



“Módulo web para la gestión de documentos institucionales en el centro de doctrina militar de la fuerza terrestre”

Morocho Benítez, Edwin David

Carrera de Tecnología en Computación

Departamento de Electrónica y Computación

Monografía, previo a la obtención del título de Tecnólogo en Computación

Ing. Caiza Caizabuano, José Rubén

02 septiembre del 2020



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE ELECTRÓNICA Y COMPUTACIÓN

CARRERA DE TECNOLOGÍA EN COMPUTACIÓN

CERTIFICACIÓN

Certifico que el trabajo de titulación, “**MÓDULO WEB PARA LA GESTIÓN DE DOCUMENTOS INSTITUCIONALES EN EL CENTRO DE DOCTRINA MILITAR DE LA FUERZA TERRESTRE**”, fue realizado por el Señor **MOROCHO BENÍTEZ, EDWIN DAVID**, el mismo que ha sido revisado y analizado en su totalidad por la herramienta de verificación de similitud de contenido; por lo tanto cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, razón por la cual me permito acreditar y autorizar para que lo sustenten públicamente.

Latacunga, 08 de septiembre de 2020

Una firma manuscrita en tinta azul que parece leer 'José Rubén Caiza Caizabano'.

Ing. Caiza Caizabano, José Rubén

C.C.: 050265429-6

DIRECTOR DE MONOGRAFÍA

REPORTE DE VERIFICACIÓN

Document Information

Analyzed document MonografíaMorochoV1.docx (D78746158)
 Submitted 9/7/2020 11:45:00 PM
 Submitted by
 Submitter email jrcaiza@espe.edu.ec
 Similarity 6%
 Analysis address jrcaiza.espe@analysis.orkund.com

Sources included in the report

SA	Universidad de las Fuerzas Armadas ESPE / MonografíaCuevaFranklinV2.docx Document MonografíaCuevaFranklinV2.docx (D78466103) Submitted by: jrcaiza@espe.edu.ec Receiver: jrcaiza.espe@analysis.orkund.com		9
SA	TESIS FARFAN-GUAMANQUISPE (1).docx Document TESIS FARFAN-GUAMANQUISPE (1).docx (D64974815)		3
SA	2018-05-04-Cifuentes-Mosquera.docx Document 2018-05-04-Cifuentes-Mosquera.docx (D38309146)		1
SA	Universidad de las Fuerzas Armadas ESPE / Pry Makrotech .docx Document Pry Makrotech .docx (D60999889) Submitted by: wwchimbrazo@espe.edu.ec Receiver: jrcaiza.espe@analysis.orkund.com		2
W	URL: https://core.ac.uk/download/pdf/200328874.pdf Fetched: 9/7/2020 12:15:42 AM		2
SA	TesisFinal - Urkund .docx Document TesisFinal - Urkund .docx (D55304360)		1
SA	2018-05-11-cifuentes-mosquera.docx Document 2018-05-11-cifuentes-mosquera.docx (D38559800)		2



JOSE
RUBEN
CAIZA
CAIZABUAN
O

Firmado digitalmente
por JOSE RUBEN CAIZA
CAIZABUANO
DN: cn=JOSE RUBEN
CAIZA CAIZABUANO
o=EC LATAACUNGA
Motivo Soy el autor de este
documento
Ubicación:
Fecha: 2020-09-07
17:11:05:00



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE ELECTRÓNICA Y COMPUTACIÓN

CARRERA DE TECNOLOGÍA EN COMPUTACIÓN

AUTORÍA DE RESPONSABILIDAD

Yo, **MOROCHO BENÍTEZ EDWIN DAVID**, con cedula de ciudadanía N° **110516256-2**, declaro que contenido, ideas y criterios de la monografía **“MÓDULO WEB PARA LA GESTIÓN DE DOCUMENTOS INSTITUCIONALES EN EL CENTRO DE DOCTRINA MILITAR DE LA FUERZA TERRESTRE”**, es de mi autoría y responsabilidad, cumpliendo con los requisitos legales, teorías, científico, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Latacunga, 10 de septiembre de 2020

Una firma manuscrita en tinta azul que parece decir 'Edwin David Morocho Benítez'.

Morocho Benítez Edwin David

C.C.: 1105162562



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA
DEPARTAMENTO DE ELECTRONICA Y COMPUTACIÓN

CARRERA DE TECNOLOGÍA EN COMPUTACIÓN

AUTORIZACIÓN

Yo, **MOROCHO BENÍTEZ, EDWIN DAVID**, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el presente trabajo de titulación "**MÓDULO WEB PARA LA GESTIÓN DE DOCUMENTOS INSTITUCIONALES EN EL CENTRO DE DOCTRINA MILITAR DE LA FUERZA TERRESTRE**", en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

Latacunga, 10 de septiembre de 2020

Morocho Benítez Edwin David

C.C.: 1105162562

DEDICATORIA

El presente trabajo lo dedico a Dios por haberme dado la vida y permitirme haber llegado a este momento tan importante de mi carrera profesional, quien como guía estuvo presente en el caminar de mi vida, bendiciéndome y dándome fuerzas para continuar con mis metas trazadas sin desfallecer.

A mis padres, por haberme dado la vida y haberme apoyado en todo momento permitiéndome lograr culminar mi formación profesional. A la Virgencita del Cisne, por ser la luz que guía mi vida ya que siempre me ha brindado toda su protección en todo momento en mi existencia. A mi padre, a pesar que el no tuvo la oportunidad de estudiar siempre ha estado presente en todo momento sin importar las circunstancias quien me han inspirado para seguir con pie firme mi propósito el cual este logro alcanzado es por ustedes y para ustedes padres amados por ser mi razón de vivir.

Edwin David, Morocho Benítez

AGRADECIMIENTO

Agradezco inmensamente al Ejército Ecuatoriano y a la Universidad de las Fuerzas Armadas ESPE por brindarme la oportunidad de seguir construyendo mis conocimientos mismos se aplicarán en cada una de las Unidades Militares para la contribuir en el cumplimiento de la misión.

En este proyecto quiero reflejar mi sincero agradecimiento a mi familia y todas las personas que me brindaron su apoyo incondicional en esta etapa de mi vida, además, agradezco infinitamente a los señores docentes de la Unidad de Gestión de Tecnologías, que, durante este tiempo en las aulas, impartieron sus conocimientos y experiencia en bien de cada uno de nosotros, en especial a mi director del proyecto José Caiza, por guiarme y ayudarme con su asesoría de una manera muy profesional en la realización de mi proyecto de titulación.

Edwin David, Morocho Benítez

ÍNDICE DE CONTENIDOS

CARÁTULA	1
CERTIFICACIÓN	2
REPORTE DE VERIFICACIÓN	3
RESPONSABILIDAD DE AUTORÍA	5
AUTORIZACIÓN	5
DEDICATORIA	1
AGRADECIMIENTO	2
RESUMEN	11
ABSTRACT	12
CAPÍTULO I	13
1. PROYECTO TÉCNICO	13
1.1 Tema	13
1.2 Antecedentes	13
1.3 Planteamiento Del Problema	14
1.4 Justificación.....	15
1.5 Objetivo General	15
1.6 Objetivos Específicos.....	15
1.7 Alcance.....	16
CAPÍTULO II	17
2. MARCO TEÓRICO	17
2.1 Software	17
2.1.1. Licencias de Software	17

2.1.2.	Categorías de Software.....	18
2.1.3.	Software libre	18
2.1.4.	Software Open Source.....	18
2.1.5.	Software propietario	18
2.2.	Dominios de Aplicación del Software	19
2.2.1.	Software de Sistema.....	19
2.2.2.	Software de Aplicación:.....	20
2.2.3.	Software de Ingeniería Y Ciencias:	20
2.2.4.	Software Incrustado	20
2.2.5.	Software de línea de Productos.....	20
2.3.	Aplicación Web	21
2.4.	Software de inteligencia artificial	21
2.5.	Aplicación Web.....	21
2.5.1.	Introducción de Aplicación Web.....	22
2.5.2.	Tipos de Aplicaciones Web.....	23
2.6.	¿Qué es una API?	25
2.6.1.	API RES	26
2.6.2.	Importancia de las API'S.....	27
2.7.	Módulo Web y sus tipos.....	28
2.8.	Sistema de Gestión Documental	29
2.8.1.	Sistema de Información	30

2.9. Arquitectura y diseño de Sistema Web.....	33
2.10. Tipos de Lenguaje de Programación	35
2.10.1. Definición de Programa	35
2.10.2 Conceptos de lenguaje de programación	35
2.10.3. El Lenguaje Java	38
2.10.4. Java Enterprise Edition (Java EE).....	39
2.10.5. Arquitectura Java EE	40
2.10.6. Componentes de Java EE	41
2.11. Herramientas de Desarrollo	42
2.11.1. Servidor Web Apache Tomcat.....	42
2.12. Ubuntu Server.....	44
2.13. PostgreSQL	45
2.14. Apache Maven.....	46
2.15. Alfresco.....	46
2.16. Repositorios de Documentos	47
2.17. Entorno de Desarrollo Integrado	48
2.18. Apache Netbeans	49
2.19. Base de datos.....	49
2.19.1. SGBD	49
2.19.2. MySQL	50
2.20. Servidor de Aplicaciones Java EE	50

2.21. Metodologías Ágiles de Software.....	51
CAPÍTULO III.....	54
3. DESARROLLO	54
3.1. Scrum	54
3.1.1. Equipo de Scrum	55
3.1.2 Product Owner	55
3.1.3 Extreme Programming XP	56
3.2. Levantamientos de Requisitos.....	57
3.2.1. Visión del Proyecto	57
3.2.2. Identificación de Épicas	58
3.2.3. Historias de Usuarios	58
3.3. Roles	60
3.3.1. Ejecución de Scrum	60
3.3.1.1. Product Backlog.....	60
3.3.1.2. Pila del Sprint.....	61
3.3.1.3. Gráfico Burn Down	66
3.4. Pruebas	67
3.4.1. Pruebas de Aceptación de Usuario.....	68
3.5. Pruebas de módulo web	69
CAPÍTULO IV.....	76
4. COMPROBACIÓN DEL SOFTWARE.....	76
4.1. Verificación y Validación del Software.....	76

4.2. Implementación del Sistema.....	76
4.2.1. Pruebas Unitarias – Sistema Web	76
4.2.2. Pruebas de Integración – Sistema Web	77
4.2.3. Pruebas de Aceptación	78
4.2.4. Ejecución de Pruebas.....	79
4.3. Validación de la Aplicación Web.....	80
4.3.1. Introducción	80
4.3.2. Encuesta de Validación del Proyecto	80
4.3.3. Análisis de Resultados de la Aplicación Web.....	81
CAPÍTULO V	87
5. CONCLUSIONES Y RECOMENDACIONES	87
5.1 Conclusiones	87
5.2 Recomendaciones.....	87
REFERENCIAS BIBLIOGRÁFICAS	89
ANEXOS	92

ÍNDICE DE FIGURAS

Figura 1: Esquema básico de una aplicación Web.....	23
Figura 2: Lista de categorías de Aplicaciones Web de Softaculous	25
Figura 3: REST API Services	27
Figura 4: Arquitectura y diseño de un sistema web	34
Figura 5: Etapa MODELO-VISTA-CONTROLADOR	35
Figura 6: Programación fuente y máquina	37
Figura 7: Adaptación del programa para cada sistema operativo	39
Figura 8: Arquitectura multicapas en JAVA EE.....	41
Figura 9: Esquema del funcionamiento de un servidor web.....	43
Figura 10: Función de un Servidor en sus tres capas	44
Figura 11: Plataforma Alfresco y sus servicios	47
Figura 12: Componentes Básicos de un IDE	49
Figura 13: Servidor de aplicaciones Java	51
Figura 14: Etapas y procesos de las metodologías tradicionales	52
Figura 15: Cuadro Comparativo de las metodologías	53
Figura 16: Flujo para un Sprint.....	55
Figura 17: Ciclo de vida de la Metodología XP.....	578
Figura 18: Burn Down del Sprint 1	66
Figura 19: Burn Down del Sprint 2	66
Figura 20: Burn Down del Sprint 3	67
Figura 21: Burn Down del Sprint 4	67
Figura 22: Página de Inicio de Módulo.....	70
Figura 23: Prueba de Ingreso a la Herramienta Alfresco	70
Figura 24: Visualización de Actividades.....	71

Figura 25: Página Principal del Sitio Gestión de Documentos	72
Figura 26: Carpeta Principal de Documentos Digitales	73
Figura 27: Clasificación de los documentos digitales	73
Figura 28: Página de descarga y de versión de documentos	74
Figura 29: Opción de búsqueda dentro del documento.....	75
Figura 30: Propiedades y versión del documento.....	75
Figura 31: Prueba en el navegador Chrome.....	80
Figura 32: Implementación del Módulo Web.....	81
Figura 33: Interfaz del Sistema Web	82
Figura 34: Gestión de Documentos Institucionales	83
Figura 35: Gestión de disponibilidad de documentos digitales	84
Figura 36: Ingreso al Sistema.....	85
Figura 37: Comunicación de Administrador de Usuarios	85
Figura 38: Comunicación del módulo web con el usuario final	86

ÍNDICE DE TABLAS

Tabla 1: Historia de usuario 1. Vista de catálogos en triage.....	59
Tabla 2: Historia de Usuario 2. Evaluación triage de Manchester	59
Tabla 3: Roles de Scrum.....	60
Tabla 4: Estimación de tiempos de producto backlog.....	60
Tabla 5: Sprint 1	62
Tabla 6: Prueba de aceptación: Doctrina Militar	68
Tabla 7: Pruebas Unitarias Sistema Web.....	76
Tabla 8: Pruebas de Integración Sistema Web.....	77
Tabla 9: Prueba de Aceptación 1: Registro de Documentos Institucionales.....	78
Tabla 10: Prueba de aceptación 2: Visualización de Documentos Institucionales	78
Tabla 12: Prueba de Aceptación 3: Edición de Documentos Institucionales	79

RESUMEN

El Centro de doctrina militar "CEDE" necesita un módulo web que ayude a la gestión de documentos institucionales y a obtener conocimientos militares dentro de los principales reglamentos, manuales, notas de aula, legislación civil y militar. El módulo web está dirigido a Personal Militar desde el Grado de Soldado el mismo que será amigable, puesto que admite el manejo de información e interacción entre Administrador del Sistema y el Soldado mediante una interfaz gráfica (pantallas), diseñada con los siguientes lenguajes de programación HTML, CSS, PHP. El módulo web cuenta con la utilización de Responsive Web Desing, donde permite adaptarse a dispositivos móviles en función a la resolución de la Pantalla, se ha optado por la utilización de este método para que el personal militar también pueda acceder desde un dispositivo móvil, además de las herramientas de desarrollo, se utilizó los Servicios de Alfresco para la gestión de Documentos Instituciones. Un servidor de prueba como servidor web para cumplir con los requerimientos del usuario y finalmente se utilizó una metodología ágil como Scrum que trabaja en desarrollo incremental como estrategia que consiste en desarrollo de software especializado para dar respuesta a las debilidades del modelo tradicional en cascada. El software utilizado es gratuito y reduce los costos de desarrollo.

PALABRAS CLAVE:

- **MÓDULO WEB**
- **DISEÑO**
- **ALFRESCO**
- **METODOLOGÍA SCRUM**
-

ABSTRACT

The Centro de Doctrina Militar "CEDE" needs a web module that helps to manage institutional documents and to obtain military knowledge within the main regulations, manuals, classroom notes, civil and military legislation. The web module is aimed at Military Personnel from the Soldier's Degree which will be friendly, since it supports information management and interaction between System Administrator and the Soldier through a graphic interface (screens), designed with the following programming languages HTML, CSS, PHP. The web module has the use of Responsive Web Design, where it can be adapted to mobile devices depending on the resolution of the screen, we have chosen to use this method so that military personnel can also access from a mobile device, in addition to development tools, Alfresco Services were used for the management of institutional documents. A test server as a web server to meet user requirements and finally used an agile methodology such as Scrum that works in incremental development as a strategy that consists of specialized software development to address the weaknesses of the traditional model cascade. The software used is free and reduces development costs.

KEYWORDS:

- **WEB MODULE**
- **WEB DESIGN**
- **ALFRESCO**
- **SCRUM METHODOLOGY**

CAPÍTULO I

1. PROYECTO TÉCNICO

1.1 Tema

Módulo Web para la Gestión de documentos Institucionales en el Centro de Doctrina Militar de la Fuerza Terrestre”.

1.2 Antecedentes

Actualmente el uso de repositorios virtuales de documentación Institucional en el Comando de Educación y Doctrina de la Fuerza Terrestre es indispensable el cual se digitalizará todos los documentos que se trabajaría y tenga accesibilidad todo el personal militar desde el Sistema Integrado de la Fuerza Terrestre (SIFTE).

De acuerdo al trabajo de investigación presentado por Barahona Do Santo, Fernando, (2014), cuyo tema es: **“DISEÑO E IMPLEMENTACIÓN DE UN PORTAL WEB Y UN REPOSITORIO VIRTUAL DE CONTENIDO MULTIMEDIA ACADÉMICO PARA EL ÁREA DE INGENIERÍA ELECTRÓNICA Y REDES”**, mencionan que:

Las empresas presentan alta complejidad en la gestión de repositorios virtuales, si dicho control no es eficiente el mismo se exponen a pérdidas de tiempo al no contar con el material adecuado en el momento de gestionar dicha documentación, lo que puede acarrear el incremento de costos por la pérdida o deterioro de recursos no controlados.

De acuerdo al trabajo de investigación presentado por Hidalgo Nilve, Lorena Alexandra, (2016) cuyo tema es **IMPLEMENTACIÓN DE UN REPOSITORIO DIGITAL OPEN SOURCE PARA LA GESTIÓN DE RECURSOS MULTIMEDIA, EN LA UNIDAD EDUCATIVA MILENIO DEL CANTON GUANO**, expone que: “Con la implementación del repositorio digitales facilita el trabajo a los administradores de la unidad educativa y disponer de un reporte diario del centro educativo, así como la gestión de recursos multimedia”.

Por lo expuesto es fundamental que la Fuerza Terrestre dentro del sistema integrado disponga del módulo de doctrina militar para que este almacene y controle todo tipo de documento digital institucional aprobado por el escalón superior del Ejército.

