto de Manabí”, el cual está enfocado en mejorar su estilo de llevar el control manual y llevarlo a un ambiente de tecnología actual, donde se facilite el acceso a la información de los Albergues de damnificados de forma integrada y segura.

1.2 ANTECEDENTES

A pesar de los estándares y los cambios de perspectiva implantados en la Universidad de las Fuerzas Armadas Espe al momento no se ha identificado una librería de componentes (framework) para la capa de presentación en Java que permita desarrollar software de forma ágil que proporcione mayor facilidad en la elaboración de pantallas de forma sencilla y, sobre todo, fácil de mantener. (ESPE, 2017)

“La elevada complejidad de muchas de las aplicaciones informáticas de hoy en día no orientan a un camino correcto en el desarrollo de aplicaciones por lo que no se posee mecanismos de reutilización, los cuales permitan a los programadores evitar partir desde cero en cada proyecto. Para ello, una solución ampliamente extendida es la utilización de frameworks de desarrollo”. (Morcillo, 2010)

Los objetivos principales que persigue un framework son: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones. Un framework Web, por tanto, podemos definirlo como un conjunto de componentes (por ejemplo, clases en java y descriptores y archivos de configuración en XML) que componen un diseño reutilizable que facilita y agiliza el desarrollo de sistemas Web. (Gutiérrez, 2015)

Debido que en el mercado existe una amplia variedad de frameworks para la capa de presentación se va a realizar un análisis comparativo de las diferentes alternativas para elegir la que mejor se ajuste a las necesidades del caso de estudio. (Foces, 2009)

Es por esto que se va a desarrollar un sistema de Albergues para damnificados de la provincia de Manabí donde se aplique la Investigación realizada sobre el Estudio comparativo de librerías de componentes JSF para la implementación de un sistema web utilizando arquitectura JEE 7.

La implementación del caso de estudio permitirá a las Organizaciones y fundaciones de ayuda social ecuatorianas apoyar al Comando Conjunto de las Fuerzas Armadas a cumplir con la disposición del gobierno para que administre los albergues de los damnificados del terremoto que ocurrió en la Provincia de Manabí el 16 de Abril del 2016 (Vera, 2017)

Con el fin de realizar el presente proyecto de investigación de manera eficiente y eficaz se ha considerado utilizar una metodología de desarrollo web que garantice la calidad del proyecto, la misma que a su vez brinde las mejores prestaciones para solucionar éste problema, por este motivo se ha considerado pertinente realizar un estudio previo utilizando la Metodología Xp al presente Caso de estudio, orientada específicamente a la colaboración con el cliente y al desarrollo incremental del software. (Beck,Kent.,Fowler,Martin, 2004)

1.3 PLANTEAMIENTO DEL PROBLEMA.

¿Es necesario realizar un Estudio comparativo de librerías de componentes JSF para la implementación de un sistema web utilizando arquitectura JEE 7, caso de estudio albergues de damnificados del terremoto de Manabí?

En relación a la pregunta anterior, se podría decir que actualmente en la Universidad de las Fuerzas Armadas- ESPE, no se dispone de Herramientas de Diseño de Java orientadas a Librerías de Componentes JSF que permitan utilizar en Sistemas Informáticos para gestionar y controlar la información de manera eficiente en lo referente a la Administración de Albergues de Damnificados.

En base a los estudios técnicos realizados por la Universidad de las Fuerzas Armadas- ESPE se considera que los problemas de una mejor gestión con la información puede darse porque no se utiliza Herramientas de desarrollo de software adecuadas como lo son los frameworks JSF, tomado en cuenta que cualquier cambio de herramientas para el desarrollo reduce el efecto de productividad en la fase inicial; por lo tanto tomar la decisión de la herramienta que se utilizará es de vital importancia para el buen desarrollo del proyecto. (Foces, 2009)

La administración de los Albergues requiere llevar a cabo un mejor control de la información para que la ayuda que se brinde a los damnificados del terremoto de Manabí sea un servicio de asistencia social real. Requiere que exista un control físico, referido a la infraestructura de cada albergue, tipo de construcción, área en la que se encuentra, si dispone servicios básicos como son agua, luz, alimentación , internet, servicios de salud para ayuda de los damnificados, se necesita llevar a cabo un control del personal que labora en los albergues, en la actualidad en los albergues se encuentra trabajando personal de diferentes instituciones públicas como son del (Ministerio de Inclusión Económica y Social (MIES), Ministerio de Salud y de otros Ministerios, personal de servidores públicos de los diferentes Ministerios son la parte Administrativa por lo tanto se desea llevar un control administrativo del personal que trabaja para los albergues. (Vera, 2017)

Por otro lado existen los damnificados que son las personas que por causa del terremoto perdieron su casa, y no tienen suficientes recursos para poder subsistir , por lo que asisten y forman parte ya de los albergues, como damnificados, aquí hay hombres, mujeres y niños, personas discapacitadas, personas de la tercera edad, a las mismas que se les da atención médica , alimentación diaria, esto hace que se deba llevar un control de quienes son las personas que están habitando en los albergues de damnificados y por último se necesita un sistema que permita emitir informes y reportes tanto en lo referente al personal que labora administrativamente en los albergues como del personal que vive, que habita en los albergues como damnificados. (Vera, 2017)

Los registros de información de los datos obtenidos de los usuarios se realizan de forma manual (Excel, Word, Correo electrónico), lo cual no presta ninguna clase de ayuda eficiente, lo cual da origen a que se dificulte el ingreso de información para los administradores lo que impide tomar decisiones y coordinar tareas hacia los partes diarios. (ESPE, 2017)

1.4 JUSTIFICACIÓN.

Establecer un conjunto de librerías JSF con el fin de implementar aplicaciones web con mejores características de prestaciones como son usabilidad, funcionabilidad, mantenimiento y portabilidad. (Donell, Martin&Wallace,Mark&McGuckin,Tony, 2011)

Proponer el desarrollo de un sistema de apoyo, para la administración de los Albergues de damnificados del terremoto de Manabí, con el fin de evidenciar las librerías JSF seleccionadas en la investigación propuesta, de manera que permitan mantener una comunicación activa entre el usuario y la información, mediante el uso de un navegador web como cliente ligero, además que sea multiplataforma con consumo de recursos bajo, que ocupe poco espacio de almacenamiento en el disco, además , que sea portable y con una alta disponibilidad.

El sistema contendrá los módulos de Usuarios, Albergues, Damnificados, Personal Administrativo, Parte Diario, los mismos que deben interactuar con el sistema de administración de albergues, con el objetivo de evitar posibles errores en los registros.

1.5 OBJETIVOS.

1.5.1 OBJETIVO GENERAL.

- Realizar un estudio comparativo de librerías de componentes JSF para la implementación de un sistema web utilizando arquitectura JEE7, aplicado a un

caso de estudio de albergues de damnificados del terremoto de Manabí, para la Universidad de las Fuerzas Armadas - ESPE.

1.5.2 OBJETIVOS ESPECÍFICOS.

- Realizar el levantamiento y análisis de la información de administración de albergues que contempla los módulos de Usuarios, Albergues, Damnificados, Personal Administrativo, Parte Diario con el fin de recolectar la información necesaria de los Albergues.
- Realizar el análisis comparativo de librerías de componentes JSF, para determinar aquella que brinde mayores beneficios en el desarrollo del sistema.
- Diseñar, desarrollar el Sistema de Albergues de damnificados del terremoto de Manabí utilizando la metodología de desarrollo ágil Xp.
- Implementar y realizar las pruebas del Sistema de Albergues de damnificados del terremoto de Manabí.
- Elaborar el manual técnico y de usuario.

CAPÍTULO II

ESTADO DEL ARTE

2.1 Herramientas de desarrollo

Se realizará una investigación con el fin de optimizar la fase de desarrollo del proyecto. Esta investigación consiste en la realización de una comparativa de librerías de componentes JSF más utilizados para la capa de presentación utilizados actualmente para realizar aplicaciones web,

Uno de los principios de diseño de la capa de presentación en aplicaciones Java EE es separar el control de flujo de la capa de negocios y las llamadas de la capa de presentación. Para este fin, las aplicaciones Java EE usualmente usan la arquitectura Modelo-2, que incluye el conocido patrón de diseño Modelo-Vista-Controlador (MVC). Las ventajas del patrón MVC son una clara separación de preocupaciones, lo que resulta en aplicaciones más flexibles, que son más fáciles de administrar y actualizar. (Caballé, Santi and Ortega, Jose-Arturo and Camps, Josep-Maria and Barolli, Leonard and Kulla, Elis and Spaho, Evjola, 2014)

Con esta investigación se pretende analizar los marcos Java EE más comunes existentes con propósitos similares en términos de funcionalidad, aplicabilidad y comparación.

La realización de este estudio permite profundizar en el conocimiento de la capa de presentación de las aplicaciones Java EE y recopilar los requisitos adecuados para establecer un marco de trabajo con el enfoque en la facilidad de uso. (Mois, 2015)

Con la realización de esta investigación, no se pretende establecer cuál es el componente de librerías más eficiente, puesto que cada framework tiene funciones útiles para el desarrollo web, lo que se pretende es justificar la elección de la tecnología

usada para el desarrollo de este proyecto en concreto. (Caballé, Santi and Ortega, Jose-Arturo and Camps, Josep-Maria and Barolli, Leonard and Kulla, Elis and Spaho, Evjola, 2014)

Un framework de Java, por tanto, podemos definirlo como un conjunto de clases relacionadas y otros elementos de soporte que hacen que las aplicaciones web Java a desarrollar sean más fáciles de mantener mediante el suministro de piezas pre-construidas. (Gutierrez, 2017)

Para llevar a cabo esta investigación se ha considerado realizar un análisis de frameworks basados en tecnología Java Server Faces (JSF) y hacer comparativa de ellos, que ayude a obtener información relevante en base a sus características, cabe destacar que algunos de estos frameworks ya se conocían durante la carrera, lo que ha facilitado adaptarse a ellos.

2.2 Normas ISO

Son un conjunto de normas orientadas a guiar la gestión de una empresa en sus distintos ámbitos. La alta competencia internacional acentuada por los procesos globalizadores de la economía y el mercado y el poder e importancia que ha ido tomando la figura y la opinión de los consumidores, ha propiciado que dichas normas, pese a su carácter voluntario, hayan ido ganando un gran reconocimiento y aceptación internacional (Isotools, 2015).

En el año 2005 se da a conocer la Norma Internacional ISO/IEC 25000, conocido también como SQuaRE (Ingeniería de Software - Requisitos de Calidad y Evaluación de Productos de Software), la cual reemplaza a la ISO/IEC 9126(Evaluación de la Calidad de Software) e ISO/IEC 14598(Evaluación del Producto de Software). El objetivo de crear SQuaRE es cubrir dos procesos importantes como son: la especificación de

requerimientos y la evaluación de calidad de software, apalancados por un proceso de medición de la calidad de software. (INEN NTE INEN-ISO/IEC 25000, 2014).

Además de ayudar a aquellos productos software en desarrollo o en adquisición. Se debe resaltar que las series de Normas Internacionales SQuaRE están dedicadas únicamente a la calidad del producto de software

SQuaRE consta de cinco divisiones (INEN NTE INEN-ISO/IEC 25000, 2014):

- ISO/IEC 2500n. División para la gestión de la calidad
- ISO/IEC 2501n. División de modelo de calidad
- ISO/IEC 2502n. División de medición de calidad
- ISO/IEC 2503n. División para los requisitos de la calidad
- ISO/IEC 2504n. División para la evaluación de la calidad

La notación tiene la forma ISO/IEC 250mn, donde m representa un apartado o división del estándar y n un documento dentro del mismo.

En la familia ISO/IEC 25000 también pertenecen las siguientes: (INEN NTE INEN-ISO/IEC 25000, 2014)

- 25001: Planificación y gestión
- 25010: Modelo de calidad
- 25020: Modelo de referencia de medición y guía
- 25021: Elementos de medida de calidad
- 25022: Medición de calidad interna
- 25023: Medición de calidad externa
- 25024: Medición de calidad en uso
- 25030: Requisitos de calidad
- 25040: Modelo de referencia de evaluación de la calidad y guía
- 25041: Módulos de evaluación
- 25042: Proceso para desarrolladores

- 25050 - 25099: Extensiones de SQuaRE y Reportes Técnicos.

Los mayores beneficios de la Norma Internacional ISO/IEC 25000 es la coordinación de la guía de medición y evaluación de calidad la cual proporciona la guía para la especificación de los requerimientos de calidad y la concertación con la Norma ISO/IEC 15939 como Modelo de referencia de medición de calidad, presentada en ISO/IEC 25020 – Modelo de referencia y guía de medición de evaluación y requerimientos de calidad del producto software (INEN NTE INEN-ISO/IEC 25000, 2014)

El objetivo general de crear el conjunto de estándares internacionales de SQuaRE fue pasar a una serie lógicamente organizada, enriquecida y unificada que cubra dos procesos principales: la especificación de los requisitos de calidad del software y la evaluación de la calidad de los sistemas y el software, respaldada por un proceso de medición de la calidad del sistema y del software.

El propósito del conjunto de estándares internacionales de SQuaRE es ayudar a aquellos que desarrollan y adquieren sistemas y productos de software con la especificación y evaluación de los requisitos de calidad. Establece criterios para la especificación de los requisitos de calidad de los productos de software y sistemas, su medición y evaluación. Incluye un modelo de calidad de dos partes para alinear las definiciones de calidad del cliente con los atributos del proceso de desarrollo. Además, la serie proporciona medidas recomendadas de los atributos de calidad de los productos de software y sistemas que pueden ser utilizados por los desarrolladores, adquirentes y evaluadores.

El Ecuador adopta esta Norma Internacional dando origen a la NORMA TÉCNICA ECUATORIANA. NTE INEN - ISO/IEC 25000 REQUERIMIENTOS Y EVALUACIÓN DE CALIDAD DEL PRODUCTO DE SOFTWARE, con la cual se realizará el presente proyecto de Investigación.

2.2.1 Requerimientos y Evaluación de Calidad del Producto de Software

Esta Norma Ecuatoriana es una traducción idéntica a la Norma Internacional ISO/IEC 25000:2005 (Ingeniería de sistemas y software - Sistemas y software Requisitos de calidad y evaluación (SQuaRE) - Guía para SQuaRE).

Los responsables de preparar esta norma y su traducción es el Comité Técnico Conjunto ISO/IEC JTC 1, Tecnologías de la Información, Subcomité SC 7, Software e Ingeniería de Sistemas (INEN NTE INEN-ISO/IEC 25000, 2014). En el año 2014 fue publicada la última versión vigente.

En esta traducción también se pone de manifiesto los predecesores de SQuaRE, ISO/IEC 9126:1991(calidad del producto de software) e ISO/IEC 14598:1999 (evaluación del producto de software), (INEN NTE INEN-ISO/IEC 25000, 2014).

Su objetivo principal es guiar el desarrollo de los productos de software con la especificación y evaluación de requisitos de calidad.

2.2.2 Proceso de transición entre ISO/IEC 9126, ISO/IEC 14598

Como las características de calidad y las medidas asociadas pueden ser útiles no solo para evaluar sistemas y productos de software, sino también para definir los requisitos de calidad, el predecesor de SQuaRE, ISO / IEC 9126: 1991 ha sido reemplazado por dos estándares internacionales relacionados de múltiples partes: ISO / IEC 9126 (Calidad del producto del software) e ISO / IEC 14598 (Evaluación del producto del software). Los siguientes puntos derivados del uso práctico de ambas series dieron el impulso lógico para crear la nueva serie de Estándares Internacionales SQuaRE:

- Tanto ISO / IEC 9126 como ISO / IEC 14598 tienen raíces comunes normativas, referenciales y funcionales.

- **ISO / IEC 9126 e ISO / IEC 14598** forman un conjunto complementario de estándares.

Los ciclos de vida independientes de ambas series han creado inconsistencias entre ellas.

- **ISO/IEC 9126-1: 2001**: Presenta el modelo de calidad que clasifica la calidad del software en un conjunto estructurado de características: funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad.
- **ISO/IEC 9126-2:2003** : Describe las métricas externas que se pueden utilizar para evaluar el comportamiento del software cuando es operado por el usuario.
- **ISO/IEC 9126-3:2003** : Describe las métricas internas que se pueden utilizar para describir propiedades estáticas de la interfaz, que puede ser evaluadas por la inspección sin poner en funcionamiento el software.
- **ISO/IEC 9126-4:2004** : Describe las métricas de calidad que se pueden utilizar para evaluar el efecto del producto software cuando son operados por el usuario en determinados contextos de uso.
- La **ISO 14598** es un estándar que proporciona un marco de trabajo para evaluar la calidad de todo tipo de producto software e indica los requisitos para los métodos de medición y el proceso de evaluación. Consta de seis partes:
- **ISO/IEC 14598-1: 1999** : provee una visión general de las otras cinco partes y explica la relación entre la evaluación del producto software y el modelo de calidad definido en la ISO/IEC 9126.
- **ISO/IEC 14598-2: 2000** : contiene requisitos y guías para las funciones de soporte tales como la planificación y gestión de la evaluación del producto del software.
- **ISO/IEC 14598-3: 2000** : provee los requisitos y guías para la evaluación del producto software cuando la evaluación es llevada a cabo en paralelo con el desarrollo por parte del desarrollador.
- **ISO/IEC 14598-4:1999** : provee los requisitos y guías para que la evaluación del producto software sea llevada a cabo en función a los compradores que planean adquirir o reutilizar un producto de software existente o pre-desarrollado.

- **ISO/IEC 14598-5: 1998** : provee los requisitos y guías para la evaluación del producto software cuando la evaluación es llevada a cabo por evaluadores independientes.
- **ISO/IEC 14598-6:2001** : provee las guías para la documentación del módulo de evaluación.

Cabe indicar que en la familia de la NORMA TÉCNICA ECUATORIANA INEN-ISO/IEC 25000 REQUERIMIENTOS Y EVALUACIÓN DE CALIDAD DEL PRODUCTO DE SOFTWARE se encuentra la NORMA TÉCNICA ECUATORIANA. NTE INEN-ISO/IEC 25040 SISTEMAS E INGENIERÍA DE SOFTWARE - REQUERIMIENTOS Y EVALUACIÓN DE SISTEMAS Y CALIDAD DE SOFTWARE - PROCESO DE EVALUACIÓN, ésta última se utiliza en el proceso de evaluación del producto software para el rol evaluador que es el objetivo principal de este trabajo Realizar un estudio comparativo de librerías de componentes JSF para la implementación de un sistema web utilizando arquitectura JEE7.

2.3 Java Server Faces (JSF)

La tecnología Java Server Faces es un framework de interfaz de componentes de usuarios del lado del servidor para las aplicaciones web que utiliza Java Server Pages (JSP) como la tecnología que permite hacer el despliegue de las páginas. (Mann, 2005) JSF asegura que Las aplicaciones están bien diseñadas con mayor facilidad de mantenimiento Integrando el modelo, la vista y el controlador (MVC) en su arquitectura. (Kurniawan, 2013)

Las ventajas de usar un framework a la hora de realizar un proyecto son diversas, entre otras, se disminuye el tiempo de creación de las aplicaciones, facilita el mantenimiento del código y hace uso de patrones. El patrón más utilizado por casi todos los frameworks es el conocido como Modelo Vista Controlador (MVC). (Pantoja, Libardo & Pardo, Cesar, 2016)

La conexión de la vista con el controlador se hace por medio de los denominados Beans Gestionados (Managed Beans), los cuales toman valor en la vista para ser usados en la lógica, la que a su vez les proporciona un valor para que se muestren en la vista. (Schalk, 2010)

A continuación, en la Tabla 1 se muestran los principales componentes de la tecnología JSF.

Tabla 1

Componentes de la tecnología Java Server Faces

API	Librería de etiquetas (tags)
Definición y Control de estado de componentes.	Cabeceras predefinidas (tags) para componentes visuales.
Manejar eventos, validar en el servidor y conversión de datos.	Etiquetas JSP personalizadas para expresar componentes en una página JSP y enlazar los componentes a objetos del servidor.
Validación del lado del Servidor	
Conversión de datos del servidor	
Definir la navegación de páginas.	
Soporte de internacionalización y accesibilidad.	

Fuente: (Schalk, 2010)

A continuación, en la Figura 1 se ilustra una arquitectura MVC creada, utilizando tecnologías basadas en Java. El patrón de diseño MVC separa las vistas y los modelos estableciendo protocolo de suscripción / notificación entre ellos.

En esta arquitectura, las solicitudes HTTP de los usuarios se enrutan a través de un controlador, que es el encargado de manejar la interacción del usuario con la aplicación y es típicamente implementado como un servlet, el modelo encapsula datos de aplicación y lógica empresarial; la vista maneja el renderizado de los datos de la aplicación y la interfaz visual para el usuario.

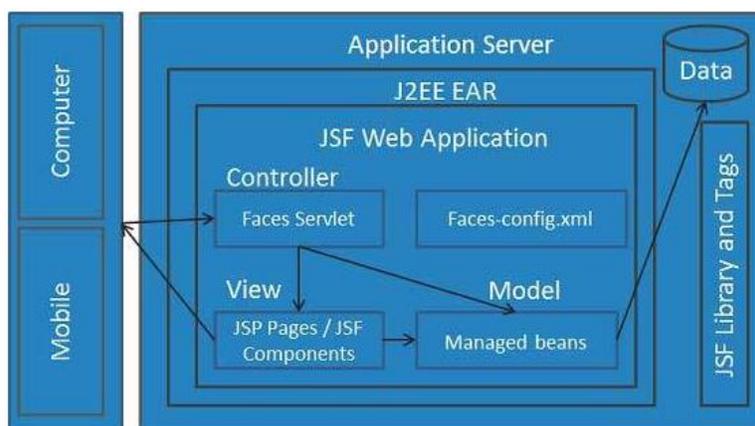


Figura 1. Arquitectura JSF

Fuente: (Costa, 2014)

De acuerdo a la Investigación realizada por (Harshad B, 2014). Se puede indicar que cuando el usuario interactúa con el patrón de diseño MVC basado en Formulario GUI o página, todos los eventos enviados, son capturados por el objeto controlador. El controlador entonces decide si el evento enviado es relacionado con el cambio de estado del modelo o el cambio de estado de la vista, como ejemplo, cuando el usuario envía un evento relacionado con el valor de cambio en el campo de texto, el controlador llama al método del modelo para cambiar su contenido. Pero si el usuario realiza un desplazamiento horizontal o vertical hacia arriba o hacia abajo, el contenido del modelo no cambia y solo la vista debe ser notificada para reflejar los cambios en su apariencia.

En la Figura 1 se puede apreciar como La tecnología JSF se adopta al patrón arquitectónico Modelo-Vista-Controlador (MVC) el mismo que divide la estructura de la aplicación en tres capas (Costa, 2014)

- **Modelo** – Se encarga de controlar el comportamiento y los datos del dominio de la aplicación respondiendo las solicitudes e instrucciones para cambiar su estado, además de contener las reglas de negocio.
- **Vista**– Es la capa de visualización, responsable de la interfaz con el usuario y control de las entradas y salidas gráficas y textuales.

- **Controlador** - Controla los eventos generados por el usuario solicitando los datos al modelo y de comunicárselos a la vista. Esta división en capas tiene objetivo de aumentar la flexibilidad y la reutilización del código

2.3.1 Ciclo de vida de Java Server Faces.

El ciclo de vida JSF está compuesto por una secuencia de fases por las que pasa una petición JSF desde que se hace una solicitud http en el servidor hasta que se genera la página HTML resultante. (Mann, 2005)

Durante el ciclo de vida de una aplicación web, se realizan tareas comunes, incluidas las siguientes: Manejo de solicitudes entrantes, Parámetros de decodificación, Modificar y guardar estado y renderizando páginas web al navegador. (JEE7, 2014)

En la Figura 2, se muestra los pasos del ciclo de vida orientado a la solicitud – respuesta de Java Server Faces, Las seis fases muestran el orden en que JSF procesa un formulario. La Tabla 2 muestra las fases en su orden de ejecución probable con el procesamiento de eventos en cada fase.

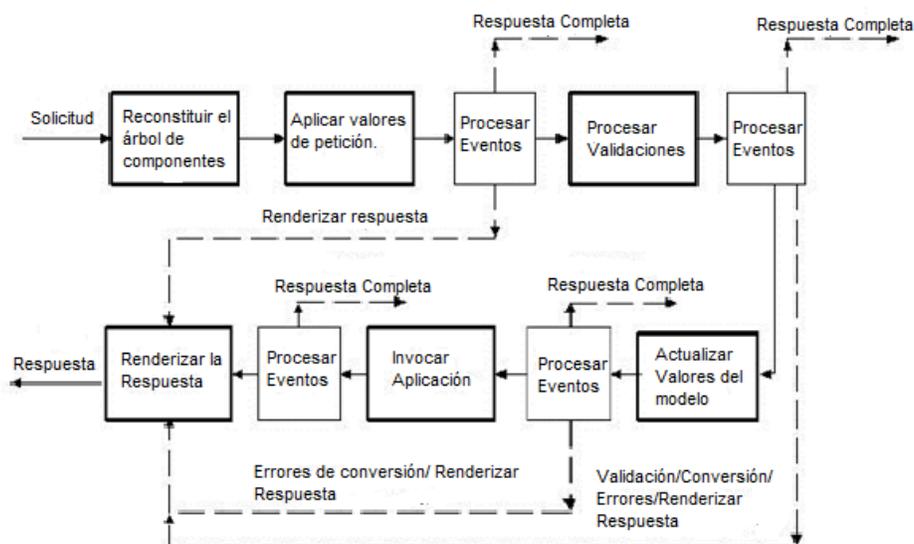


Figura 2. Ciclo de vida de Java Server Faces

Fuente: (Costa, 2014)

A continuación, en la Tabla 2, se muestran los Pasos del Ciclo de vida de Java Server Faces.

Tabla 2

Ciclo de Vida JSF

Pasos del Ciclo de Vida JSF	Detalle
Reconstituir el árbol de componentes o Restaurar la Vista.	Inicia en el momento que se realiza la petición http (request) en el servidor, En esta fase de Implementación se construye el árbol con todos los componentes de la página.
Aplicar valores de petición	Cada componente creado en la fase anterior obtiene el valor que le corresponde a la petición y lo almacena.
Procesar eventos	Durante esta fase se llevan a cabo todos los cambios realizados en los valores de los componentes asociados a las acciones realizadas por el usuario.
Procesar validaciones	Una vez que se han almacenado los valores del componente, se valida en base a las reglas que se ha declarado.
Actualizar valores del modelo.	Después de que se comprueba la validez de los datos, se recorre el árbol de componentes de manera que los valores locales de los componentes sean utilizados para actualizar las propiedades del bean correspondiente.
Invocar aplicación.	En esta fase de implementación se ejecuta cualquier evento inicial a nivel de aplicación para dar comienzo al proceso de envío o enlace de una página o un formulario.
Renderizar la respuesta	La respuesta se renderiza es decir retorna al cliente, de esta forma la vista se actualiza.

Fuente: (Schalk, 2010)

Según (Madeja, 2018) existen numerosas ventajas que hacen que JSF sea una tecnología apropiada para el desarrollo de aplicaciones web.

2.3.2 Ventajas

- Permite la separación entre la capa lógica y de presentación.
- Fácil manejo de las API's para administrar, manejar, eventos, mediante el uso de librerías propias del framework.
- Permite crear etiquetas por parte de la vista que facilitan el uso de técnicas de desarrollo web como Ajax, jQuery, y JavaScript.
- Los flujos de JSF permite a los desarrolladores agrupar de forma lógica las vistas de JSF para representar los módulos de las funciones. Cada módulo tiene un conjunto de puntos de entrada y salida bien definidos. La combinación de módulos crea un flujo de funciones. Un ejemplo de un flujo de funciones es un proceso finalización de compra de un pedido. Utilizada en la especificación JSF 2.2+ y versiones posteriores.
- Resource Library Contracts permite que las bibliotecas de recursos residan en el directorio contracts del directorio raíz web-app, o en el nombre de entrada META-INF/contracts del archivo JAR. Utilizada en la especificación JSF 2.2.
- JSF 2.2 permite a las aplicaciones utilizar vistas sin estado. Utilizada especificación JSF 2.2.
- JSF 2.2 puede manejar atributos HTML5 a través de elementos y de atributos de paso a través. Lea la sección 10.1.4 de la especificación JSF 2.2.

Según (Madeja, 2018) entre las desventajas que posee JSF se encuentran las siguientes.

2.3.3 Desventajas:

- Posee poca flexibilidad de los componentes propios en la vista si es que los componentes provienen de una librería externa.
- Su naturaleza como estándar hace que la evolución de JSF no sea tan rápida como pueda ser la de otros entornos como WebWork, Wicket, Spring, etc.