1.3 Planteamiento Del Problema

El presente estudio parte de la necesidad de encontrar soluciones óptimas para la elaboración de software basados en estándares de calidad. La historia del ejército ecuatoriano va de la mano con la gesta imperecedera del 10 de agosto de 1809, cuando al albor de la libertad, nace el Ejército ecuatoriano, cuya labor en más de dos siglos ha contribuido indiscutiblemente a la edificación del Ecuador democrático y soberano.

El sistema integrado de la fuerza terrestre fue creado en el año 2002, con la finalidad de sistematizar los procesos que existen dentro del Ejército ecuatoriano.

Actualmente el sistema integrado de la fuerza terrestre no dispone del módulo de doctrina militar para almacenar los documentos digitales aprobados por el escalón superior de la fuerza terrestre y varios problemas que se detallan a continuación:

El sistema no dispone del módulo de doctrina militar que permita almacenar los documentos digitales y documentos Aprobados y publicados en la orden general por el escalón superior, no están disponibles para los miembros de la fuerza terrestre.

- Falta de confianza en la robustez del código.
- Inexistencia de indicadores de versión en la documentación institucional.

De no solucionarse lo antes mencionado el sistema Integrado de la Fuerza Terrestre no dispondrá del módulo de doctrina militar que permitirá el almacenamiento de la documentación oficial publicados en la orden general, el personal de la institución militar no tendría acceso a la documentación actualizada, afectando de manera directa al cumplimiento de la misión institucional.

Por lo tanto, es de vital importancia que en el Sistema Integrado de la Fuerza Terrestre se implemente el módulo de doctrina militar, que facilite la gestión documental al Comando de Educación y Doctrina permitiendo realizar el almacenamiento y las versiones de la documentación.

1.4 Justificación

Con el desarrollo del módulo de doctrina militar se busca mantener un adecuado almacenamiento y de las versiones de la documentación digitales aprobados por la Fuerza Terrestre. Teniendo como objetivo principal disponer de un módulo en el Sistema Integrado de la Fuerza Terrestre.

Los documentos alojados en el módulo de doctrina militar puedan estar disponibles para todo el personal de la institución. Dentro de la búsqueda de la documentación se podrá realizar por tipo, categoría y versión del documento, el cual contará con documentación actualizada y aprobada por el escalón superior.

Del presente trabajo de investigación se beneficiará de manera directa el personal de la Fuerza Terrestre de manera especial al equipo de desarrollo y miembros del centro generador de Doctrina Militar.

El personal militar tendrá acceso a la documentación oficial actualizada permitiendo adquirir nuevos conocimientos los mismos que contribuirán a un mejor cumplimiento de la misión Institucional.

1.5 Objetivo General

- Desarrollar un módulo para la gestión de documentos institucionales en el Centro de Doctrina Militar de la Fuerza Terrestre.

1.6 Objetivos Específicos

- Realizar la especificación de requerimientos del software.

- Implementar una metodología para el desarrollo del módulo y utilización de la herramienta Alfresco para la gestión de documentos institucionales.
- Realizar las respectivas pruebas de funcionalidad del módulo gestión documental de doctrina militar.

1.7 Alcance

El presente trabajo de investigación abarca el desarrollo de un módulo web para la gestión de documentos institucionales que será implementados como parte del Sistema Integrado de la Fuerza Terrestre (SIFTE).

Así mismo se espera levantar los requerimientos funcionales y no funcionales, de las necesidades que tenga el módulo en cuanto al comportamiento y la calidad del software. Para el modelamiento de la estructura de la base de datos se utilizará software libre, para la implementación del módulo de gestión de documentos se utilizará la herramienta Alfresco.

El módulo permitirá almacenar, versionar, visualizar, buscar y descargar la documentación actualizada aprobada por el escalón superior. Adicional servirá de fuente de información y consulta para todas aquellas personas relacionadas o interesadas en el tema.

CAPÍTULO II

2. MARCO TEÓRICO

2.1 Software

El software es el conjunto de componentes lógicos que no son tangibles, pero si muy necesarios para llevar a cabo una tarea específica en nuestro sistema, se menciona que el software se desarrolla, el software no se perjudica y el software puede ser construido a medida según los requerimientos del cliente (Moreno & Ramos, 2014).

Para operar adecuadamente, los programas requieren medios que permitan su funcionamiento, tales como las estructuras de datos que les permiten manipularlos y la documentación, que describe la operación de los mismos. Para referenciar a todo el conjunto se utiliza el termino Software, al que parte de un sistema de computación constituido por los programas, procedimientos, reglas informáticas, documentación y datos asociados, necesarios para ejecutar ciertas tareas. El vocablo software surge por oposición a la palabra hardware empleada para denominar a los componentes físicos de la computadora en un intento de señalar la intangibilidad del mismo (Brocca & Camasaqui, 2005).

2.1.1. Licencias de Software

Por lo general, todo software está fundamentado bajo una licencia, la misma establece todas las políticas que limitan el uso y distribución del software, de incumplir dichas políticas el creador podría iniciar un proceso legal en contra del usuario final que no respeto las políticas de la licencia bajo la cual se encontraba el software.

Al instalar, utilizar o copiar un producto de software bajo licencia, el usuario está aceptando las condiciones estipuladas en la misma y queda obligado por los términos de dicho contrato. Es importante, entonces, leer detenidamente la licencia de un programa para conocer las condiciones para su utilización. Debe tenerse en cuenta que algunas licencias son tan

restrictivas que incluso el producido con la herramienta en cuestión no es de libre disponibilidad, mientras que otras invitan a su modificación o duplicación (Brocca & Camasaqui, 2005).

2.1.2. Categorías de Software

El software está clasificado en tres grandes categorías basadas en las limitaciones que tienen cada software para el uso de los usuarios finales.

2.1.3. Software libre

Es el software que respeta la libertad de los usuarios y la comunidad. A grandes rasgos, significa que los usuarios tienen la libertad de ejecutar, copiar, distribuir, estudiar, modificar y mejorar el software, permitiendo de esta manera que los usuarios controlan el programa y lo que este hace. Para que un software sea libre debe cumplir con las 4 libertades esenciales que son: ejecutar, estudiar, redistribuir, distribuir (Arteaga , 2019).

2.1.4. Software Open Source

Originalmente, la expresión open sources (o código abierto) hacía referencia al software open source (OSS). El software open source es un código diseñado de manera que sea accesible al público: todos pueden ver, modificar y distribuir el código de la forma que consideren conveniente. El software open source se desarrolla de manera descentralizada y colaborativa, así que dependen de la revisión entre compañeros y la producción de la comunidad. Además, suele ser más económico, flexible y duradero que sus alternativas propietarias, ya que las encargadas de su desarrollo son las comunidades y no un solo autor o una sola empresa (Free Software Foundation, 2019).

2.1.5. Software propietario

Es cualquier programa informático en el que los usuarios tienen limitadas las posibilidades de usarlo, modificarlo o redistribuirlo siendo uno de los más conocidos pese a que su código

fuente no está disponible, con la finalidad de evitar que cualquier usuario pueda realizar cambios (EAE Business School, 2020).

Como el software propietario responde fundamentalmente a los intereses comerciales de quien lo desarrolla, es necesario adquirir las correspondientes licencias para usarlo dentro de la legalidad. Además, y por el mismo motivo, periódicamente se liberan al mercado nuevas versiones de los productos, que corrigen los defectos encontrados e incorporan nuevas funcionalidades. Son justamente estas nuevas presentaciones las que, tarde o temprano, obligan al usuario a migrar a las nuevas versiones actualizadas, con el consiguiente costo asociado. También debe tenerse en cuenta que, pasado cierto tiempo, la empresa comercializadora deja de dar soporte técnico al producto. Esta es también una frecuente causa de migración (Brocca & Camasaqui, 2005).

2.2. Dominios de Aplicación del Software

Debido a la gran variedad de software que, aparecido en los últimos tiempos, se ha categorizado tomando en cuenta el área al que está dirigido, permitiendo de esta manera determinar siete dominios de aplicación de software que son:

2.2.1. Software de Sistema

Permite mantener gran interacción con el hardware de la computadora, uso intensivo por parte de usuarios múltiples, operación concurrente que requiere la secuenciación, recursos compartidos y administración de un proceso sofisticado. Estructuras complejas de datos e interfaces externas múltiples (compiladores, editores de texto, software de redes, componentes de sistemas operativos), (Pressman, 2010).

2.2.2. Software de Aplicación:

Son programas aislados que resuelven una necesidad específica de negocios. Las aplicaciones en esta área procesan datos comerciales o técnicos en una forma que facilita las operaciones de negocios o la toma de decisiones administrativas o técnicas. Además de las aplicaciones convencionales de procesamiento de datos, este software se usa para controlar las funciones de negocio en tiempo real (Pressman, 2010).

2.2.3. Software de Ingeniería Y Ciencias:

La característica principal es que se encuentra formado por algoritmos "devoradores de números". Sin embargo, las aplicaciones modernas dentro del área de la ingeniería y las ciencias están abandonando los algoritmos numéricos convencionales. El diseño asistido por computadora, la simulación de sistemas y otras aplicaciones interactivas, han comenzado a hacerse en tiempo real e incluso han tomado características del software de sistemas (Pressman, 2010).

2.2.4. Software Incrustado

Se encuentra dentro de un producto o sistemas y se usa para implementar y controlar características y funciones para el usuario final y para el sistema. Las funciones de este software son limitadas y particulares o provee una capacidad significativa de funcionamiento y control (Pressman, 2010).

2.2.5. Software de línea de Productos

Este software proporciona de manera específica una capacidad para uso de diferentes consumidores, este software está disponible para un mercado en particular y también para mercados masivos de consumidores que manejan grandes cantidades de datos (Pressman, 2010).

2.3. Aplicación Web

A estas aplicaciones también se las conoce con el nombre de “webapps”, en su forma más sencilla, las webapps son poco más que un conjunto de archivos de hipertexto vinculados que muestran información con uso de texto y graficas limitadas. Sin embargo, desde que surgió la web 2.0, las webapps están evolucionando hacia ambientes de cómputo sofisticados que no solo proveen características aisladas, funciones de cómputo y contenido para los usuarios, sino que también están integradas con bases de datos corporativas y aplicaciones de negocios (Pressman, 2010).

2.4. Software de inteligencia artificial

Hace uso de algoritmos no numéricos para solucionar problemas complejos que no resultan fáciles tratar computacionalmente o con el análisis directo. Las aplicaciones en esta área incluyen robótica, sistemas expertos, reconocimiento de patrones (imagen y voz), redes neutrales artificiales, demostración de teoremas y juegos (Pressman, 2010).

2.5. Aplicación Web

En los primeros días de la red mundial (entre 1990 y 1995), los sitios web consistían en poco más que un conjunto de archivos de hipertexto vinculados que presentaban la información con el empleo de texto y graficas limitadas. Al pasar el tiempo, el aumento de HTML por medio de herramientas de desarrollo (XML, Java) permitió a los ingenieros de la web brindar capacidad de cómputo junto con contenido de información.

Habían nacido los sistemas y aplicaciones basados en la web (denominado a estas en forma colectiva como webapps). En la actualidad, las webapps se han convertido en herramientas sofisticadas de cómputo que no solo proporcionan funciones aisladas al usuario final, sino que también se han integrado con base de datos corporativas y aplicaciones de negocios, existen aplicaciones web: Estáticas, Dinámicas, Animadas, E-comerse, Portal-web y

aplicaciones web con Gestor de contenidos. La gran mayoría de las webapps tienen las siguientes propiedades (Pressman, 2010).

- Uso intensivo de redes
- Concurrencia
- Carga impredecible
- Rendimiento
- Disponibilidad
- Orientada a los datos
- Evolución continua
- Seguridad
- Estética

En las aplicaciones Web suelen distinguirse tres niveles (como en las arquitecturas cliente/ servidor de tres niveles): el nivel superior que interacciona con el usuario (el cliente web, normalmente un navegador), el nivel inferior que proporciona los datos (la base de datos) y el nivel intermedio que procesa los datos (el servidor web). El cual se describe el cliente y el servidor web y se comentan los entornos web en los que se ejecutan las aplicaciones web. También se describen las arquitecturas típicas de las aplicaciones web. Por último, se presenta una metodología de desarrollo de sitios web (Luján Mora , 2002).

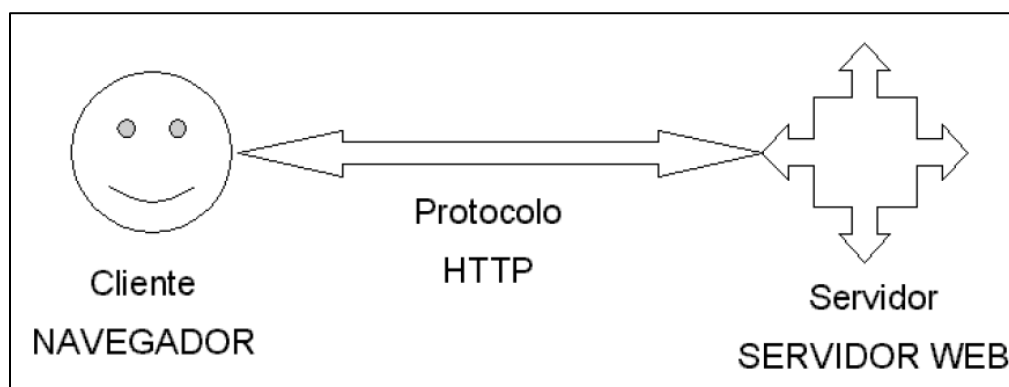
2.5.1. Introducción de Aplicación Web

Una aplicación web (web-based application) es un tipo de aplicación cliente/servidor, donde tanto el cliente (el navegador, explorador o visualizador) como el servidor (el servidor web) y el protocolo mediante el que se comunican (HTTP) están estandarizados y no han de ser creados por el programador de aplicaciones.

EL protocolo HTTP forma parte de la familia de protocolos de comunicación TCP/IP, que son los empleados de internet. Estos protocolos permiten la conexión de sistemas heterogéneos, lo que facilita el intercambio de información entre distintos ordenadores. HTTP sitúa en el nivel 7 (aplicación) del modelo OSI (Luján Mora , 2002).

Figura 1:

Esquema básico de una aplicación Web



Nota: La imagen describe el modelo básico de una aplicación web. Figura tomada de (Lujan-Moran,2002).

2.5.2. Tipos de Aplicaciones Web

Las aplicaciones Web presentan una serie de características que la convierten en productos software de gran valor, de entre ellas podemos destacar alguna como la capacidad de ser accesibles y operables desde cualquier plataforma y ubicación, puesto que al no ser necesario su descarga para una posterior instalación o configuración, son capaces de distribuirse con mayor facilidad q un elevado número de usuarios. Otra característica relevante podría ser el hecho de disponer siempre una versión actualizada de la aplicación, sin involucrar al usuario en tareas de actualización. Todas estas características han proporcionado que las aplicaciones web

hayan adaptadas actualmente como un elemento esencial e indispensable en toda actividad empresarial, intercambio de información y redes sociales (Hattori, 2007)

En la actualidad existen multitud de tipos de aplicaciones Web (también llamados familias o dominios de aplicaciones Web), que se pueden clasificar siguiendo diferentes criterios.

Hay clasificaciones que dividen sitios Web en multitud de categorías, basándose en los temas que tratan (como salud, noticias o deportes), que a su vez pueden contener otras categorías. Ejemplos de esto son los llamados directores Web, como son el Open Directory Project (Open Directory Project) y Yahoo! Directory (Yahoo Inc.). Otro caso es la compañía Alexa (Alexa Internet, Inc.), que clasifica los sitios Web por relevancias a través de las estadísticas de visitas y emplea un sistema de categorías similares (Hattori, 2007).

Otra posibilidad consiste en clasificar basándose en la motivación principal del sitio Web (Powell, Jones, &Cutts, 1998). Dado que un sitio Web puede tener varias metas y ser dividido a su vez en microsítios, la motivación principal no debe ser demasiado compleja.


























Se identifica y se clasifican aplicaciones Web automáticamente basándose en el análisis del tráfico HTTP. Para ello emplean una lista de 5 aplicaciones Web:

- Social networks
- Online news
- Email
- Photo sharing
- Online videos

Existen también en la nube servicios de instalación de aplicaciones Web que disponen de una amplia variedad de la misma. Softaculous (Softaculous Ltd.), ha confeccionado una interesante lista de categorías que puede verse a continuación.

Figura 2:

Lista de categorías de Aplicaciones Web de Softaculous

 Blogs (13) Wordpress, Serendipity, Pixie, Chyrp, b2 evolution...	 Micro Blogs (4) StatusNet, PageCookery, Sharetronic, Storytr.	 Portals/CMS (31) Joomla, Drupal, Zikula, Mambo, phpwcms, Contao...
 Forums (13) phpBB, Phorum, bbPress, UseBB, Vanilla, Mybb...	 Image Galleries (11) Coppermine, LinPHA, Gallery, ZenPhoto, Shutter...	 Wikis (5) MediaWiki, DokuWiki, PhpWiki, PmWiki, WikkaWiki.
 Social Networking (7) Elgg, Dolphin, Jcow, Oxwall, Etano, PeoplePods, Beatz.	 Ad Management (7) Noahs Classifieds, OpenX, Kamads, PHPads, OSClass...	 Calendars (4) WebCalendar, phpicalendar, phpSchedulit, ExtCalendar.
 Gaming (4) Shadows Rising, Multiplayer Checkers...	 Mails (7) phpList, poMMo, SquirrelMail, ccMail, Roundcube...	 Polls and Surveys (9) Lime Survey, phpESP, EasyPoll, Piwik, Piwigo...
 Project Management (12) dotProject, SugarCRM, Mound, Collabtive...	 E-Commerce (18) Zen Cart, osCommerce, Magento, WHMCS, Axis...	 ERP (8) Vtiger, OrangeHRM, BlueERP, Dolibarr, SimpleInvoices...
 Guest Books (7) phpBook, Lazarus, VX Guestbook, BellaBook...	 Customer Support (11) Trellis Desk, Crafty Syntax, osTicket, phpOnline...	 Frameworks (15) CodeIgniter, DIY, xAjax, Zend, yii, PRADO, Symfony...
 Educational (8) Moodle, Claroline, TCEExam, Omeka, eFront...	 DB Tools (5) phpMyAdmin, Adminer, MyWebSQL, Vty...	 Music (4) kPlaylist, Impleo, Podcast Generator, AmpJuke.
 Video (3) ClipBucket, VidScript, videoDB.	 RSS (4) Gregarius, SimplePie, Feed On Feeds, selfoss.	 Others (29) OpenRealty, PhpGedView, Arfoo, Seo Panel, OpenBiblio...
 JavaScript Libraries (21) jQuery, MooTools, ExtJS, MochiKit, Dojo, jqPlot...		