Para seleccionar las herramientas de software libre que se utilizaran en lo referente a: Lenguaje de Programación, Servidor de Aplicaciones, Entorno de Desarrollo Integrado (IDE), Motor de Base de Datos, Reportes, Arquitectura de Software y Frameworks Java de seguridad del “Sistema web para los albergues de damnificados del terremoto de Manabí” se consideró la evaluación de un análisis comparativo de características, en base a los criterios de Likert (Bertram, 2018).

Para realizarla presente comparativa se ha utilizado criterios y subcriterios en base a una evaluación comparativa de información de herramientas de desarrollo web, utilizando clasificaciones de Likert, orientadas cuestiones derivadas empíricamente, obtenidas de foros, blogs publicaciones, estadísticas posicionales de mejor rango en cuanto a aceptación (cuadrante de garner), ambiente de pruebas instalando cada uno de ellos, validando en base a obtener información relevante de su uso y funcionalidad en base a cada una de las características.

A continuación, en la Tabla 3, se muestra el cuadro comparativo de Criterios en base a las categorías de acuerdo a la valoración dada a cada Nivel de Satisfacción, la misma que será utilizada para evaluar las herramientas utilizadas en el desarrollo del Sistema para damnificados del terremoto de Manabí. (Bertram, 2018)

Tabla 3

Valoración por Categorías - Escala de Likert

Categorías (Nivel de Satisfacción)	Valoración
Insuficiente	0-3
Regular	4-5
Bueno	6-8
Muy Bueno/Satisfactorio	9-10

Fuente: (Bertram, 2018)

2.3.4 Lenguaje de Programación.

Los lenguajes de programación son la base de toda la tecnología existente con la que estamos rodeados. Los desarrolladores, personas interesadas en la tecnología mantienen actualizados sus conocimientos respecto a los lenguajes de programación utilizados para la creación de sitios web de acuerdo a los avances que se producen dentro de cada uno de ellos mediante sitios web de encuestas populares como TIOBE, Redmonk, StackOverflow, IEEE spectrum (Lobo, 2019).

De estos sitios, el espectro de IEEE sintetiza clasificaciones de 10 fuentes, incluyendo Búsqueda en Google de "programación X", Tendencias de Google, Gorjeo, GitHub, Desbordamiento de pila, Reddit, Noticias Hacker, Constructor de la carrera, Dado y Biblioteca digital IEEE Xplore (Lobo, 2019).

Actualmente existe una gran variedad de lenguajes de programación orientados a desarrollar aplicaciones web, aplicaciones en dispositivos móviles o de escritorio.

A continuación, en la Tabla 4 se muestra un cuadro comparativo de los Lenguajes de Programación más representativos al momento atendiendo sus características.

Tabla 4

Comparativo entre Lenguajes de Programación

Características	Java	Php	Python
Seguridad	10	8	8
Multipataforma	9	5	5
Robusto	9	8	8
Portable	9	10	8
Escalable	10	10	10
Flexible	10	10	10
Orientado a Objetos	10	8	6

Continúa



Multihilo	10	7	8
Dinámico	10	10	0
Interpretado	10	0	0
Promedio	9,7	7,6	6,3
Porcentaje	97%	76%	63%

Fuente: (Adecco, 2018)

Ante la consulta, se puede analizar que **Java** es el lenguaje más relevante con el **(97%)**, ubicándose en el primer lugar por ser un lenguaje de código abierto muy popular y versátil y muestra de ello es que existen multitud de aplicaciones y sitios webs que no funcionarán a menos que tenga Java instalado. Java es rápido, seguro y fiable. Desde portátiles hasta centros de datos, desde consolas para juegos hasta súper computadoras, desde teléfonos móviles hasta Internet (Adecco, 2018).

Una vez que se ha tomado como referencia estas características se observa que Java mantiene liderazgo de aceptación ante los demás lenguajes analizados es por este motivo que se ha utilizado este lenguaje para el desarrollo del “Sistema web para los albergues de damnificados del terremoto de Manabí”

2.3.5 Entorno de Desarrollo Integrado (IDE).

En la actualidad existe una gran variedad de entornos de desarrollo integrados (IDES), los mismos que pueden ser utilizados para la creación, modificación, compilación, implementación y depuración de software.

A continuación. En la Tabla 5, se muestra un Cuadro comparativo de IDE's, atendiendo sus características más representativas.

Tabla 5*Comparación de Entornos de Desarrollo Integrado (IDE's)*

Características	Eclipse IDE	Netbeans 8.1	Intellij IDEA
Facilidad en la Configuración.	10	9	8
Multiplataforma	9	9	9
Capacidad para integrarse con un sistema de versiones.	9	8	8
Reconocimiento de sintaxis.	9	8	9
Permite añadir extensiones y otros componentes,	9	9	8
Permite realizar pruebas unitarias con (JUnit)	9	9	9
Control de versiones con (CVS).	10	9	8
La compilación es en tiempo real	10	9	9
Rapidez en tiempo de ejecución	9	9	8
Documentación	10	8	9
Promedio	9,4	8,7	8,5
Porcentaje	94%	87%	85%

Fuente: (Quora, 2018)

De acuerdo al análisis de la Tabla 5 se puede observar que **Eclipse** esta con la mejor aceptación (94%), en relación a los demás IDE's comparados, puede integrar el IDE de Spring STS , siendo el más recomendado para la utilización en el desarrollo del "Sistema web para los albergues de damnificados del terremoto de Manabí" Específicamente se utilizó la tecnología JSF para implementar la capa de presentación integrada con Hibérnate para la lógica de negocio de manera de administrar los objetos de negocio y gestionar los recursos. (Eclipse, 2019)

2.3.6 Servidor de Aplicaciones.

Entre los servidores de Aplicaciones más conocidos para utilizar en el “Sistema web para los albergues de damnificados del terremoto de Manabí” se encuentran clasificados los siguientes de acuerdo a sus características más representativas:

A continuación, en la Tabla 6, se muestra un cuadro comparativo de los Servidores de Aplicaciones.

Tabla 6

Cuadro Comparativo entre Servidores de Aplicaciones

Características	Wildfly 11	Tomcat 8	Glassfish 4.1
Rápida puesta en marcha.	10	8	9
Ahorro de memoria.	10	9	8
Motor capaz de configurarse a las necesidades.	9	8	8
Administración unificada.	9	9	8
Basado en estándares.	9	8	9
Seguridad.	9	9	9
Control de versiones.	10	8	9
Modificación en caliente.	10	9	8
Soporte con Hibernate.	9	10	8
Volumen de Usuarios.	10	9	8
Promedio	9,5	8,7	8,4
Porcentaje	95%	87%	84%

Fuente: (Quora, 2018)

En cuanto a la información que provee la Tabla 3, el **Servidor de Aplicación Wildfly**, tiene el mayor grado de aceptación con el 95%, , lo cual indica que es el mejor recomendado para utilizar en el despliegue del Sistema de Albergues de damnificados del terremoto de Manabí. . (Wildfly, 2018)

2.3.7 Motor de Base de Datos

Un sistema gestor de base de datos (SGBD) es un conjunto de programas que permiten el almacenamiento, modificación y extracción de la información en una base de datos.

El motor de la base de datos provee ayuda de tal manera que la información este organizada y sea fácilmente accesible gestionada y actualizada.

En la Tabla 7 se muestran las características de tres motores de base de datos que servirán como marco de referencia para comparar y realizar la mejor elección.

Tabla 7

Cuadro Comparativo de Motores de Base de Datos

Características	María DB	Oracle9i	PostgresSQL9.4
Almacenamiento de Memoria	10	8	8
Multiplataforma	9	9	8
Seguridad	9	8	8
Encriptación	9	9	8
Control de Concurrencia	10	9	9
Seguridades	9	8	9
Soporte para transacciones	10	9	9
Reducción de la redundancia	10	9	9
Alta Disponibilidad	9	10	9
Funciones y procedimientos almacenados.	10	9	8
Promedio	9,5	8,8	8,5
Porcentaje	95%	88%	85%

Fuente: (DB-Engines, 2018)

De acuerdo a la Tabla 7, las características indican que el motor de base de datos MySQL (**MariaDB**) es el más recomendado dado que tiene el mayor porcentaje de

aceptación con un **95%**, seguido de Oracle 9i. En base a esta información se ha escogido MariaDB, por poseer características necesarias como son: compatibilidad, licenciamiento, posicionamiento, documentación, posee herramientas de utilerías y una base de conocimiento (MariaDB, 2019)

2.3.8 Reportes.

Para la creación de informes, en la Tabla 8 se realizó un cuadro comparativo entre las herramientas de reportes más recomendadas atendiendo sus características, y funcionalidad.

Tabla 8

Cuadro Comparativo de Generadores de Reportes

Características	Jasper Reports Suite	Eclipse Birt	Pentaho BI
Uso stand-alone desde otras aplicaciones	10	8	8
Interfaz de usuario muy sencilla de usar y personalizable	9	5	5
Entorno grafico Drag & Drop.	9	8	8
Provee el punto de ingreso para los reportes y análisis de datos	9	8	10
Conexión de Datos desde Script.	10	10	10
Motor Olap	10	10	10
Manejo de Consultas Hibernate Query	10	6	8
Amplia Documentación de Plantillas	10	8	7
Integración con Entornos de Desarrollo Integrados (IDES)	10	10	0
Permite la creación de Subinformes.	10	0	0
Promedio	9,7	7,3	6,6
Porcentaje	97%	73%	66%

Fuente: (Innovent-Solutions, 2018)

De acuerdo a la Tabla 8, las características indican que el Generador de Reportes Jasper Reports con el 97%, es el más recomendado para usar, porque tiene el mayor índice de aceptación, de acuerdo a las funcionalidades que brinda y por poseer características adicionales como son: está escrito completamente en Java y puede usar datos provenientes de cualquier tipo de fuente de datos y producir documentos de píxeles perfectos que se pueden ver, imprimir o exportar en una variedad de formatos de documentos, incluidos HTML, PDF, Excel, OpenOffice y Word. lo que ayudara en el despliegue de Informes en el “Sistema de albergues de damnificados del terremoto de Manabí”. (JasperReports, 2019)

2.3.9 Arquitectura de Software

Como marco de referencia para guiar la construcción del software, a continuación, en la Tabla 9 se realizó un cuadro comparativo entre las Arquitecturas de software más recomendadas atendiendo sus características y funcionalidad.

Tabla 9

Comparativo de Arquitecturas de Software

Características	JEE7	Spring Framework
Madurez.	9	8
Rapidez.	8	9
Flexibilidad de servicios	8	9
Conectividad.	9	9
Pruebas Unitarias	8	9
Manejo de transacciones	9	9
Persistencia de entidades	8	9
Seguridad	9	9
Uso de interfaces	8	9
Contenedor ligero	10	7
Promedio	8,6	8,7
Porcentaje	86%	87%

Fuente: (Reddit, 2018)

De acuerdo a la Tabla 9, las características indican que las dos Arquitecturas están a la vanguardia de las tecnologías informáticas, por mantener altos porcentajes de aceptación en cuanto a características y funcionalidad es por este motivo que se ha escogido la **Arquitectura JEE7** como la mejor opción de acuerdo a las funcionalidades que brinda como son : es abierta, basada en estándar (JCP, 2018), hardware y plataforma independiente del sistema operativo apta para aplicaciones que pueden ser desarrolladas y ejecutadas en un entorno empresarial distribuido, en donde cada componente de la plataforma se define en una especificación separada que describe la API, javadocs y el comportamiento de tiempo de ejecución esperado.

Además, se la ha elegido en base a la madurez de su crecimiento, siendo una característica muy importante, proporcionando alta calidad a la aplicación web, satisfaciendo cambios en los requerimientos del cliente y cambios en los requerimientos del negocio. En atención a estas características es recomendable para utilizar en el “Sistema web para los albergues de damnificados del terremoto de Manabí”. (Gupta, 2016)

2.3.10 Frameworks Java para Seguridad

De acuerdo con Open Web Application Security Project (OWASP, 2019), los riesgos de seguridad más críticos son:

- Inyección SQL.
- Autenticación rota y gestión de sesión.
- Referencias de objetos directos inseguros.
- Mala configuración de la seguridad.
- Exposición a datos sensibles.
- Falta el control de acceso a nivel de función.
- Uso de componentes con vulnerabilidades conocidas.
- Redirecciones y reenvíos no validados.

A continuación, en la Tabla 10 se muestran las características de los frameworks

Tabla 10
Frameworks Java para Seguridad

Características	Spring	JEE	Apache
Validaciones	<p>Realiza las validaciones a partir de un sistema propio que no pertenece a ninguna capa. Permite que el usuario integre su propia lógica de interfaz a partir de una interfaz. Posee tratamiento de errores y añade, con Spring Security otro paquete de validaciones</p>	<p>Las validaciones son definidas mediante anotaciones permitidas en la especificación JPA. Se pueden realizar validaciones individuales o completas (todos los campos de un formulario).</p>	<p>Las sesiones deben validarse para que las sesiones no válidas (caducadas o detenidas) se puedan eliminar del almacén de datos de la sesión. Esto garantiza que el almacén de datos no se llene con el tiempo con sesiones que nunca se volverán a utilizar.</p>
Seguridad	<p>Spring posee un framework de seguridad asociado, Spring Security, que gestiona todo el mecanismo de login, autenticación y autorización en aplicaciones empresariales basadas en Java. La autenticación es el proceso de establecer que un principal es quien dice ser (puede ser un usuario, dispositivo o algún otro sistema que pueda realizar una acción en su aplicación). La autorización es un proceso para decidir si un Administrador General puede realizar una acción dentro de su aplicación. (Spring, 2018)</p>	<p>La especificación de seguridad de Java EE Security es compatible con un conjunto de funcionalidades de seguridad requeridas que incluyen autenticación, autorización, integridad de datos y seguridad de transporte. La seguridad se define por Grupo y por Rol. (Blasco, 2018)</p>	<p>Apache Shiro es un marco de seguridad de Java potente y fácil de usar que realiza autenticación, autorización, criptografía y administración de sesiones. Con la API fácil de entender de Apache Shiro se puede asegurar cualquier aplicación, desde las aplicaciones móviles más pequeñas hasta las aplicaciones web y empresariales más grandes. (Blasco, 2018)</p>

Fuente: (Blasco, 2018)

Dadas las características funcionales que los frameworks presentan en base a los aspectos de Seguridad en que se centran se ha escogido **Spring Security**, porque posee una serie de clases y parámetros, que hay que personalizar y adecuar para que se ajuste de la mejor manera posible al Sistema de damnificados del terremoto de Manabí.

2.4 Metodología de Investigación

Existen dos tipos de metodologías:

Ágiles: Las metodologías ágiles son flexibles, pueden ser modificadas para que se ajusten a la realidad de cada equipo y proyecto. Los proyectos son altamente colaborativos y se adaptan mejor a los cambios en los requerimientos considerándose una característica esperada y deseada, al igual que las entregas constantes realizadas al cliente y la retroalimentación por parte de él. Tanto el producto como el proceso son mejorados frecuentemente. En cada ciclo de desarrollo, las características a ser implementados son decididas conjuntamente por el cliente y el resto del equipo de desarrollo (Navarro, A.,Fernandez,J.,&Morales,J, 2013).

La "Metodología Ágil" funciona mejor en la actualidad utiliza escenarios de requisitos que cambian con frecuencia".

Ágil "significa un enfoque rápido e incremental para desarrollo de proyectos con un fuerte enfoque y la implicación del cliente. El sistema ágil está más centrado en la codificación. En esta metodología, el código fuente es considerado como la forma de comunicación y documentación entre el usuario y el equipo de trabajo. Ágil no es más que un grupo de métodos que se basan principalmente en la iterativa del flujo de trabajo La meta principal de la metodología ágil es entrega continua y rápida del proyecto. La principal ventaja de este método sobre el tradicional es aumento en la velocidad de desarrollo, La metodología ágil está diseñada de tal manera que funciona bien en la

frecuencia de requisitos cambiantes., en la velocidad de desarrollo y la reducción en el costo (Navarro, A.,Fernandez,J.,&Morales,J, 2013).

Tradicional: Se resisten al cambio.

A continuación, en la tabla 11 se realizó un Cuadro Comparativo orientado a las características que presentan ambas metodologías.

Tabla 11

Comparativo de las Metodologías Tradicionales vs Agiles

Propiedades	Tradicional	Agiles
Actitud	Predictivo	Adaptativo
Tamaño de proyecto	Largo	Corto
Tamaño / mentalidad del equipo	Largo/disciplinado	Pequeño / innovador
Modelo de Gestión de proyectos	Autocrático	Descentralizado
Cambio de actitud	Resistencia contra el cambio	No resistencia al cambio
Documentación	Comprensivo	Ligero y abstracto
Planificación por adelantado	Comprensivo	Limitado
Ciclo de vida	Atado y atado	Iteración sin limite
Cultura Organizacional	Comando y control	Líder y colaboración
Inversión de retorno	Al final del proyecto	Durante las primeras etapas

Fuente: (Navarro, A.,Fernandez,J.,&Morales,J, 2013)

De acuerdo a la Tabla 11, En el presente proyecto se utilizó, una metodología de desarrollo ágil con la finalidad de guiar el ciclo de vida del software permitiendo mitigar y resolver los errores existentes respecto a cada etapa de desarrollo.

2.4.1 Programación Extrema (XP).

La programación extrema (XP) es un marco de desarrollo de software ágil que tiene como objetivo producir software de mayor calidad de vida para el equipo de desarrollo.

Fue presentado por "Ward Cunningham, Ron Jeffries, Kent Beck " (Joskowicz, 2008)

Xp es el marco ágil más específico en relación con las prácticas de ingeniería adecuadas para el desarrollo de software.

Es un modelo de desarrollo simple que maneja con eficacia el incierto de cambiar con frecuencia los requisitos del cliente. Destacándose en las prácticas de ingeniería con el fin de desarrollar un producto de calidad y adaptarse rápidamente a la inestable o requisitos cambiantes. Los equipos de desarrolladores de XP siguen el concepto de programación en pares. La programación de par se refiere a un par de desarrolladores o programadores trabajando juntos en una estación de trabajo única donde uno (el controlador) es responsable de la codificación y otros (el observador) es responsable de observar el código línea por línea, ya que está escrito y sugiere mejoras o cambios, cada vez que se escribe. Los dos programadores cambian de rol con frecuencia. (Joskowicz, 2008) Debido a la práctica de la programación de pares éxito del proyecto depende en gran medida de la comunicación entre los miembros del equipo.

A continuación, en la Tabla 12, se muestra los Valores que posee la Programación Extrema (Xp).

Tabla 12
Valores de la Programación Extrema XP

Valores	Detalle
Comunicación	La Comunicación directa con el cliente es la mejor forma de comunicación, entre los desarrolladores y el cliente.
Simplicidad.	La simplicidad ayuda a que se encuentren soluciones más simples a problemas según los requerimientos del cliente, por medio del desarrollo se crean características en el diseño que pueden ser útiles para resolver problemas en el futuro.
Retroalimentación	La retroalimentación continua permite llevar y dirigir el proyecto en una dirección correcta orientada a los requerimientos del cliente.
Valentía	Cuando se considera ir a la par con el cambio, porque es inevitable. Los equipos de XP tienen la valentía de ser transparentes porque solo a partir de la transparencia obtendrán una retroalimentación real de parte de los clientes.

Fuente: (Joskowicz, 2008)

2.4.2 Roles de Programación Extrema (XP).

Como en todo equipo de trabajo se deben asignar o tener roles y en la Programación extrema, existen varios roles (actores) y responsabilidades asignadas para controlar diferentes tareas y propósitos durante el proceso de desarrollo del proyecto.

A continuación, en la Tabla 13, se muestra los roles asignados al equipo de trabajo.

Tabla 13

Roles de Programación Extrema XP

Roles	Detalle
Programador	Es el productor del código responsable de planificar, coordinar y revisar todas las actividades técnicas del equipo.
Cliente	Se encarga de decidir qué Sistema se va a construir y que funcionalidad va a aportar.
Encargado de pruebas (Tester)	Evalúan, planifican y llevan a cabo pruebas de software para comprobar si funcionan correctamente
Encargado de seguimiento (Tracker)	Coordina el seguimiento, la ejecución y la verificación entre las estimaciones realizadas en cada fase y el tiempo real dedicado.

Fuente: (Åkerblom, 2018)

2.4.3 Fases de Programación Extrema (XP).

El proceso de desarrollo de software comienza con la planificación, y todas las iteraciones constan de cuatro fases básicas en su ciclo de vida: diseño, codificación, pruebas y planeación.

En la Figura 3 se muestra el detalle del proceso generado por las fases de desarrollo del ciclo de vida del proyecto utilizando la Metodología de Programación Extrema XP.

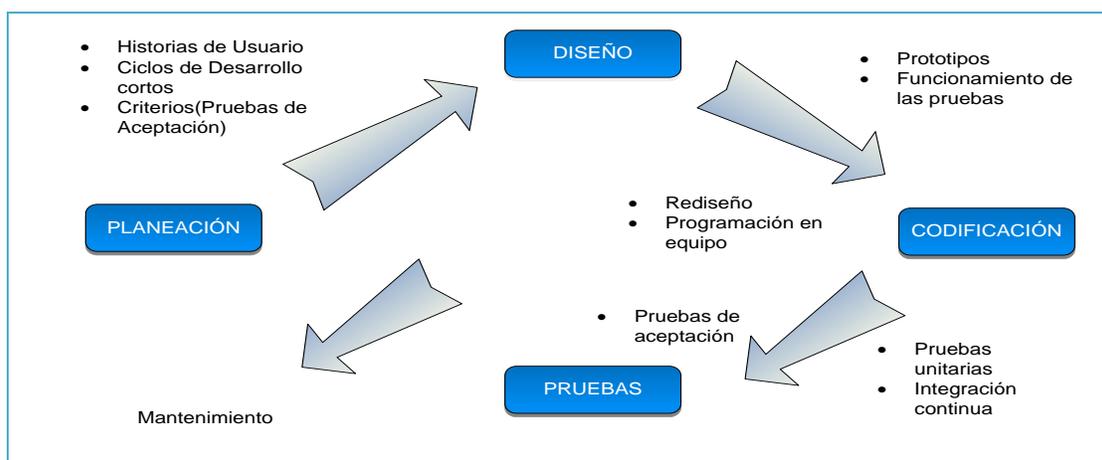


Figura 3. Fases de la Programación Extrema (XP)

Fuente: (Pressman, 2010)

A continuación, en la Tabla 14, se detallan las fases de la programación extrema, y los pasos que comprende cada una.

Tabla 14

Fases de la Programación Extrema (XP)

Planificación	Diseño	Desarrollo	Pruebas
1. Historias de Usuario	1. Metáfora del Sistema	1. Disponibilidad del Cliente	1. Implantación
2. Plan de Entregas	2. Tarjetas CRC	2. Unidad de Pruebas	2. Pruebas de Aceptación
3. Velocidad del Proyecto	3. Soluciones Puntuales	3. Programación por pares	
4. Iteraciones	4. Funcionalidad mínima	4. Integración	
5. Rotaciones	5. Reciclaje		
6. Reuniones			

Fuente: (Borja, 2018)

2.4.3.1 Planificación.

La Planificación, se realizó posteriormente a la realización del estudio del problema en base a reunir los requerimientos necesarios se procedió a interactuar con el cliente y los demás integrantes del equipo de desarrollo se procedió a identificar el tamaño y

número de las iteraciones y de esta forma realizar los correctivos necesarios a la metodología en base a las características presentes en el proyecto.

Las características que tienen cada uno de estos elementos dentro de la de planificación se muestran, a continuación, en la Tabla 15.

Tabla 15

Características de la Etapa de Planificación

Elemento	Características
Historias de usuario	Las historias de usuario son herramientas utilizadas por el equipo de desarrollo para dar a conocer los requerimientos del sistema. Representan las actividades que realizara el sistema atendiendo la necesidad del cliente de forma clara y sencilla sin profundizar detalles.
Velocidad del proyecto	La velocidad del proyecto es utilizada para establecer cuántas historias se pueden implementar antes de una fecha determinada o cuánto tiempo tomará implementar un conjunto de historias.
Iteraciones	Son las etapas en las que se divide el desarrollo del sistema para facilitar su realización. La duración varia, entre una o tres semanas. Para cada iteración se define un módulo o conjunto de historias a implementar.
Plan de entregas	El equipo de trabajo y el cliente definen los tiempos que se utilizaran en la realización de las diferentes etapas del proyecto, los cuales no son datos exactos, pero proporcionan información importante que sirve de base para el cronograma. A partir de las historias de usuarios, se comprueba que cada actividad ha sido correctamente implementada, el cliente será el encargado de plantear las pruebas de aceptación.
Reuniones	Se realiza una revisión continua del plan de trabajo poniendo énfasis en la organización del trabajo.
Roles en XP	Mediante la asignación de roles se planifica la organización de quienes se encargarán de la realización cada una de las actividades que forman parte del proyecto, sin descartar la posibilidad de rotar los roles entre el equipo si fuera necesario.
Cambio del personal y ajuste a XP.	Todos los miembros del equipo deben tener conocimiento de la estructura del código para poder avanzar en la realización de manera que se evite perdida

de información. A medida que todos los programadores entienden todas las partes del programa se puede hacer balance en la carga de trabajo, de manera que el trabajo sea equitativo.

Fuente: (Beck,Kent.,Fowler,Martin, 2004)

2.4.3.2 Diseño.

En esta fase de la metodología Xp se diseñan las historias de usuario basadas en la funcionalidad del sistema, sirve para realizar correctivos permanentes de diseño en el proyecto desde la realización hasta la implementación,

A continuación, en la Tabla 16 se detallan las características de esta fase:

Tabla 16

Características de la Etapa de Diseño

Elemento	Características
Simplicidad en el Diseño	Un aspecto importante en la metodología XP es la sencillez en el diseño. La idea es un diseño más sencillo que cumpla con los requerimientos de las historias de usuario.
Metáfora del sistema	El sistema es definido por una metáfora o conjunto de metáforas compartidas entre el cliente y el equipo de desarrollo donde se describe como debería funcionar el sistema,
Tarjetas de clase , responsabilidad, colaboración (CRC Cards)	Ayudan al desarrollo orientado a objetos olvidándose de los malos hábitos de la programación procedimental clásica. Describen las clases utilizadas en la programación de una historia. El esquema con la que se representa está dado por el nombre de la clase en la parte superior, responsabilidades a la izquierda, colaboradores a la derecha.
Soluciones puntuales(Spike Solution)	Es una pequeña aplicación desconectada del proyecto que sirve para solucionar inconvenientes que generen problemas de diseño e implementación en los requerimientos del cliente (historias de usuario)
No Solucionar antes de tiempo.	Se analiza lo que se está desarrollando en la iteración actual, olvidando cualquier necesidad que se pueda presentar en el futuro.



Funcionalidad Extra	Se puede añadir dado los requerimientos del Sistema.
Refactorización	Refactorizar es mejorar y modificar la estructura de códigos ya creados sin alterar su funcionalidad de manera de conservar su sencillez y que sea fácil de mantener.

Fuente: (Beck,Kent.,Fowler,Martin, 2004)

2.4.3.3 Codificación.

Esta fase de la metodología Xp, se la realiza con el equipo de trabajo luego se lleva a cabo una integración paralela con el diseño, debido a que no se garantiza la consistencia y la calidad existe la necesidad de hacer pruebas de supervisión exhaustivas, dejando la optimización al final una vez que el código requerido este completo. (Joskowicz, 2008)

A continuación, en la Tabla 17 se detallan las características de la fase de codificación.

Tabla 17

Características de la Etapa de Codificación

Elemento	Características
Disponibilidad del Cliente	Uno de los requerimientos de XP es que el cliente este siempre disponible. No solamente para solucionar las dudas del grupo de desarrollo, debería ser parte de este.
Elaboración de pruebas unitarias.	Cuando se realiza una prueba, se ahorra mucho tiempo elaborando el código que la haga pasar, siendo menor el tiempo de hacer ambos procesos que crear el código solamente.
Programación en parejas.	En esta fase de la metodología Xp, se utiliza una técnica empleada en el desarrollo ágil de software, que consiste en trabajar en el mismo equipo dos programadores de forma conjunta. Uno de ellos (desarrollador) escribe el código, mientras que el otro (observador) lo supervisa.
Integración secuencial.	Los inconvenientes presentados en proyectos de software tienen que ver con la integración. XP, para ello se debe emplear un sistema de turnos, para saber cuál es la última versión, con lo que se evita problemas de compatibilidad.



Integraciones frecuentes.	Se deben realizar integraciones periódicas cada corto tiempo siempre que sea posible de manera que no transcurra más de un día entre una iteración y otra.
----------------------------------	--

Fuente: (Joskowicz, 2008)

2.4.3.4 Pruebas.

En esta fase de la metodología Xp, las pruebas se realizan cuando no existe claridad en la interfaz de un método, la implementación presenta dificultades, las entradas muestran condiciones inusuales, o luego de modificar algo. Éstas deben contemplar cada módulo del sistema que pueda generar fallas. Para poder integrar el código modificado al ya existente, el mismo debe aprobar satisfactoriamente todos los casos de prueba definidos. (J. J. Gutiérrez, 2018)

Lo que se pretende con las pruebas es aumentar la calidad del sistema reduciendo el número de errores no detectados y disminuyendo el tiempo transcurrido entre la aparición de un error y su detección, lo que permitirá aumentar la seguridad de evitar efectos colaterales no deseados a la hora de realizar modificaciones y refactorizaciones.

A continuación, En la Tabla 18 se detallan las características de esta fase.

Tabla 18

Características de la Etapa de Pruebas

Elemento	Características
Pruebas Unitarias	Las pruebas unitarias sirven para verificar el correcto funcionamiento de una unidad de código, de manera que permiten tomar los correctivos si algo está mal.
Pruebas de Aceptación	Las pruebas de aceptación se realizaran para comprobar si el software cumple con las expectativas del cliente basadas en los requerimientos de las historias de usuario.

Fuente: (Joskowicz, 2008)

CAPÍTULO III

ANÁLISIS COMPARATIVO DE LIBRERÍAS DE COMPONENTES

Hoy en día se puede encontrar una variedad de Librerías de componentes o frameworks JSF en el mercado, de los cuales algunos son de licencia libre y otros de propietaria, cada uno cuenta con características específicas de acuerdo a la necesidad del usuario, es por tal motivo que la elección de un framework que cumpla con las características y requerimientos que un desarrollador de software desea es de vital importancia y por tanto su selección una tarea difícil.

3.1 Librería de componentes JSF

Según (Mann, 2005), señala que los componentes permiten al usuario interactuar con la aplicación creando extensiones dinámicas al servidor web proporcionando información desde el programa al usuario sobre el estado del programa, pueden ser expresados de diferentes formas simples como: botones, barras de desplazamiento, etiquetas, campos de texto, o componentes compuestos: como tablas, menús, árboles, Ajax, entre otros, los componentes nunca se encuentran de forma aislada, sino agrupados dentro de contenedores que son los encargados de organizarles de acuerdo a su situación, que además, en sí mismo son considerados componentes y como tales pueden ser situados dentro de otros contenedores.

A continuación, se definirán algunos componentes considerando sus características más importantes:

3.1.1 OpenFaces

OpenFaces (Figura 4) es una biblioteca de JSF de código abierto que proporciona un conjunto extendido de componentes JSF alimentados por AJAX para crear la interfaz de usuario para aplicaciones web. Se basa totalmente en el conjunto de componentes

JSF anteriormente conocido como QuipuKit. Contiene una base de código totalmente revisada de QuipuKit e introduce muchos nuevos componentes y características funcionales para la capa de presentación.



Figura 4. Logo Open Faces

Fuente: (Openfaces, 2018)

El paquete de componentes de OpenFaces incluye componentes visuales y auxiliares no visuales que brindan un mayor control sobre la usabilidad en aplicaciones web.

Existe una diversidad de componentes que ofrece Open Faces como son: Calendario, Gráfico, Confirmación, Tabla de datos, Selector de fechas, Imagen dinámica, Panel desplegable, Texto gráfico, Etiqueta de sugerencia, Capa emergente, Campo de sugerencias, Conjunto de pestañas, Panel con pestañas, Tabla de árbol, Selección de dos listas, Componentes de utilidades (Enfoque, Posición de desplazamiento, Cargar paquete).

El marco de validación OpenFaces proporciona soporte completo para la validación del lado del cliente como aporte en el marco JSF. Utiliza la lógica de validación del lado del servidor de los validadores JSF estándar para el lado del cliente y proporciona validadores y mensajes adicionales que le permiten validar una gama más amplia de datos con un mínimo esfuerzo del servidor para datos de entrada y salida. Facilita la forma de validar de manera detallada, cumpliendo con las necesidades más específicas de su aplicación y mejorando la experiencia del usuario.

Entre sus características se tiene:

- Componentes JSF avanzados impulsados por Ajax.

- Marco Ajax fácil de usar.
- Marco de validación del lado del cliente con validadores personalizados.
- Estilo enriquecido web (RIAS) y flexible basado en CSS.
- Experiencia mejorada para el usuario final.
- Proyecto de código abierto.

3.1.2 RichFaces

RichFaces, (Figura 5) es un framework JSF implementado como una biblioteca de componentes que agrega capacidad Ajax a las páginas existentes, por lo que no es necesario generar código JavaScript o reemplazar los componentes existentes con nuevos widgets Ajax.

RichFaces permite la compatibilidad con Ajax en toda la página en lugar del soporte tradicional de todo el componente. Por lo tanto, puede definir el evento en la página que invoca una solicitud Ajax y las áreas de la página que deben sincronizarse con el Árbol de componentes de JSF después de que la solicitud Ajax cambie los datos en el servidor según los eventos activados en el cliente.



Figura 5. Logo RichFaces
Fuente: (RichFaces, 2018)

El objetivo del framework RichFaces es definir (mediante etiquetas JSF) diferentes partes de una página JSF que desea actualizar con una solicitud Ajax y proporcionar algunas opciones para enviar solicitudes Ajax al servidor.

Además, la página JSF no cambia de una página JSF "normal" y no necesita escribir ningún objeto JavaScript o XMLHttpRequest, el framework lo hace automáticamente.

Según (RichFaces, 2018) atendiendo la funcionalidad provee las siguientes ventajas:

- Incluye funcionalidades Ajax en su contenedor.
- Integra funcionalidad JavaScript
- Contiene un marco de componentes visuales, con un número bastante amplio, siendo los más comunes para el desarrollo de una aplicación web rica (Rich Internet Application).
- Soporta Facelets (Plantillas Jsf).
- Crea una moderna interfaz de usuario 'look-and-feel', con el soporte de hojas de estilo css, themes o skins.
- Permite de manera rápida y fácil generar recursos binarios. El marco de recursos puede generar imágenes, sonidos, hojas de cálculo de Excel, etc. en tiempo real, de manera que permite, por ejemplo, crear imágenes con la biblioteca Java Graphics 2D y otros recursos similares.
- Crear rápidamente crear mascarar personalizadas de vistas complejas basándose en la caja de componentes. La librería UI (Interfaz de usuario) que contiene componentes para agregar características de interfaz de usuario a aplicaciones JSF. RichFaces incluye una serie de máscaras predefinidas para poner en marcha el desarrollo de aplicaciones.
- Su ámbito o alcance de envío /recepción de peticiones hace que se integre perfectamente al ciclo de vida de Jsf.

Arquitectura de RichFaces

La arquitectura de RichFaces consiste en un filtro Ajax, componentes de acción Ajax, contenedores Ajax y un motor JavaScript. Filtro Ajax: para obtener todos los beneficios de RichFaces, un desarrollador debe registrar un filtro en el archivo web.xml de la aplicación (JBoss, 2010).

La arquitectura de RichFaces consta de los siguientes elementos como se muestra a continuación, en la Figura 6.

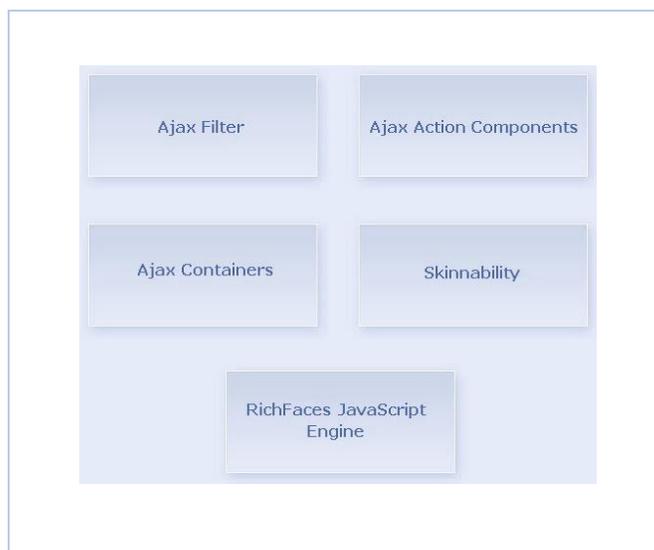


Figura 6. Arquitectura RichFaces

Fuente: (JBoss, 2010)

- **Ajax filtro.** Ajax Filter., registra el filtro en el archivo web.xml de la aplicación. El filtro reconoce varios tipos de request (envió de petición), a continuación, mediante un Diagrama de secuencia se muestra el procesamiento de una petición de envió tanto Jsf como Ajax.
- **Componentes Ajax de acción.** Los componentes de solicitud de acción son: AjaxCommandButton, AjaxCommandLink, AjaxPoll y AjaxSupport El uso de estos componentes se pueden utilizar para enviar peticiones Ajax desde el lado del cliente.
- **Contenedores Ajax.** El contenedor de Ajax es una interfaz que define un área de una página JSF. <AjaxViewRoot> y <AjaxRegion> son implementaciones de esta interfaz.

- **Skinnability.** Personaliza la apariencia en la interfaz de usuario de las aplicaciones JSF.

RichFaces le permite definir y administrar fácilmente diferentes esquemas de color y otros parámetros en la interfaz de usuario mediante el uso de parámetros de skin con nombre. Esto significa que puede acceder a los parámetros de UI desde JSP y código Java para ajustar su UI en tiempo real.

- **Motor de JavaScript.** Los motores de Java Script son exclusivos de cada navegador, en cuanto a la velocidad se refiere, interpretan instrucciones y renderizado de página al cargar, actualizan las áreas de Jsf, basadas en la petición de respuesta, se ejecutan en el lado del cliente. El motor de JavaScript proporciona una API para que los desarrolladores no necesiten crear su propia funcionalidad de JavaScript.

3.1.3 PrimeFaces

PrimeFaces, (Figura 7) es un conjunto de componentes JSF de código abierto que cuenta con varias extensiones. Estos componentes aportan una abstracción para el uso de la tecnología AJAX ya soportada en JSF 2. Es decir, el desarrollador puede estar centrado en la parte funcional sin tener que preocuparse del código JavaScript que se ejecutará en el cliente o de que partes de la pantalla serán necesarias refrescar en respuesta de un evento en la interfaz de usuario.



Figura 7. Logo PrimeFaces
Fuente: (PrimeFaces, 2018)

Según (PrimeFaces, 2018) las principales características que provee el framework son:

- Soporte de Ajax incorporado basado en API Ajax JSF estándar.
- Ligero, considerado un contenedor, configuración cero y sin dependencias requeridas.
- Se basa en librerías que utilizan el soporte de jQuery y jQuery UI para los efectos visuales y la manipulación de eventos AJAX se basa en la API del usuario.
- Simplicidad y Rendimiento.
- Skinning Framework con más de 35 temas incorporados y soporte para la herramienta de diseño de tema visual.
- Temas y diseños premium
- Amplia documentación.
- Paginación por demanda, donde rellene los datos de una lista de forma fragmentada, sólo después de la solicitud del usuario;

3.1.4 ButterFaces

Butterfaces, (Figura 8) es un marco JSF liviano y sensible que combina las ventajas de Bootstrap, jQuery y HTML 5 para desarrollar aplicaciones web rápidas, sencillas y modernas usando JSF 2



Figura 8. Logo ButterFaces
Fuente: (ButterFaces, 2018)

Según (ButterFaces, 2018) las principales características que provee el framework son:

- Posee componente de inserción que se extienden por una vista previa de lectura y tiene la etiqueta de responsabilidad y los atributos de validación asociados a Ajax.
- Basado en Bootstrap & jQuery.

- Agrega etiquetas, vistas de solo lectura, información sobre herramientas, marcador de posición (placeholders), marcador de clase de estilo de validación y otras características agradables a los componentes JSF 2 estándar.
- Diseño simple, e intuitivos, con agilidad a la hora de cargar y adaptarse a otros dispositivos.
- Posee varios elementos con estilos predefinidos fáciles de configurar: Botones, Menús desplegables, Formularios incluyendo todos sus elementos e integración jQuery para ofrecer ventanas y tooltips dinámicos.

Butterfaces es compatible con los últimos estándares de seguridad y es compatible con el ancho de banda de los exploradores en la industria.

3.1.5 OmniFaces

Omnifaces, (Figura 9) es una biblioteca de utilidad del código abierto para el marco Java Server Faces 2.2. Fue desarrollado utilizando el API de JSF, y su objetivo es hacer que JSF sea más fácil de utilizar.

Esta librería representa sencillamente una respuesta a problemas frecuentes que los desarrolladores encuentran al trabajar con JSF (p. ej. corrección de errores, obstáculos al desarrollo, características y utilidades faltantes, cuestiones comunes, etc.)



Figura 9. Logo OmniFaces

Fuente: (OmniFaces, 2018)

Según (OmniFaces, 2018) las principales características que provee el framework son:

- Conversión automática de objetos modelo en menús desplegable y otros componentes seleccionados
- Validadores de campo múltiple, para validaciones.
- Kit de renderizado HTML5 que agrega soporte para varios atributos específicos de HTML5 a los componentes UIForm y UIInput
- Manipulador de excepciones Ajax completo
- Árbol con marcado personalizado completo por nivel (utilizable para varios casos de uso recursivo)
- Colecciones de funciones EL para manejar matrices, conversiones, fechas y cadenas.
- Filtro de compresión GZIP para respuestas HTTP
- Incluir servlets y páginas JSP en Facelets.

3.1.6 BootsFaces

BootsFaces, (Figura 10) es una biblioteca de utilidad del código abierto JSF potente y liviana basado en Bootstrap 3 y jQuery UI que le permite desarrollar aplicaciones empresariales de Front-end de forma rápida y sencilla.



Figura 10. Logo BootsFaces

Fuente: (BootsFaces, 2018)

La misión del proyecto es llevar el diseño receptivo de Bootstrap al ambiente de JSF. (BootsFaces, 2018).

El primer objetivo de BootsFaces fue ayudar al desarrollo de JSF mediante la creación de una interfaz de usuario bien diseñada y receptiva, sin forzarlos a sumergirse en los detalles forzados de CSS y HTML5. Además, BootsFaces tiene como

objetivo salvarlo de la tarea tediosa y propensa a errores de escribir una gran cantidad de lenguaje de marcado (HTML) y código de JavaScript.

BootsFaces tiene la capacidad de integrarse con los marcos JSF existentes antes que convertirse en una alternativa exclusiva orientada al Diseño. Hoy en día, BootsFaces puede integrarse con AngularFaces, PrimeFaces y OmniFaces.

Considerando las características descritas anteriormente se escogió a Bootfaces como el más recomendado para el diseño de la capa de presentación para utilizar en el desarrollo de este proyecto.

3.2 Estudio general de las librerías de componentes

En la actualidad existe una gran variedad de librerías de componentes JSF, como son OpenFaces, RichFaces, PrimeFaces, ButterFaces, OmniFaces y BootsFaces. (Notas&Codigo, 2017)

Para realizar un análisis comparativo se han seleccionado bibliotecas de acuerdo al nivel de madurez y a la popularidad que han presentado.

En este estudio, la calidad de productos de software fue evaluada a través de las siguientes características: Documentación, Configuración, Showcase, Soporte, Nivel de Popularidad, Infraestructura.

A continuación, En la Tabla 19 se presentan los criterios de selección que se compararan para realizar el respectivo análisis.

Tabla 19*Criterios Comparativos de Librerías de Componentes JSF*

Criterio	Requisitos	Detalle
Documentación	<ol style="list-style-type: none"> 1. Documentación Oficial 2. Configuración 3. Showcase 4. Soporte 5. Popularidad 	Mediante este criterio se podrá analizar la disponibilidad de información existente en la web en cuanto a instalación, uso y características para preveer el grado de dificultad de la herramienta.
Infraestructura	<ol style="list-style-type: none"> 1. Navegadores 2. Servidores 3. Soporte de Ajax 4. Controles UI 	Este criterio permite analizar el soporte ofrecido por las bibliotecas en relación a la infra estructura
Soporte a Plataformas móviles	<ol style="list-style-type: none"> 1. Soporte a Plataformas móviles 	En este criterio se analiza la gestión hacia herramientas móviles.

3.3 Descripción de Requisitos**Tabla 20***Documentación*

Requisito	Descripción
Documentación Oficial	Disponibilidad de Documentación Oficial, guías de usuario, manuales, cursos.
Configuración	configuraciones del descriptor de despliegue en el archivo web.xml;
Showcase	Son las librerías de utilidades cuentan con información integrada para el uso de sus componentes.
Soporte	Aumentar características, mejorar o reparar
Popularidad	Índices de aceptación (Ranking) de la Librería de Componentes,

Tabla 21*Infraestructura*

Requisito	Descripción
Navegadores	Disponibilidad de Documentación Oficial, guías de usuario, manuales, cursos.
Servidores	configuraciones del descriptor de despliegue en el archivo web.xml;
Soporte Ajax	Ofrece soporte para la interacción asíncrona cliente-servidor. Este es el proceso de intercambio de datos con el servidor y actualizar partes de una página web sin volver a cargar toda la página.

Tabla 22*Plataformas Móviles*

Requisito	Descripción
Soporte a Plataformas móviles	Existencia de plataformas móviles, mejorar características de usabilidad., capacidad de funcionalidad.

3.4 Comparación

A continuación, se realiza una evaluación comparativa de los criterios en base a los requisitos seleccionados.

Criterio: Documentación**3.4.1 Documentación Oficial**

Para realizar el análisis comparativo se ha seleccionado las librerías de componentes OpenFaces, RichFaces, PrimeFaces, ButterFaces, OmniFaces y BootFaces, considerando el nivel de madurez y popularidad en la creación de interfaces enriquecidas (RÍAS). (Notas&Codigo, 2017)

A continuación, en la Tabla 23, se presenta el análisis comparativo de la Documentación Oficial, en base a sus requisitos.

Tabla 23

Análisis comparativo de la Documentación Oficial

Requisitos	Open Faces	RichFaces	PrimeFaces	ButterFaces	OmniFaces	BootFaces
Documentación Completa	No	No	No	No	No	No
Manual de la Api	Si	Si	Si	Si	Si	Si
Guía de Usuario	Si	Si	Si	Si	Si	Si
Guía rápida	Si	Si	Si	Si	Si	Si
Cursos	No	No	Si	No	No	Si

3.4.2 Configuración

Ninguno de los frameworks de presentación requiere algún tipo de instalación, solo se debe descargar los archivos indicados en el sitio de la página oficial de la Documentación y / o manual de usuario. Adicionalmente se necesita configurar el descriptor de despliegue web.xml, para gestionar recursos de las librerías de componentes.

3.4.3 Showcase

Todas las opciones están organizadas con una amplia demostración de sus utilidades y facilidad de uso. Permite encontrar el modelo y el código fuente de un determinado componente de interfaz de usuario para Java Server Faces.

3.4.4 Soporte

Los frameworks ofrecen en su sitio nuevas funcionalidades, a través de presupuestos estimados por esfuerzo y complejidad de la solución. Cada sitio oficial ofrece soporte profesional con servicios de consultoría, y desarrollo de componentes personalizables, para versiones comerciales.

3.4.5 Popularidad

Para analizar los índices de popularidad presentes durante el año 2018 se utilizó Herramientas de medición de la web encargada de medir las variantes de popularidad de rangos de datos que presenta cada framework.

La fórmula para calcular la cantidad de visitantes mensuales es la siguiente, tal como se muestra a continuación:

$$\text{Visitantes mensuales} = 104\,943\,144\,672 \times \text{Índice de Alexa}^{-1'008}$$

Otra forma de obtener el dato es usar el siguiente formulario como se muestra en la Figura 11.



The image shows a web form for calculating monthly visitors based on the Alexa Rank. The form is enclosed in a blue border. Inside, there is a rounded rectangle with an orange border containing three elements: an input field for 'Alexa Rank' with the value '792862', an output field for 'Visitantes mensuales' with the value '118,731', and a 'Calcular' button.

Figura 11. Formulario Alexa para calcular visitantes mensuales
Fuente: (norfipc.com, 2017)

A continuación, mediante gráficos generados del sitio <https://siterankdata.com/> se analiza los rangos de popularidad que presenta cada framework.

3.4.5.1 OpenFaces

De acuerdo a los datos investigados (Figura12) en el sitio web de OpenFaces se ha obtenido los siguientes resultados (SiteRankData, 2019)

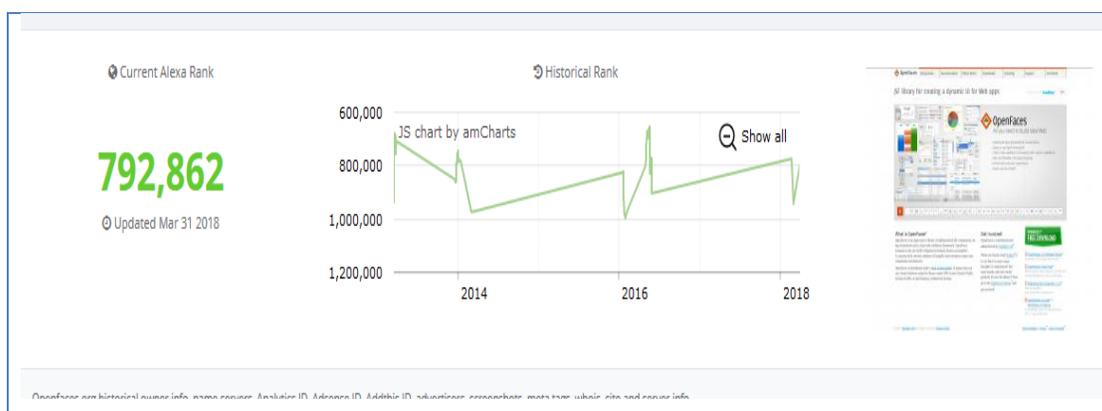


Figura 12. Visitas acumuladas al sitio OpenFaces
Fuente: (SiteRankData, 2019)

Utilizando la Formula Alexa para calcular las visitas mensuales se obtiene los siguientes resultados, como se muestra en la Tabla 24.

Tabla 24

Estimación de Visitas OpenFaces

Visitas Diarias	Visitas Mensuales	Visitas Anuales
3,830	118,731	1,424,772

Fuente: (SiteRankData, 2019)

A partir de los resultados obtenidos en las visitas diarias, se puede establecer las visitas mensuales y las visitas anuales.

A continuación, en la Figura 13 se muestra el creciente número de visitas producidas en el sitio oficial de OpenFaces.

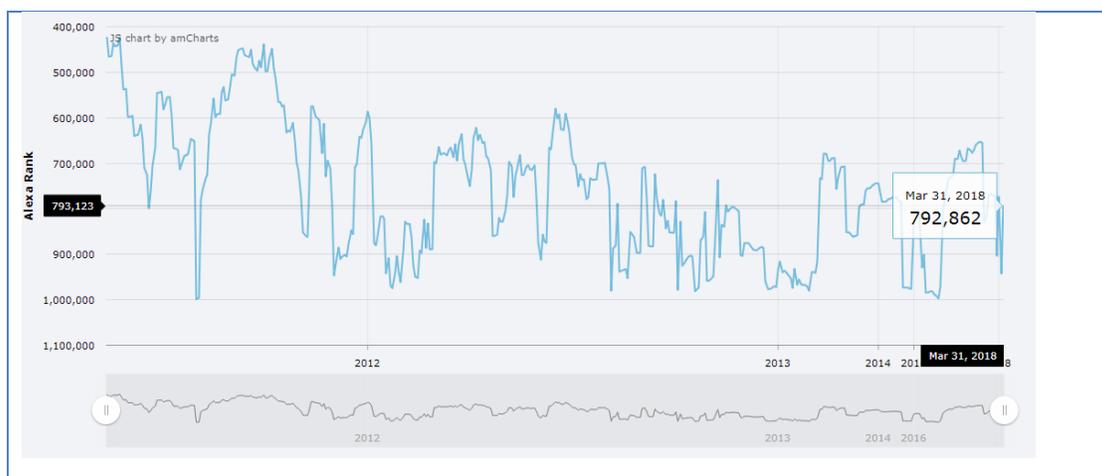


Figura 13. Datos Obtenidos de los visitantes a OpenFaces
Fuente: (SiteRankData, 2019)

3.4.5.2 RichFaces

De acuerdo a los datos investigados (Figura14) en el sitio web de RichFaces se ha obtenido los siguientes resultados (SiteRankData, 2019).



Figura 14. Visitas acumuladas al sitio RichFaces
Fuente: (SiteRankData, 2019)

A partir de las visitas diarias generadas en la página oficial de RichFaces, utilizando la Formula Alexa se puede calcular las visitas mensuales.

A continuación, en la Tabla 25, mediante la fórmula de probabilidades Alexa se muestran los resultados obtenidos.

Tabla 25

Datos Obtenidos de visitas RichFaces

Visitas Diarias	Visitas Mensuales	Visitas Anuales
5,880	182,265	2,187,180

Fuente: (SiteRankData, 2019)

A partir de los resultados obtenidos en las visitas mensuales, se puede establecer las visitas diarias y las visitas anuales.

A continuación, en la Figura 15 se muestra el creciente número de visitas producidas en el sitio oficial de RichFaces.

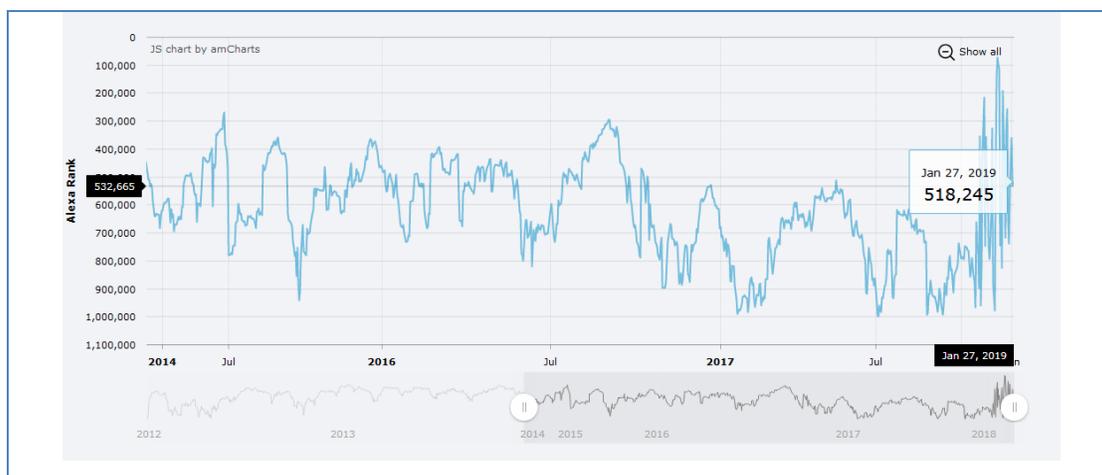


Figura 15. Datos Obtenidos de los visitantes a RichFaces
Fuente: (SiteRankData, 2019)

3.4.5.3 PrimeFaces

De acuerdo a los datos investigados (Figura16) en el sitio web de PrimeFaces se ha obtenido los siguientes resultados (SiteRankData, 2019)



Figura 16. Visitas acumuladas al sitio PrimeFaces
Fuente: (SiteRankData, 2019)

Utilizando la Formula Alexa para calcular las visitas mensuales se obtiene los siguientes resultados, como se muestra en la Tabla 26.

Tabla 26

Datos Obtenidos de visitas a PrimeFaces

Visitas Diarias	Visitas Mensuales	Visitas Anuales
87,146	2,701,513	32,418,156

Fuente: (SiteRankData, 2019)

A partir de los resultados obtenidos en las visitas mensuales, se puede establecer las visitas diarias y las visitas anuales.