Nota: La imagen describe la Categoría de aplicaciones web de Softaculous. Figura tomada de (Hattori,2007).

2.6. ¿Qué es una API?

Según (Gonzalo Soto & Nuria, 2018) En términos más generales, una API es una Interfaz de Programación de Aplicaciones (Application Programming Interface), es decir, es un conjunto de funciones, procedimientos o clases que un sistema operativo, librería o servicio proporciona para soportar peticiones realizadas por un software. Permite que las funcionalidades de un servicio inicial puedan ser amplia a otras, pudiendo integrar otras APIs haciendo que el servicio/software sea extensible.

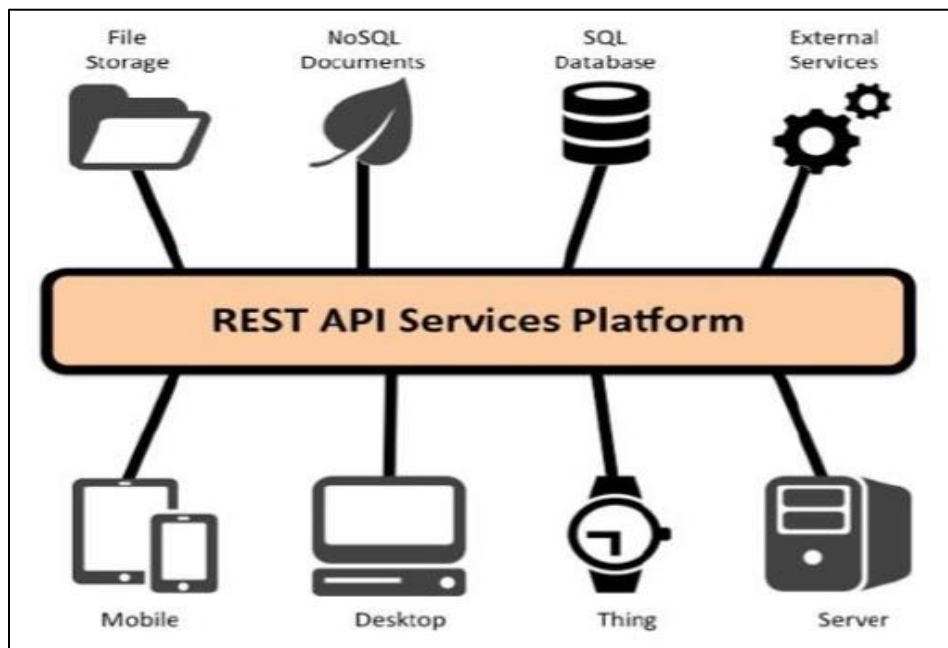
En conclusión, las APIs facilitan considerablemente la construcción de aplicaciones más complejas; un aspecto que estimula a una nueva generación de soluciones de tipo “quick fix” y aplicaciones cliente que operan en sistemas de mayor extensión.

Según (TutorialsPoint , 2019), SDLC es el proceso que se lleva a cabo para un proyecto de software seguido por los equipos de desarrollo de software, consiste en describir cómo se va a desarrollar, mantener, reemplazar o mejorar un software.

De igual se define la metodología a seguir para mejorar la calidad de software y el proceso de desarrollo general, en cada etapa se obtienen documentos que son las bases de partida de la etapa siguiente es decir no se puede comenzar la siguiente etapa sin haber terminado la anterior (Gonzalo Soto & Nuria, 2018).

2.6.1. API RES

Según (Gonzalo Soto & Nuria, 2018) Una API REST permite la utilización de servicios web disponibles a través de Internet utilizando el protocolo estándar HTTP; en otras palabras, una API REST es un servicio que al igual que el termino inicial de una API provee de funciones que dan capacidad de usar un servicio web de terceros dentro de nuestro servicio o aplicación propia de un modo seguro.

Figura 3:*REST API Services*

Nota: La imagen describe las funciones que dan la capacidad de usar un servicio web Figura tomada de página web integración de Google Maps (<https://developers.google.com/maps/?hl=es-419>).

Mediante un método GET se obtiene una respuesta a una consulta a una API REST. El contenido de las respuestas son datos en diferentes formatos como pueden ser: texto plano, ficheros XML, JSON, CSV, etc. Los servicios de una REST API son los intermediarios entre el contenedor de los datos con la parte.

2.6.2. Importancia de las API'S

Uno de los primeros puntos que hacen referencia al porque las APIs son importantes es la relevancia para el desarrollo de aplicaciones web y de las aplicaciones móviles de hoy en día. Gran parte del éxito que ha cosechado las aplicaciones móviles ya sea para iPhone como para

Android, es gracias a una potente API en el backend para realizar el manejo de los datos que a posteriori muestran estas aplicaciones.

Actualmente, se está hablando de una tercera revolución industrial y, por lo tanto, de un nuevo ecosistema económico. Esta nueva era está enfocada a la digitalización de los datos, cada persona genera una gran cantidad de información provocando transferencias intensivas de datos digitales. Los ejemplos más claros de la existencia de una necesidad de desarrollado de recursos informáticos de un modo rápido y ágil son las citadas APIs. A un nivel primario una API permite que exista una comunicación bidireccional entre los productos y los servicios; dependiendo de las características del desarrollo van a satisfacer diferentes necesidades como: el comercio electrónico, las redes sociales o los pagos en línea. (Gonzalo Soto & Nuria, 2018).

2.7. Módulo Web y sus tipos

Existen diferentes tipos de módulos dependiendo de su contenido y el contenedor donde se vaya a ejecutar. Los distintos módulos van empaquetados en un fichero JAR (Java Archive).

No obstante, la extensión de dicho fichero dependerá del tipo de modulo:

- **Módulo Web (Web Module).** - Contiene normalmente Java Servlets, JavaServer Pages (JSP), JavaServer Faces (JSF), contenidos estáticos como imágenes, HTMLs, CSSs la extensión del fichero empaquetado será WAR (Web Archive).
- **Módulo de EJBs (EJB Module).** - Como su nombre lo indica, contiene Enterprise JavaBeans (EJB). La extensión del fichero empaquetado es la de por defecto, JAR (Java Archive).

- **Modulo cliente (Application Client Module).** - Contiene el código de la aplicación cliente. La extensión del fichero empaquetado es la de por defecto, JAR (Java Archive).
- **Modulo Adaptador (Resource Adapter Modulo).** - Contiene un conector para comunicarse con un Sistema de Información Empresarial (EIS). La extensión del fichero empaquetado será RAR (Resource Archive).

La aplicación Java EE a su vez, también es un empaquetado de los distintos módulos que la forman. La extensión del fichero empaquetado será EAR (Enterprise Archive) (Cassá, 2012).

Un módulo web representa una aplicación web. El módulo web se crea ensamblando servlets, archivos JPS (JavaServer Pages) y contenido estático como paginas HTML (Hypertext Markup Language) en una sola unidad desplegable. Los módulos web se amacena en archivos WAR, que son archivos JAR (Java Archive) estándar (Center, 2006).

Un módulo contiene:

- Uno o varios servlets, archivos JSP y archivos HTML.
- Un descriptor y despliegue, almacenado en un archivo XML (Extensible Markup Language).
- El archivo, denominado web.xml, declara el contenido del módulo. Contiene información acerca de la estructura y las dependencias externas de los componentes web en el módulo y describe como se utilizan los componentes en el tiempo de ejecución. (Center, 2006)

2.8. Sistema de Gestión Documental

Los sistemas de gestión documental moderna, acorde con los niveles de desarrollo de las diferentes entidades del Distrito, de forma que los procesos, actividades y tareas de los sistemas de archivos, garantices que los documentos producidos por una entidad y sus dependencias se acoplen, procesen y conserven de tal forma que cuando sean requeridos se puedan buscar, localizar y poner al servicio de los usuarios en el menor tiempo posible (Zapata, 2012).

La industria informática, por motivos de marketing o, simplemente, por mal conocimiento del sector, ofrece como documentales soluciones que sirven perfectamente para gestionar documentos administrativos, pero no para gestionar documentación científico-técnica (Codina, 2013).

Por otro lado, en los departamentos de informática de las empresas, es frecuente que el personal informático desconozca el hecho de que existen programas especialmente diseñados para ese tipo de documentos y, normalmente, tienden a imponer aquellas soluciones que conocen bien, pero que no proporcionan a los documentalistas las herramientas adecuadas para su trabajo.

Sin embargo, tanto los profesionales de la información como la industria deberían estar interesados en no confundir estos dos tipos de sistemas, porque lo contrario conduce a los profesionales a errores de implantación; y a la industria, a errores de marketing (Codina, 2013).

2.8.1. Sistema de Información

El Sistema formal es para recabar, integrar, comprar, analizar y difundir información interna y externa de la empresa en forma oportuna, eficaz y eficiente.

2.8.1.1. Elementos de un sistema de información

Un sistema de información (SI) es un conjunto organizado de elementos, estos elementos son de 4 tipos:

- Personas.
- Datos.
- Actividades o técnicas de trabajo.
- Recursos materiales en general (típicamente recursos informáticos y de comunicación, aunque no tienen por qué ser de este tipo obligatoriamente).

Todo ese conjunto de elementos interactúa entre sí para procesar los datos y la información (incluyendo procesos manuales y automáticos) y distribuirla de la manera más adecuada posible en una determinada organización en función de sus objetivos. Normalmente el término es usado de manera errónea como sinónimo de sistema de información informático, estos son el campo de estudio de la tecnología de la información (IT), y aunque puedan formar parte de un sistema de información (como recurso material), por sí solos no se pueden considerar como sistemas de información, este concepto es más amplio que el de sistema de información informático (Pachano, 2010).

No obstante, un sistema de información puede estar basado en el uso de computadoras, según la definición de Langefors este tipo de sistemas son:

Un medio implementado tecnológicamente para grabar, almacenar y distribuir expresiones lingüísticas,

- Así como para extraer conclusiones a partir de dichas expresiones.

2.8.1.2. Tipos de sistemas de información

- a. Indexación.** - Consiste en poner identificadores únicos a los documentos de forma que sean más sencillos de localizar. Además de esto, cuando el volumen de documentos es grande conviene utilizar otras técnicas como categorización y agrupaciones de índices

para que se puedan hacer búsquedas en árbol y no sea necesario recorrer todos los elementos en una búsqueda. (Pachano, 2010).

- b. Almacenamiento.** - Almacenamiento de los documentos electrónicos, en una base de datos. Contemplará funcionalidad para definir cuánto tiempo se mantienen, cómo hacer migraciones a otro sistema de almacenamiento, sistema de copias de seguridad, para recuperar en caso de fallo del primer sistema.
- c. Recuperación.** - Herramientas para recuperar un determinado archivo almacenado. No siempre se conocerá el identificador único del archivo, por lo que la aplicación debería proporcionar la posibilidad de encontrarlo por título, u otros metadatos como tamaño, autor, etc., pudiendo hacer búsquedas más eficientes (Pachano, 2010).
- d. Bases De Datos.** - Un sistema gestor de bases de datos (SGBD) consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a dichos datos. La colección de datos, normalmente denominada base de datos, contiene información relevante para una empresa. El objetivo principal de un SGBD es proporcionar una forma de almacenar y recuperar la información de una base de datos de manera que sea tanto *práctica* como *eficiente*. Los sistemas de bases de datos se diseñan para gestionar grandes cantidades de información. La gestión de los datos implica tanto la definición de estructuras para almacenar la información como la provisión de mecanismos para la manipulación de la información.

Además, los sistemas de bases de datos deben proporcionar la fiabilidad de la información almacenada, a pesar de las caídas del sistema o los intentos de acceso sin autorización. Si los datos van a ser compartidos entre diversos usuarios, el sistema debe evitar posibles resultados anómalos.

Dado que la información es tan importante en la mayoría de las organizaciones, los científicos informáticos han desarrollado un amplio conjunto de conceptos y técnicas para la gestión de los datos. En este capítulo se presenta una breve introducción a los principios de los sistemas de bases de datos. (Silberschatz, Korth, & Sudarshan, 2002).

Los gestores de bases de datos permiten gestionar la información referida a personas o artículos, realizando operaciones de edición, ordenación, búsqueda, etc.

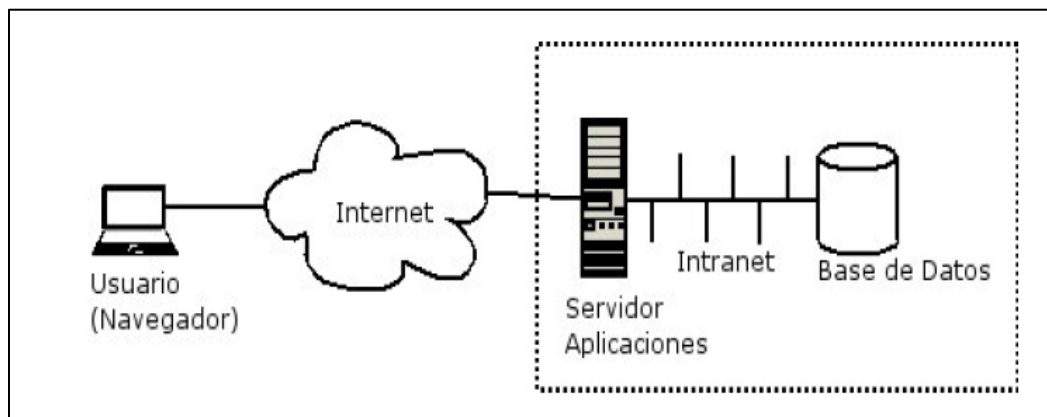
Los generadores de gráficos facilitan la creación de distintos tipos de gráficos, a partir de datos de hojas de cálculo o bases de datos, o directamente introducidos por el usuario. Existen muchas otras aplicaciones, de uso menos general, como son los programas matemáticos, estadísticos, de CAD (Diseño Asistido por Computador), aplicaciones contables y de gestión de empresas, comunicaciones, juegos, etc. (Cristóbal Pareja, 1994).

2.9. Arquitectura y diseño de Sistema Web

Según Garrido (2004), manifiesta que las aplicaciones web se han convertido en pocos años en complejos sistemas con interfaces de usuario cada vez más parecidas a las aplicaciones de escritorio, dando servicio a procesos de negocio de considerable envergadura y estableciéndose sobre ellas requisitos estrictos de accesibilidad y respuesta. Esto ha exigido reflexiones sobre la mejor arquitectura y las técnicas de diseño más adecuadas.

Figura 4:

Arquitectura y diseño de un sistema web



Nota: La imagen describe la arquitectura y diseño de un sistema web por (Garrido, 2004).

En todos los sistemas de este tipo y ortogonalmente a cada una de las capas de despliegue comentadas, podemos dividir la aplicación en tres áreas o niveles:

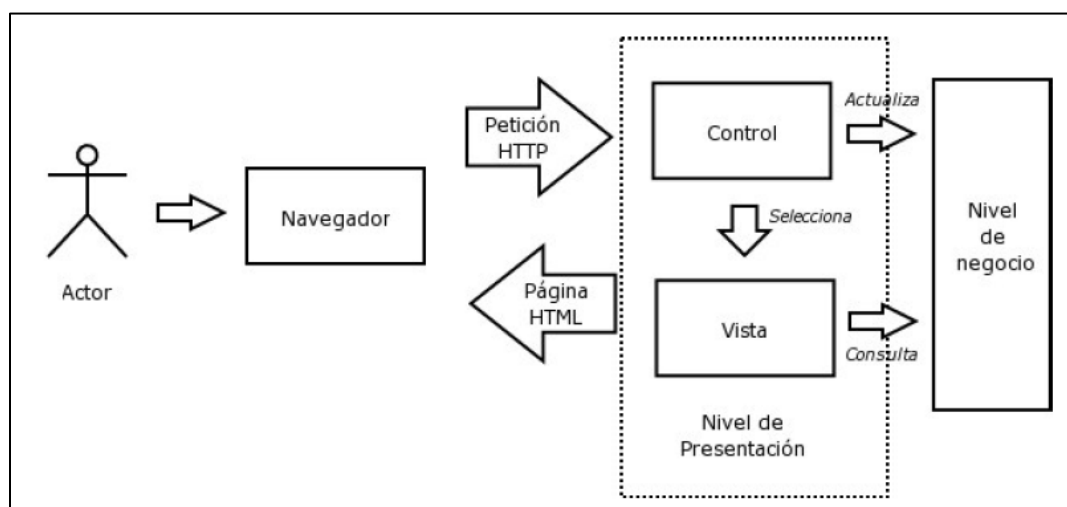
- **Nivel de presentación.** - Es el encargado de generar la interfaz de usuario en función de las acciones llevadas a cabo por el mismo.
- **Nivel de negocio.** - Contiene toda la lógica que modela los procesos de negocio y ese donde se realiza todo el procedimiento necesario para atender a las peticiones del usuario.
- **Nivel de administración de datos.** - Encargado de hacer persistencia toda la información, suministra y almacena información para el nivel de negocio.

Uno de los patrones que han demostrado ser fundamental a la hora de diseñar aplicaciones web es el **Modelo-Vista-Control (MVC)**. Este patrón propone la separación en distintos componentes de la interfaz de usuario (vistas), el modelo de negocio y la lógica de control. Una vista es una "fotografía" del modelo (o una parte del mismo) en un determinado momento. Un control recibe un evento disparado por el usuario a través de la interfaz.

En una aplicación web, las vistas serían las páginas HTML que el usuario visualiza en el navegador. A través de estas páginas el usuario interactúa con la aplicación, enviando eventos, que en función del evento concreto actúa sobre el modelo convenientemente. Los resultados de la acción se devuelven al usuario en forma de páginas HTML mediante la respuesta HTTP.

Figura 5:

Etapa MODELO-VISTA-CONTROLADOR



Nota: La imagen describe el modelo-vista-controlador (Garrido, 2004).