A continuación, en la Figura 17 se muestra el creciente número de visitas producidas en el sitio oficial de PrimeFaces

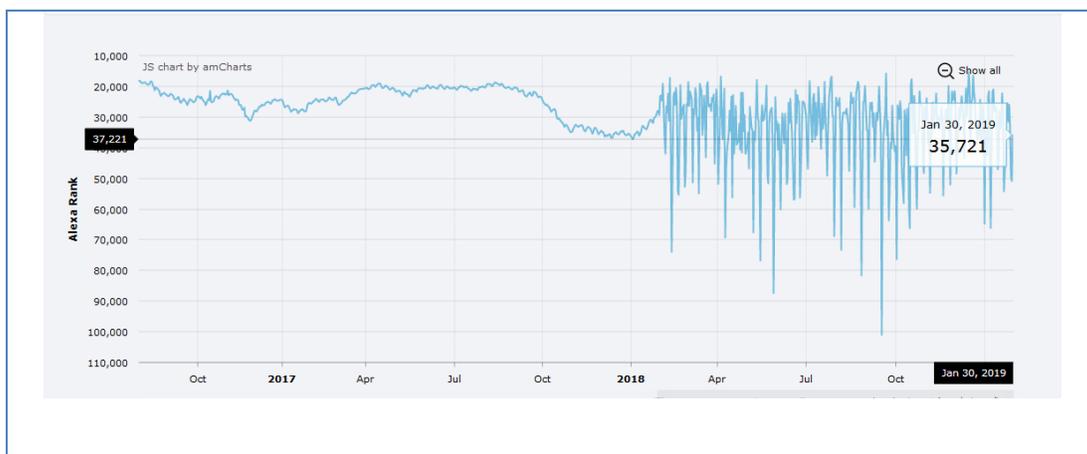


Figura 17. Datos Obtenidos de los visitantes a PrimeFaces
Fuente: (SiteRankData, 2019)

3.4.5.4 ButterFaces

De acuerdo a los datos investigados (Figura18) en el sitio web de ButterFaces se ha obtenido los siguientes resultados (SiteRankData, 2018)

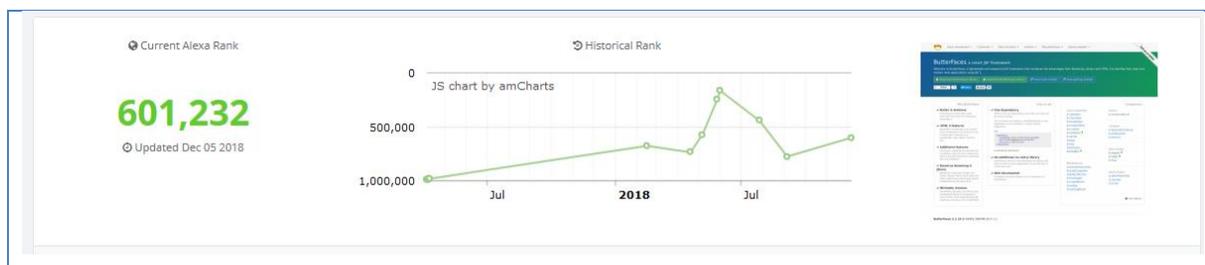


Figura 18. Visitas al sitio ButterFaces
Fuente: (SiteRankData, 2018)

Utilizando la Formula Alexa para calcular las visitas mensuales se obtiene los siguientes resultados, como se muestra en la Tabla 27.

Tabla 27

Datos Obtenidos de visitas ButterFaces

Visitas Diarias	Visitas Mensuales	Visitas Anuales
5,062	156,920	1,883,040

Fuente: (SiteRankData, 2018)

A partir de los resultados obtenidos en las visitas mensuales, se puede establecer las visitas diarias y las visitas anuales.

A continuación, en la Figura 19 se muestra el creciente número de visitas producidas en el sitio oficial de ButterFaces

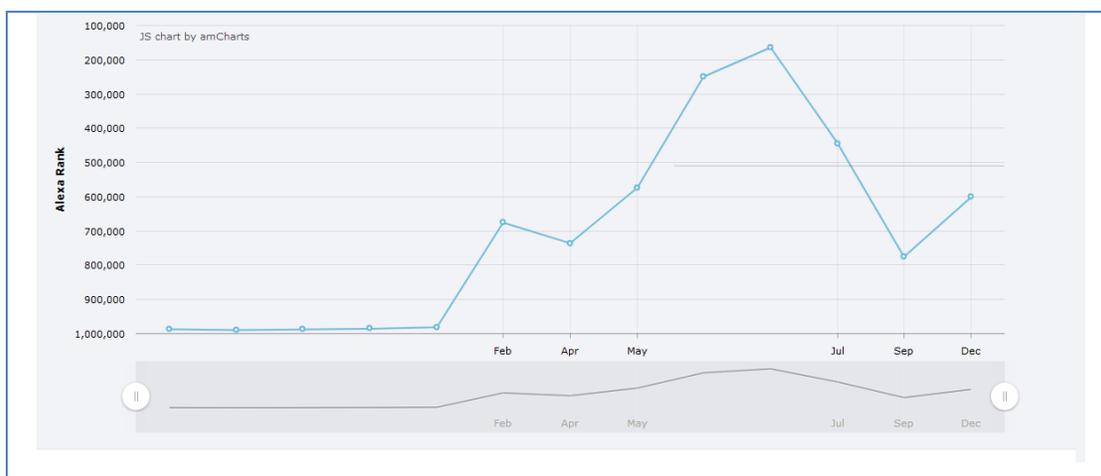


Figura 19. Datos Obtenidos de los visitantes a ButterFaces
Fuente: (SiteRankData, 2018)

3.4.5.5 OmniFaces

De acuerdo a los datos investigados (Figura 20) en el sitio web de OmniFaces se ha obtenido los siguientes resultados (SiteRankData, 2019)

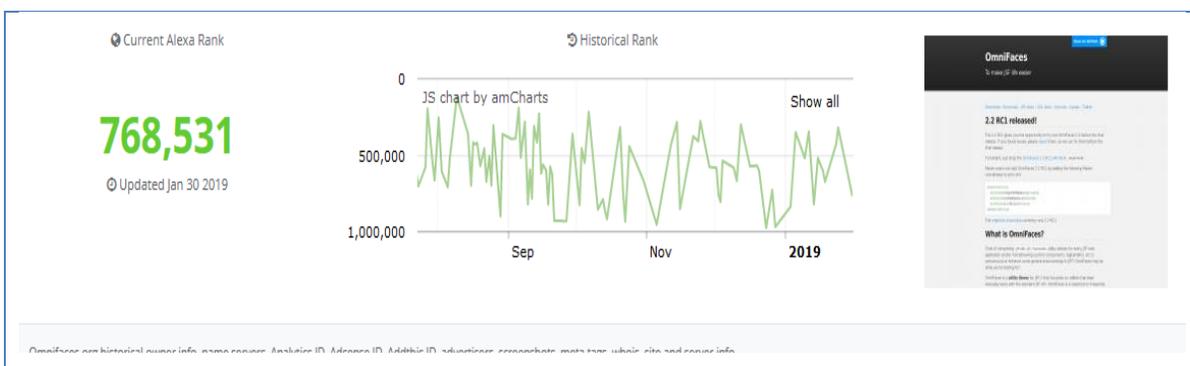


Figura 20. Cantidad de Visitas al sitio OmniFaces
Fuente: (SiteRankData, 2019)

Utilizando la Formula Alexa para calcular las visitas mensuales se obtiene los siguientes resultados, como se muestra en la Tabla 28.

Tabla 28

Datos Obtenidos de visitas OmniFaces

Visitas Diarias	Visitas Mensuales	Visitas Anuales
3,952	122,520	1,470,240

Fuente: (SiteRankData, 2019)

A partir de los resultados obtenidos en las visitas mensuales, se puede establecer las visitas diarias y las visitas anuales.

A continuación, en la Figura 21 se muestra el creciente número de visitas producidas en el sitio oficial de OmniFaces.

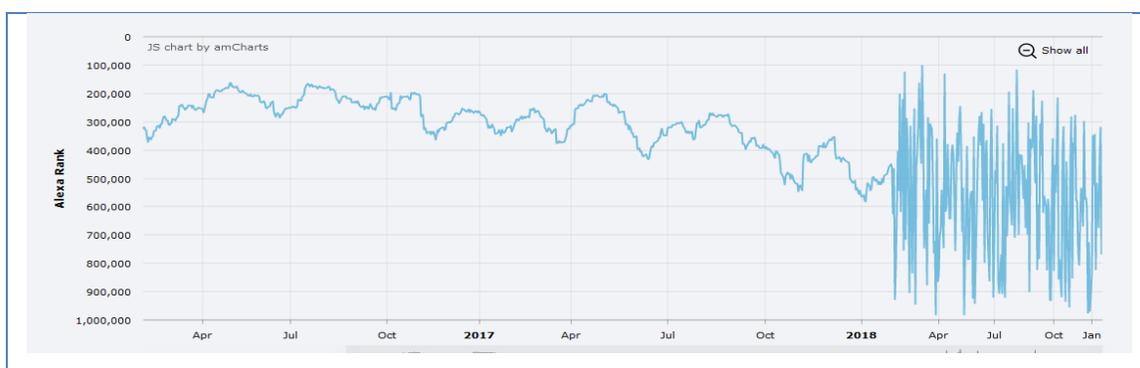


Figura 21. Datos Obtenidos de los visitantes a ButterFaces

Fuente: (SiteRankData, 2019)

3.4.5.6 BootsFaces

De acuerdo a los datos investigados (Figura 22) en el sitio web BootsFaces se ha obtenido los siguientes resultados (SiteRankData, 2019)



Figura 22. Cantidad de Visitas al sitio BootsFaces
Fuente: (SiteRankData, 2019)

Utilizando la Formula Alexa para calcular las visitas mensuales se obtiene los siguientes resultados, como se muestra en la Tabla 29.

Tabla 29

Datos Obtenidos de visitas BootsFaces

Visitas Diarias	Visitas Mensuales	Visitas Anuales
4,485	139,027	1,668,324

Fuente: (SiteRankData, 2019)

A partir de los resultados obtenidos en las visitas mensuales, se puede establecer las visitas diarias y las visitas anuales.

A continuación, en la Figura 23 se muestra el creciente número de visitas producidas en el sitio oficial de BootsFaces.

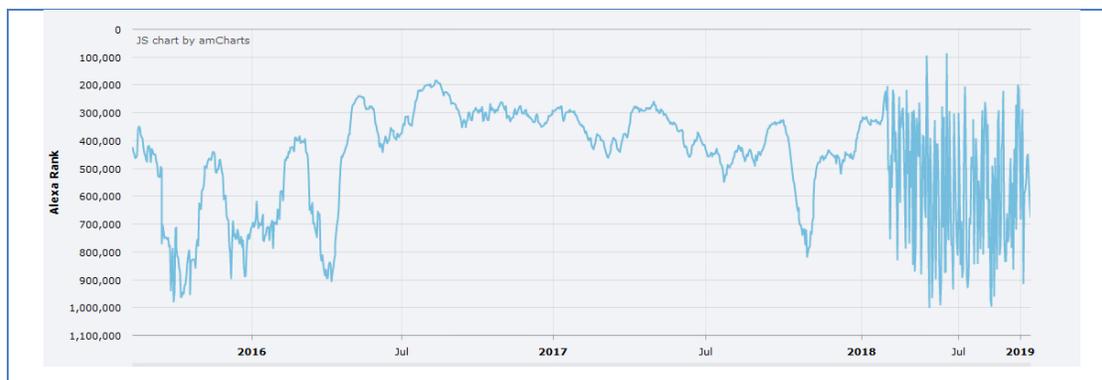


Figura 23. Datos Obtenidos de los visitantes a BootsFaces
Fuente: (SiteRankData, 2019)

3.4.6 Criterio: Infraestructura

3.4.6.1 Navegadores

Tabla 30

Navegadores soportados por frameworks JSF

Navegador	Open Faces	RichFaces	PrimeFaces	ButterFaces	OmniFaces	BootsFaces
Firefox	V3	V3+	V53	V52+	V52+	V52+
Safari	V5	V7+	V10.1	V12	V12	V12+
Opera	V10.0	V10.0+	V56	V50	V50	V50+
Google Chrome	Todas las versiones	V1+	V58	V58+	V58+	V58+
Internet Explorer	IE 7	V8+	IE 8-9-10-11,12	IE 7, IE 10	IE 7, IE 10	IE 8-9-10-11-12

3.4.6.2 Servidores

Tabla 31

Servidores soportados Frameworks JSF

Servidor	Open Faces	RichFaces	PrimeFaces	ButterFaces	OmniFaces	BootFaces
GlassFish (Oracle)	V 4.1.1+	V 4.1.1+	V 4.1.1+	V 4.1.1+	V 4.1.1+	V 4.1.1+
Tomcat (Apache)	V 8.5.31	V 8.5.31	V 8.5.31	V 8.5.31	V 8.5.31	V 8.5.31
Wildfly (RedHat)	V9+	V9+	V9+	V9+	V9+	V9+

3.4.6.3 Soporte Ajax

Tabla 32

Soporte Ajax en Frameworks JSF

Framework	Soporte de Ajax
OpenFaces	Proporciona la capacidad de invocar acciones al servidor con Ajax.
RichFaces	Los componentes de RichFaces están construidos con soporte Ajax. Utiliza



	la implementación A4JSF
PrimeFaces	Proporciona soporte Ajax incorporado
ButterFaces	Cada componente de entrada se amplía mediante una vista de solo lectura y tiene una etiqueta receptiva y atributos de información sobre herramientas que admiten Ajax y validación
OmniFaces	Los programas JSF frecuentemente usan widgets JavaScript nativos que son pequeños programas que brindan fácil acceso a funciones frecuentemente usadas y proveen de información visual. Al usar OmniFaces, es fácil agregar AJAX a estos widgets.
BootsFaces	Los programas JSF frecuentemente usan widgets JavaScript nativos. Al usar BootsFaces, es fácil agregar AJAX a estos widgets.

Criterio: Plataformas Móviles

Soporte a Plataformas móviles

Tabla 33

Soporte de Frameworks JSF a plataformas móviles

Requisito	OpenFaces	RichFaces	PrimeFaces	ButterFaces	OmniFaces	BootsFaces
Soporte a Plataformas móviles	Permite mostrar la distancia del dispositivo al GPS. HTML5 permite obtener las coordenadas del móvil.	Técnicas compatibles HTML5 / CSS3 y Javascript.	Compatible con jQuery Mobile, compatible con HTML5 optimizado al tacto que brinda soporte para varias plataformas	Compatible con JQuery, twitters bootstrap y html5	Soporte Bootstrap y jQuery UI, html5 Para brindar el mejor diseño en el desarrollo aplicaciones móviles.	Soporte Bootstrap y jQuery UI, html5.

3.5 Construcción del Modelo de Calidad

Para realizar el Modelo de calidad se utilizará los lineamientos que plantea la Norma ISO 25000 respecto a guiar el desarrollo de los productos de software con la especificación y evaluación de requisitos de calidad con el fin de evaluar la comparación de librerías de componentes JSF bajo la plataforma JEE 7.

La matriz para la realización del estudio comparativo tiene como objetivo evaluar las características generales y los atributos específicos de cada uno de los frameworks de librerías de componentes JSF.

3.5.1 Criterios de decisión para la evaluación

En esta actividad se definen los criterios para las diferentes características evaluadas a partir de las subcaracterísticas y métricas de calidad, los cuales se han establecido de acuerdo al Nivel de importancia como: A (Alto), M (Medio), B (Bajo), N/A No Aplica.

A continuación, en la Tabla 34 se detallan los Indicadores que permiten realizar la valoración de la calidad del producto de software de forma general.

Tabla 34

Criterios de Decisión para la evaluación de Calidad

Escala de medición	Niveles de importancia	Grado de Satisfacción	Ponderación	Criterio
Alto	A	Alto	7-10	Se debe evaluar
Medio	M	Medio	4 - 6	Se puede evaluar/No se puede evaluar
Bajo	B	Bajo	2-3	No se evalúa
No Aplica	N/A	No Aplica	0 -1	No aplica

3.5.2 Matriz de Evaluación de calidad

La Matriz hace referencia a los indicadores en una perspectiva general de acuerdo a las características y subcaracterísticas de calidad para la construcción del nuevo modelo, coherente a un estándar de Calidad.

El procedimiento para realizar la matriz de evaluación de calidad consta de las siguientes partes como se detalla a continuación en la Tabla 35:

- a) Se define la ponderación de las características, el nivel de importancia el grado de satisfacción.

Tabla 35 *Ponderación asignada para la calidad interna y externa*

Características	Nivel de importancia	Grado de Satisfacción	Ponderación
Adecuación Funcional	A	Alto	25%
Fiabilidad	M	Medio	15%
Rendimiento	M	Medio	5%
Usabilidad	M	Medio	10%
Compatibilidad	M	Medio	15%
Portabilidad	A	Alto	25%
Mantenibilidad	M	Medio	5%
Porcentaje de Ponderación			100%

Como se puede apreciar en la tabla la fiabilidad y la compatibilidad de peso menor considerable al 15%, debido a que no se perciben, pero son importantes indicadores de calidad. Por último, el Rendimiento, la mantenibilidad y la Usabilidad son los que indican un menor peso con el 5% y 10% respectivamente, estos datos son probatorios puesto que en cuanto al Servidor de Aplicaciones Wildfly aumenta la capacidad de Rendimiento en Java.

3.5.3 Evaluación de los Frameworks

La evaluación de los frameworks JSF 2.2 seleccionados se realizó tomando en cuenta la información disponible en Internet que incluye foros de usuarios, publicaciones y ambiente de pruebas, realizando instalaciones de cada una de las librerías de componentes.

- b)** Posteriormente se describe cada una de las características y subcaracterísticas, como se muestra en la (Tabla 36) especificando la ponderación respectiva del Framework las características internas, externas y de uso según el criterio de medición de calidad del componente, al igual que las métricas a evaluar lo cual ayudara a definir el comportamiento y efectividad del aplicativo.

3.5.4 Aplicación de los criterios de decisión para las métricas

Los criterios de decisión para las métricas seleccionadas sobre los valores obtenidos en la medición del producto han sido desarrollados de acuerdo a las consideraciones descritas en base a las características, subcaracterísticas y métricas a evaluar de acuerdo al nivel de importancia

Tabla 36

Aplicación de Criterios de Decisión en base a una matriz de calidad

ISO/ IEC 25000													
Funcionalidad: Capacidad del software de proveer los servicios necesarios para cumplir con los requisitos funcionales.													
Funcionalidad	Característica	Su característica	Métrica	Nivel de Importancia	Motivo de Selección	Componentes	Criterio del Componente	OpenFaces	RichFaces	PrimeFaces	ButterFaces	OmniFaces	BootsFaces
	Adecuación Funcional						Ajax Core	más de 10=2 entre 5 y 9=1 menos de 5=0	2	2	2	2	2
						Button	más de 10=2 entre 5 y 9=1	0	0	1	0	0	1

Continúa



Completi tud funcional	Capacidad de contener todas las funciones especificadas por el usuario /Completi tud de la implementa ción fun cional.	A	Evaluar las Funcionalida des de los componentes.		menos de 5=0							
				Carousel	más de 10=2 entre 5 y 9=1 menos de 5=0	0	0	1	0	0	1	
				Charts	más de 10=2 entre 5 y 9=1 menos de 5=0	0	0	2	0	0	2	
				Data(tables/ grids)	más de 10=2 entre 5 y 9=1 menos de 5=0	1	1	2	0	0	2	
				Drag&Drop	más de 10=2 entre 5 y 9=1 menos de 5=0	0	0	1	0	1	1	
				Dialog	más de 10=2 entre 5 y 9=1 menos de 5=0	0	0	0	0	0	0	
				File	más de 10=2 entre 5 y 9=1	0	0	1	0	0	1	

Continúa



				neous	más de 10=2 entre 5 y 9=1 menos de 5=0						
				Multimedia	más de 10=2 entre 5 y 9=1 menos de 5=0	0	0	1	0	0	2
				Overlay	más de 10=2 entre 5 y 9=1 menos de 5=0	0	1	1	1	0	1
				Panel	más de 10=2 entre 5 y 9=1 menos de 5=0	1	0	2	0	0	2
				Total Completitud Funcional		8	9	22	7	6	24
	Capacidad de obtener los resultados esperados /Exactitud.	A	Evaluar el Comparativo de Librerías de componentes para cumplir los requerimientos	Button	Si = 1 No = 0	1	1	1	1	1	1
				Carousel	Si = 1 No = 0	0	0	1	0	0	1

Continúa



			funcionales.								
	Corrección funcional			Charts	Si = 1 No = 0	1	1	1	0	0	1
				Data(tables/ grids)	Si = 1 No = 0	1	1	1	0	0	1
				Drag & Drop	Si = 1 No = 0	0	0	1	0	1	1
				Dialog	Si = 1 No = 0	0	0	1	0	0	1
				File	Si = 1 No = 0	0	0	1	0	0	1
				Icon	Si = 1 No = 0	0	0	1	0	0	1
				Input	Si = 1 No = 0	1	1	1	1	1	1
				Link	Si = 1 No = 0	0	0	1	0	0	1
				Menú	Si = 1 No = 0	1	1	1	0	0	1
				Message	Si = 1 No = 0	1	1	1	1	1	1
				Miscellaneous	Si = 1 No = 0	0	1	1	1	0	1

Continúa



				Multimedia	Si = 1 No = 0	0	0	1	0	0	1
				Overlay	Si = 1 No = 0	0	1	1	1	0	1
				Panel	Si = 1 No = 0	1	0	1	0	0	1
				Total Corrección Funcional		7	8	15	5	4	16
Pertinencia Funcional	Resultado de Pruebas	A	Capacidad de Responder efectivamente a pruebas.	Button	Si = 1 No = 0	0	1	1	0	0	1
				Input	Si = 1 No = 0	1	1	1	1	1	1
				Message	Si = 1 No = 0	0	1	1	0	0	1
				Resultado de pruebas aleatorias individuales		1	3	3	3	1	3
				Button	Si = 1 No = 0	1	1	1	0	0	1
				Input	Si = 1	1	1	1	1	1	1

Continúa



						No = 0							
						Message	Si = 1 No = 0	0	1	1	1	1	1
						Resultado de pruebas aleatorias Grupales		2	3	3	2	2	3
						Total de Adecuación Funcional		18	23	43	17	13	46

Fiabilidad: Capacidad de un sistema o componente para desempeñar las funciones especificadas, cuando se usa bajo unas condiciones y periodo de tiempo determinados

Fiabilidad	Característica	Subcaracterística	Métrica	Nivel de Importancia	Motivo de Selección	Detalle de la Métrica	Criterio de la Métrica	OpenFaces	RichFaces	PrimeFaces	ButterFaces	OmniFaces	BootsFaces
------------	----------------	-------------------	---------	----------------------	---------------------	-----------------------	------------------------	-----------	-----------	------------	-------------	-----------	------------

Fiabilidad	Madurez	Tiempo en el Mercado	M	Tiempo de existencia del Componente	Más de 4 años=1 Menos de 3 años=2	1	1	1	2	2	2
		Base de Conocimiento de software		Mantiene una Base de Conocimiento de software.	Si=1 No=0	1	1	1	1	1	1
		Actualizaciones Disponibles		Frecuencia de las Actualizaciones	Periodo>=4 años	2	2	3	2	2	3
		Capacidad de Recuperación de errores.		De acuerdo al componente	Si=1 No=0	0	1	1	0	0	1
		Total Madurez				4	5	6	5	5	7
Disponibilidad	Disponibilidad	Disponibilidad de componentes en Aplicación.	A	Orientado al Componente	90% - 100%=2 50%-89%=1 Menos del 50%= 0	0	0	2	0	0	2
		Disponibilidad de documentación.		Ayuda en línea, tutoriales, videos	Si =1 No=0	1	1	1	1	1	1
				En el IDE	Si =1 No=0	1	1	1	1	1	1

Continúa



				Total Disponibilidad			2	2	4	2	2	4
	Tolerancia a fallos	Capacidad de mantener la funcionalidad en caso de falla de un componente/ prestaciones en casos de fallos.	M	Evaluar la prestación en caso de fallos	Funcionalidad en caso de falla de un componente	Si =1 No=0	1	1	1	1	1	1
				Total Tolerancia a fallos			1	1	1	1	1	1
	Recuperación	Capacidad de restablecer un determinado nivel de prestaciones.	M	Evaluar como la herramienta mantiene la capacidad de recuperación en fallas de interrupción	Tiempo de Respuesta por falla de interrupción de un componente	más de 1h=0 entre 1/2h - 1/4h=1 menos de 1/4h=2	1	1	1	1	1	1
		Tiempo de recuperación de diferente índole a fallo en componente.		Tiempo de recuperación de un componente En depurar fallos de daños	Tiempo de Respuesta por falla de un componente	más de 1h=0 entre 1/2h - 1/4h=1 menos de 1/4h=2	1	1	2	0	0	2

					Total Recuperación			2	2	3	1	1	3
						Total de Pruebas de Fiabilidad		9	10	14	9	9	15

Rendimiento: Esta característica representa el desempeño relativo a la cantidad de recursos utilizados bajo determinadas condiciones.

Rendimiento		Característica	Subcaracterística	Métrica	Nivel de Importancia	Motivo de Selección	Detalle de la Métrica	Criterio de la Métrica	OpenFaces	RichFaces	PrimeFaces	ButterFaces	OmniFaces	BootsFaces
Rendimiento	Comportamiento temporal	Tiempo de respuesta en realizar procesos en determinadas condiciones.	M	Evaluar el tiempo en finalizar una tarea determinada	Tiempo de renderización de un componente	más de 2s= 0 Entre 1s y 2s= 1 Menos de 1s=2	1	1	2	0	0	0	2	
					Tiempo de renderización de un componente con rol de usuario	más de 2s= 0 Entre 1s y 2s= 1 Menos de 1s=2	1	2	1	0	0	2		

Continúa 

				Total Comportamiento temporal			2	3	3	0	0	4
Utilización de recursos	Utilización de frameworks instalados en el IDE	M	Conocer el uso de recursos que utilizan el IDE.	Capacidad de utilizar otros frameworks instalados en el IDE	Si=1 No=0		1	1	1	1	1	1
	Utilización de herramientas externas al IDE.			Capacidad de utilizar herramientas externas al IDE	Si=1 No=0		1	1	1	1	1	1
				Total Utilización de recursos			2	2	2	2	2	2
Capacidad	Sistema de transmisión de ancho de banda	M	Capacidad de ancho de banda durante la utilización del componente	Cantidad de CPU, memoria y ancho de banda.	Aplica Si=1 No=0		0	0	1	1	1	1
				Total Capacidad			0	0	1	1	1	1
				Total Rendimiento			4	5	6	3	3	7

Continúa



Usabilidad: Capacidad de un sistema o componente para desempeñar las funciones especificadas, cuando se usa bajo unas condiciones y periodo de tiempo determinados.

Característica	Subcaracterística	Métrica	Nivel de Importancia	Motivo de Selección	Detalle de la Métrica	Criterio de la Métrica	OpenFaces	RichFaces	PrimeFaces	ButterFaces	OmniFaces	BootsFaces
Usabilidad	Capacidad de reconocer su adecuación	Capacidad de permitir su uso y funcionalidad por parte del usuario / Capacidad de demostración.	M	Conocer si el software es adecuado para las necesidades del usuario	Los componentes funcionan correctamente	Aplica Si =1 No=0	1	1	1	1	1	1
			Total Capacidad de reconocer su adecuación				1	1	1	1	1	1
	Capacidad de aprendizaje	Eventos de Capacitación Cursos específicos de la herramienta	M	Es factible el uso de ayudas del uso de la herramienta hacia el	Se tiene planificado eventos de capacitación	Si=1 No=0	0	0	0	0	0	0

			usuario	Existe Cursos Específicos de la herramienta	Si=1 No=0	0	1	1	0	0	1
	Cursos on-line			Se tiene acceso a cursos on-line	Si=1 No=0	1	1	1	1	1	1
	Video tutoriales			Existe videos tutoriales de los componentes	Si=1 No=0	1	1	1	1	1	1
			Total Capacidad de Aprendizaje			2	3	3	2	2	3
Capacidad para ser usado	Documentación Clara para ser operado y controlado /Claridad de mensajes.	M	Existe Documentación para facilitar las instalaciones del framework. La personalización y apariencia son propias de cada herramienta	Herramienta intuitiva	Aplica Si=1 No=0	1	1	1	1	1	1
	Posibilidad de personalización			Opciones de personalización	Aplica Si=1 No=0	1	1	1	1	1	1
			Total Capacidad para ser usado			2	2	2	2	2	2
Protección contra errores del usuario	Manual de Usuario	M	Usar el manual de Usuario ayuda a prevenir errores involuntarios por parte de	Disponibilidad de la Información	Si=1 No=0	1	1	1	1	1	1
	Prevención del uso incorrecto de la herramienta.			Herramienta intuitiva	Si=1 No=0	1	1	1	1	1	1

Continúa



			los usuarios.								
			Total de Protección contra errores del usuario			2	2	2	2	2	2
Estética de la Interfaz del usuario	Capacidad de la interfaz de facilitar su comprensión y uso.	M	Facilidad en la comprensión y uso	Opciones de personalización	Aplica Si=1 No=0	1	1	1	1	1	1
	Interfaces visualmente atractivas		Armonía de colores, consistencia en patrones de diseño.	Estética de la Interfaz mejorada	Aplica Si=1 No=0	1	0	1	1	1	1
			Total Estética de la Interfaz del usuario			2	1	2	2	2	2
Accesibilidad	Ayuda en línea	A	Se provee facilidad en línea, soporte para desarrolladores. Se provee facilidad en línea, soporte para personas con capacidades especiales y cognitivas	Existe Ayuda en línea	Aplica Si=1 No=0	1	1	1	1	1	1
	Comunidad de soporte			Existe Comunidad de soporte	Aplica Si=1 No=0	1	1	1	1	1	1
	Accesibilidad física de personas con técnicas HCI.			Existe ayuda para personas con capacidades especiales y cognitivas	Aplica Si=1 No=0	0	0	0	0	0	0
			Total Accesibilidad			2	2	2	2	2	2
				Total Usabilidad		11	11	12	11	11	12

Continúa



Compatibilidad: Capacidad de un producto de software para intercambiar información con otros sistemas o componentes y/o llevar a cabo sus funciones requeridas cuando comparten el mismo entorno hardware o software.													
Compatibilidad	Característica	Subcaracterística	Métrica	Nivel de Importancia	Motivo de Selección	Detalle de la Métrica	Criterio de la Métrica	OpenFaces	RichFaces	PrimeFaces	ButterFaces	OmniFaces	BootsFaces
Compatibilidad	Coexistencia	Coexistencia disponible.	Coexistencia disponible.	M	Facilidad de adaptación con sistemas o componentes externas	Interacción con otros componentes	Aplica Si=1 No=0	0	0	1	0	0	1
		Generadores de Reportes	Generadores de Reportes			Interacción con Reportes	Aplica Si=1 No=0	1	0	1	0	0	1
		Coexistencia con IDEs de desarrollo	Coexistencia con IDEs de desarrollo			Interacción con IDEs	Aplica Si=1 No=0	1	1	1	1	1	1
				Total Coexistencia				2	1	3	1	1	3
	Interoperabilidad	Conectividad con sistemas externos	Conectividad con sistemas externos	M	Los	Capacidad de Interoperabilidad con sistemas externos	Aplica Si=1 No=0	1	1	1	1	1	1

Continúa



	Interoperar con productos del fabricante.	frameworks instalados facilitan la conectividad con algunos componentes externos.	Interoperar con IDE's de desarrollo	Aplica Si=1 No=0	1	1	1	1	1	1
	Interoperar con los formatos de texto.		Capacidad de Interoperar con Formatos de Texto	Aplica Si=1 No=0	1	1	1	1	1	1
	Informes PDF, Excel.		Capacidad de Interoperar con Formatos PDF, Excel	Aplica Si=1 No=0	1	1	1	1	1	1
	Gráficos		Capacidad de Interoperar con Gráficos	Aplica Si=1 No=0	1	0	1	0	1	0
			Total Interoperabilidad		5	4	5	4	4	5
			Total Compatibilidad		11	11	12	11	11	12

Portabilidad: Capacidad del producto de software de ser transferido de forma efectiva y eficiente de un entorno hardware a otro.