2.10. Tipos de Lenguaje de Programación

2.10.1. Definición de Programa

Un programa es una serie de ordenes o instrucciones ordenadas con una finalidad concreta que realizan una función determinada.

2.10.2 Conceptos de lenguaje de programación

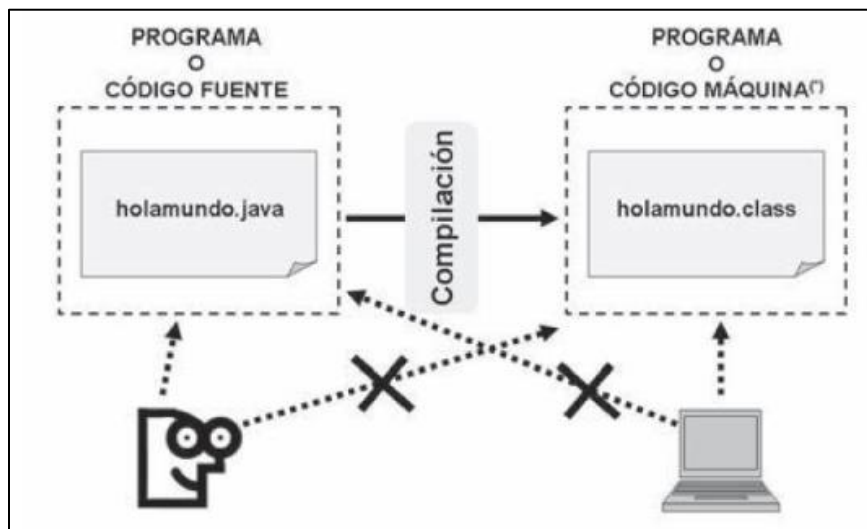
Lenguaje artificial que utiliza para expresar programas de ordenadores. Cada ordenador, según su diseño, "entiende" un cierto conjunto de instrucciones elementales (lenguaje maquina). No obstante, para facilitar la tarea del programador, se dispone también de lenguajes de alto

nivel más fáciles de manejar y que no dependen del diseño específico de cada ordenador mientras no sean traducidos al lenguaje propio de este.

Para definir un lenguaje de programación es necesario especificar:

- Conjunto de símbolos y palabras clave utilizables.
 - Reglas gramaticales para construir sentencias (instrucciones, ordenes) sintácticas y semánticas correctas.
- a. Sintaxis.** - Conjuntos de normas que determinan como escribir las sentencias del lenguaje.
- b. Semántica.** - Interpreta de las sentencias. Indica el significado de las mismas.

Si una vez has tenido la curiosidad de abrir un programa con un bloc de notas o editor de texto te habrás dado cuenta que aparece algo horrible en el editor, una serie de símbolos ininteligibles (por los humanos). Eso es porque los programas están en binario, que es el lenguaje que entienden las máquinas. Entonces te preguntarás: si al final de este libro seré capaz de escribir programas, ¿podré entender esos códigos? La respuesta es No. En este libro vamos a aprender un lenguaje de programación para escribir programas de manera entendible por los humanos que luego traduciremos al lenguaje máquina entendible por los ordenadores mediante otros programas llamados intérpretes o compiladores. En la siguiente figura se verá todo esto de modo más gráfico: (Moreno Pérez, 2015).

Figura 6:*Programación fuente y máquina*

Nota: La imagen describe la programación fuente y maquina por Moreno Pérez J.C (2015).

Como se puede observar, el código fuente es el que escribe el programador que luego lo compila a código máquina. Compilar equivale a transformar el programa inteligible por el programador al programa inteligible por la máquina. El código fuente o programa fuente está escrito en un lenguaje de programación y el compilador es un programa que se encarga de transformar el código fuente en código máquina (Moreno Pérez, 2015).

Los compiladores son programas específicos para un lenguaje de programación, los cuales transforman el programa fuente en un programa directa o indirectamente ejecutable por la máquina destino. No es posible compilar un programa escrito en lenguaje Java con un compilador de C porque éste no lo entendería.

El lenguaje máquina que genera Java es un lenguaje intermedio interpretable por una máquina virtual instalada en el ordenador donde se va a ejecutar. Una máquina virtual es una máquina ficticia que traduce las instrucciones máquina ficticias en instrucciones para la máquina real. La ventaja de la misma es que los programas se pueden ejecutar en cualquier tipo de

hardware siempre y cuando tenga instalada la máquina virtual correspondiente. Los programas no van a cambiar, lo que cambiará es la máquina virtual dependiendo del hardware (no será igual la máquina virtual de un smartphone que la de un PC). a Compiladores e Intérpretes A diferencia de los compiladores, los intérpretes leen línea a línea el código fuente y lo ejecutan (Moreno Pérez, 2015).

Este proceso es muy lento y requiere tener cargado en memoria el intérprete. La ventaja de los intérpretes es que la depuración y corrección de errores del programa es mucho más sencilla que con los compiladores (Moreno Pérez, 2015).

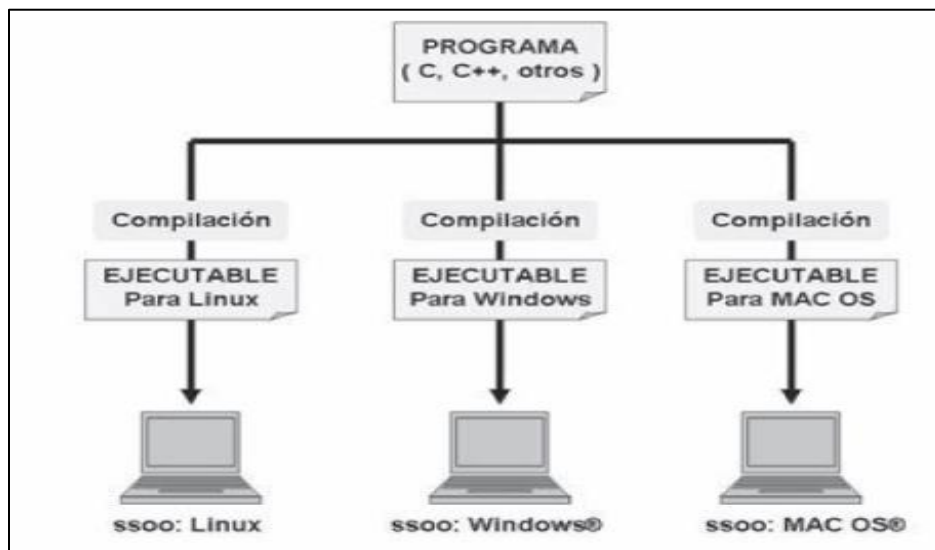
2.10.3. El Lenguaje Java

Según Moreno (2015), expresa que Java es uno de los lenguajes más utilizados en la actualidad. Es un lenguaje de propósito general y su éxito radica en que es el lenguaje de Internet. Applets, Servlets, páginas JSP o JavaScript utilizan Java como lenguaje de programación. El éxito de Java radica en que es un lenguaje multiplataforma.

Java utiliza una máquina virtual en el sistema destino y por lo tanto no hace falta recompilar de nuevo las aplicaciones para cada sistema operativo. Java, por lo tanto, es un lenguaje interpretado que para mayor eficiencia utiliza un código intermedio (bytecode). Este código intermedio o bytecode es independiente de la arquitectura y por lo tanto puede ser ejecutado en cualquier sistema.

Figura 7:

Adaptación del programa para cada sistema operativo



Nota: La imagen describe la recompilación de programa para cada sistema operativo por (Moreno, Pérez)

Como puede apreciarse en las figuras anteriores, en Java, una vez compilado el programa, se puede ejecutar en cualquier plataforma solamente con tener instalada la máquina virtual (Virtual Machine – VM) de Java. Sin embargo, en C, C++ u otro lenguaje, deberemos recompilar el programa para el sistema destino con la consiguiente pérdida de flexibilidad (Moreno Pérez, 2015).

2.10.4. Java Enterprise Edition (Java EE)

La plataforma Java Enterprise Edition define las API's (Interfaz de Programación de Aplicaciones) y requerimientos necesarios para poder ejecutar aplicaciones Java servidoras, enfocadas a brindar servicios empresariales como: transaccionalidad, seguridad, interoperabilidad entre muchos más servicios, características que no eran importantes o imprescindibles en aplicaciones de escritorio (Ordax Cassá, 2013).

Actualmente Java EE se encuentra en la versión 8.0 cabe mencionar que las versiones anteriores a Java EE 8.0 son:

- Java EE 1.2
- Java 2 EE 1.3
- Java 2 EE 1.4
- Java EE 5.0
- Java EE 6.0
- Java EE 7.0
- Java EE 8.0

2.10.5. Arquitectura Java EE

El modelo de aplicaciones Java EE define una arquitectura que está compuesta de distintas capas, cada capa tiene una función muy específica, esto tiene varias ventajas como son separación de responsabilidades, un mejor mantenimiento a la aplicación entre otras. La versión empresarial de java brinda un API distinta para cada capa (Ordax Cassá, 2013).

Las aplicaciones en Java se componen de una serie de ficheros, clases en bytecode que contienen las clases del programa. Estos ficheros no tienen por qué estar situados en un directorio concreto, sino que pueden estar distribuidos en varios discos o incluso en varias máquinas. La aplicación se ejecuta desde el método principal o main () situada en una clase. A partir de aquí se van creando objetos a partir de las clases y se va ejecutando la aplicación. El main () es un método estático (ya se explicará esto más adelante) el cual puede empezar a crear los objetos, incluidos los de su propia clase (Moreno Pérez, 2015).

Figura 8:

Arquitectura multicapas en JAVA EE



Nota: La imagen describe la arquitectura multicapas en java por (Ordax Cassa,2013)

- **Capa cliente.** - Esta capa el cliente interactúa por medio de un navegador web, un cliente móvil, una aplicación de escritorio entre otros.
- **Capa presentación.** – Reside en un servidor web y es la responsable del control de la aplicación aquí encontramos tecnologías como JSP y JSF.
- **Capa de negocios.** - Es la responsable de la lógica de la aplicación en esta capa podemos encontrar tecnologías como EJB.
- **Capa de datos.** - Conocida también con el nombre de Sistema de Información Empresarial, esta capa es la responsable de la persistencia de datos aquí se encuentran presente las tecnologías como JDBC o JPA.

2.10.6. Componentes de Java EE

Las aplicaciones Java EE están formada por varios componentes siendo estos una unidad de software funcional auto contenida que se ensambla como parte de una aplicación java EE y que puede interactuar con otros componentes (Ordax Cassá, 2013).

Componente cliente. Estos componentes son desplegados del lado del cliente (capa cliente).

- **Cliente web.** - Consiste en desplegar el contenido de la página web dinámica en cualquier navegador web.
- **Aplicación cliente.** - Son aplicaciones Java SE que se ejecutan en la maquina cliente la cual tiene una interfaz gráfica con el usuario, creada con Swing o Awt.
- **Componente web.** - En esta capa podemos encontrar Servlets o paginas creadas utilizando tecnología Java Server Faces (JSF) y/o o tecnologías Java Server Pages (JSP). Estos componentes se despliegan en la capa web.
- **Componente de negocios.** - Los Enterprise Java Beans (EJB) es código Java del lado del servidor. Normalmente tienen la lógica de negocio de nuestra aplicación, y por lo tanto cubren el rol de la capa de servicio de nuestras aplicaciones Java, los EJB's al día de hoy es una tecnología muy probada que brinda beneficios tales como seguridad, transaccionalidad, multi-threading entre muchas características más, todo esto con ayuda de un servidor de aplicaciones Java.

2.11. Herramientas de Desarrollo

Según (Ordax Cassá, 2013) Las herramientas de desarrollo de software son programas informáticos que usa un programador para crear, depurar, gestionar o mantener un programa, cabe mencionar que la mayoría de las herramientas son de tipo software libre, a continuación, se detalla las diferentes herramientas que serán empleadas en el desarrollo de la aplicación web.

2.11.1. Servidor Web Apache Tomcat

Apache proporciona contenidos al cliente web o navegador como:

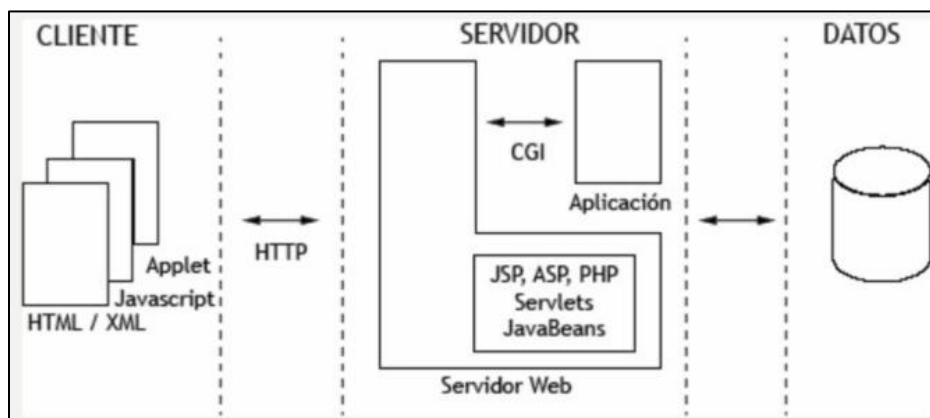
- **Paginas estáticas.** - Es el modo más básico y antiguo, pero también es el uso más generalizado que se hace de un servidor web de esta forma se transfiere

archivos HTML, imágenes, etc., y no se requiere en servidor muy potente en lo que al hardware se refiere. (Mifsuf Talón, 2013.)

- **Páginas dinámicas.** - La información que se muestra las páginas que sirve Apache cambia continuamente ya que se obtiene a partir de consultas a bases de datos u otras fuentes de datos son páginas con contenido dinámico, cambiante. (Mifsuf Talón, 2013.)

Figura 9:

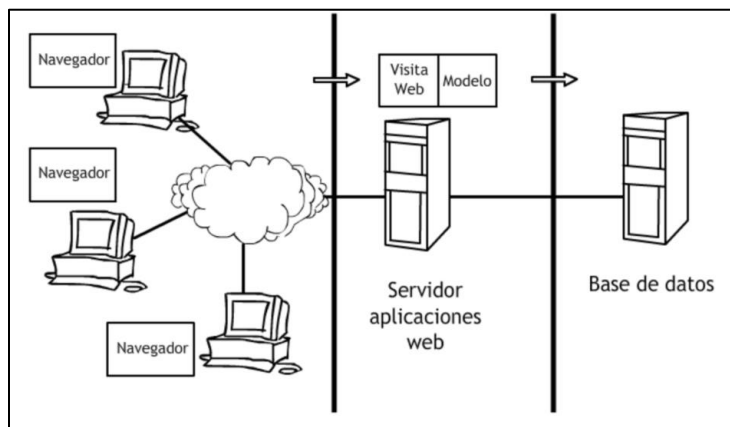
Esquema del funcionamiento de un servidor web



Nota: La imagen describe los tres elementos básicos que componen las peticiones web dinámicas ((Apache Netbeans,2019). (Mifsuf Talon,2013)).

Figura 10:

Función de un Servidor en sus tres capas



Nota: La imagen describe las tres capas de un servidor web (Apache Netbeans,2019).

2.12. Ubuntu Server

Existen en la actualidad gran variedad de distribuciones libres del Sistema Operativo Linux, el cual ha ganado un espacio preponderante por sus características de multiusuario, multitarea, estabilidad, seguridad, conectividad, escalabilidad y compatibilidad con gran variedad de aplicaciones. Una de las distribuciones más usadas en diferentes ámbitos, entre ellos el científico, académico, industrial y comercial, es la distribución UBUNTU, ésta ha sido patrocinada por la empresa Canonical Ltda., organización británica propiedad del sudafricano Mark Shuttleworth. UBUNTU posee múltiples herramientas de configuración de servicios tales como DHCP (Dynamic Host Configuration Protocol), DNS (Domain Name System), LDAP y SAMBA, PROXY y el servidor WEB APACHE, entre otros. Por ello su funcionalidad es bastante amplia en lo referente a procesos de configuración de servicios para estaciones de trabajo y servidores. Organización británica propiedad del sudafricano Mark Shuttleworth. (De la Hoz Correa, 2009.)

UBUNTU posee múltiples herramientas de configuración de servicios tales como DHCP (Dynamic Host Configuration Protocol), DNS (Domain Name System), LDAP y SAMBA, PROXY

y el servidor WEB APACHE, entre otros. Por ello su funcionalidad es bastante amplia en lo referente a procesos de configuración de servicios para estaciones de trabajo y servidores (De la Hoz Correa, 2009.).

Ubuntu Server 20.04 LTS (soporte a largo plazo) está aquí con estabilidad, resistencia e incluso mejor seguridad de clase empresarial. Como versión LTS, será compatible con Canonical hasta 2025. Sin embargo, los clientes pueden extender las actualizaciones de seguridad por cinco años adicionales a través del servicio ESM (Mantenimiento de seguridad extendido) como parte de su suscripción UA-I (Ventaja de Ubuntu para infraestructura). Todo eso hace que Ubuntu Server 20.04 LTS sea una de las distribuciones de Linux más estables y seguras, (De la Hoz Correa, 2009.)

2.13. PostgresSQL

PostgreSQL es un gestor de bases de datos orientadas a objetos (SGBDOO o ORDBMS en sus siglas en inglés) muy conocido y usado en entornos de software libre porque cumple los estándares SQL92 y SQL99, y también por el conjunto de funcionalidades avanzadas que soporta, lo que lo sitúa al mismo o a un mejor nivel que muchos SGBD comerciales. (Ginestá, 2012)

El origen de PostgreSQL se sitúa en el gestor de bases de datos POSTGRES desarrollado en la Universidad de Berkeley y que se abandonó en favor de PostgreSQL a partir de 1994. Ya entonces, contaba con prestaciones que lo hacían único en el mercado y que otros gestores de bases de datos comerciales han ido añadiendo durante este tiempo. PostgreSQL se distribuye bajo licencia BSD, lo que permite su uso, redistribución, modificación con la única restricción de mantener el copyright del software a sus autores, en concreto el PostgreSQL Global Development Group y la Universidad de California. (Ginestá, 2012)

2.14. Apache Maven

Maven es una herramienta de software para la gestión de proyectos Java y la automatización de compilaciones. Es similar en funcionalidad a la herramienta Apache Ant, pero se basa en conceptos diferentes. Maven está alojado en Apache Software Foundation, donde anteriormente formaba parte del Proyecto Jakarta (Miller, 2010).

Maven utiliza una construcción conocida como Project Object Model (POM) para describir el proyecto de software que se está construyendo, sus dependencias en otros módulos y componentes externos y el orden de construcción. Viene con objetivos predefinidos para realizar ciertas tareas bien definidas, como la compilación de código y su empaque. Maven descarga dinámicamente las bibliotecas de Java y los complementos de Maven desde uno o más repositorios. Maven proporciona soporte integrado para recuperar archivos del Repositorio central de Maven 2 y otros repositorios de Maven (Miller, 2010).

2.15. Alfresco

Alfresco Software Inc. Es el líder en el mercado de código abierto para la gestión de contenidos empresariales. La empresa integra la innovación del código abierto con la estabilidad de una verdadera plataforma de clase empresarial a un costo cero. La plataforma de contenido de Alfresco utiliza una arquitectura flexible para proveer gestión de documentos, gestión de contenidos web, gestión de registro y software colaborativos a clientes empresariales y socios en 40 países (Navarro Bustos, 2012).

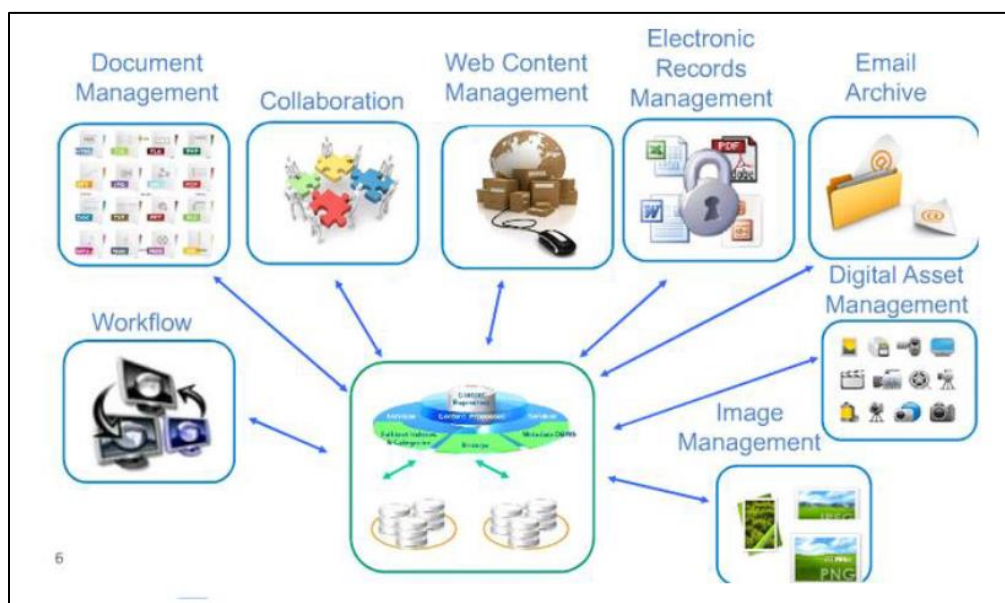
Igual Alfresco ofrece una gestión de documentos que utiliza interfaces habituales para que el usuario las adopte rápidamente, basadas en un repositorio que proporciona servicios transparentes e invisibles para una completa gestión de contenido empresarial (Navarro Bustos, 2012).

- Sistema de archivos virtuales
- Gestor de Documentos

- Reglas tipo correo electrónico
- Búsqueda tipo Google: busca directamente desde Firefox o IE7
- Exploración tipo Yahoo: extracción y categorización automática de metadatos.
- Espacios inteligentes.
- Asistencia de ciclo de vida transparente.

Figura 11:

Plataforma Alfresco y sus servicios



Nota: La imagen describe todos los servicios que nos presenta la herramienta Alfresco (Navarro Bustos,2012).

2.16. Repositorios de Documentos

Los Repositorios de Documentos Administrativos describieron un proyecto para mejorar el acceso y la longevidad de los documentos del gobierno electrónico usando DSpace y concluyen que este software puede ser usado para dicho objetivo, porque facilita la preservación

y el acceso a los objetos digitales, utiliza estándares abiertos para facilitar la interoperabilidad y está organizado por el concepto de comunidades-colecciones (Texier, 2013).

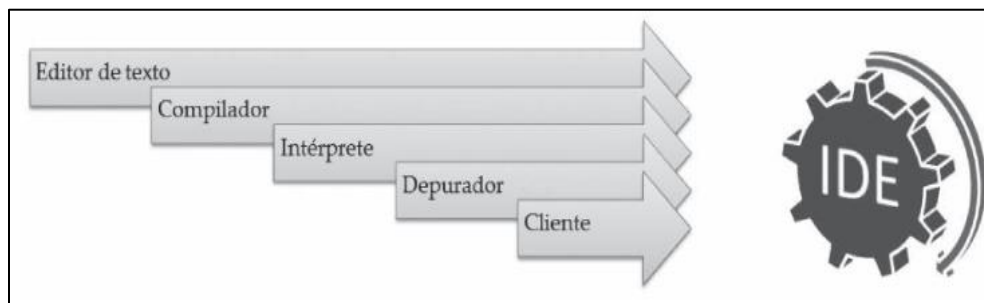
De esta manera los Repositorios de Documentos Administrativos se están convirtiendo progresivamente en los lugares en los que numerosas organizaciones almacenan y organizan el resultado de sus actividades administrativas. Esas experiencias sirven de base para consolidar un RDA, que ofrezca los servicios de catalogación, acceso, difusión y preservación de la información. Los principales beneficios de un RDA son:

- Establecer un depósito institucional de documentos administrativos que ofrezca acceso a toda la comunidad universitaria y al público en general.
- Obtener un acceso más fácil y rápido a los documentos para la toma de decisiones en diferentes ámbitos.
- Ofrecer acceso remoto a los documentos, con el fin de evitar trasladarse a la dependencia de la institución físicamente (Texier, 2013).

2.17. Entorno de Desarrollo Integrado

La definición más sencilla de entorno de desarrollo. “Un entorno de desarrollo integrado o IDE (Integrated Development Environment) es un programa informático que tiene el objetivo de asistir al programador en la tarea de diseñar y codificar un software mediante la inclusión de múltiples herramientas destinadas para dicha tarea” (Casado Iglesias, 2015, pág. 45).

En la actualidad la gran mayoría de entornos de desarrollo integrado traen consigo varias características y funcionalidades particulares, todos los IDE's poseen componentes en común.

Figura 12:*Componentes Básicos de un IDE*

Nota: La imagen describe los componentes básicos del entorno de Desarrollo integrado por (Apache Netbeans,2019)

2.18. Apache Netbeans

El entorno de desarrollo Apache Netbeans trae nuevas características trae soporte para las ultimas características de java, integra nuevas donaciones de código de Oracle para el desarrollo web de java, mejora sus herramientas para Apache Maven, todas las funciones mencionadas son compatibles de inmediato sin necesidad de instalar complementos adicionales (Apache Netbeans, 2019).

2.19. Base de datos

Las bases de datos según Cabello (2015), afirma que, “Conjunto de datos interrelacionados y no redundantes que se estructuran de forma independiente en cuanto a su uso e implementación, facilitando a los usuarios un acceso concurrente a unos datos que cumplen ciertas restricciones de integridad”.

2.19.1. SGBD

El sistema gestor de base de datos es un conjunto de programas según lo afirma Cabello (2015), menciona que “Un SGBD es el software que facilitara al usuario definir consultar y

mantener la información de una base de datos, proporcionando la interfaz necesaria para ello y garantizando las restricciones de integridad y seguridad definidas”. En la actualidad existe una gran variedad de SGBD’s que se dividen en dos grupos relacionales y no relacionales.

2.19.2. MySQL

Según Cabello, (2015), describe que MySQL es un sistema gestor de base de datos relacional de propiedad de Oracle Corporation siendo multi plataforma la licencia que maneja es dual, este SGBD se caracteriza por ser multiusuario y muy fácil de manejar, contiene varias funciones para los usuarios novatos, y más sencillo que otros sistemas.

2.20. Servidor de Aplicaciones Java EE

El servidor de aplicaciones es en donde se ejecutan ciertas aplicaciones que van hacer usadas por los clientes de una red como lo afirma Cabello (2015) expresa lo siguiente:

Un servidor de aplicaciones, por tanto, es un producto de software que implementa todas las especificaciones Java EE. De manera que al desplegar o instalar una aplicación Java EE en el servidor, sabemos seguro que va a encontrarse con todos los contenedores y servicio definidos por la especificación y que seguramente utiliza y necesita la aplicación.

Figura 13:

Servidor de aplicaciones Java



Nota: La imagen describe el Servidor por (Apache Netbeans,2019)

2.21. Metodologías Ágiles de Software

Las metodologías ágiles están basadas en métodos de desarrollo iterativo e incremental. Donde los requisitos y soluciones evolucionan durante el proyecto según la necesidad del proyecto.

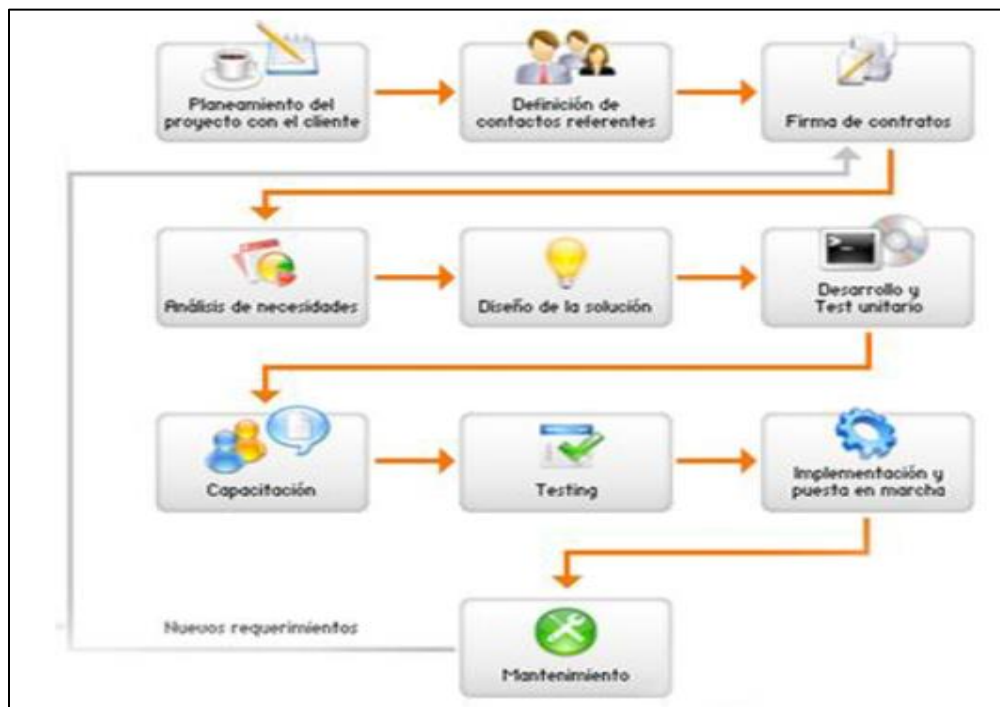
Las metodologías ágiles usan iteraciones y cada iteración incluye una planificación, análisis de requisitos, diseño, codificación, pruebas y documentación (Letelier & Penades, 2006).

Dentro de la Ingeniería de Software, para la creación de un software se aplica un conjunto de procesos, métodos y herramientas de forma organizada, a todo esto, lo conocemos como la metodología de desarrollo la cual cumple un papel muy importante en todo el proceso. Comúnmente se suele utilizar metodologías "tradicionales" para la elaboración o desarrollo de un software, donde han demostrado ser efectivas y fundamentales en proyectos de gran tamaño

con relación al tiempo y a los recursos, ya que utilizan un control riguroso en los procesos, además de incluir el modelado de datos y la documentación detallada (Letelier & Penades, 2006).

Figura 14:

Etapas y procesos de las metodologías tradicionales



Nota: El gráfico describe las metodologías tradicionales

Sin embargo, en la actualidad los proyectos presentan entornos variantes, se espera reducir los tiempos drásticamente y a la vez manteniendo la calidad del software, y es así como todas estas problemáticas de tiempo y flexibilidad ya no pueden ser cubiertas por las metodologías tradiciones.

Figura 15:*Cuadro Comparativo de las metodologías*

Metodologías Ágiles	Metodologías Tradicionales
Se fundamenta en el conjunto de conocimientos obtenidos previamente a través de la experiencia en el desarrollo de software.	Se fundamenta en las normas y estándares establecidos por un ambiente de desarrollo.
Se aceptan cambios durante el proceso de desarrollo del proyecto y son utilizados retroalimentación.	Resistencia a cambios planteados por el cliente durante el proceso de desarrollo del proyecto.
Control mínimo requerido durante el proceso, dando mayor facilidad al equipo.	Proceso más rígido y controlado por políticas que se apegan un modelo establecido.
No existe un Contrato predeterminado o este es más flexible.	Existencia de un Contrato definido desde el principio.
Se considera al cliente como parte del equipo durante el desarrollo, estableciendo una relación más directa y personalizada.	Se realizan reuniones documentadas donde el cliente interactúa indirectamente con el equipo de desarrollo.
Orientada a pequeños grupos donde todos trabajan de forma cooperada.	Se presenta un grupo grande de trabajo donde están distribuidos de acuerdo a las diferentes etapas.
Menor relevancia en la arquitectura y mayor énfasis en la funcionalidad del software.	Mayor relevancia en la utilización de modelos en la arquitectura del software.

Nota: El grafico describe las diferencias entre metodologías ágiles y tradicionales (José Moran,2019)

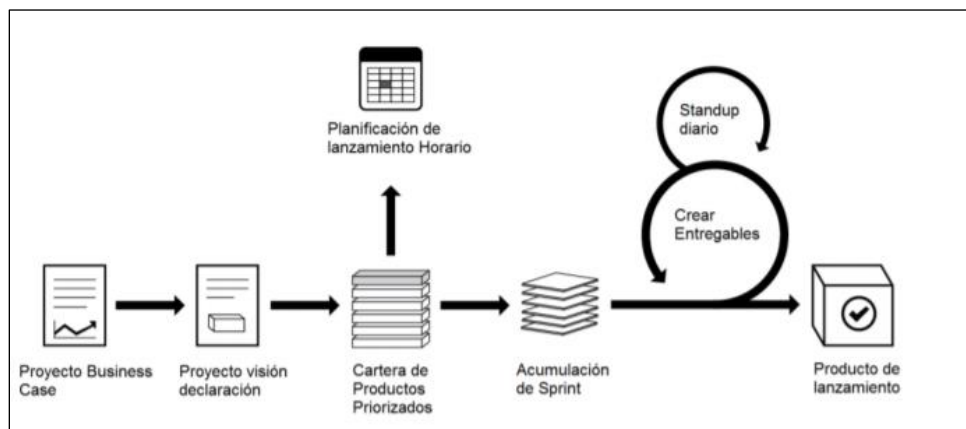
CAPÍTULO III

3. DESARROLLO

3.1. Scrum

Un proyecto Scrum implica un esfuerzo de colaboración para crear un nuevo Producto o, servicio, o cualquier otro resultado como se define en el Declaración de la Visión del Proyecto. Los proyectos se ven afectados por las limitaciones de tiempo, costo, alcance, calidad, recursos, capacidades organizativas, y otras limitaciones que los hacen difíciles de planificar, ejecutar, administrar y finalmente tener éxito. Sin embargo, la implementación exitosa de los resultados de un proyecto acabado le proporciona ventajas económicas significativas a una organización. Por lo tanto, es importante que las organizaciones seleccionen y practiquen una metodología adecuada de gestión de proyectos (Tridibesh Satpathy, 2013).

Scrum es una de las metodologías ágiles más populares. Es una metodología de adaptación, iterativa, rápida, flexible y eficaz, diseñada para ofrecer un valor significativo de forma rápida en todo el proyecto. Scrum garantiza transparencia en la comunicación y crea un ambiente de responsabilidad colectiva y de progreso continuo. El marco de Scrum, tal como se define en la Guía SBOK™, está estructurado de tal manera que es compatible con los Productos y el desarrollo de servicio en todo tipo de industrias y en cualquier tipo de proyecto, independientemente de su complejidad. Una fortaleza clave de Scrum radica en el uso de equipos multifuncionales, auto-organizados, y con poder que dividen su trabajo en ciclos de trabajo cortos y concentrados llamados Sprints. Figura 1-1 proporciona una visión general de flujo de un proyecto Scrum (Tridibesh Satpathy, 2013).

Figura 16:*Flujo para un Sprint*

Nota: El grafico describe las diferencias entre metodología Scrum para un Sprint Elaborado por (Tridibesh Satpathy,2013).

3.1.1. Equipo de Scrum

El equipo de Scrum está confirmado:

- el Dueño del Producto (Product Owner).
- el Equipo de Desarrollo.
- el Scrum Master.

3.1.2 Product Owner

Es el responsable de maximizar el valor del producto y del trabajo del Equipo de Desarrollo. El Dueño del producto está a cargo de gestionar la Lista del Producto (Product Backlog) cuya gestión incluye:

- Expresar claramente los elementos de la lista del Producto.
- Ordenar los elementos en la lista del Producto para alcanzar los objetivos y misiones de la mejor manera posible.
- Optimizar el valor del trabajo desempeñado por el Equipo de Desarrollo.

- Asegurar que la Lista del Producto es visible, transparente y clara para todos, y que muestra aquellas en lo que el equipo trabajara a continuación
- Asegurar que el Equipo de Desarrollo entiende los elementos de la Lista del Producto al nivel necesario.

EL Product Owner es una única persona. Toda la organización tiene que respetar las decisiones del Dueño del Producto, de lo contrario será más difícil alcanzar el objetivo esperado (Sutherland, 2016).