Portabilidad	Característica	Subcaracterística	Métrica	Nivel de Importancia	Motivo de Selección	Detalle de la Métrica	Criterio de la Métrica	OpenFaces	RichFaces	PrimeFaces	ButterFaces	OmniFaces	BootsFaces
--------------	----------------	-------------------	---------	----------------------	---------------------	-----------------------	------------------------	-----------	-----------	------------	-------------	-----------	------------

Continúa



Portabilidad	Adaptabilidad	Adaptabilidad sobre Navegadores	A	El uso de componentes permite adaptabilidad en cualquier arquitectura de hardware/ entorno de software.	Adaptabilidad sobre navegadores conocidos	Aplica Si=1 No=0	1	1	1	1	1	1
		Mozilla Firefox			Adaptabilidad sobre navegador Mozilla Firefox	Aplica Si=1 No=0	1	1	1	1	1	1
		Internet Explorer			Adaptabilidad sobre navegador Internet Explorer	Aplica Si=1 No=0	1	1	0	1	1	1
		Safari			Adaptabilidad sobre navegador Safari	Aplica Si=1 No=0	1	1	1	1	1	1
		Opera			Adaptabilidad sobre navegador Opera	Aplica Si=1 No=0	0	0	1	1	1	1
		Idioma				Aplica	1	1	1	1	1	1

	disponible: (inglés).			Adaptabilidad sobre idiomas disponibles	Si=1 No=0							
				Total Adaptabilidad		6	6	6	7	7	7	
Capacidad de ser instala do	Facilidad de instalación	A		Existe facilidad en la instalación	Aplica Si=1 No=0	1	1	1	1	1	1	
	Manuales de Instalación			Existe manuales de instalación	Aplica Si=1 No=0	1	1	1	1	1	1	
	Ayuda en línea			Facilidad de ayuda en línea	Aplica Si=1 No=0	1	1	1	1	1	1	
	Comunidad de Soporte			Existe Comunidad de Soporte	Aplica Si=1 No=0	1	1	1	1	1	1	
				Total Capacidad de ser instalado		4	4	4	4	4	4	
Capacidad de ser reemplazado	Capacidad de ser usado reemplazando con el mismo propósito que otro producto software /Uso continuo de	M	Puede ser reemplazado pero si requerirá de un estudio previo para la implementación	El componen te puede ser reemplaza do	Aplica Si=1 No=0	1	1	1	1	1	1	

Continúa



		datos											
				Total Capacidad de ser reemplazado									
						Total Portabilidad							
								1		1		1	
								11		11		12	
Mantenibilidad: Capacidad del producto de software para ser modificado.													
Mantenibilidad	Característica	Subcaracterística	Métrica	Nivel de Importancia	Motivo de Selección	Detalle de la Métrica	Criterio de la Métrica	OpenFaces	RichFaces	PrimeFaces	ButterFaces	OmniFaces	BootsFaces
	Mantenibilidad	Modularidad	División en partes más pequeñas que funcionan de forma independiente.	M	El framework se adapta a implementación de interfaces y plantillas..	Estructura del framework adaptable por módulos	Aplica Si=1 No=0	1	1	1	1	1	1
				Total Modularidad				1		1		1	
		Reutilización	Ejecución de reusabilidad	M	Los componentes pueden ser reutilizados en la misma vista u otras	Componentes reutilizables en otras vistas.	Aplica Si=1 No=0	1	1	1	1	1	1
				Total Ejecución de reusabilidad				1		1		1	

Continúa



	Capacidad de ser analizado	Facilidad para diagnosticar causas de errores semánticos o para identificar donde se debe realizar cambios.	M	Se maneja código liberado y varios desarrolladores del mundo pueden aportar sus conocimientos respecto a versionamiento.	Permite Identificar un componente con errores semánticos.	Aplica Si=1 No=0	1	1	1	1	1	1
				Total Facilidad para diagnosticar causas de errores semánticos			1	1	1	1	1	1
	Capacidad de ser modificado	Complejidad/ Facilidad de modificación incluye en el diseño, código, documentación.	M	Renderización del componente para evaluar el comportamiento	Facilidad en el mantenimiento del componente. Permite modificar el código fuente de los componentes.	Aplica Si=1 No=0	0	1	1	1	1	1
				Total Capacidad de ser modificado			0	1	1	1	1	1
	Capacidad de ser probado	Capacidad de reinicio de pruebas /Facilidad para evaluar las partes modificadas/	M	Reinicio de pruebas	Evaluar el componente mediante pruebas	Aplica Si=1 No=0	1	1	1	1	1	1
				Total Capacidad de ser probado			1	1	1	1	1	1
					Total Mantenibilidad		4	5	5	5	5	5

Es importante señalar que el porcentaje de evaluación para el parámetro Facilidad para el desarrollo como se muestra en la Tabla 38, ha sido considerado en base a los criterios de investigación de diversas fuentes blogs, foros, publicaciones, ambiente de pruebas instalado, cuadrante de garner, frameworks mejor puntuados en el sitio web (SiteRankData, 2019) que han servido para guiar la evaluación de cada framework..

Tabla 38

Ponderación de los Criterios del Parámetro Facilidad para el desarrollo

Parámetro	Criterios	Open Faces	RichFaces	PrimeFaces	ButterFaces	OmniFaces	BootsFaces
Facilidad para el desarrollo	Facilidad de Instalación	5%	10%	10%	5%	5%	10%
	Facilidad para actualizar componentes	5%	5%	10%	10%	10%	10%
	Facilidad de uso	10%	5%	10%	10%	10%	5%
	Compatibilidad Ajax	20%	20%	20%	20%	20%	20%
	Líneas de código	10%	10%	10%	5%	5%	5%
Total		50%	50%	50%	50%	50%	50%

En cuanto al Aprendizaje, la documentación disponible que permita realizar una aplicación con OpenFaces, ButterFaces y OmniFaces en Eclipse IDE, es escasa, puesto que para ButterFaces y OmniFaces existe información orientada a Maven y Wildfly y para OpenFaces es difícil encontrar un material útil que ayude con la inicialización de dicha aplicación, no siendo este el caso de RichFaces, PrimeFaces o BootsFaces, pues se posee documentación que permite crear aplicaciones ahorrando tiempo y esfuerzo en el desarrollo.

3.6.2 Selección del Framework de presentación

Como resultado de aplicar la matriz para los seis frameworks seleccionados, se pudo obtener los puntajes promedio como se detallan a continuación en la Tabla 39.

Tabla 39

Selección del framework de presentación

Características de evaluación de calidad	Open Faces	RichFaces	PrimeFaces	ButterFaces	OmniFaces	BootFaces
Adecuación Funcional	18	23	43	17	13	46
Fiabilidad	9	10	14	9	9	15
Rendimiento	4	5	6	3	3	7
Usabilidad	11	11	12	11	11	12
Compatibilidad	11	11	12	11	11	12
Portabilidad	11	11	11	12	12	12
Mantenibilidad	4	5	5	5	5	5
Total	68	76	103	68	64	109

3.6.3 Selección del framework en base a puntajes promedio

De acuerdo a los valores obtenidos de las Características de evaluación de calidad se ha establecido crear una escala de valores que ayude a obtener la Selección del mejor framework que se ajuste a las necesidades del Sistema de Albergues de damnificados del terremoto de Manabí,

A continuación en la Tabla 40 se detalla el nivel de ponderación (Indicador de Selección) de acuerdo al criterio de selección del framework.

Tabla 40*Selección del framework de acuerdo al mejor puntaje*

Criterio de Selección	Nivel de Importancia	Grado de satisfacción	Indicador de Selección
Mayor a 15	A	Alto	5
Entre 12-15	M	Medio	3
Entre 5 Y 11	B	Bajo	2
Menor a 4	N/A	No Aplica	0

3.6.4 Resultados Totales de los frameworks

Una vez que se ha asignado valores a las características de evaluación de calidad de acuerdo al indicador de selección, se procede a obtener el total promedio de cada framework, el mismo que servirá para evaluar el mejor framework que cumpla con los requisitos de calidad.

A continuación, en la Tabla 41, se presenta los Resultados Totales de los frameworks de acuerdo a las características de evaluación de calidad

Tabla 41*Resultados Totales de los frameworks*

Características de evaluación de calidad	Open Faces	RichFaces	PrimeFaces	ButterFaces	OmniFaces	BootsFaces
Adecuación Funcional	5	5	5	5	3	5
Fiabilidad	2	2	3	3	2	3
Rendimiento	0	2	2	0	0	2
Usabilidad	2	2	3	2	2	3
Compatibilidad	2	2	3	2	2	3
Portabilidad	2	2	2	3	3	3
Mantenibilidad	0	2	2	2	2	2
Total	13	14	17	13	11	21

3.6.5 Calificación del framework

Los Criterios de Selección han sido valorados tomando en cuenta el promedio total de los frameworks en relación a las características de evaluación.

A continuación, en la Tabla 42, se presenta el detalle de parámetros para la evaluación de frameworks de acuerdo al grado de satisfacción respectivo.

Tabla 42

Calificación del framework

Criterio	Nivel de Importancia	Grado de Satisfacción
≥ 17	A	Alto
Entre 14-17	M	Medio
Entre 11 - 13	B	Bajo
≤ 10	N/A	No Aplica

En base al Criterio de selección que cumpla con los requisitos de calidad de evaluación se ha escogido a **BootsFaces**, puesto que tiene un Grado de Satisfacción: Alto, y además cumple con todos los requisitos necesarios para el desarrollo del Sistema de damnificados del terremoto de Manabí, ofreciendo Funcionalidad, Fiabilidad, Usabilidad, Compatibilidad, Portabilidad, en cuanto a mejorar el rendimiento dependerá del servidor de aplicaciones, para el presente caso se utiliza **Wildfly 11** que es uno de los más recomendados, está diseñado para un alto rendimiento en cuanto a características ofrece administración de gran alcance, Configuración y gestión unificadas. Soporta los últimos estándares y tecnología. Java EE 8 y Java modular. Es capaz de arrancar los procesos críticos en paralelo, eliminando las esperas innecesarias y aprovechando la potencia de los procesadores multi-núcleo, es Incrustable, orientado a arquitectura de servicios, Flexibilidad consistente, Servicios del middleware para cualquier objeto de Java, además, brinda soporte completo para JMX.

CAPÍTULO IV

DISEÑO, DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA DE ALBERGUES

4.1 Planificación

El Sistema de albergues permitirá ayudar a llevar de mejor manera el control automatizado de los damnificados del terremoto de Manabí para determinar el número de damnificados de acuerdo a cada uno de los albergues de esta manera poderles brindar la ayuda necesaria.

4.1.1 Especificación de Requerimientos

Para cumplir con esta fase se aplicará la Norma (ISO/IEC/IEEE 29148, 2011)

El propósito de utilizar este estándar es definir las funcionalidades y restricciones del sistema a construir.

El estándar ayudara a establecer las características funcionales y requerimientos necesarios tanto a nivel de Software como de Hardware y estará dirigido a los clientes y equipo de desarrollo como un medio que sirva de guía en el proceso de desarrollo del sitio web, mediante el uso Historias de usuario se describe lo que el usuario desea hacer. , se centran en el valor en lugar de una especificación detallada de lo que el sistema debe hacer. Están concebidos como un medio para fomentar la colaboración. (ISO/IEC/IEEE 29148, 2011).

De esta forma se podrá determinar si los requisitos están de acuerdo con las necesidades del cliente que plantea la Norma (ISO/IEC/IEEE 29148, 2011).

- Mediante la Verificación de Requisitos — se determina si el producto de alguna actividad de desarrollo cumple los requisitos (hacer las cosas bien).
- Validación — evalúa si un producto satisface las necesidades del cliente (hacer la cosa correcta).

Después de la planificación de los requisitos necesarios con el cliente del Sistema de Albergues de Damnificados del Terremoto de Manabí se procedió a establecer los diferentes requisitos Funcionales y No funcionales.

4.1.1.1 Requerimientos Funcionales: Historias de Usuarios

Las características o requerimientos que debe poseer el proyecto, así como la Documentación respectiva se muestran a continuación:

Tabla 43

Historia 1- iteración 1

Historia de Usuario	
HU-001	Actores: Usuario Administrador General, Administrador de Albergue.
Nombre historia: Iniciar Sesión	
Prioridad en negocio: Alta	Riesgo de desarrollo: Alto
Esfuerzo: Medio	Iteración asignada: 1
Programador Responsable: Patricio Arrobo	
Descripción:	
<ol style="list-style-type: none"> 1. Para ingresar en la aplicación el usuario tiene que autenticarse mediante un usuario y clave pudiendo ingresar según el tipo de usuario. 2. Debe existir cuentas de usuario creadas 	
Observaciones: El registro de usuarios es para uso exclusivo de personal Administrativo autenticado por medio de contraseña.	
El Registro ingresado de Usuario Administrador es almacenado en la base de datos.	

Tabla 44*Historia 2- iteración 1*

Historia de Usuario	
HU-002	Actores: Usuario Administrador General, Administrador de Albergue.
Nombre historia: Gestionar Procesos	
Prioridad en negocio: Alta	Riesgo de desarrollo: Alto
Esfuerzo: Medio	Iteración asignada: 1
Programador Responsable: Patricio Arrobo	
Descripción:	
<ol style="list-style-type: none"> 1. Para ingresar en la aplicación el usuario tiene que autenticarse mediante un usuario y clave pudiendo ingresar según el tipo de usuario. 2. Debe Seleccionar Procesos que desee Controlar (Usuarios, Albergues, Damnificados, Personal Administrativo y Parte Diario). 	
Observaciones: El registro de usuarios es para uso exclusivo de personal Administrativo autenticado por medio de contraseña..	

Tabla 45*Historia 3- iteración 1*

Historia de Usuario	
HU-003	Actores: Administrador General (Súper Administrador),
Nombre historia: Gestionar Albergues.	
Prioridad en negocio: Alta	Riesgo de desarrollo: Alto
Esfuerzo: Alto	Iteración asignada: 1
Programador Responsable: Patricio Arrobo	
Descripción: El usuario autenticado ingresa en el Sistema y desde el menú Procesos elige la opción Crear Albergues desplegado desde el submenú Albergues, desde allí, se presentan las acciones que desea realizar Crear un Nuevo albergue, Modifica, Buscar o Eliminar.	
Observaciones: Para realizar esta actividad se debe ingresar como administrador General,. Se debe tener datos ingresados para seleccionar el Tipo de Comando, Tipo de Fuerza, Provincia, Cantón, Parroquia, y el Tipo de Albergue. El Registro ingresado de Albergues es almacenado en la base de datos.	

Tabla 46*Historia 4- iteración 1*

Historia de Usuario	
HU-004	Actores: Administrador General (Súper Administrador), Administrador de Albergue.
Nombre historia: Gestionar Administradores de Albergue.	
Prioridad en negocio: Alta	Riesgo de desarrollo: Alto
Esfuerzo: Medio	Iteración asignada: 1
Programador Responsable: Patricio Arrobo	
<p>Descripción: El usuario autenticado ingresa en el Sistema y desde el menú Procesos elige la opción Albergues, se presentan tres alternativas elige Administración del albergue donde se despliega el listado de administradores, desde allí, se presentan las acciones que desea realizar Crear Nuevo Administrador, Modificar, Buscar o Eliminar. El mismo.</p>	
<p>Observaciones: Para realizar esta actividad se debe tener ingresado un usuario. El Registro ingresado de Administrador de Albergue es almacenado en la base de datos.</p>	

Tabla 47*Historia 5- iteración 1*

Historia de Usuario	
HU-005	Actores: Administrador General (Súper Administrador), Administrador de Albergue.
Nombre historia: Gestionar Cabecera de Servicios	
Prioridad en negocio: Alta	Riesgo de desarrollo: Alto
Esfuerzo: Medio	Iteración asignada: 1
Programador Responsable: Patricio Arrobo	
<p>Descripción: El usuario autenticado ingresa en el Sistema y desde el menú Servicios elige la opción Cabecera de Servicios, desde allí, se presentan las acciones que desea realizar Crear Nueva Cabecera, Modificar, Buscar o Eliminar.</p>	
<p>Observaciones: Para realizar esta actividad se debe tener ingresado el albergue al que pertenece. El Registro ingresado de Cabecera de Servicios es almacenado en la base de datos.</p>	

Tabla 48*Historia 6- iteración 1*

Historia de Usuario	
HU-006	Actores: Administrador General (Súper Administrador), Administrador de Albergue.
Nombre historia: Gestionar Servicios Generales.	
Prioridad en negocio: Alta	Riesgo de desarrollo: Alto
Esfuerzo: Medio	Iteración asignada: 1
Programador Responsable: Patricio Arrobo	
Descripción: El usuario autenticado ingresa en el Sistema y desde el menú Servicios elige la opción Servicios Generales, desde allí, se presentan las acciones que desea realizar : Crear Nuevo Servicio, Modificar, , Buscar o Eliminar.	
Observaciones: Para realizar esta actividad se debe tener ingresado el Albergue al que pertenece, el Servicio,(Servicio, Tipo de Ingreso, Subtipo-servicio) que se desea controlar. El Registro ingresado de Servicios Generales es almacenado en la base de datos.	

Tabla 49*Historia 7- iteración 1*

Historia de Usuario	
HU-007	Actores: Administrador General (Súper Administrador), Administrador de Albergue.
Nombre historia: Gestionar Damnificados	
Prioridad en negocio: Alta	Riesgo de desarrollo: Alto
Esfuerzo: Medio	Iteración asignada: 1
Programador Responsable: Patricio Arrobo	
Descripción: El usuario autenticado ingresa en el Sistema y desde el menú Procesos elige la opción Damnificados, desde allí, se presentan las acciones que desea realizar Crear Nuevo, Modificar, Buscar o Eliminar un Usuario Damnificado.	
Observaciones: Para realizar esta actividad se debe tener ingresado el Albergue al que pertenece, el Damnificado, (Genero, Tipo de Genero, Lugar de Nacimiento, Estado Civil, Nivel de Educación, Tipo de discapacidad, Estado de embarazo, Etapa de la persona,) que se desea controlar. El Registro ingresado de Damnificados es almacenado en la base de datos.	

Tabla 50*Historia 8- iteración 1*

Historia de Usuario	
HU-008	Actores: Administrador General (Súper Administrador),
Nombre historia: Gestionar Personal Administrativo	
Prioridad en negocio: Alta	Riesgo de desarrollo: Alto
Esfuerzo: Medio	Iteración asignada: 1
Programador Responsable: Patricio Arrobo	
Descripción: El usuario autenticado ingresa en el Sistema y desde el menú Procesos elige la opción Personal Administrativo desde allí, se presentan las acciones que desea realizar Crear Nuevo Personal Militar, Modificar, Buscar o Eliminar.	
Observaciones: Para realizar esta actividad se debe tener ingresado el Albergue (Tipo de Personal, Tipo de Fuerzas, Estado al que pertenece, Tipo de Militar, Tipo de Rango, Genero, Estado Civil, Instrucción) que se desea controlar. El Registro ingresado de Personal Administrativo es almacenado en la base de datos.	

Tabla 51*Historia 9- iteración 1*

Historia de Usuario	
HU-009	Actores: Administrador General (Súper Administrador).
Nombre historia: Gestionar Mantenimiento.	
Prioridad en negocio: Alta	Riesgo de desarrollo: Alto
Esfuerzo: Medio	Iteración asignada: 2
Programador Responsable: Patricio Arrobo	
Descripción: El usuario autenticado ingresa como Administrador General en el Sistema y desde el menú Mantenimiento elige el ítem que desea realizar Mantenimiento, desde el ítem seleccionado, se controlan las acciones que desea realizar Crear Nuevo, Modificar o Eliminar.	
Observaciones: Para realizar esta actividad se debe tener ingresado lo que se solicite en el ítem seleccionado, Los Registros ingresados de ítems de Mantenimiento son almacenados en la base de datos.	

Tabla 52*Historia 10- iteración 2*

Historia de Usuario	
HU-010	Actores: Administrador General (Súper Administrador), Administrador de Albergue.
Nombre historia: Gestión de Reportes.	
Prioridad en negocio: Alta	Riesgo de desarrollo: Alto
Esfuerzo: Alto	Iteración asignada: 2
Programador Responsable: Patricio Arrobo	
Descripción: El usuario autenticado ingresa como Administrador General o Administrador de Albergue en el Sistema y desde el menú Reportes se despliegan tres alternativas elige la alternativa, que desea se despliegan los Listados respectivos, desde ahí, se controlan las acciones que desea realizar Crear Nuevo..., Modificar, Buscar o Eliminar.	
Observaciones: Los Registros ingresados en los Formulario son presentados en los Reportes.	

Tabla 53*Historia 11- iteración 2*

Historia de Usuario	
HU-011	Actores: Administrador General (Súper Administrador), Administrador de Albergue.
Nombre historia: Gestionar parte diario	
Prioridad en negocio: Alta	Riesgo de desarrollo: Alto
Esfuerzo: Medio	Iteración asignada: 2
Programador Responsable: Patricio Arrobo	
Descripción: El usuario autenticado ingresa en el Sistema y desde el menú Procesos elige la opción Parte Diario de Albergue, se presentan tres alternativas elige Crear Parte Diario de Albergue donde se despliegan diferentes ítems elige Listado de Parte Diario, desde allí, se presentan las acciones que desea realizar Crear Nuevo Parte Diario, Modificar, Buscar o Eliminar.	
Observaciones: Para realizar esta actividad se debe tener ingresado el Usuario y el Albergue al que pertenece. El Registro ingresado de Parte Diario del Albergue es almacenado en la base de datos.	

Tabla 54*Historia 12- iteración 2*

Historia de Usuario	
HU-012	Actores: Administrador General (Súper Administrador), Administrador de Albergue.
Nombre historia: Gestionar parte diario militar	
Prioridad en negocio: Alta	Riesgo de desarrollo: Alto
Esfuerzo: Medio	Iteración asignada: 2
Programador Responsable: Patricio Arrobo	
Descripción: El usuario autenticado ingresa en el Sistema y desde el menú `Procesos elige la opción Parte Diario, se presentan tres alternativas elige Parte Diario Militar donde se despliega el Parte Diario Militar , desde allí, se controlan las acciones que desea realizar Crear Nuevo Parte Diario Militar , Modificar, Buscar o Eliminar.	
Observaciones: Para realizar esta actividad se debe tener ingresado el Administrador del Parte Diario y el Estado al que pertenece. El Registro ingresado de Parte Diario Militar es almacenado en la base de datos.	

Tabla 55*Historia 13- iteración 2*

Historia de Usuario	
HU-013	Actores: Administrador General (Súper Administrador), Administrador de Albergue.
Nombre historia: Gestionar parte diario de personas	
Prioridad en negocio: Alta	Riesgo de desarrollo: Alto
Esfuerzo: Medio	Iteración asignada: 2
Programador Responsable: Patricio Arrobo	
Descripción: El usuario autenticado ingresa en el Sistema y desde el menú Procesos elige la opción Parte Diario, se presentan tres alternativas elige Parte Diario Personas, donde se despliega el Parte Diario de Personas , desde allí, se controlan las acciones que desea realizar Crear Nuevo Parte Diario de Personas , Modificar, Buscar o Eliminar.	
Observaciones: Para realizar esta actividad se debe tener ingresado el Albergue, el Administrador del Parte Diario y la Etapa Persona a la que pertenece.. El Registro ingresado de Parte Diario de Personas es almacenado en la base de datos.	

Tabla 56*Historia 14- iteración 3*

Historia de Usuario	
HU-014	Actores: Usuario Administrador General, Administrador de Albergue.
Nombre historia: Gestionar Usuarios	
Prioridad en negocio: Alta	Riesgo de desarrollo: Alto
Esfuerzo: Medio	Iteración asignada: 3
Programador Responsable: Patricio Arrobo	
Descripción:	
<ol style="list-style-type: none"> 1. Para ingresar en la aplicación el usuario tiene que autenticarse mediante un usuario y clave pudiendo ingresar según el tipo de usuario. 2. Debe Seleccionar Usuarios del Menú Procesos que desee Controlar, desde allí se despliegan cuatro alternativas (Crear Usuarios, Crear Roles-Usuario, Crear Roles, Cambiar Clave) 	
Observaciones: El registro de usuarios es para uso exclusivo de personal Administrativo autenticado por medio de contraseña.	

Tabla 57*Historia 15- iteración 3*

Historia de Usuario	
HU-015	Actor:: Administrador General (Súper Administrador),
Nombre historia: Crear Usuarios	
Prioridad en negocio: Alta	Riesgo de desarrollo: Alto
Esfuerzo: Alto	Iteración asignada: 3
Programador Responsable: Patricio Arrobo	
Descripción: El usuario autenticado ingresa en el Sistema y desde el menú Procesos elige la opción Crear Usuarios desplegado desde el submenú Usuarios, desde allí, se presentan las acciones que desea realizar Crear un Nuevo usuario, Modifica, Buscar o Eliminar.	
Observaciones: Para realizar esta actividad se debe ingresar como administrador General. El Registro ingresado de Usuarios es almacenado en la base de datos.	