3.1.3 Extreme Programming XP

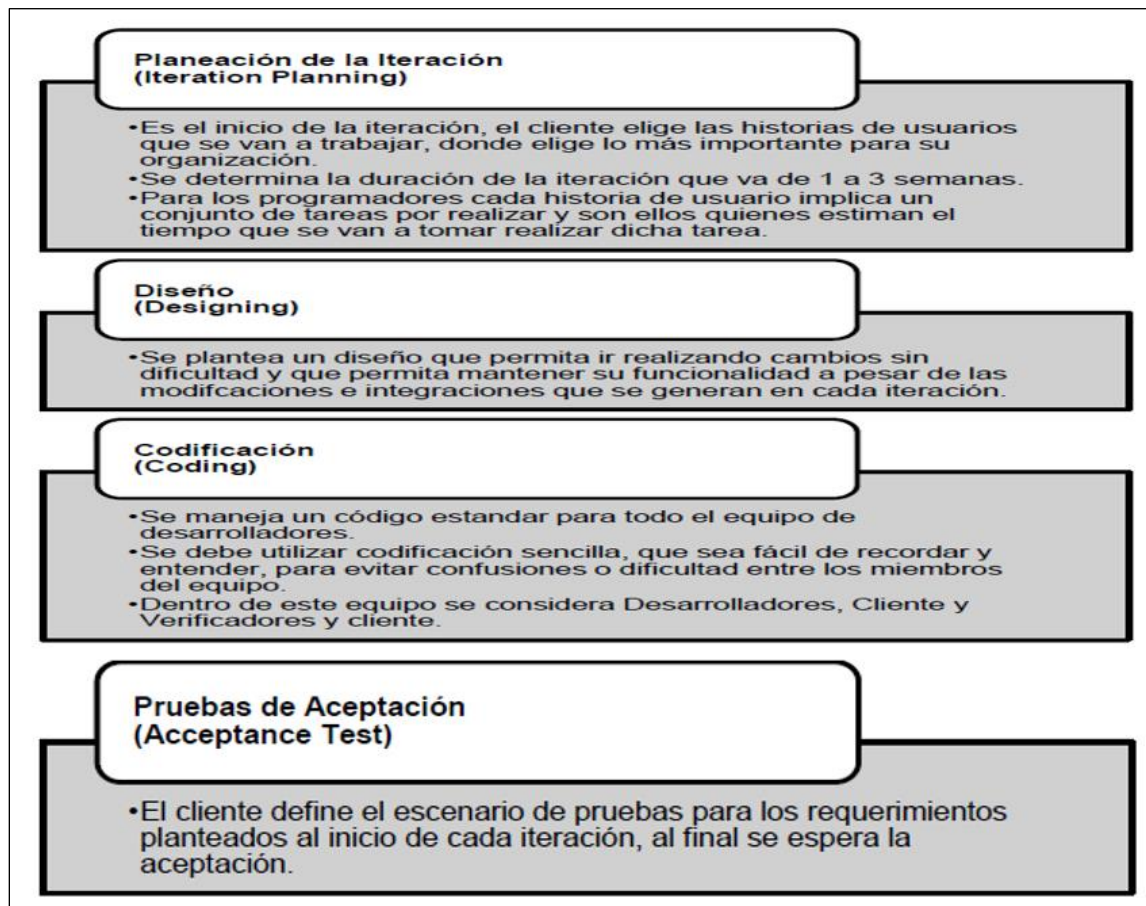
Dentro de este proceso que tiene la metodología XP, existe un ciclo con etapas o frases que cumple cada iteración como son: la Planificación, el Diseño, la Codificación y las Pruebas. Las detallaremos en el cuadro a continuación:

También conocida como Programación Extrema, tiene sus orígenes a mediados de la década del 90 de la mano de Kent Beack un Ingeniero de Software y Master en Ciencias de la Computación, el cual también forma parte de los autores del Manifiesto Ágil. Entre sus principales beneficios se menciona la elaboración del sistema por iteraciones con entregas frecuentes y continuas, durante el desarrollo del software se considera la participación activa del cliente y esto se realiza por medio de las historias de usuarios, obteniendo al final de dicha iteración una versión para validar el producto (Manzaba, 2014).

Dentro de este proceso que tiene la metodología XP, existe un ciclo con etapas o frases que cumple cada iteración como son: la Planificación, el Diseño, la Codificación y las Pruebas. Las detallaremos en el cuadro a continuación:

Figura 17:

Ciclo de vida de la Metodología XP



Nota: La imagen describe la metodología XP Elaborado por Juan Moran Sánchez fuentes metodológicas para el desarrollo de software.

3.2. Levantamientos de Requisitos

3.2.1. Visión del Proyecto

EL COMANDO DE EDUCACIÓN Y DOCTRINA MILITAR tiene por objetivo principal disponer de un módulo web para la gestión de documentos para mantener un adecuado almacenamiento, búsqueda y las versiones de la documentación digitales aprobados y actualizada por el escalón superior con un diseño amigable.

El CEDE tiene por objetivo principal disponer de un módulo web para la gestión de documentos para mantener un adecuado almacenamiento, búsqueda y las versiones de la documentación digitales aprobados y actualizada por el escalón superior con un diseño amigable al usuario.

3.2.2. Identificación de Épicas

El levantamiento de las épicas del proyecto DOCTRINA MILITAR se describe de forma general la intención y propósito de cada una de las áreas que se buscan cubrir y desarrollar este sistema, así como el funcionamiento de estas en conjunto, mismas que se describen a continuación:

ÉPICA 1: Yo DM-1 necesito reemplazar el MODULO DOCTRINA MILITAR por una herramienta actual que permita visualizar los diferentes catálogos, que brinde confiabilidad y permita una operación eficaz del módulo de doctrina militar.

ÉPICA 2: Yo DM-1 necesito una aplicación web sea más amigable e interactiva al usuario y al administrador.

ÉPICA 3: Yo DM-1 necesito una aplicación web que me permita almacenar, versionar buscar los diferentes tipos de documentación digitales y manuales al usuario.

ÉPICA 4: Yo DM-1 necesito una aplicación web que me permita imprimir un reporte del documento descargado por el usuario.

ÉPICA 5: Yo DM-1 necesito una aplicación web permita al actualizar la diferente documentación y manuales.

3.2.3. Historias de Usuarios

Tabla 1:*Historia de usuario 1. Vista de catálogos en triage*

HISTORIA DE USUARIO			
Número:	1	Usuario:	ADMINISTRADOR
Nombre historia:	Vista lista de catálogos documentales en triage.		
Prioridad en negocio:	10	Riesgo en desarrollo:	bajo
Puntos estimados:	69	Iteración asignada	1
Programador responsable:	Edwin Morocho		
Descripción:	Un módulo web que me permita visualizar la lista de documentos digitales a ser ingresados.		
Validación:	El sistema permite visualizar la lista de documentación digital subida		

Nota: La tabla describe la historia de usuario 1 de Vista de documentos en triage

Tabla 2:*Historia de Usuario 2. Evaluación triage de Manchester*

HISTORIA DE USUARIO			
Número:	2	Usuario:	USUARIO 1
Nombre historia:	Evaluación Triage de Manchester		
Prioridad en negocio:	10	Riesgo en desarrollo:	Medio
Puntos estimados:	116	Iteración asignada	1
Programador responsable:	Edwin Morocho		
Descripción:	Una aplicación web que me permita visualizar los diferentes documentos digitales y su respectiva descarga.		
Validación:	El Sistema determina la visualización de los diferentes documentos digitales		

Nota: La tabla describe la documentación de usuario 2 de triage de Manchester

3.3. Roles

Tabla 3:

Roles de Scrum

PROYECTO DOCTRINA MILITAR ROLES EQUIPO SCRUM		
Rol	Nombre	Contacto
Project Owner	Mayo. Guerrero S	rfierro@he1.mil.ec
Master Scrum	Sgos. Cruz w	jfcruz@he1.mil.ec
Desarrollador Equipo Scrum	Cbos. Morocho Edwin	eemorocho@espe.edu.ec

Nota: Esta tabla representa los roles de Scrum con sus participantes del proyecto

3.3.1. Ejecución de Scrum

3.3.1.1. Product Backlog

Tabla 4:

Estimación de tiempos de producto backlog

	HISTORIA DE USUARIO		TAREA	OTROS DATOS DE LA HISTORIA DE USUARIO O TAREA					
Gestión Documental	Una aplicación web que me permita visualizar la lista de documentos institucionales para el centro de doctrina militar	Iniciar su sesión y visualización de los documentos institucionales	1.1	Diseño de la interfaz de triage (para el ingreso al módulo web)	El sistema permite visualizar la lista de documentos almacenados en Alfresco	10	10	1	Terminado
			1.2	Programar la funcionalidad de "Modulo web"	El sistema permite llamar e ingresar al inicio de sesión de los servicios de Alfresco	6	4	1	

HISTORIA DE USUARIO	TAREA	OTROS DATOS DE LA HISTORIA DE USUARIO O TAREA				
	1.3 Programar la funcionalidad de "Sitio"	El sistema abre el formulario de Triage	10	1	1	Terminado
	1.4 Programar la funcionalidad de Gestión de Documentos Institucionales"	El sistema crea el sitio gestión para guardar la información de cada repositorio documental.	4	4	1	Terminado
	1.5 Diseño de la interfaz de triage (Doctrina Militar)	El sistema permite visualizar la lista de documentos institucionales	8	8	1	Terminado
	1.6 Programar la funcionalidad de búsquedas discriminatorias por campos	El sistema permite realizar búsquedas	6	16	1	Terminado
	1.7 Diseño de la interfaz de triage (Catalogo de documentos)	El sistema permite visualizar los documentos que existen en cada carpeta	6	8	1	Terminado
	1.8 Programar la funcionalidad de búsquedas discriminatorias por campos	El sistema permite realizar búsquedas	6	16	1	Terminado

Nota: La tabla muestra la Estimación de tiempos de productos backlog por cada Sprint con las pruebas de aceptación por el usuario.

3.3.1.2. Pila del Sprint

Duración del Sprint		15	Esfuerzo Faltante en los días siguientes...																																																															
Tendencia calculada en los últimos		15	<table border="0"> <tr> <td></td><td>2</td><td>2</td><td>2</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>9</td><td>7</td><td>6</td><td>4</td><td>3</td><td>1</td> </tr> <tr> <td></td><td>2</td><td>2</td><td>1</td><td>9</td><td>8</td><td>6</td><td>4</td><td>3</td><td>1</td><td>5</td><td>9</td><td>4</td><td>9</td><td>3</td><td>7</td><td></td> </tr> <tr> <td></td><td>7</td><td>7</td><td>2</td><td>6</td><td>0</td><td>5</td><td>9</td><td>3</td><td>8</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> </table>														2	2	2	1	1	1	1	1	1	1	9	7	6	4	3	1		2	2	1	9	8	6	4	3	1	5	9	4	9	3	7			7	7	2	6	0	5	9	3	8							
	2	2	2	1	1	1	1	1	1	1	9	7	6	4	3	1																																																		
	2	2	1	9	8	6	4	3	1	5	9	4	9	3	7																																																			
	7	7	2	6	0	5	9	3	8																																																									
			Días	Totales																																																														
ID Tarea	Tarea	ID Historia	Responsable	Estado	Est.	1	2	3	4	5	6	7	8	9	1	0	1	1	1	1	1																																													
1.5	Diseño de la interfaz de triage (Creación de carpeta Catalogo)	1	Morocho Edwin	Terminado	8	8	8	8	4	0	0	0	0	0	0	0	0	0	0	0	0	0																																												
1.6	Programar la funcionalidad de búsquedas discriminatorias por campos	1	Morocho Edwin	Terminado	1	1	1	1	1	1	5	0	0	0	0	0	0	0	0	0	0	0																																												
1.7	Creación de Carpetas de Documentos Doctrinales	1	Morocho Edwin	Terminado	8	8	8	8	8	8	8	5	0	0	0	0	0	0	0	0	0	0																																												
1.8	Programar la funcionalidad de búsquedas discriminatorias por campos (Toda la Herramienta Alfresco)	1	Morocho Edwin	Terminado	1	1	1	1	1	1	1	1	1	1	1	6	0	0	0	0	0	0																																												
					6	6	6	6	6	6	6	6	6	6	4																																																			

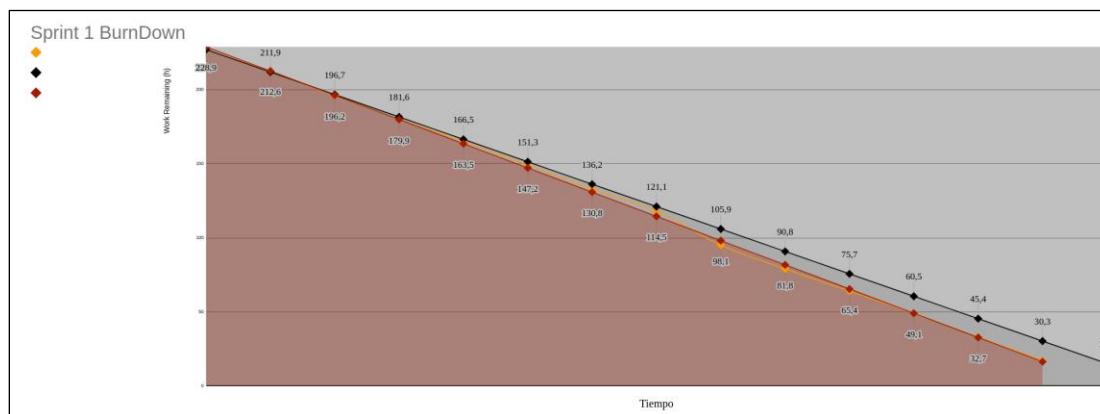
		Duración del Sprint	15	Esfuerzo Faltante en los días siguientes...																	
		Tendencia calculada en los últimos	15	2	2	2	1	1	1	1	1	1	1	9	7	6	4	3	1		
				2	2	1	9	8	6	4	3	1	5	9	4	9	3	7			
				7	7	2	6	0	5	9	3	8									
				Días Totales																	
ID Tarea	Tarea	ID Historia	Responsable	Estado	Est.	1	2	3	4	5	6	7	8	9	1	0	1	1	1	1	1
																1	2	3	4	5	
3.1	Determinar las Diferentes Funcionalidades dependiendo el tipo de Usuario Designado por el Administrador de la Herramienta Alfresco	3	Morocho Edwin	Terminado		1	1	1	1	1	8	8	8	8	8	8	0	0	0	0	0

Nota: Esta tabla muestra el Sprint 4 con todas sus tareas

3.3.1.3. Gráfico Burn Down

Figura 18:

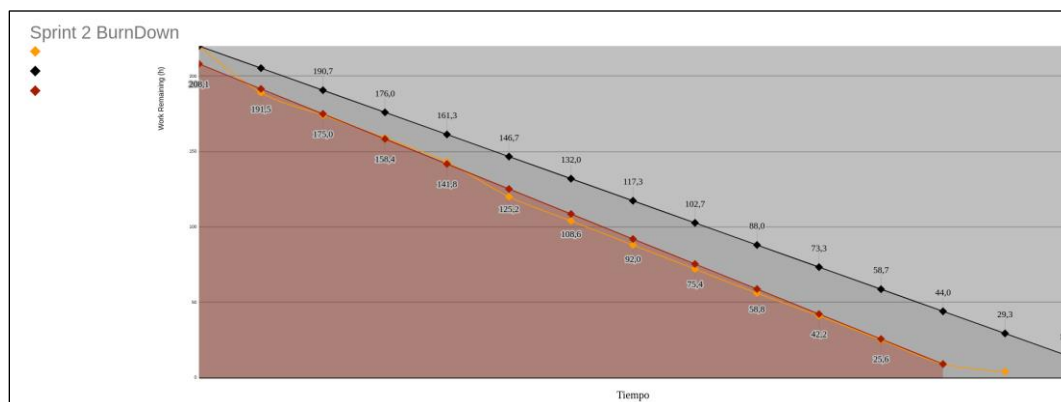
Burn Down del Sprint 1



Nota: Este gráfico presenta el Burn Down del Sprint 1 el cual podemos observar que el trabajo planificado fue cumplido poco tiempo antes de lo establecido para el desarrollo del Sprint, debido a que el esfuerzo estimado de 227 y el Sprint fue diseñado para una carga de 240 horas.

Figura 19:

Burn Down del Sprint 2

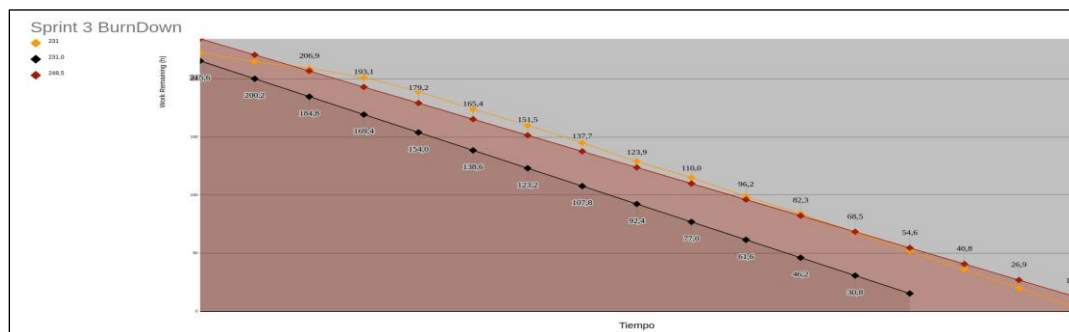


Nota: Este gráfico presenta el Burn Down del Sprint 2 el cual podemos observar que el trabajo planificado fue culminando antes del tiempo establecido para el Desarrollo del Sprint esto

también se debe a que el Sprint tiene una duración de 240 horas y el esfuerzo necesario contemplado al inicio desde 220 horas.

Figura 20:

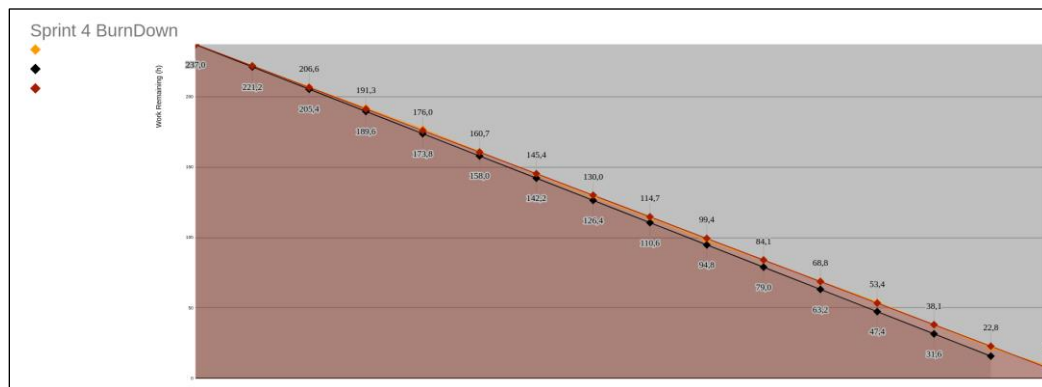
Burn Down del Sprint 3



Nota: El gráfico presenta el Burn del Sprint 3 podemos observar que el esfuerzo en este Sprint 3 podemos observar que el esfuerzo en este Sprint fue mayor al planificado tomándonos tres días más de lo establecido.

Figura 21:

Burn Down del Sprint 4



Nota: El gráfico presenta el Burn del Sprint 4 podemos observar que el esfuerzo requerido para el desarrollo de este Sprint fue mayor al planificado, tomándonos un día más de trabajo.

3.4. Pruebas

3.4.1. Pruebas de Aceptación de Usuario

Para obtener un producto terminado y que se cumpla con todos los requerimientos del cliente es necesario que la aplicación sea llevada con todos los requerimientos del cliente es necesario que la aplicación sea llevada a un Aseguramiento de Calidad (QA), en donde es importante tomar en cuenta las pruebas de aceptación de usuario.

Las pruebas de aceptación deben estar involucradas por el usuario real, tal es que el Sistema de Gestión Documental participe el Administrador del Sistema y los usuarios finales para definir los resultados obtenidos se representaran las siguientes notaciones:

- **Aceptado.** - Cuando los procesos correspondientes son evaluados y responde según lo esperado, así como la interfaz es amigable con el usuario.
- **No aceptado.** - Cuando los proceso al ser evaluados presenta errores, ejerce un mal funcionamiento o no muy entendible al usuario.

Tabla 6:

Prueba de aceptación: Doctrina Militar

	Historia de Usuario / # Requisito	Criterios de Aceptación	Aceptación	Usuario
1	Vista lista de documentos en triage.	El sistema permite visualizar la lista de documentos.	Aceptado	Administrador
2	Evaluación Triage de Manchester	El Sistema determina la clasificación y prioridad de búsqueda.	Aceptado	Usuario Final

Historia de Usuario / # Requisito	Criterios de Aceptación	Aceptación	Usuario
3 Clasificación de los tipos de documentos Institucionales y descarga del archivo deseado	El sistema permite visualizar la lista de documentos Institucionales y descarga del mismo.	Aceptado	Usuario Final
4 Permite registrar los nuevos archivos según su clasificación de Documentos Institucionales.	El sistema muestra los archivos, documentos, reglamentos ingresados en el módulo web.	Aceptado	Administrador y Usuario Final

Nota: Esta tabla muestra las pruebas de aceptación realizadas por el Administrador y el usuario final del sistema de gestión Documental por cada historia de usuario.

3.5. Pruebas de módulo web

Figura 22:

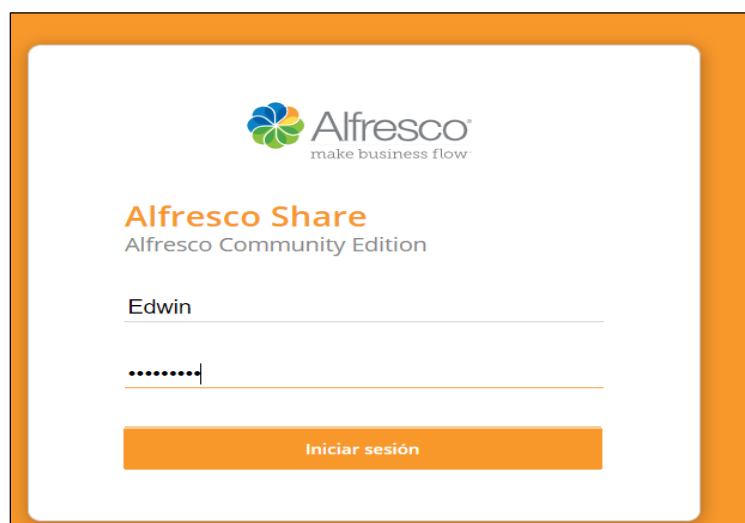
Página de Inicio de Módulo



Nota: Esta imagen permite visualizar la página de Inicio de la Gestión de Documentos Institucionales en el cual contaremos con un botón que nos llevara a utilizar los servicios de Alfresco para la gestión de Documentos Institucionales.

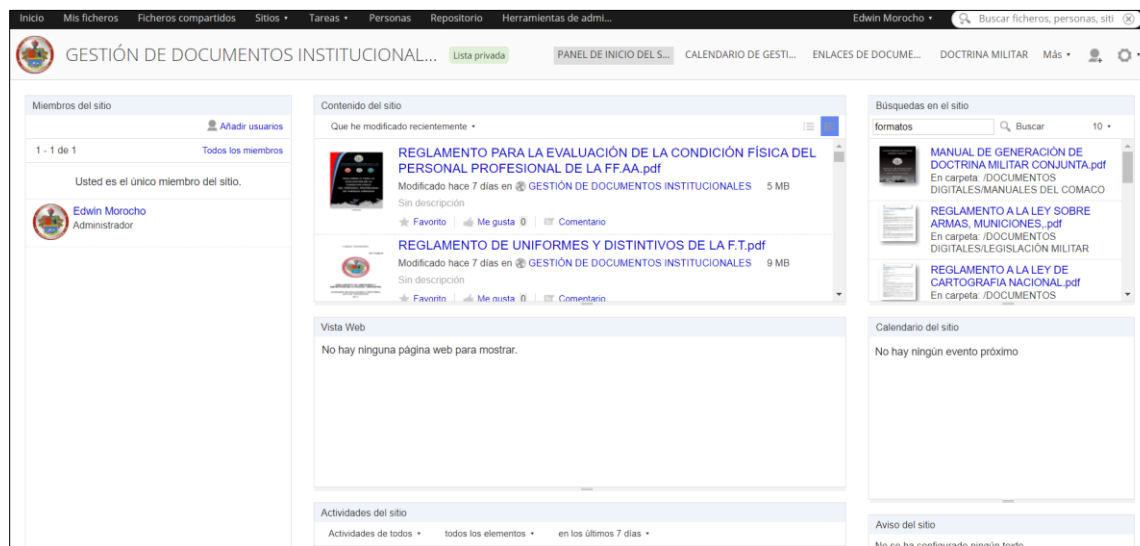
Figura 23:

Prueba de Ingreso a la Herramienta Alfresco

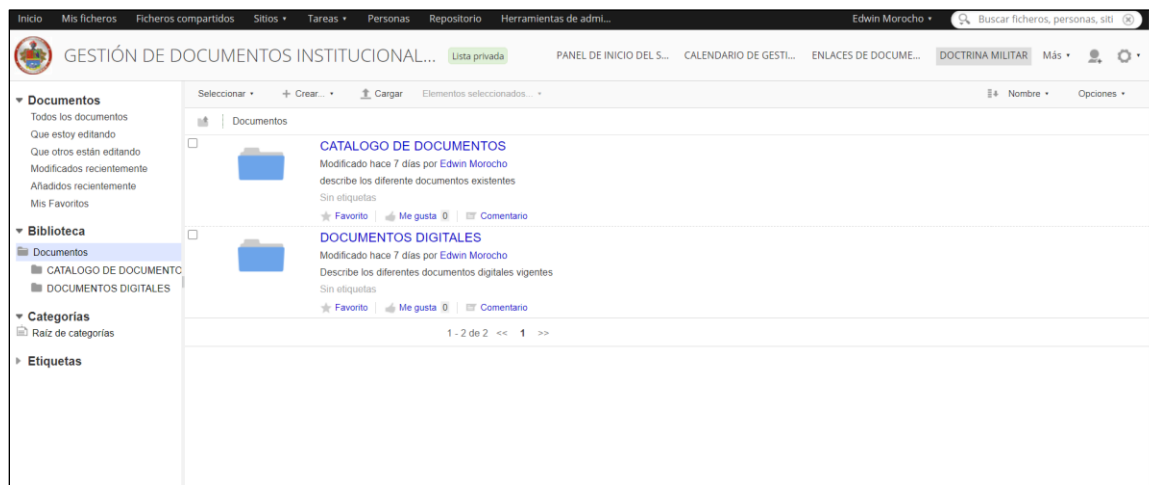


Nota: La imagen visualiza la página de inicio de sesión para utilizar los servicios de Alfresco e ingresar nuestro Usuario y Contraseña par a la Gestión de Documentos Institucionales.

Figura 24:

Visualización de Actividades

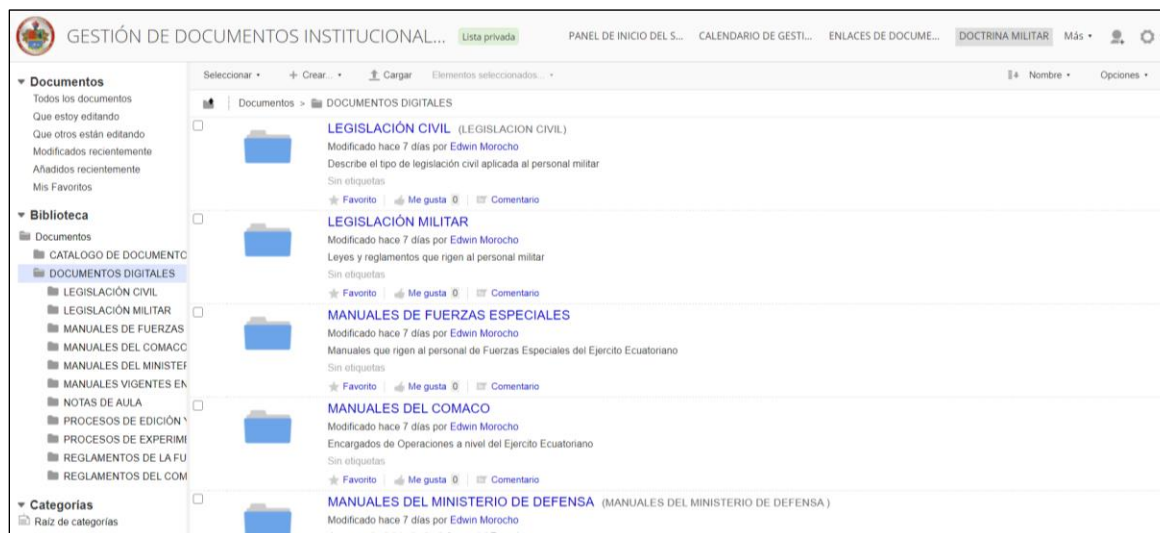
Nota: La imagen se visualiza la página de las últimas actividades realizadas por el Administrador y los usuarios del Sitio de Documentos Institucionales.

Figura 25:*Página Principal del Sitio Gestión de Documentos*

Nota: La imagen visualiza la página donde se encuentran los documentos Institucionales clasificados por sus diferentes tipos de Especificaciones como Catalogo de documentos y documentos digitales.

Figura 26:

Carpeta Principal de Documentos Digitales



Nota: La imagen visualiza las diferentes carpetas entre manuales, reglamentos, notas de aula, etc.

Figura 27:

Clasificación de los documentos digitales



Nota: La imagen describe los diferentes documentos de la carpeta legislación militar.

En la figura 27 se visualiza las diferentes carpetas clasificadas por el tipo de documentación dependiendo el tipo de documentación requerida por el Usuario.

Figura 28:

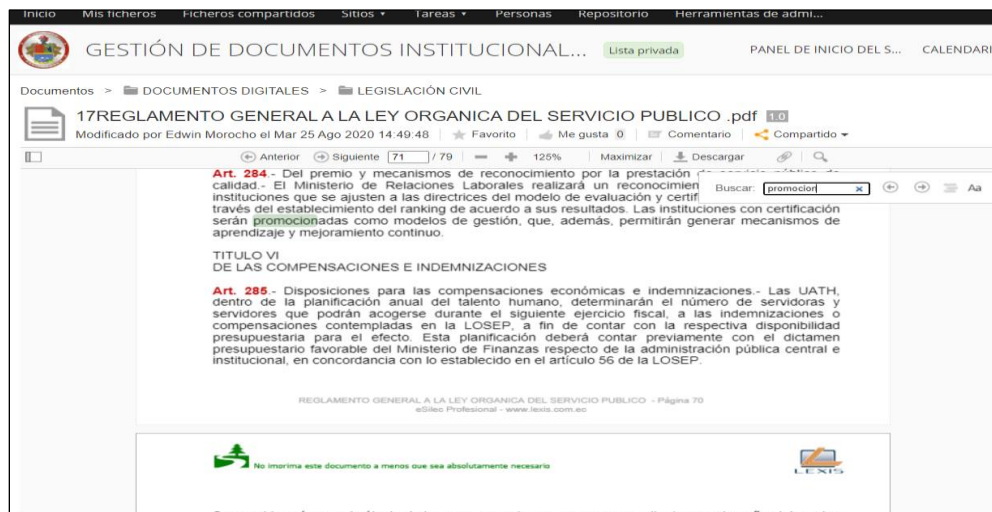
Página de descarga y de versión de documentos



Nota: La imagen describe la información del archivo donde se puede descargar, versionar y editar las propiedades de cada documento.

Figura 29:

Opción de búsqueda dentro del documento



Nota: La imagen muestra la búsqueda dentro del documento a ser imprimido.

Elaborado por autor.

Figura 30:

Propiedades y versión del documento



Nota: La imagen muestra las propiedades del documento.

En la figura 30, se verifica las propiedades, versiones, fecha de creación, modificaciones, tipo de documento, creador de dicho documento.

CAPÍTULO IV

4. COMPROBACIÓN DEL SOFTWARE

4.1. Verificación y Validación del Software

En esta parte del documento se presenta la fase de producción donde se da conocer lo que se implementa el funcionamiento del proyecto, por tal motivo cada interacción se implementa, se estabiliza y realizan las pruebas del producto, teniendo en cuenta que sea de óptima calidad y logre cumplir el ciclo de desarrollo.

4.2. Implementación del Sistema

4.2.1. Pruebas Unitarias – Sistema Web

A continuación, se verifica el eficiente funcionamiento del Sistema web por lo cual se tiene en cuenta el desempeño particular de cada función. En las pruebas unitarias se toma en cuenta el código particular de cada unidad de tal manera que se busca una estabilización en los métodos, para la página web se utilizó la herramienta Alfresco.

Tabla 7:

Pruebas Unitarias Sistema Web

Nº	PRUEBA	CUMPLE
1	Muestra Página de inicio	Si
2	Mostrar caja de texto de usuario	Si
3	Mostrar caja de texto de contraseña	Si
4	Mostrar Botón Login	Si
6	Presentar Pantalla del Sistema	Si
7	Mostrar Botón Inicio	Si
8	Mostrar Botón Sitios	Si
9	Mostrar Botón Gestión de Documentos Institucionales	Si
10	Mostrar Botón Doctrina Militar	Si
11	Mostrar Botón Añadir Usuarios	Si
12	Mostrar Opción Búsqueda en el Sitio	Si
13	Mostrar Opción Biblioteca	Si
14	Mostrar Carpeta Catalogo de Documentos	Si
15	Mostrar Carpeta Documentos Digitales	Si
16	Mostrar lista de Documentos Digitales	Si

N°	PRUEBA	CUMPLE
17	Mostrar Lista de Legislación Civil	Si
18	Mostrar Lista de Legislación Militar	Si
19	Mostrar Lista de Manuales de Fuerzas Especiales	Si
20	Mostrar Lista del Manuales del Comaco	Si
21	Mostrar Lista del Manuales del Ministro de Defensa	Si
22	Mostrar Lista de Manuales Vigentes de la FF.TT	Si
23	Mostrar Lista de Notas de Aula	Si
24	Mostrar Lista de Procesos de Edición y Difusión	Si
25	Permitir Imprimir	Si
26	Mostrar Lista de Procesos de Experimentación	Si
27	Mostrar Lista de Reglamentos de la Fuerza Terrestre	Si
28	Mostrar Lista de Reglamento del Comaco	Si
29	Permite Versionar un Documento	Si
30	Permite Buscar un Documento	Si
31	Permitir Seleccionar Carpetas	Si
32	Permite compartir los Archivos mediante un link	Si
33	Permite guardar y conservar la información segura	Si

Nota: Esta tabla muestra las pruebas unitarias. Elaborado por el autor.

4.2.2. Pruebas de Integración – Sistema Web

Ya finalizado la comprobación de las pruebas unitarias se ha comprobado la funcionalidad de cada una de ellas, por lo que se procederá a realizar las pruebas de integración, en donde se verificará de forma grupal el funcionamiento del sistema, de tal manera que se podrá constatar los resultados positivos.

Tabla 8:

Pruebas de Integración Sistema Web

N°	PRUEBA	CUMPLE
1	Permite Ingresar al Sistema	Si
2	Muestra Pantallas	Si
3	Permite subir Archivos de Documentos Institucionales	Si
4	Mostrar Información de los Documentos Existentes en el Sistema de Gestión Documental	Si
5	Permitir contacto el usuario con el administrador	Si
6	Permite al usuario Final la opción de visualizar y a su vez descargar el documento	Si

Nota: Esta tabla muestra los casos de pruebas. Elaborado por el autor.

4.2.3. Pruebas de Aceptación

Conforme a las historias de usuario realizadas, se plantea las pruebas de aceptación, con esto se va a comprobar la efectividad de la funcionalidad de cada requisito una vez que se ha implementado el Sistema Web.

Tabla 9:

Prueba de Aceptación 1: Registro de Documentos Institucionales

Prueba de aceptación 1
<p>Historia de usuario: 1</p> <p>Nombre del caso de la prueba: Registro del Documentación</p> <p>Descripción: prueba de registro del administrador.</p> <p>Condiciones de ejecución:</p> <ul style="list-style-type: none"> • Proporcionar datos del documento a ser subido a la carpeta Documentos Digitales • Tener acceso al sistema. <p>Entrada/ Pasos de ejecución: Ingresar usuario y contraseña proporcionados por el administrador.</p> <p>Resultado deseado: usuario y contraseña creada, visualización de página principal.</p> <p>Evaluación: prueba efectiva</p>

Nota: Esta tabla muestra la prueba de aceptación Registro de Documentación.