Tabla 58*Historia 16- iteración 3*

Historia de Usuario	
HU-016	Actor: Administrador General (Súper Administrador),
Nombre historia: Crear Roles-Usuario	
Prioridad en negocio: Alta	Riesgo de desarrollo: Alto
Esfuerzo: Alto	Iteración asignada: 3
Programador Responsable: Patricio Arrobo	
<p>Descripción: El usuario autenticado ingresa en el Sistema y desde el menú Procesos elige la opción Crear Roles-Usuarios desplegado desde el submenú Usuarios, desde allí, se presentan las acciones que desea realizar Crear un Nuevo usuario, Modifica, Buscar o Eliminar.</p>	
<p>Observaciones: Para realizar esta actividad se debe ingresar como administrador General, para poder elegir las alternativas de selección se debe tener ingresado el Rol de Usuario. El Registro ingresado de Roles - Usuarios es almacenado en la base de datos.</p>	

Tabla 59*Historia 17- iteración 3*

Historia de Usuario	
HU-017	Actor: Administrador General (Súper Administrador),
Nombre historia: Crear Roles	
Prioridad en negocio: Alta	Riesgo de desarrollo: Alto
Esfuerzo: Alto	Iteración asignada: 3
Programador Responsable: Patricio Arrobo	
<p>Descripción: El usuario autenticado ingresa en el Sistema y desde el menú Procesos elige la opción Crear Roles desplegado desde el submenú Usuarios, , desde allí, se presentan las acciones que desea realizar Crear un Nuevo usuario, Modifica, Buscar o Eliminar.</p>	
<p>Observaciones: Para realizar esta actividad se debe ingresar como administrador General. El Registro ingresado de Roles es almacenado en la base de datos.</p>	

Tabla 60*Historia 18- iteración 3*

Historia de Usuario	
HU-018	Actores: Administrador General (Súper Administrador),
Nombre historia: Cambiar Clave	
Prioridad en negocio: Alta	Riesgo de desarrollo: Alto
Esfuerzo: Alto	Iteración asignada: 3
Programador Responsable: Patricio Arrobo	
Descripción: Proceso mediante el cual, se modificará la contraseña de usuario registrado para acceso al sistema. Para la solicitud de la nueva contraseña, al usuario logueado se le pedirá una verificación de la contraseña (Captcha).	
Observaciones: Para realizar esta actividad se debe ingresar como administrador General. El Registro ingresado de Cambiar es almacenado en la base de datos.	

4.1.1.2 Requerimientos No Funcionales

Para el Sistema de Albergues de Damnificados de la Provincia de Manabí, se ha identificado los siguientes requerimientos no funcionales.

Tabla 61*Requerimientos No Funcionales 1 - Tiempo de Respuesta*

RNF-001	Tiempo de Respuesta
Descripción	El sistema debe presentar un tiempo de respuesta menor a 3 segundos, debiendo siempre mostrar mensajes de procesamiento que permitan al usuario mantener control del estado del sistema.
Importancia	Alta
Estado	Aprobado

Tabla 62*Requerimientos No Funcionales 2- Utilización de colores*

RNF-002	Utilización de Colores
Descripción	El sistema debe presentar una interface con patrones de colores de 3 combinaciones.
Importancia	Alta
Estado	Aprobado

Tabla 63*Requerimientos No Funcionales 3- Iconos de Opciones*

RNF-003	Iconos de opciones
Descripción	El sistema mantendrá un conjunto de iconos basados en la imagen corporativa y será ubicado en todos los botones de los formularios, además de un texto indicativo.
Importancia	Alta
Estado	Aprobado

Tabla 64*Requerimientos No Funcionales 4- Métodos de Acceso*

RNF-004	Métodos de Acceso
Descripción	El sistema permitirá la verificación de acceso mediante un usuario y una contraseña.
Importancia	Alta
Estado	Aprobado

Tabla 65*Requerimientos No Funcionales 5 - Plataforma Open Source*

RNF-005	Plataforma Open Source
Descripción	El sistema debe estar desarrollado en una plataforma Open Source en lo referente a las interfaces y la base de datos.
Importancia	Alta
Estado	Aprobado

Tabla 66*Requerimientos No Funcionales 6 - Accesibilidad*

RNF-006	Accesibilidad
Descripción	La interfaz deberá presentar en una sola pantalla toda la funcionalidad necesaria para dar mantenimiento a los datos.
Importancia	Alta
Estado	Aprobado

- El sistema contará con dos niveles de acceso: Administrador General y Administrador de Albergue.
- El sistema contará con un login de usuario y contraseña personal
- El sistema tendrá mecanismos de respaldo de información automáticos, los cuales aumentarán la integridad de los mismos.
- Emitirá mensajes de alertas cuando el usuario realice un procedimiento erróneo en la ejecución del sistema.
- Emitirá advertencias y mensajes de confirmación en el momento de registrar, modificar y eliminar usuarios, información de los albergues y damnificados.
- Los Usuarios se identifican por un nombre de usuario y clave.
- Las Búsquedas se realizarán clasificándolas según el dato solicitado.
- Los administradores podrán consultar información de albergues y damnificados ingresados.
- El administrador puede buscar datos de su albergue y damnificados a su cargo.

4.2 Iteraciones

Para los ciclos de pruebas de funcionamiento, retroalimentación y cambios en el “Sistema web para los albergues de damnificados del terremoto de Manabí”, se consideró dividir las Historias de Usuario en tres procesos iterativos. Como se puede observar en las Tablas de Historias de Usuarios que van de la Tabla 67 - Tabla 69.

4.2.1 Plan de Entrega

Mediante las reuniones establecidas con el cliente del “Sistema de albergues de damnificados del terremoto de Manabí” se logró recopilar información acerca de los requerimientos o Historias de usuario con el fin de evaluarlas en base al tiempo de duración destinado a cada una, agrupada de acuerdo al orden de importancia. El tiempo establecido para Gestionar el plan de entregas tuvo una duración de 6 meses.

A continuación, se muestra el Plan de Entregas en función del tiempo de duración de cada Historia de Usuario

Tabla 67

Plan de Entrega- Iteración 1

Número	Historia de Usuario	Duración (Semanas)
1	Iniciar Sesión	2
2	Gestionar Procesos	2
3	Gestionar Albergues	2
4	Gestionar Administradores de Albergue.	2
5	Gestionar Cabecera de Servicios.	2
6	Gestionar Servicios Generales	2
7	Gestionar Damnificados	2
8	Gestionar Personal Administrativo	2

Tabla 68*Plan de Entrega- Iteración 2*

Número	Historia de Usuario	Duración (Semanas)
9	Gestionar Mantenimiento	4
10	Gestionar Reportes	2
11	Gestionar Parte Diario	2
12	Gestionar parte diario militar	2
13	Gestionar parte diario de personas.	2

Tabla 69*Plan de Entrega - Iteración 3*

Número	Historia de Usuario	Duración (Semanas)
14	Gestionar Usuarios	2
15	Crear Usuarios	2
16	Crear Roles de Usuario	2
17	Crear Roles	2
18	Cambiar Contraseña	2

4.2.2 Reuniones

La Planificación de las reuniones, para la revisión y entrega de avances del “Sistema web para los albergues de damnificados del terremoto de Manabí” se lo realizo en las instalaciones del Departamento de Ciencias de la Computación de la Universidad de las Fuerzas Armadas – Espe, los días lunes, miércoles y viernes de 3:00 PM a 5:00 PM.

4.3 Diseño

Durante esta fase se realizaron cambios constantes dado la necesidad de proveer la mejor funcionalidad acorde a los requerimientos de la Universidad de las Fuerzas

Armadas – Espe en función que no se disponía un sistema con información integrada de Administración de Albergues.

4.4 Procesos

Los procesos utilizados en el Sistema de Albergues de damnificados del terremoto de Manabí, se indican a continuación, mediante los siguientes diagramas de flujo.

4.4.1 Proceso de Administración de Albergues

En la Figura 24, se muestra el proceso de las actividades que se realizan en la Administración de Albergues.

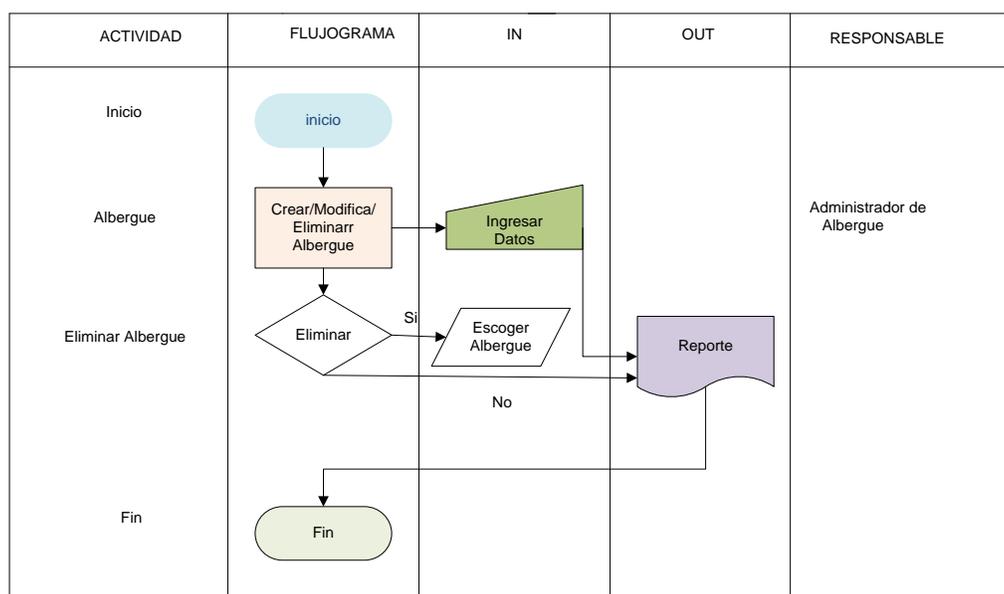


Figura 24. Proceso de Administración de Albergues

4.4.2 Proceso de Administración de Damnificados

En la Figura 25, se muestra el proceso de las actividades que se realizan la Administración de Damnificados.

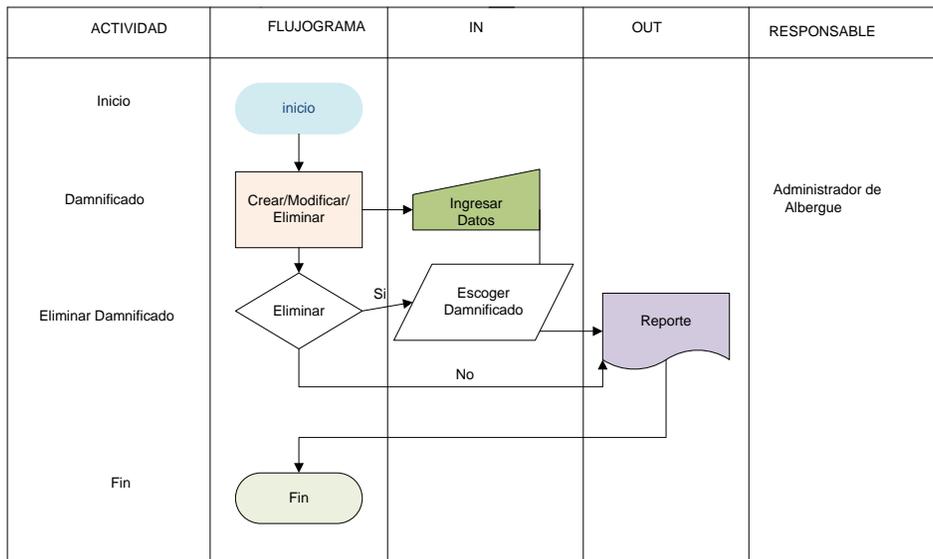


Figura 25. Proceso de Administración de Damnificados

4.4.3 Proceso de Administración de Personal Administrativo

En la Figura 26, se muestra el proceso de las actividades que se realiza en la Administración de Personal Administrativo.

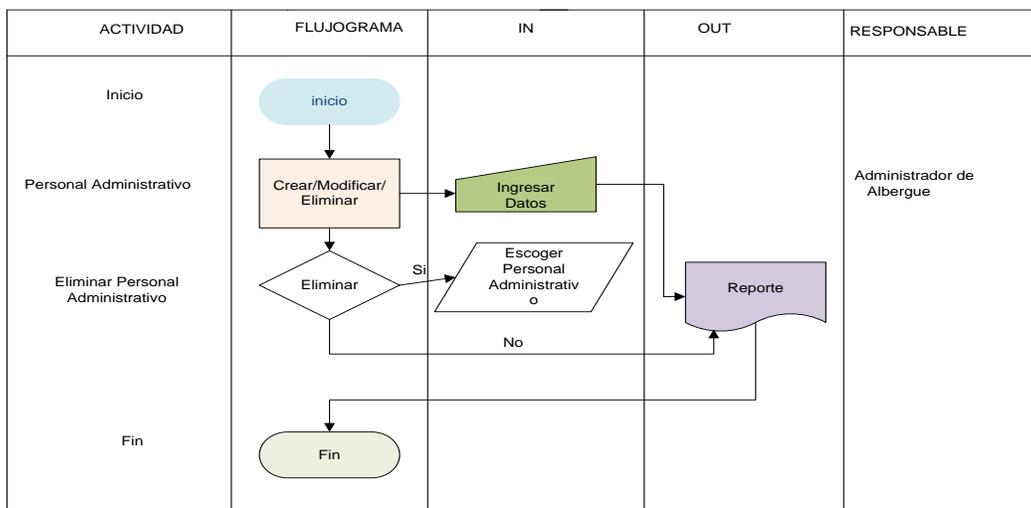


Figura 26. Proceso de Administración de Personal Administrativo

4.4.4 Proceso de Administración de Parte Diario

En la Figura 27, se muestra el Proceso de las actividades que se realizan en la Administración de Parte Diario.

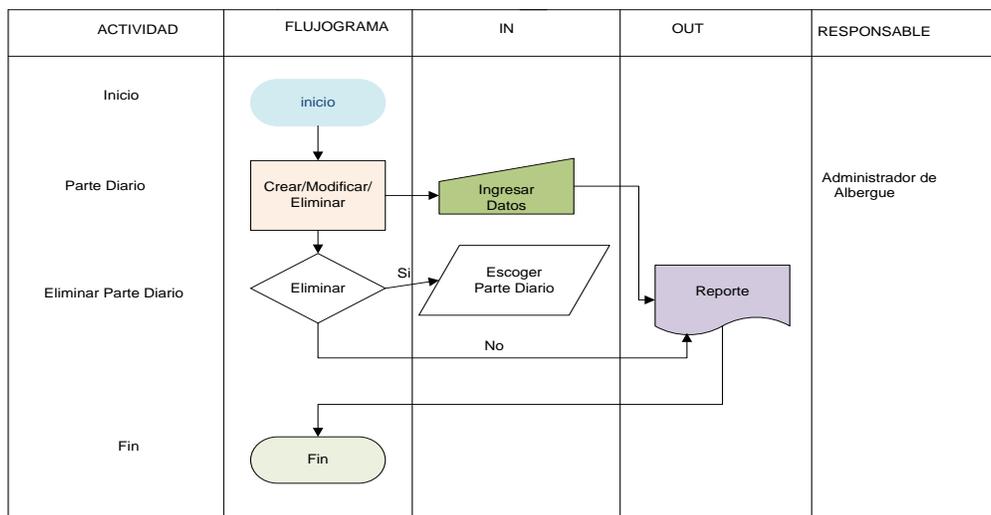


Figura 27. Proceso de Administración de Parte Diario

4.5 Desarrollo

La implementación del “Sistema web para los albergues de damnificados del terremoto de Manabí” se realizó en base a la planificación, después de reunir todos los requerimientos necesarios, posteriormente se procedió a la creación de Historias de usuario (Tabla 43 -60), que fueron modificándose o eliminándose a lo largo del proceso de desarrollo, para tener mayor claridad a lo solicitado por el cliente como parte del equipo de desarrollo.

4.5.1 Integrantes y Roles de usuarios.

La Universidad de las Fuerzas Armadas - ESPE para fines de Administración del “Sistema web para los albergues de damnificados del terremoto de Manabí” ha tomado en cuenta la participación de dos perfiles de usuario como son: **Administrador General** y **Administrador de Albergue**, planificando las actividades relacionadas con el control de los procesos de gestión y control de los Albergues.

A continuación, en la Tabla 70, se detalla las tareas que deben cumplir los Perfiles de Usuarios de acuerdo al Rol.

Tabla 70

Integrantes del Sistema y Roles de Usuario

Rol	Tareas
Administrador General	Control del sistema en las tareas referentes a Mantenimiento, Administradores de albergue y Reportes.
Administrador del Albergue	Control del sistema en las tareas referentes a Administración de su Albergue

En las siguiente Figuras (Fig. 29-30), se muestran las tareas que deben cumplir los Perfiles:

- Roles del Administrador General del Albergue
- Roles del Administrador de Albergue

- Roles del Administrador General de Albergue

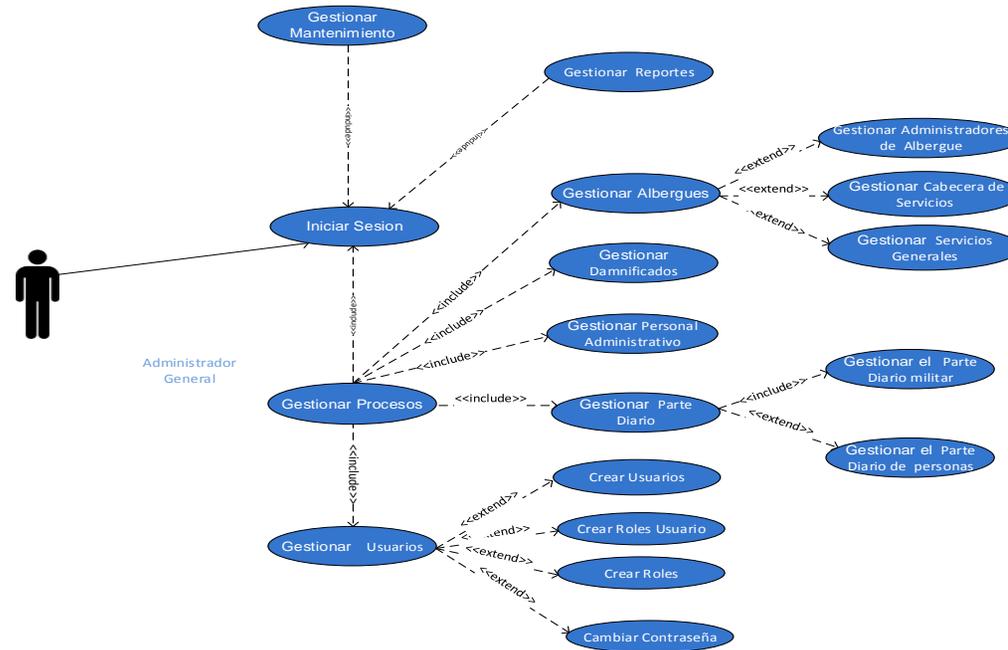


Figura 29. Roles del Administrador General

- Roles del Administrador de Albergue

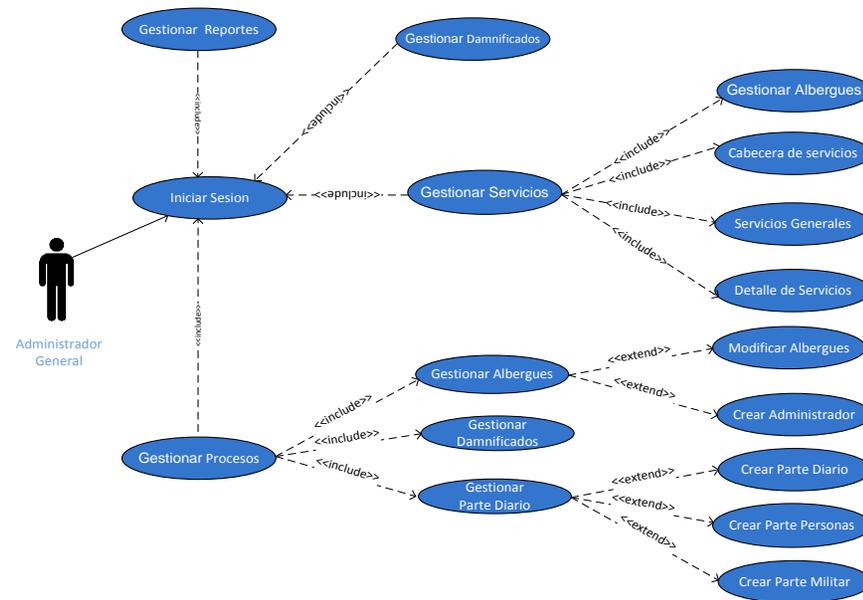


Figura 30. Roles del Administrador de Albergue

4.5.2 Prototipos

Como parte del Desarrollo del Sistema de Albergues de Damnificados de Manabí en las primeras fases del diseño se ha considerado la creación de prototipos. Según Toni Allen Analista e Investigador de Usabilidad en sitios web señala que La creación de prototipos es una gran oportunidad para ampliar su alcance, explorar problemas más amplios del propósito del sitio, la comprensión y las preferencias del usuario de tal forma de que participe el usuario en el desarrollo.

Un prototipo es una implementación parcial pero concreta del diseño de un sistema. Los prototipos pueden ser creados para explorar muchas cuestiones acerca del sistema durante el desarrollo del mismo.

Según (Rosson,Carroll, 2002) señalan que un prototipo de una interfaz de usuario se realiza con la finalidad de explorar los aspectos interactivos del sistema (incluyendo la usabilidad, la accesibilidad y/o la funcionalidad del mismo.

A continuación, se presenta los prototipos utilizados en el desarrollo del Sistema de Albergues de damnificados del terremoto de Manabí

4.5.3 Prototipo para Inicio de Sesión

Plantilla general que se utilizará para Ingresar al Sistema en donde cada usuario podrá visualizar los contenidos diferentes dependiendo de su Perfil de Usuario.

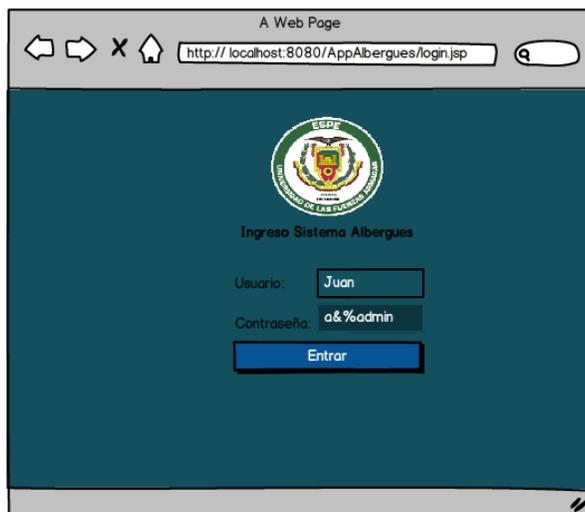


Figura 31. Prototipo de la plantilla general para Inicio de Sesión

4.5.4 Prototipo para Creación de Nuevas Cuentas

Plantilla general que se utilizara para Crear una Nueva Cuenta en donde cada usuario mediante un formulario de registro podrá ingresar sus datos, dependiendo de su Perfil de Usuario se le asignara un rol de acceso ya sea como Administrador General o Administrador de Albergue.

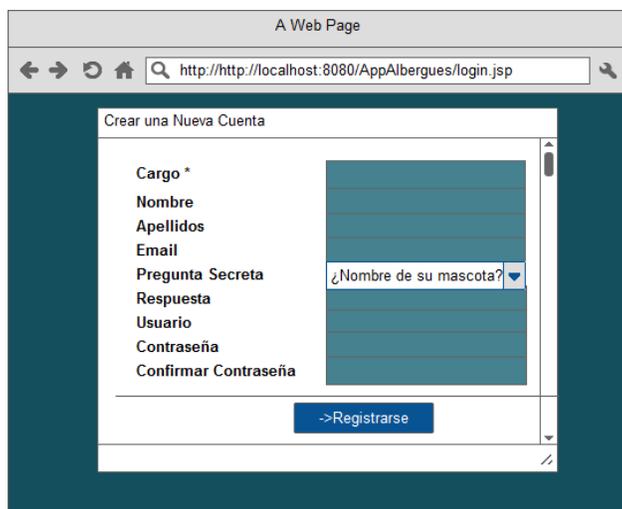


Figura 32. Prototipo de la plantilla general para la Creación de Nuevas Cuentas

4.5.5 Prototipo de la página de inicio Administrador General

Plantilla general que se utilizara para crear todas las interfaces del Administrador General, las mismas que permitirán visualizar, crear, modificar, eliminar contenidos diferentes.

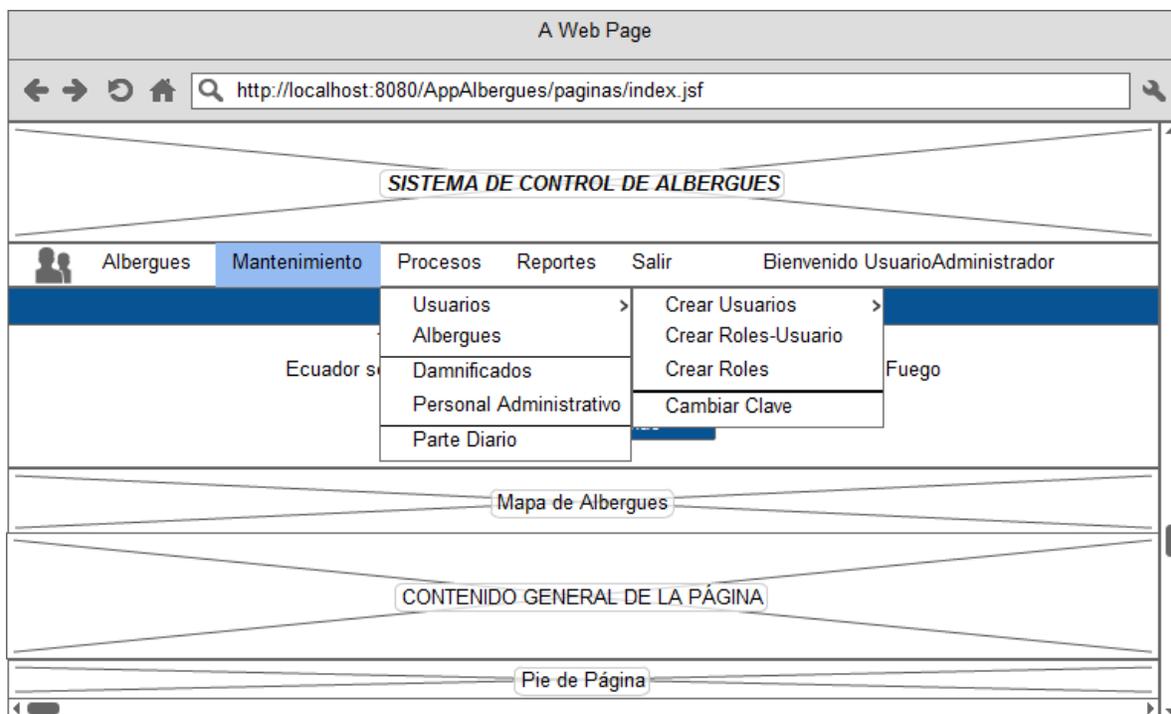


Figura 33. Prototipo de la plantilla Página de inicio del Administrador General

4.5.6 Prototipo de la página de inicio Administrador - Albergue

Plantilla general que se utilizara para crear todas las interfaces del Administrador de Albergue, las mismas que permitirán visualizar, crear, modificar, eliminar contenidos diferentes.

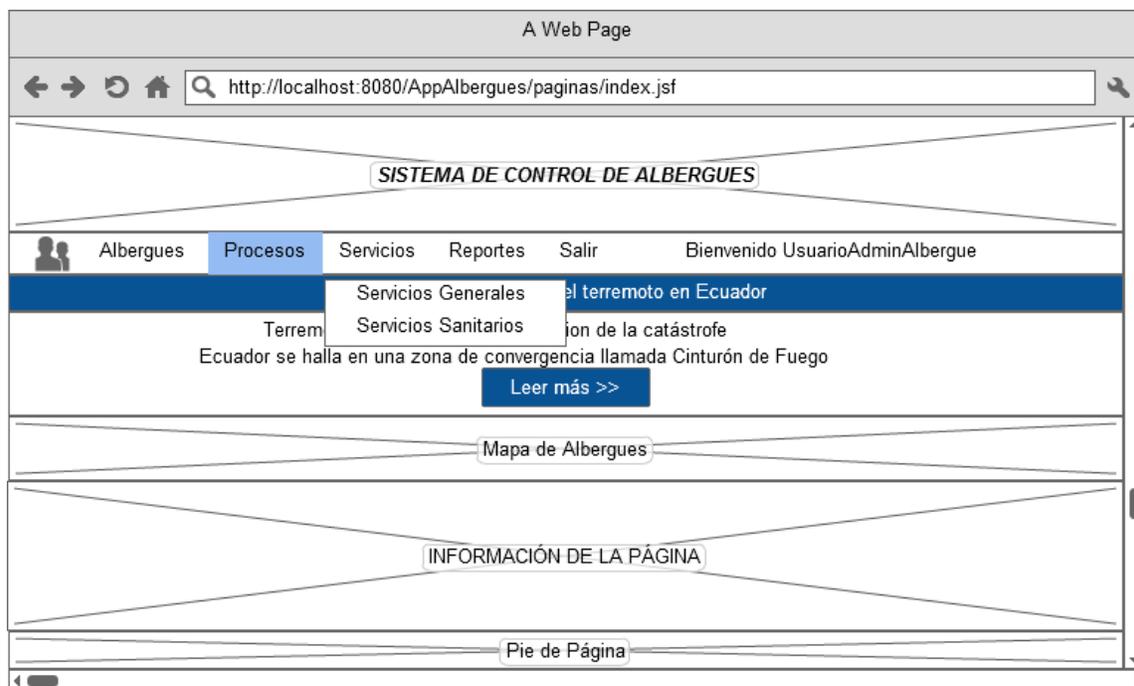


Figura 34. Prototipo de la plantilla Página de inicio del Administrador-Albergue

4.6 Interfaces de Usuario

Para el desarrollo de las páginas XHTML utilizadas para la capa de presentación relacionados con las interfaces de usuario se utilizó el framework : **Bootsfaces**, como se muestra en el **ANEXO A**, el mismo que sirvió para la creación de la plantilla principal y el contenido del Sistema web utilizando arquitectura jee7, Caso de Estudio albergues de damnificados del terremoto de Manabí ,las mismas que han sido integradas con los JavaBeans de acuerdo a las diferentes funcionalidades del sistema según el vínculo que el usuario seleccione.

4.6.1 Página de inicio

Diseñada para la validación y Registro de los datos del usuario en el sistema, misma que se puede apreciar en la Figura 35.

The image shows a web browser window displaying the login and registration interface for the 'Ingreso Sistema Albergues' system. The browser's address bar shows the URL 'localhost:8080/AppAlbergues/login.jsp?logout'. The page has a dark blue background with a network diagram and the ESPE logo. The login form includes fields for 'Usuario:' and 'Contraseña:' and an 'Entrar' button. To the right, there is a 'Crear una Nueva Cuenta' section with fields for 'Cargo', 'Nombres', 'Apellidos', 'Email', 'Pregunta Secreta' (with a dropdown menu), 'Respuesta', 'Usuario', 'Contraseña', and 'Confirmar Contraseña', along with a 'Registrarse' button.

Figura 35. Página de inicio para la validación y Registro de Inicio de Sesión

4.6.2 Plantilla principal

Permite organizar la estructura y apariencia del sistema distribuida en tres partes diferentes: banner, menú, contenido y pie de página. (Ver Figura 36), en la que se ilustra el diseño de la plantilla principal, en donde se observa el banner en la parte superior, el menú en la parte central y el contenido y pie de página, en la parte central-inferior, en base a este diseño, se realizó todas las demás interfaces, considerando que el área a cambiar será solamente el contenido (Ver Figura 37).

The screenshot displays the main interface of the 'Sistema de Control de Albergues'. At the top, there is a header with the ESPE and DECC logos and the title 'SISTEMA DE CONTROL DE ALBERGUES'. Below this is a navigation menu with options like 'Albergues', 'Mantenimiento', 'Procesos', 'Reportes', and 'Salir'. A dropdown menu is open under 'Reportes', showing options such as 'Crear Usuarios', 'Crear Roles-Usuario', 'Crear Roles', 'Personal Administrativo', and 'Cambiar Clave'. Below the navigation is a news section titled 'Terremoto en Ecuador: Una explicación de la catástrofe' with a 'Leer más' button. The main content area features a map titled 'Ubicación de Albergues' with a popup for 'Albergue Musne' showing details like 'Comando C.O.1', 'Responsable ING. JUAN PEREZ', and 'Administrador CEE'. Below the map are buttons for 'Ubicación de Albergues', 'Información del Albergue', and 'Servicios'. The bottom section is titled 'Listado de Albergues' and contains a table with columns for 'Albergue', 'Comando', and 'Administrador'. The table lists four albergues: ESMERALDAS 1, MUSNE, JAMA, and ESTEROS, each with a 'Comando' (C.O.1) and an 'Administrador' (ING. VICTOR GOMEZ BRAVO, ING. MAURICIO CAMPAÑA, ING. HENRY CORAL, and ING. JUAN FLORES). There are also buttons for 'Excel' and 'PDF' and a 'Crear Nuevo Albergue' button at the bottom.

Albergue	Comando	Administrador		
ESMERALDAS 1	C.O.1	TRIV. SU VICTOR GOMEZ BRAVO		
MUSNE	C.O.1	ING. MAURICIO CAMPAÑA		
JAMA	C.O.1	ING. HENRY CORAL		
ESTEROS	C.O.1	ING. JUAN FLORES		

Figura 36. Plantilla principal, utilizada para la estructura y apariencia del sistema.

Codigo	Fecha	Albergue	Total Familias	Total Personas	Total Discapacitados	Total Militares
2	2016-08-08	ESMERALDAS 1	6	2	4	5
3	2016-09-06	ESMERALDAS 1	1	2	4	5
4	2018-12-06	ESMERALDAS 1	2	4	7	5
1	2016-04-06	MUISNE	19	33	0	34
5	2018-12-12	JAMA	2	3	4	5

Figura 37. Contenido de la plantilla principal

4.6.3 Página de entrada de datos

Permite el ingreso o modificación de la información del Contenido, de acuerdo a los datos solicitados en el formulario, como se observa en la Figura 38.

Figura 38. Página de entrada de datos de Usuarios del Sistema de Albergues

4.7 Codificación

Una vez que se ha llegado a la fase de codificación, por medio de la programación se hará posible el cumplimiento de los requisitos descritos en las fases anteriores, para lo cual se usará un entorno integrado de desarrollo con la finalidad de crear un proyecto de forma ágil y sencilla.

Para la realización de este proyecto se eligió el IDE Eclipse como entorno de desarrollo por las características presentadas en la **Tabla 5**, Lenguaje Java en el Servidor Backend de la aplicación por las características presentadas en la **Tabla 4**, en la parte Front-End, el Framework de presentación **Bootsfaces**, de acuerdo al estudio comparativo de las librerías de componentes JsF.

Para fines de construcción, se ha previsto utilizar la Plataforma Java EE 7, por las características presentadas en la **Tabla 9**, mediante el Patrón de Diseño Modelo-Vista-Controlador se creará módulos independientes, separando la vista de la lógica de negocio (modelos), con el fin de separar los roles de usuario que se utilizan en el Sistema de Albergues de damnificados del terremoto de Manabí.

4.8 Pruebas

4.8.1 Pruebas Unitarias

4.8.2 Pruebas de Aceptación

Las pruebas de aceptación se las realizo considerando los niveles administrativos (Administrador General / Administrador de Albergue) a fin de validar que el “Sistema de albergues de damnificados del terremoto de Manabí “cumpla con las funciones descritas en la Historia de Usuario, desde el punto de vista de su funcionalidad y rendimiento.

A continuación, se detallan cada una de las pruebas realizadas:

Tabla 71

Prueba de aceptación Iniciar Sesión

Identificador:	HU-001
Historia de Usuario:	Iniciar Sesión
Descripción de la Prueba:	La prueba consiste en administrar los usuarios que interactúan con el Sistema, mediante la asignación de cuentas de ingreso.
Propósito:	Verificar el proceso de creación de un usuario Administrador General/Administrador de Albergue en el sistema de albergues.
Condición de Ejecución:	El usuario debe ingresar al sistema autenticado como Administrador General/Administrador de Albergue. El usuario Administrador General/Administrador de Albergue debe ingresar cambiada la contraseña.
Dependencias:	
Acciones	
Acción del Actor:	Respuesta del Sistema:
<ol style="list-style-type: none"> 1. Ingresar como Administrador General/Administrador de Albergue. 2. Elegir Procesos ->Usuarios- >Crear Usuarios. 3. Presionar el botón Crear Nuevo Usuario ubicado en la parte inferior izquierda. 5. Ingresar los datos solicitados del Formulario. 6. Verificar el ingreso de datos en todos los campos obligatorios para la creación de un nuevo usuario. 7. Crear el usuario a través del botón Guardar 9. Seleccionar un usuario ya creado. 	<ol style="list-style-type: none"> 4. Aparece el controlador para ingresar Nuevo usuario. 8. Si no se ingresa algún dato solicitado en el formulario de creación de usuario aparece Mensaje de Confirmación o Error.

<p>10. Editar la información del usuario (cambiar contraseña, elegir parámetros, email, de acuerdo a las necesidades del mismo)</p> <p>11. Dar clic en el botón Guardar.</p> <p>12. Cerrar sesión</p> <p>13. Ingresar al sistema con los datos de usuario creados.</p> <p>14. Cerrar sesión</p>	
Datos de Entrada:	<p>Contraseña de Administrador General/Administrador de Albergue.</p> <p>Contraseña de nuevo Administrador creado.</p>
Resultados Esperados:	<p>Después de ejecutar el paso 7, en modo CORRECTO validando y guardando los datos se puede iniciar sesión dentro del sistema, el sistema permitirá el ingreso del usuario.</p> <p>En la ejecución del paso 12, previo a la asignación de la nueva contraseña la sesión debe cerrarse y el nuevo usuario debe volver a ingresar con el nombre de usuario y contraseñas asignadas.</p>
Resultados Reales:	<p>El Registro de Usuario es CORRECTO por lo tanto es almacenado en la base de datos de usuarios.</p>

Tabla 72

Prueba de aceptación para Gestionar Procesos

Identificador:	HU-002
Historia de Usuario:	Gestionar Procesos
Descripción de la Prueba:	La prueba consiste en visualizar los Procesos que interactúan con el Sistema una vez que el usuario Administrador ha sido logueado.
Propósito:	Visualizar los procesos con los que cuenta el Sistema de albergues.
Condición de Ejecución:	El usuario debe ingresar al sistema autenticado como Administrador General/Administrador de Albergue.

Dependencias:	
Acciones	
Acción del Actor:	Respuesta del Sistema:
<ol style="list-style-type: none"> 1. Ingresar como Administrador General/Administrador de Albergue. 2. Elegir Procesos. 3. Desplegar el Menú Procesos, 5. Escoger el Proceso con el que desea interactuar. 6. Cerrar sesión 	<ol style="list-style-type: none"> 4. Se despliegan los Procesos disponibles.
Datos de Entrada:	Contraseña de Administrador General
Resultados Esperados:	Después de ejecutar el paso 5, en modo CORRECTO se podrá utilizar los Procesos que se deseen en el Sistema de Albergues.
Resultados Reales:	El ingreso al sistema es CORRECTO por lo tanto se puede utilizar cada uno de los Procesos.

Tabla 73

Prueba de aceptación para Gestionar Albergues

Identificador:	HU-003
Historia de Usuario:	Gestionar Albergues.
Descripción de la Prueba:	La prueba consiste en Gestionar Albergues por parte del Administrador de Albergue.
Propósito:	Verificar el proceso de control de Albergues por medio del Administrador en el sistema de albergues.
Condición de Ejecución:	<p>El usuario debe ingresar al sistema autenticado como Administrador de Albergue.</p> <p>El usuario Administrador de Albergue debe ingresar y controlar su albergue.</p>
Dependencias:	<p>El usuario escoge el albergue que le ha sido asignado.</p> <p>La información puede ser vista por el Administrador General/Administrador de Albergue.</p>



Acciones	
Acción del Actor:	Respuesta del Sistema:
<p>1. Ingresar como Administrador de Albergue.</p> <p>2. Elegir Procesos ->Albergues - >Crear Albergues/Crear Administrador</p> <p>3. Administrar Proceso</p> <p>5. Ingresar los datos solicitados del Formulario.</p> <p>6. Verificar el ingreso de datos en todos los campos obligatorios para la administración del proceso.</p> <p>7. Crear el proceso solicitado a través del botón Guardar</p> <p>9. Seleccionar un Albergue ya creado.</p> <p>10. Editar la información del albergue (elegir parámetros, de acuerdo a las necesidades del mismo)</p> <p>11. Dar clic en el botón Guardar.</p> <p>12. Cerrar sesión</p> <p>13. Ingresar al sistema con los datos de usuario creados.</p> <p>14. Cerrar sesión</p>	<p>4. Aparece el controlador para Administrar el proceso seleccionado.</p> <p>8. Si no se ingresa algún dato solicitado en el formulario de administración del proceso aparece Mensaje de Confirmación o Error en datos requeridos.</p>
Datos de Entrada:	Contraseña de Administrador de Albergue.
Resultados Esperados:	<p>Después de ejecutar el paso 7, en modo CORRECTO validando y guardando los datos se puede Administrar el Albergue.</p> <p>En la ejecución del paso 12, previo a la asignación del proceso, podrá ser modificado de acuerdo a la necesidad del Administrador.</p>
Resultados Reales:	El control del proceso es CORRECTO por lo tanto es almacenado en la base de datos de Albergues y puede ser utilizado por el Administrador.

Tabla 74*Prueba de Aceptación Gestionar Administradores de Albergue*

Identificador:	HU-004
Historia de Usuario:	Gestionar Administradores de Albergue
Descripción de la Prueba:	La prueba consiste en administrar la Información de los Administradores de Albergue por parte de los Administradores Generales.
Propósito:	Verificar el proceso de Ingreso de Información de los Administradores de Albergue por parte de los Administradores Generales.
Condición de Ejecución:	El usuario debe ingresar al sistema autenticado como Administrador General. El Usuario Administrador General debe controlar la Información de los Administradores de Albergue.
Dependencias:	La información ingresada por el Administrador de Albergue es controlada por el Administrador General
Acciones	
Acción del Actor:	Respuesta del Sistema:
<ol style="list-style-type: none"> 1. Ingresar como Administrador General 2. Elegir Albergue que desea controlar Información ingresada. 3. Solicitar Reporte (Pdf/Excel) de Datos ingresados. 5. Visualizar Reporte. 6. Verificar el ingreso de datos de acuerdo al Albergue solicitado. 7. Crear Administradores de Albergue /Albergues a través del botón Guardar 9. Seleccionar un Administrador ya creado 10. Editar la información del Administrador (cambiar contraseña, elegir parámetros, contactos, de acuerdo a las necesidades del mismo) 	<ol style="list-style-type: none"> 4. Aparece el Reporte de Datos Ingresados por el Administrador de Albergue. 8. Si no se ingresa algún dato solicitado en el formulario de creación de Administradores de Albergue/Albergues aparece Mensaje de Confirmación o Error.



11. Dar clic en el botón Guardar. 12. Cerrar sesión	
Datos de Entrada:	Contraseña de Administrador General.
Resultados Esperados:	Después de ejecutar el paso 7, en modo CORRECTO validando y guardando los datos se puede controlar la información tanto de Administradores de Albergue como de Albergues.
Resultados Reales:	El Registro de Administradores de Albergue /Albergues es CORRECTO por lo tanto es almacenado en la base de datos de Administración de Albergues.

Tabla 75*Prueba de Aceptación Gestionar Cabecera de Servicios*

Identificador:	HU-005
Historia de Usuario:	Gestionar Cabecera de Servicios
Descripción de la Prueba:	La prueba consiste en administrar la Cabecera de Servicios por parte del Administrador de Albergue.
Propósito:	Verificar el servicio de Cabecera de servicios, de acuerdo al Albergue seleccionado.
Condición de Ejecución:	El usuario debe ingresar al sistema autenticado como Administrador de Albergue. El usuario Administrador de Albergue debe ingresar la Cabecera de Servicios para poder ingresar el Detalle de Servicios.
Dependencias:	
Acciones	
Acción del Actor:	Respuesta del Sistema:
1. Ingresar como Administrador de Albergue. 2. Elegir Servicios -> Cabecera de Servicios 3. Presionar el botón Crear Nueva Cabecera de Servicios ubicado en la parte inferior izquierda. 5. Ingresar los datos solicitados del Formulario.	4. Aparece el controlador para ingresar Nueva cabecera de servicios.



<p>6. Verificar el ingreso de datos en todos los campos obligatorios para la creación de una nueva Cabecera de Servicios.</p> <p>7. Crear la Cabecera de Servicios a través del botón Guardar</p> <p>9. Seleccionar una cabecera ya creada.</p> <p>10. Editar la información de la cabecera de acuerdo al Usuario y Albergue.</p> <p>11. Dar clic en el botón Guardar.</p> <p>12. Cerrar sesión</p> <p>13. Ingresar al sistema con los datos creados.</p> <p>14. Cerrar sesión</p>	<p>8. Si no se ingresa algún dato solicitado en el formulario de creación de usuario aparece Mensaje de Confirmación o Error.</p>
Datos de Entrada:	Contraseña de Administrador de Albergue.
Resultados Esperados:	Después de ejecutar el paso 7, en modo CORRECTO validando y guardando los datos se puede controlar la cabecera de servicio.
Resultados Reales:	El Registro de cabecera de servicio es CORRECTO por lo tanto es almacenado en la base de datos de cabeceras de servicio.

Tabla 76

Prueba de Aceptación Gestionar Servicios Generales

Identificador:	HU-006
Historia de Usuario:	Gestionar Servicios Generales
Descripción de la Prueba:	La prueba consiste en administrar Servicios Generales por parte del Administrador de Albergue.
Propósito:	Verificar el servicio de Servicios Generales..
Condición de Ejecución:	El usuario debe ingresar al sistema autenticado como Administrador de Albergue. El usuario Administrador de Albergue debe ingresar el detalle de Servicios Generales.



Dependencias:	
Acciones	
Acción del Actor:	Respuesta del Sistema:
<ol style="list-style-type: none"> 1. Ingresar como Administrador de Albergue. 2. Elegir Servicios -> Servicios Generales 3. Presionar el botón Nuevo Servicio ubicado en la parte inferior izquierda. 5. Ingresar los datos solicitados del Formulario. 6. Verificar el ingreso de datos en todos los campos obligatorios para la creación de una nuevo Servicio. 7. Crear el Servicio General a través del botón Guardar 9. Seleccionar un servicio ya creado. 10. Editar la información del servicio de acuerdo al servicio deseado 11. Dar clic en el botón Guardar. 12. Cerrar sesión 13. Ingresar al sistema con los datos de servicios creados. 14. Cerrar sesión 	<ol style="list-style-type: none"> 4. Aparece el controlador para ingresar Nuevo servicio. 8. Si no se ingresa algún dato solicitado en el formulario de creación de servicio aparece Mensaje de Confirmación o Error.
Datos de Entrada:	Contraseña de Administrador de Albergue. .
Resultados Esperados:	Después de ejecutar el paso 7, en modo CORRECTO validando y guardando los datos se puede controlar el servicio ingresado.
Resultados Reales:	El Registro de servicio es CORRECTO por lo tanto es almacenado en la base de datos de servicios generales.

Tabla 77*Prueba de Aceptación Gestionar Damnificados*

Identificador:	HU-007
Historia de Usuario:	Gestionar Damnificados
Descripción de la Prueba:	La prueba consiste en administrar Damnificados por parte del Administrador de Albergue.
Propósito:	Verificar el Proceso de Gestión de Damnificados.
Condición de Ejecución:	El usuario debe ingresar al sistema autenticado como Administrador General/ Administrador de Albergue El usuario Administrador de Albergue debe ingresar los Datos de Damnificados.
Dependencias:	
Acciones	
Acción del Actor:	Respuesta del Sistema:
<ol style="list-style-type: none"> 1. Ingresar como Administrador de Albergue. 2. Elegir Procesos -> Damnificados 3. Presionar el botón Crear Nuevo Damnificado ubicado en la parte inferior izquierda. 5. Ingresar los datos solicitados del Formulario. 6. Verificar el ingreso de datos en todos los campos obligatorios para la creación de un nuevo Damnificado. 7. Crear el Nuevo Damnificado a través del botón Guardar 9. Seleccionar un usuario Damnificado ya creado. 10. Editar la información del Damnificado de acuerdo al Damnificado deseado. 11. Dar clic en el botón Guardar. 	<ol style="list-style-type: none"> 4. Aparece el controlador para ingresar Nuevo Damnificado. 8. Si no se ingresa algún dato solicitado en el formulario de creación de Damnificado aparece Mensaje de Confirmación o Error.



12. Cerrar sesión	
13. Ingresar al sistema con los datos de usuario creado.	
14. Cerrar sesión	
Datos de Entrada:	Contraseña de Administrador de Albergue. .
Resultados Esperados:	Después de ejecutar el paso 7, en modo CORRECTO validando y guardando los datos se puede controlar el Damnificado ingresado.
Resultados Reales:	El Registro de Damnificado es CORRECTO por lo tanto es almacenado en la base de datos de Damnificados.

Tabla 78 Prueba de Aceptación Gestionar Personal Administrativo

Identificador:	HU-008
Nombre:	Gestionar Personal Administrativo
Descripción de la Prueba:	Proceso en el cual el Administrador General administra el Personal Administrativo del Sistema de albergues
Propósito:	Verificar el proceso de Control de Personal Administrativo.
Condición de Ejecución:	El usuario debe ingresar al Sitio autenticado como Administrador General
Dependencias:	
Acciones	
Acción del Actor:	Respuesta del Sistema:
1. Acceder autenticado como administrador General. 2. Seleccionar el Menú Procesos->Personal Administrativo del menú principal. 3. Presionar el botón Nuevo Personal Militar ubicado en la parte inferior izquierda. 4. Ingresar los datos solicitados del Formulario.	5. El sistema muestra el controlador para ingresar Nuevo Personal Administrativo



<p>6. Verificar el ingreso de datos en todos los campos obligatorios para la creación de un nuevo Personal Administrativo.</p> <p>7. Dar clic en guardar.</p> <p>9. Seleccionar un Personal Administrativo ya creado.</p> <p>10. Editar la información del Personal Administrativo seleccionado.</p> <p>11. Dar clic en el botón Guardar</p> <p>12. Cerrar sesión</p> <p>13. Ingresar al sistema con los datos de usuario creado.</p> <p>14. Cerrar sesión</p>	<p>8. Si no se ingresa algún dato solicitado en el formulario de creación de Personal Administrativo aparece Mensaje de Confirmación o Error solicitando algún dato requerido.</p>
<p>Datos de Entrada:</p>	<p>Contraseña de Administrador General. Ingresar los Datos solicitados en el Formulario de Personal Administrativo</p>
<p>Resultados Esperados:</p>	<p>Una vez que se ejecuta el paso 6, se valida los datos y se guarda. El sistema muestra un mensaje de confirmación de acciones realizadas siempre y cuando no haya ingresado un campo requerido posteriormente el usuario podrá modificar/eliminar/Crear nuevo Personal Administrativo.</p> <p>Si se desea eliminar un usuario Administrativo :</p> <p>a. El sistema muestra un mensaje de advertencia Si desea eliminar el usuario seleccionado.</p> <p>Caso contrario.</p> <p>b. El usuario mantiene su información en el sistema.</p>
<p>Resultados Reales:</p>	<p>Se visualiza la Información del Usuario en el Sistema La publicación del Usuario es CORRECTA por lo tanto puede utilizar Información el Administrador del Albergue.</p>

Tabla 79*Prueba de Aceptación Gestionar Mantenimiento*

Identificador:	HU-009
Nombre:	Control de Mantenimiento
Descripción de la Prueba:	Proceso en el cual el Administrador General Controla/Administra la Información para dar soporte a los Administradores de Albergue.
Propósito:	Verificar el proceso de Control de Mantenimiento de los Datos almacenados en el Sistema de Albergues.
Condición de Ejecución:	El usuario Administrador debe ingresar al Sistema autenticado como Administrador General

Dependencias:	
Acciones	
Acción del Actor:	Respuesta del Sistema:
<ol style="list-style-type: none"> 1. Acceder autenticado como administrador General. 2. Seleccionar el Menú Mantenimiento Del Menú principal. 3. Se despliega los ítems que se va a Controlar para dar Mantenimiento. 4. Seleccionar un ítem. 5. Presionar el botón Nuevo ubicado en la parte inferior izquierda del sistema de acuerdo al ítem seleccionado. 6. Ingresar los datos solicitados del Formulario del ítem de Mantenimiento. 8. Verificar el ingreso de datos en todos los campos obligatorios para la creación de un nuevo registro en el ítem de Mantenimiento. 9. Dar clic en guardar. 11. Cerrar sesión 	<ol style="list-style-type: none"> 7. El sistema muestra el controlador para ingresar los Datos en el Formulario. 10. Si no se ingresa algún dato solicitado en el formulario el sistema muestra Mensaje/s de Confirmación o Error solicitando algún dato requerido que no se ha ingresado.



12. Ingresar al sistema con los datos de usuario creado. 13. Cerrar sesión	
Datos de Entrada:	Contraseña de Administrador del Albergue.
Resultados Esperados:	Una vez que se ejecuta el paso 8, se valida los datos y se guarda. El sistema muestra un mensaje de confirmación de acciones realizadas siempre y cuando no haya ingresado un campo requerido posteriormente el usuario podrá modificar/eliminar/Crear nuevo registro del ítem de Mantenimiento seleccionado. Si se desea eliminar un registro creado en el ítem de Mantenimiento. a. El sistema muestra un mensaje de advertencia Si desea eliminar el registro ingresado del ítem de Mantenimiento. Caso contrario. b. El usuario mantiene su información en el sistema.
Resultados Reales:	Se visualiza la Información del registro de Mantenimiento del ítem seleccionado que se maneja en el sistema. La publicación del registro de Mantenimiento es CORRECTA por lo tanto puede utilizar la Información almacenada.

Tabla 80

Prueba de aceptación Gestionar Reportes

Identificador:	HU-010
Nombre:	Gestionar Reportes
Descripción de la Prueba:	Proceso en el cual el Administrador General Controla/Administra la Información de Reportes de acuerdo a la información almacenada en el Sistema.
Propósito:	Verificar el proceso de Generación de Reportes de los Datos almacenados en el Sistema de Albergues.
Condición de Ejecución:	El usuario Administrador debe ingresar al Sistema autenticado como Administrador General
Dependencias:	



Acciones	
Acción del Actor:	Respuesta del Sistema:
<p>1. Acceder autenticado como administrador General.</p> <p>2. Seleccionar el Menú Reportes del Menú principal.</p> <p>3. Se despliega los ítems que se va a Controlar para la Generación de Reportes.</p> <p>4. Seleccionar un ítem.</p> <p>5. Presionar el botón Crear Nuevo ubicado en la parte inferior izquierda del sistema de acuerdo al ítem seleccionado.</p> <p>6. Ingresar los datos solicitados del Formulario del ítem de Reportes.</p> <p>8. Verificar el ingreso de datos en todos los campos obligatorios para la creación de un nuevo Reporte disponible en el ítem del Menú Reportes.</p> <p>9. Dar clic en guardar</p> <p>11. Dar clic en el botón Guardar.</p> <p>12. Comprobar que el proceso se ha guardado</p> <p>13. Cerrar sesión.</p>	<p>7. El sistema muestra el controlador para ingresar los Datos en el Formulario para generar Reportes.</p> <p>10. Si no se ingresa algún dato solicitado en el formulario para la generación de Reportes (PDF/ EXCEL) se muestra Mensaje/s de Confirmación o Error solicitando algún dato requerido que no se ha ingresado.</p>
Datos de Entrada:	Contraseña de Administrador del Albergue.
Resultados Esperados:	<p>Una vez que se ejecuta el paso 8, se valida los datos y se guarda. El sistema muestra un mensaje de confirmación de acciones realizadas siempre y cuando no haya ingresado un campo requerido posteriormente el usuario podrá modificar/eliminar/Crear nuevo registro del ítem de Reportes seleccionado.</p> <p>Se puede generar el Reporte respectivo sea en formatos (PDF/EXCEL).</p> <p>Si se desea eliminar un registro creado en el ítem de Reportes.</p> <p>a. El sistema muestra un mensaje de advertencia Si</p>



	desea eliminar el registro ingresado del ítem de Reportes. Caso contrario. b. El usuario mantiene su información en el sistema.
Resultados Reales:	Se visualiza la Información del registro de Reporte del ítem seleccionado del Menú Reportes. La publicación del registro de Mantenimiento es CORRECTA por lo tanto puede utilizar la Información almacenada.

Tabla 81

Prueba de aceptación Gestionar Parte Diario

Identificador:	HU-011
Nombre:	Gestionar parte diario del Albergue
Descripción de la Prueba:	Proceso en el cual el Administrador General administra el Parte Diario del Albergue del Sistema de albergues.
Propósito:	Verificar el proceso de Control de parte diario del Albergue
Condición de Ejecución:	El usuario Administrador debe ingresar al Sistema autenticado como Administrador General
Dependencias:	
Acciones	
Acción del Actor:	Respuesta del Sistema:
<ol style="list-style-type: none"> 1. Acceder autenticado como administrador General. 2, Seleccionar el Menú Procesos->Parte Diario-> Crear Parte Diario del Menú principal. 3. Presionar el botón Crear Nuevo Parte ubicado en la parte inferior izquierda de la Página principal del sistema. 4. Ingresar los datos solicitados del Formulario del Parte Diario del Albergue. 6. Verificar el ingreso de datos en todos los campos obligatorios para la creación de un 	<ol style="list-style-type: none"> 5. El sistema muestra el controlador para ingresar Nuevo Parte Diario



nuevo Parte Diario. 7. Dar clic en guardar. 9. Seleccionar un proceso ya creado. 10. Editar la información del proceso de acuerdo a las necesidades del mismo. 11. Dar clic en el botón Guardar. 12. Comprobar que el proceso se ha guardado 13. Cerrar sesión	8. Si no se ingresa algún dato solicitado en el formulario de Nuevo Parte Diario del Albergue aparece Mensaje de Confirmación o Error solicitando algún dato requerido.
Datos de Entrada:	Contraseña de Administrador del Albergue.
Resultados Esperados:	Una vez que se ejecuta el paso 6, se valida los datos y se guarda. El sistema muestra un mensaje de confirmación de acciones realizadas siempre y cuando no haya ingresado un campo requerido posteriormente el usuario podrá modificar/eliminar/Crear nuevo Parte Diario del Albergue. Si se desea eliminar un Parte Diario ya creado: a. El sistema muestra un mensaje de advertencia Si desea eliminar el Parte Diario de Albergue seleccionado. Caso contrario. b. El usuario mantiene su información en el sistema.
Resultados Reales:	Se visualiza la Información del Parte Diario que se maneja en el sistema. La publicación del Parte Diario del Albergue es CORRECTA por lo tanto puede utilizar la Información del Parte Diario del Albergue.

Tabla 82

Prueba de aceptación Gestionar parte diario militar

Identificador:	HU-012
Nombre:	Gestionar parte diario militar
Descripción de la Prueba:	Proceso en el cual el Administrador General administra el Parte Diario Militar del Sistema de albergues.



Propósito:	Verificar el proceso de Control de parte diario militar
Condición de Ejecución:	El usuario Administrador debe ingresar al Sistema autenticado como Administrador General
Dependencias:	
Acciones	
Acción del Actor:	Respuesta del Sistema:
<ol style="list-style-type: none"> 1. Acceder autenticado como administrador General. 2, Seleccionar el Menú Procesos->Parte Diario-> Crear Parte Militar del Menú principal. 3. Presionar el botón Crear Nuevo Parte ubicado en la parte inferior izquierda de la Página principal del sistema. 4. Ingresar los datos solicitados del Formulario del Parte Diario Militar. 6. Verificar el ingreso de datos en todos los campos obligatorios para la creación de un nuevo Parte Diario. 7. Dar clic en guardar. 9. Seleccionar un proceso ya creado. 10. Editar la información del proceso de acuerdo a las necesidades del mismo. 11. Dar clic en el botón Guardar. 12. Comprobar que el proceso se ha guardado 13. Cerrar sesión 	<ol style="list-style-type: none"> 5. El sistema muestra el controlador para ingresar Nuevo Parte Diario 8. Si no se ingresa algún dato solicitado en el formulario de Nuevo Parte Diario Militar aparece Mensaje de Confirmación o Error solicitando algún dato requerido.
Datos de Entrada:	Contraseña de Administrador del Albergue.
Resultados Esperados:	Una vez que se ejecuta el paso 6, se valida los datos y se guarda. El sistema muestra un mensaje de confirmación de acciones realizadas siempre y cuando no haya



	<p>ingresado un campo requerido posteriormente el usuario podrá modificar/eliminar/Crear nuevo Parte Diario Militar.</p> <p>Si se desea eliminar un Parte Diario ya creado:</p> <p>a. El sistema muestra un mensaje de advertencia Si desea eliminar el Parte Diario seleccionado.</p> <p>Caso contrario.</p> <p>b. El usuario mantiene su información en el sistema.</p>
Resultados Reales:	<p>Se visualiza la Información del Parte Diario que se maneja en el sistema.</p> <p>La publicación del Parte Diario Militar es CORRECTA por lo tanto puede utilizar la Información del Parte Diario Militar.</p>

Tabla 83

Prueba de aceptación Gestionar parte diario de personas

Identificador:	HU-013
Nombre:	Gestionar parte diario de personas
Descripción de la Prueba:	Proceso en el cual el Administrador General administra el Parte Diario de Personas del Sistema de albergues.
Propósito:	Verificar el proceso de Control de parte diario de personas.
Condición de Ejecución:	El usuario Administrador debe ingresar al Sistema autenticado como Administrador General
Dependencias:	
Acciones	
Acción del Actor:	Respuesta del Sistema:
<ol style="list-style-type: none"> 1. Acceder autenticado como administrador General. 2, Seleccionar el Menú Procesos->Parte Diario-> Crear Parte Personas del Menú principal. 3. Presionar el botón Crear Nuevo Parte 	



<p>ubicado en la parte inferior izquierda de la Página principal del sistema.</p> <p>4. Ingresar los datos solicitados del Formulario del Parte Diario de Personas.</p> <p>6. Verificar el ingreso de datos en todos los campos obligatorios para la creación de un nuevo Parte Diario de Personas.</p> <p>7. Dar clic en guardar.</p> <p>9. Seleccionar un proceso ya creado.</p> <p>10. Editar la información del proceso de acuerdo a las necesidades del mismo.</p> <p>11. Dar clic en el botón Guardar.</p> <p>12. Comprobar que el proceso se ha guardado</p> <p>13. Cerrar sesión</p>	<p>5. El sistema muestra el controlador para ingresar Nuevo Parte Diario de Personas.</p> <p>8. Si no se ingresa algún dato solicitado en el formulario de Nuevo Parte Diario de Personas el sistema muestra Mensaje/s de Confirmación o Error solicitando algún dato requerido que no se ha ingresado.</p>
Datos de Entrada:	Contraseña de Administrador del Albergue.
Resultados Esperados:	<p>Una vez que se ejecuta el paso 6, se valida los datos y se guarda. El sistema muestra un mensaje de confirmación de acciones realizadas siempre y cuando no haya ingresado un campo requerido posteriormente el usuario podrá modificar/eliminar/Crear nuevo Parte Diario de Personas.</p> <p>Si se desea eliminar un Parte Diario de Personas ya creado:</p> <p>a. El sistema muestra un mensaje de advertencia Si desea eliminar el Parte Diario seleccionado.</p> <p>Caso contrario.</p> <p>b. El usuario mantiene su información en el sistema.</p>
Resultados Reales:	<p>Se visualiza la Información del Parte Diario de Personas que se maneja en el sistema.</p> <p>La publicación del Parte Diario de Personas es CORRECTA por lo tanto puede utilizar la Información almacenada.</p>



Tabla 84*Prueba de aceptación Gestionar Usuarios*

Identificador:	HU-014
Historia de Usuario:	Gestionar Usuarios
Descripción de la Prueba:	La prueba consiste en Gestionar Usuarios por parte del Administrador General.
Propósito:	Verificar el proceso de control de Usuarios por parte del Administrador en el sistema de albergues.
Condición de Ejecución:	El usuario debe ingresar al sistema autenticado como Administrador General. El usuario Administrador General debe ingresar al Sistema y controlar los Usuarios creados.
Dependencias:	El usuario escoge el usuario que desea modificar/eliminar./ Ver Reporte.
Acciones	
Acción del Actor:	Respuesta del Sistema:
<ol style="list-style-type: none"> 1. Ingresar como Administrador de Albergue. 2. Elegir Procesos ->Usuarios - >Crear Usuarios/Crear Roles-Usuario/Crear Roles/Cambiar Clave 3. Administrar Proceso 5. Ingresar los datos solicitados del Formulario. 6. Verificar el ingreso de datos en todos los campos obligatorios para la administración del proceso. 7. Crear el proceso solicitado a través del botón Guardar 9. Seleccionar un proceso ya creado. 10. Editar la información del proceso de 	<ol style="list-style-type: none"> 4. Aparece el controlador para Administrar el proceso seleccionado. 8. Si no se ingresa algún dato solicitado en el formulario de administración del proceso aparece Mensaje de Confirmación o Error en datos requeridos.



<p>acuerdo a las necesidades del mismo.</p> <p>11. Dar clic en el botón Guardar.</p> <p>12. Comprobar que el proceso se ha guardado.</p> <p>13. Cerrar sesión</p>	
Datos de Entrada:	Contraseña de Administrador de Albergue.
Resultados Esperados:	<p>Después de ejecutar el paso 7, en modo CORRECTO validando y guardando los datos se puede Administrar el Usuario.</p> <p>En la ejecución del paso 12, previo a la asignación del proceso, podrá ser modificado de acuerdo a la necesidad del Administrador.</p>
Resultados Reales:	El control del proceso es CORRECTO por lo tanto es almacenado en la base de datos de Usuarios y puede ser utilizado por el Administrador.

Tabla 85

Prueba de aceptación Crear Usuarios

Identificador:	HU-015
Historia de Usuario:	Crear Usuarios
Descripción de la Prueba:	La prueba consiste en Crear Usuarios por parte del Administrador General
Propósito:	Verificar el proceso de creación de usuarios por parte del Administrador en el sistema de albergues.
Condición de Ejecución:	<p>El usuario debe ingresar al sistema autenticado como Administrador General.</p> <p>El usuario Administrador General debe ingresar sistema y controlar los usuarios creados.</p>
Dependencias:	El usuario Administrador General crea un Nuevo usuario para la Administración de Albergue asignado. La información de cada Administrador de Albergue , solo puede ser vista por el Administrador General
Acciones	
Acción del Actor:	Respuesta del Sistema:



<ol style="list-style-type: none"> 1. Ingresar como Administrador General. 2. Elegir Procesos ->Usuarios - >Crear Usuario 3. Administrar Proceso 5. Ingresar los datos solicitados del Formulario. 6. Verificar el ingreso de datos en todos los campos obligatorios para la administración del proceso. 7. Crear el proceso solicitado a través del botón Guardar 9. Seleccionar un proceso ya creado. 10. Editar la información del proceso de acuerdo a las necesidades del mismo. 11. Dar clic en el botón Guardar. 12. Comprobar que el proceso se ha guardado. 13. Cerrar sesión 	<ol style="list-style-type: none"> 4. Aparece el controlador para Administrar el proceso seleccionado. 8. Si no se ingresa algún dato solicitado en el formulario de administración del proceso aparece Mensaje de Confirmación o Error en datos requeridos.
Datos de Entrada:	Contraseña de Administrador de Albergue.
Resultados Esperados:	<p>Después de ejecutar el paso 7, en modo CORRECTO validando y guardando los datos se puede Administrar el Usuario creado.</p> <p>En la ejecución del paso 12, previo a la asignación del proceso , podrá ser modificado de acuerdo a la necesidad del Administrador.</p>
Resultados Reales:	El control del proceso es CORRECTO por lo tanto es almacenado en la base de datos de Albergues y puede ser utilizado por el Administrador.

Tabla 86

Prueba de aceptación Crear Roles de Usuario

Identificador:	HU-016
Historia de Usuario:	Crear Roles de Usuarios
Descripción de la Prueba:	La prueba consiste en Crear Roles de Usuarios por parte del Administrador General



Propósito:	Verificar el proceso de creación de roles de usuarios por parte del Administrador en el sistema de albergues.
Condición de Ejecución:	El usuario debe ingresar al sistema autenticado como Administrador General. El usuario Administrador General debe ingresar sistema y controlar los roles de usuarios creados.
Dependencias:	El usuario Administrador General crea un Nuevo rol de usuario para la Administración de Usuarios.
Acciones	
Acción del Actor:	Respuesta del Sistema:
<ol style="list-style-type: none"> 1. Ingresar como Administrador General. 2. Elegir Procesos ->Usuarios - >Crear Roles de Usuario 3. Administrar Proceso 5. Ingresar los datos solicitados del Formulario. 6. Verificar el ingreso de datos en todos los campos obligatorios para la administración del proceso. 7. Crear el proceso solicitado a través del botón Guardar 9. Seleccionar un Rol de Usuario ya creado. 10. Editar la información del proceso de acuerdo a las necesidades del mismo. 11. Dar clic en el botón Guardar. 12. Comprobar que el proceso se ha guardado. 13. Cerrar sesión 	<ol style="list-style-type: none"> 4. Aparece el controlador para Administrar el proceso seleccionado. 8. Si no se ingresa algún dato solicitado en el formulario de administración del proceso aparece Mensaje de Confirmación o Error en datos requeridos.
Datos de Entrada:	Contraseña de Administrador de Albergue.
Resultados Esperados:	Después de ejecutar el paso 7, en modo CORRECTO validando y guardando los datos se puede Administrar el Rol de Usuario creado. En la ejecución del paso 12, previo a la asignación del proceso, el proceso podrá ser modificado de acuerdo a la necesidad del Administrador.



Resultados Reales:	El control del proceso es CORRECTO por lo tanto es almacenado en la base de datos de Roles de Usuario y puede ser utilizado por el Administrador.
---------------------------	---

Tabla 87

Prueba de aceptación Crear Roles

Identificador:	HU-017
Historia de Usuario:	Crear Roles
Descripción de la Prueba:	La prueba consiste en Crear Roles por parte del Administrador General
Propósito:	Verificar el proceso de creación de roles por parte del Administrador en el sistema de albergues.
Condición de Ejecución:	El usuario debe ingresar al sistema autenticado como Administrador General. El usuario Administrador General debe ingresar sistema y controlar los roles creados.
Dependencias:	El usuario Administrador General crea un Nuevo rol para la Administración de Usuarios.
Acciones	
Acción del Actor:	Respuesta del Sistema:
<ol style="list-style-type: none"> 1. Ingresar como Administrador General. 2. Elegir Procesos ->Usuarios - >Crear Roles 3. Administrar Proceso 5. Ingresar los datos solicitados del Formulario. 6. Verificar el ingreso de datos en todos los campos obligatorios para la administración del proceso. 7. Crear el proceso solicitado a través del botón Guardar 9. Seleccionar un Rol ya creado. 10. Editar la información del proceso de acuerdo a las necesidades del mismo. 11. Dar clic en el botón Guardar. 12. Comprobar que el proceso se ha 	<ol style="list-style-type: none"> 4. Aparece el controlador para Administrar el proceso seleccionado. 8. Si no se ingresa algún dato solicitado en el formulario de administración del proceso aparece Mensaje de Confirmación o Error en datos requeridos.



guardado. 13. Cerrar sesión	
Datos de Entrada:	Contraseña de Administrador de Albergue.
Resultados Esperados:	Después de ejecutar el paso 7, en modo CORRECTO validando y guardando los datos se puede Administrar el Rol creado. En la ejecución del paso 12, previo a la asignación del rol, el proceso podrá ser modificado de acuerdo a la necesidad del Administrador.
Resultados Reales:	El control del rol es CORRECTO por lo tanto es almacenado en la base de datos de Roles y puede ser utilizado por el Administrador.

Tabla 88

Prueba de aceptación Cambiar Clave

Identificador:	HU-018
Historia de Usuario:	Cambiar Clave
Descripción de la Prueba:	La prueba consiste en Cambiar Clave por parte del Administrador General
Propósito:	Verificar el proceso de cambio de clave por parte del Administrador en el sistema de albergues.
Condición de Ejecución:	El usuario debe ingresar al sistema autenticado como Administrador General. El usuario Administrador General debe ingresar sistema y controlar la asignación de usuarios ya creados.
Dependencias:	El usuario Administrador General crea un Nuevo Usuario para la Administración de Albergues. La clave asignada será cambiada.
Acciones	
Acción del Actor:	Respuesta del Sistema:
1. Ingresar como Administrador General. 2. Elegir Procesos ->Usuarios ->Cambiar Clave	



<p>3. Administrar Proceso</p> <p>5. Ingresar los datos solicitados del Formulario.</p> <p>6. Verificar el ingreso de datos en todos los campos obligatorios para la administración del proceso.</p> <p>7. Crear el proceso solicitado a través del botón Guardar</p> <p>9. Seleccionar un Usuario ya creado.</p> <p>10. Editar la información de Cambiar Clave reemplazada por una nueva clave.</p> <p>11. Dar clic en el botón Guardar.</p> <p>12. Comprobar que el proceso se ha guardado/ validado.</p> <p>13. Cerrar sesión</p>	<p>4. Aparece el controlador para Administrar el proceso seleccionado.</p> <p>8. Si no se ingresa algún dato solicitado en el formulario de administración del proceso aparece Mensaje de Confirmación o Error en datos requeridos.</p>
Datos de Entrada:	Contraseña de Administrador de Albergue.
Resultados Esperados:	<p>Después de ejecutar el paso 7, en modo CORRECTO validando y guardando los datos se puede Administrar el Rol creado.</p> <p>En la ejecución del paso 12, previo a la asignación del usuario, el usuario podrá ingresar con la nueva clave.</p>
Resultados Reales:	El control de Cambio de Clave es CORRECTO por lo tanto es almacenado en la base de datos de Usuarios y puede ser utilizado por el Administrador.

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

- ✓ Existen actualmente gran cantidad de librerías de componentes para JSF la utilización de una u otra librería de componentes o framework, dependerá en gran medida del rendimiento, eficiencia y calidad acordes a la funcionalidad que nos ofrezca, tomando en cuenta factores como la estabilidad en los cambios de versión, el disponer de una buena documentación y la utilización e integración con otros estándares.

- ✓ Para el Desarrollo del proyecto se utilizó la Metodología Xp, mediante la determinación de requisitos en las Historias de usuarios se pudo entender claramente lo que el usuario realmente necesita revisando los avances acordes al tiempo de desarrollo lo que permitió dar solución de manera clara y precisa de acuerdo a las reuniones establecidas en los horarios planificados.

- ✓ Con el levantamiento de requisitos por medio de las Historias de usuarios se pudo planificar la arquitectura acerca de cómo va a interactuar el usuario con la aplicación.

- ✓ Las pruebas de instalación y funcionamiento ayudaron a validar que la aplicación cumpla con las funciones escritas en las Historias de usuario.

5.2 RECOMENDACIONES

- ✓ Se recomienda estudiar, analizar, comparar librerías de componentes, antes de ser implementadas para considerar la elección de la librería que mayor se adapte al desarrollo de la aplicación.
- ✓ Se recomienda siempre recopilar información de los usuarios al momento de obtener los requerimientos del sistema, ya que de esta manera se facilita la comprensión y documentación de los mismos.
- ✓ Para no tener inconvenientes al momento de seleccionar componentes Jsf para ser instalados durante el desarrollo del proyecto, se recomienda escoger los que tengan mayores comentarios positivos en los foros y con mayores puntuaciones de aceptación, ya que estos son buenos indicadores de estabilidad y funcionalidad y rendimiento, que cuente con una amplia documentación.

5.3 TRABAJO FUTURO

En relación a las aportaciones de investigación que se podrían dar en el futuro, está la utilización del framework de presentación Bootstrap, el mismo que podría ser utilizado en la creación de páginas web orientadas a sitios de desastres naturales como: catástrofes, inundaciones, terremotos, o todo lo concerniente a brindar ayuda social, de modo que se utilice los recursos del framework, que incluye un conjunto de hojas de estilo (archivos .css) y JavaScript (archivos .js), barras de título, columnas, y una gran variedad de componentes, que pueden ser útiles en interacción con cualquiera de los lenguajes de programación ya sea Python, Node.js, Angular.js, Php con la finalidad de lograr la creación de sitios bien diseñados y funcionales, utilizando hibernate como medio de persistencia, dada las características que este orm posee.

BIBLIOGRAFÍA

- Adecco. (12 de 04 de 2018). Obtenido de <https://www.adeccorientaempleo.com/los-lenguajes-de-programacion-mas-demandados-en-2018/>
- Ákerblom, B. (2018). XP & Scrum.
- Beck, Kent., Fowler, Martin. (2004). *Planning Extreme Programming*. Addison- Wesley Professional.
- Bertram, D. (1 de 10 de 2018). *Likert Scales*. Obtenido de <http://poincare.matf.bg.ac.rs/~kristina/topic-dane-likert.pdf>
- Blasco, J. (2018). *Implementación de un sistema de acceso seguro a web con JEE*. Valencia.
- BootsFaces. (13 de 01 de 2018). <https://www.bootsfaces.net/>. Obtenido de <https://www.bootsfaces.net/>: <https://www.bootsfaces.net/>
- Borja, Y. (2018). Metodología Ágil de Desarrollo de Software – XP.
- ButterFaces. (02 de 01 de 2018). <http://www.butterfaces.org/>. Obtenido de <http://www.butterfaces.org/>: <http://www.butterfaces.org/>
- Caballé, Santi and Ortega, Jose-Arturo and Camps, Josep-Maria and Barolli, Leonard and Kulla, Elis and Spaho, Evjola. (2014). A presentation Framework to Simplify the Development of Java EE application thin Clients. *Researchgate Conference*, 3-4.
- Caballe, Santi et al. (2014). A presentation framework to simplify the development of java ee application thin clients. *Researchgate conference*, 3-4.
- Correa, R. (2009). *Estrategia para la Implantación de Software Libre en la Administración Pública Central*. Quito.
- Costa, R. (02 de 09 de 2014). <http://www.numeroreal.com.br/jsf-java-server-faces/>. Obtenido de <http://www.numeroreal.com.br/jsf-java-server-faces/>: <http://www.numeroreal.com.br/jsf-java-server-faces/>
- DB-Engines. (01 de 02 de 2018). Obtenido de Comparativo de Motores de Base de Datos

Donell, Martin&Wallace,Mark&McGuckin,Tony. (2011). *Matering XPages A step by step guide to Xpages Application Development and XSP Languge*. Mexico: IBM Press.

Eclipse. (01 de 01 de 2019). Obtenido de <https://www.eclipse.org/documentation/>

ESPE. (13 de 07 de 2017). <http://www.espe.edu.ec>. Obtenido de <http://www.espe.edu.ec>

Foces, J. (14 de 01 de 2009). <https://www.researchgate.net>. Obtenido de <https://www.researchgate.net>:
https://www.researchgate.net/profile/Jose_Foces_Moran/publication/43668513_Framework_para_la_capa_de_presentacion_de_aplicaciones_web/links/00b7d526f875195515000000/Framework-para-la-capade-presentacion-de-aplicaciones-web.pdf

Gupta, A. (01 de 01 de 2016). *Java EE 7 Essentials*. Oreilly. Obtenido de <https://docs.oracle.com/javase/7/docs/api/>

Gutiérrez, J. (10 de 06 de 2015). *Framework Web*. Obtenido de Framework Web: http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf

Gutierrez, J. (10 de 11 de 2017).
http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf. Obtenido de http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf:
http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf

Harshad B, V. (2014). *High Quality Web-Application Development on Java EE Platform*. Vipul Dabhi, 8.

INEN NTE INEN-ISO/IEC 25000, 2. (15 de 03 de 2014). Obtenido de <https://www.sis.se/api/document/preview/917156/>

Innovent-Solutions. (01 de 02 de 2018). Obtenido de <http://www.innoventsolutions.com/comparison-matrix.html>

ISO/IEC/IEEE 29148. (01 de 12 de 2011). Obtenido de https://webstore.iec.ch/preview/info_isoiecieee29148%7Bed1.0%7Den.pdf

Isotools. (2015 de 03 de 2015). <https://www.isotools.org>. Obtenido de <https://www.isotools.org/2015/03/19/que>.

J. J. Gutiérrez, M. J. (01 de 01 de 2018). *Pruebas del Sistema en Programación Extrema (Xp)*. Obtenido de http://www.lsi.us.es/~javierj/investigacion_ficheros/PSISEXTREMA.pdf

- JasperReports. (01 de 01 de 2019). <https://community.jaspersoft.com>. Obtenido de <https://community.jaspersoft.com/project/jasperreports-library>
- JBoss. (29 de 07 de 2010). *RichFaces Developer Guide*. Obtenido de <https://docs.jboss.org/richfaces/>
- JCP. (01 de 01 de 2018). <https://jcp.org/>. Obtenido de <https://jcp.org/en/home/index>
- JEE7. (01 de 01 de 2014). Obtenido de <https://docs.oracle.com/javaee/7/tutorial/jsf-intro006.htm>
- Joskowicz, J. (2008). Reglas y Prácticas en eXtreme Programming .
- Kurniawan, B. (2013). *Java Server Faces Programming*. New York: McGraw-Hill.
- Lobo, S. (19 de 08 de 2019). Obtenido de <https://hub.packtpub.com/what-the-ieee-2018-programming-languages-survey-reveals-to-us/>
- Madeja. (01 de 12 de 2018). <http://www.juntadeandalucia.es>. Obtenido de <http://www.juntadeandalucia.es>:
<http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/101>
- Mann, K. (2005). *JSF in Action*. Greenwich: manning Publications Co.
- MariaDB. (01 de 01 de 2019). <https://mariadb.com>. Obtenido de <https://mariadb.com/kb/en/library/heidisql/>
- Mois, M. (07 de 12 de 2015). <http://enos.itcollege.ee>. Obtenido de <http://enos.itcollege.ee>:
<http://enos.itcollege.ee/~jpoial/allalaadimised/reading/JSF-2.0-Programming-Cookbook.pdf>
- Morcillo, E. (17 de 11 de 2010). *Opinion Tecnológica Web*. Obtenido de Opinion Tecnológica Web: <http://www.opiniontecnologica.com>
- Navarro, A., Fernandez, J., & Morales, J. (2013). Revisión de metodologías ágiles para el desarrollo de software. 39.
- norfipc.com*. (21 de 06 de 2017). Recuperado el 21 de 06 de 2018, de <https://norfipc.com/herramientas/como-saber-visitantes-diarios-sitio-web-internet.php>
- Notas&Codigo. (24 de 08 de 2017). Obtenido de <https://notasecodigos.wordpress.com/2017/08/24/biblioteca-de-componentes-jsf-para-interfaces-ricas/>

- OmniFaces. (02 de 01 de 2018). <http://showcase.omnifaces.org/>. Obtenido de <http://showcase.omnifaces.org/>: <http://showcase.omnifaces.org/>
- Openfaces. (2 de 01 de 2018). <http://openfaces.org>. Obtenido de <http://openfaces.org>: <http://openfaces.org/documentation/tagReference/o/tld-summary.html>
- OWASP, 2. (10 de 02 de 2019). <https://www.owasp.org/>. Obtenido de https://www.owasp.org/index.php/Main_Page
- Pantoja, Libardo & Pardo, Cesar. (01 de 01 de 2016). *Evaluando La Facilidad De Aprendizaje De Frameworks Mvc En El Desarrollo De Aplicaciones Web*. Obtenido de <http://hemeroteca.unad.edu.co>: <http://hemeroteca.unad.edu.co/index.php/publicaciones-e-investigacion/article/view/1592/1939>
- Pressman, R. (2010). *Ingeniería de Software Un enfoque práctico*. Mexico: Mc Graw-Hill.
- PrimeFaces. (02 de 01 de 2018). <https://www.primefaces.org/>. Obtenido de <https://www.primefaces.org/>: <https://www.primefaces.org/>
- Quora. (2018). Obtenido de <https://www.quora.com/What-is-the-best-IDE-for-Java-at-2018-Eclipse-or-IntelliJ-or-other>
- Quora. (02 de 01 de 2018). Obtenido de <https://www.quora.com/Which-one-is-the-better-Java-EE-application-server-Glassfish-or-JBOSS>
- Reddit. (01 de 02 de 2018). Obtenido de https://www.reddit.com/r/java/comments/2lzujk/should_i_learn_java_ee_7_or_spring_framework_4x/
- RichFaces. (02 de 01 de 2018). <http://richfaces.jboss.org/>. Obtenido de <http://richfaces.jboss.org/>: <http://richfaces.jboss.org/>
- Rivadeneira, F. (2007). *Breves Fundamentos sobre terremotos en Ecuador*. Quito: Cooperación Editora Nacional.
- Rosson, Carroll. (10 de 01 de 2002). <http://lsi.ugr.es>. Obtenido de <http://lsi.ugr.es>: <http://lsi.ugr.es/~mgea/workshops/coline02/Articulos/toni.pdf>
- Schalk. (2010). *Introduction to JavaServer Faces-the complete reference*. New York: McGraw Hill.
- SiteRankData. (05 de 12 de 2018). Obtenido de <https://siterankdata.com/butterfaces.org>

SiteRankData. (30 de 01 de 2019). Obtenido de <https://siterankdata.com/primefaces.org>

SiteRankData. (28 de 01 de 2019). Obtenido de <https://siterankdata.com/bootsfaces.net>

SiteRankData. (02 de 01 de 2019). Obtenido de <https://siterankdata.com/openfaces.org>

SiteRankData. (30 de 01 de 2019). Obtenido de <https://siterankdata.com/omnifaces.org>

SiteRankData. (29 de 01 de 2019). Obtenido de <https://siterankdata.com/richfaces.org>

Spring. (01 de 02 de 2018). <https://spring.io>. Obtenido de
<https://spring.io/projects/spring-security>

Trenkamp et al. (2002). Wide plate margin deformation, southern Central America and northwestern South America, CASA GPS observations. *Journal of south American Earth Sciences*, 159.

Vera, D. (2017). *Reconstruyendo las cifras luego del sismo - Memorias*. Quito: INEC.

Wildfly. (01 de 01 de 2018). <https://docs.wildfly.org/>. Obtenido de <https://docs.wildfly.org/>

ANEXOS

ANEXO A: MANUAL DE USUARIO

ANEXO B: MANUAL TÉCNICO