Tabla 10:

Prueba de aceptación 2: Visualización de Documentos Institucionales

Prueba de aceptación 2
<p>Historia de usuario: 2</p> <p>Nombre del caso de la prueba: Visualización del Documentación</p> <p>Descripción: prueba de visualización de los Documentos Institucionales.</p> <p>Condiciones de ejecución:</p> <ul style="list-style-type: none"> • Proporcionar datos del documento a ser visualizados por el Usuario • Tener acceso al sistema. • Descargar el documento por el Usuario Final <p>Entrada/ Pasos de ejecución: Ingresar usuario y contraseña proporcionados por el administrador.</p> <p>Resultado deseado: usuario y contraseña creada, visualización de página principal.</p> <p>Evaluación: prueba efectiva</p>

Nota: Esta tabla muestra la prueba de aceptación Registro de Documentación.

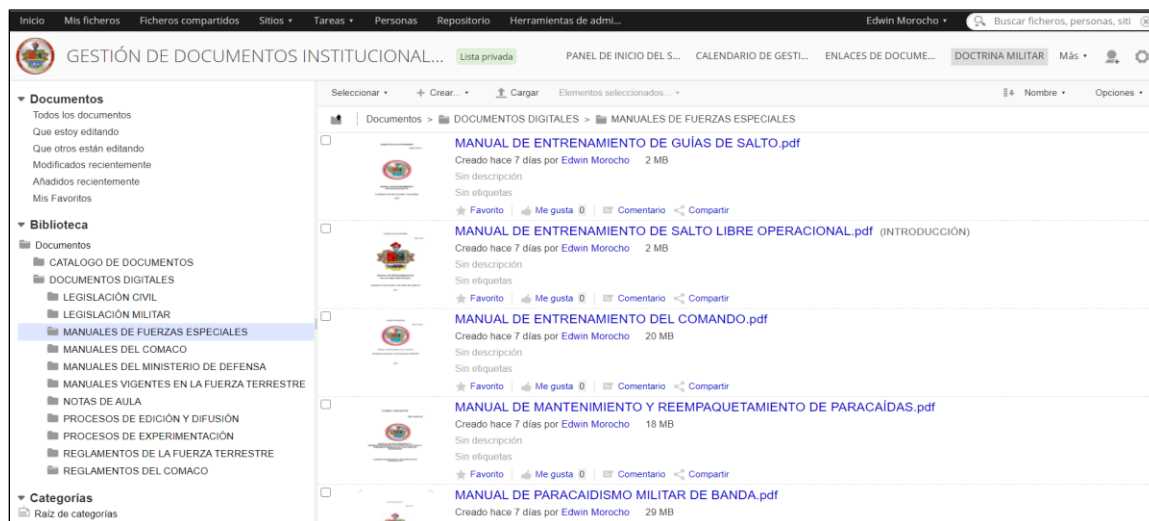
Tabla 11:*Prueba de Aceptación 3: Edición de Documentos Institucionales*

Prueba de aceptación 3
Historia de usuario: 3
Nombre del caso de la prueba: Edición del Documentación y Versionamiento
Descripción: prueba de Edición de los Documentos Institucionales.
Condiciones de ejecución: <ul style="list-style-type: none"> • Proporcionar datos del documento a ser editados por el Administrador • Tener acceso al sistema. • Edición del documento por el Administrador o colaborador
Entrada/ Pasos de ejecución: Ingresar usuario y contraseña proporcionados por el administrador.
Resultado deseado: usuario y contraseña creada, visualización de página principal.
Evaluación: prueba efectiva

Nota: Esta tabla muestra la prueba de aceptación Registro de Documentación.

4.2.4. Ejecución de Pruebas

Al presentar la ejecución de pruebas se presenta la demostración del módulo web a través de evidencias ya completamente integradas en el equipo respectivamente al módulo web que esta designada para El Comando de Educación y Doctrina del Ejercito.

Figura 31:*Prueba en el navegador Chrome*

Nota: Esta imagen muestra la prueba en el navegador. Elaborado por el autor

4.3. Validación de la Aplicación Web

4.3.1. Introducción

En este capítulo se realiza la validación del sistema mediante una encuesta donde se considera como una investigación técnica y este procedimiento se toma en cuenta por la efectividad de los resultados para calcular el impacto de validez que ofrece el módulo web de gestión de documentos Institucionales; se plantea el presente cuestionario que permite observar la calidad del software.

4.3.2. Encuesta de Validación del Proyecto

Tiene como objetivo de realizar la encuesta para llegar a conocer el impacto del funcionamiento del sistema, donde se analiza la manipulación que ejercen los usuarios mediante la gestión de los Usuarios Finales del módulo web para la Gestión de Documentos Institucionales del Centro de Doctrina Militar.

La muestra que se consideró para la encuesta del Módulo web fue de 30 personas que pertenecen al Centro de Doctrina Militar y al personal que labora en el área de Sistemas y Desarrollo, así como al personal militar que son los usuarios finales del módulo web.

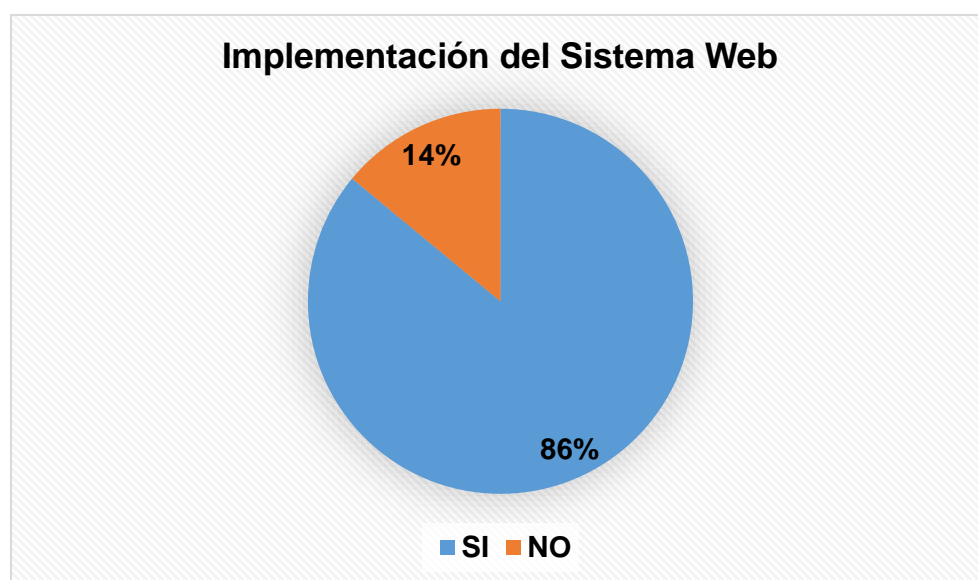
4.3.3. Análisis de Resultados de la Aplicación Web

Pregunta 1

¿Cree usted que con la implementación del módulo web va ayudar a tener lo documentos Institucionales disponibles para los Usuarios Finales?

Figura 32:

Implementación del Módulo Web



Nota: Este gráfico indica los resultados de la pregunta 1. Elaborado por el autor

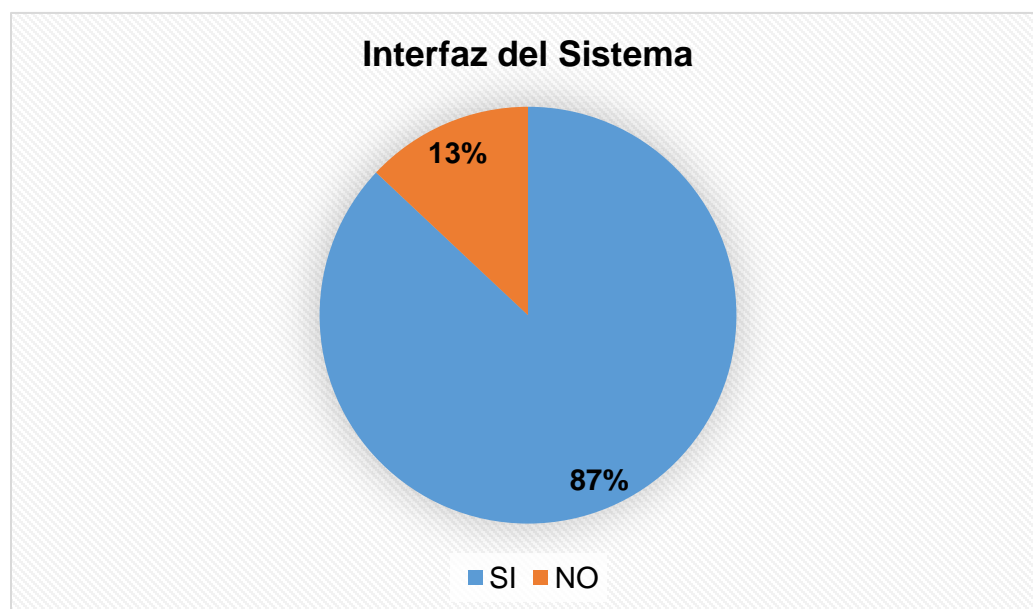
En el presente resultado se considera en un 86%, que la implementación del sistema va ayudar a mejorar la Gestión de Documentos Institucionales. De tal manera que se observa gran acogida del uso y la implementación de esta herramienta, por otra parte, existe un porcentaje del 14% que considera que el módulo no va a ser de gran ayuda.

Pregunta 2

¿Cree usted que la interfaz del módulo web es atractiva y de fácil uso para usted?

Figura 33:

Interfaz del Sistema Web



Nota: Este gráfico indica los resultados de la pregunta 2. Elaborado por el autor

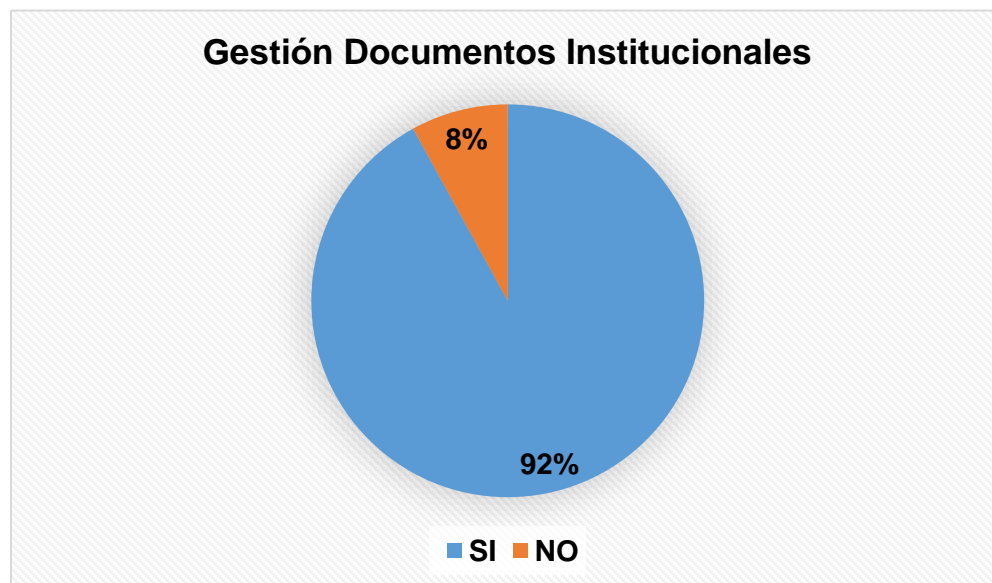
En el presente resultado se considera en un 87%, que la interfaz gráfica del módulo web es atractiva y de fácil uso. De tal manera que se observa gran acogida por parte de los usuarios, por otra parte, existe un porcentaje del 15% que considera que el módulo no es atractivo ni de fácil uso para el usuario final.

Pregunta 3

¿Cree que con la implementación del módulo web va a ayudar a mejorar la Gestión de Documentos Institucionales?

Figura 34:

Gestión de Documentos Institucionales



Nota: Este gráfico indica los resultados de la pregunta 3. Elaborado por el autor

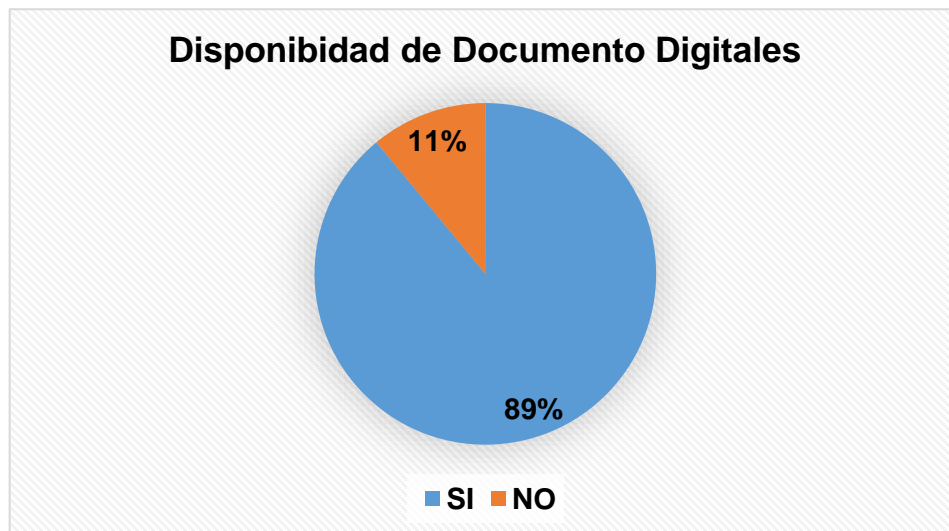
En el presente resultado se considera en un 92%, que el módulo web va a ayudar a gestionar los documentos Institucionales en el Centro de Doctrina Militar. De tal manera que se observa gran acogida por parte de los usuarios, por otra parte, existe un porcentaje del 8% que considera que el sistema no va a ser de gran ayuda.

Pregunta 4

¿Cree que con la implementación del módulo web va a ayudar a mejorar la disponibilidad de los Documentos Institucionales?

Figura 35:

Gestión de disponibilidad de documentos digitales



Nota: Este gráfico indica los resultados de la pregunta 4. Elaborado por el autor

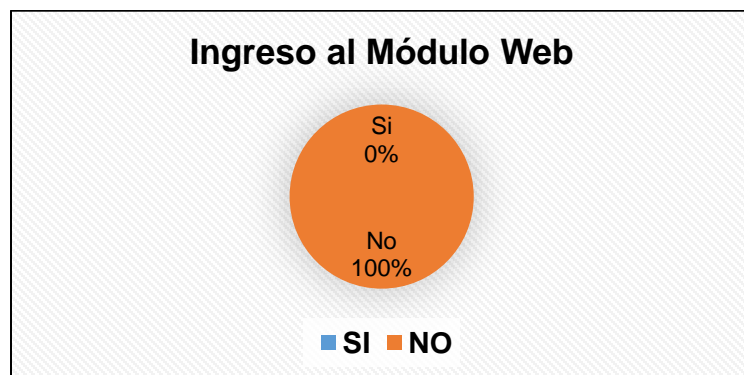
En el presente resultado se considera en un 89%, que el sistema web va a ayudar a gestionar el módulo Parte Operatorio. De tal manera que se observa gran acogida por parte de los usuarios, por otra parte, existe un porcentaje del 11% que considera que el sistema no va a ser de gran ayuda.

Pregunta 5

¿Usted logró identificar algún problema al momento de ingresar al sistema?

Figura 36:

Ingreso al Sistema



Nota: Este gráfico indica los resultados de la pregunta 5. Elaborado por el autor

En el presente resultado se considera en un 100%, que el sistema no tuvo ningún problema en el momento de realizar el ingreso al sistema.

Pregunta 6

¿Considera usted que el módulo web tiene una buena comunicación con el personal de Administradores de Usuarios?

Figura 37:

Comunicación de Administrador de Usuarios



Nota: Este gráfico indica los resultados de la pregunta 6. Elaborado por el autor

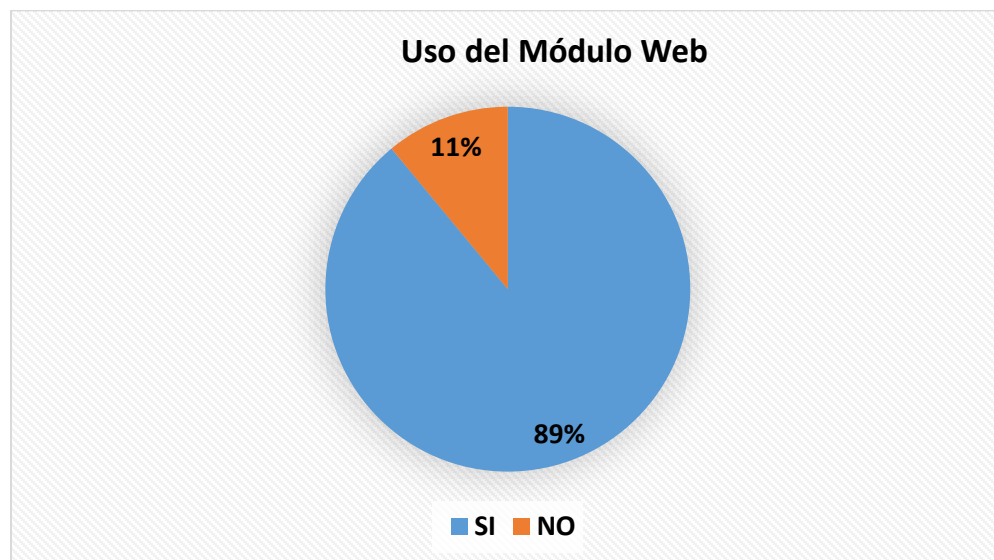
En el presente resultado se considera en un 96%, que la comunicación entre el administrador de usuarios y el personal de usuarios finales dentro del módulo web, es buena. De tal manera que se observa gran acogida por parte de los usuarios, por otra parte, existe un porcentaje del 4% no considera que la comunicación entre el administrador del módulo y los usuarios finales no es buena.

Pregunta 7

¿Usted volvería a usar el módulo web, la cual está orientada a la gestión de documentos institucionales en el Centro de Doctrina Militar?

Figura 38:

Comunicación del módulo web con el usuario final



Nota: Este gráfico indica los resultados de la pregunta 7. Elaborado por el autor

En el presente resultado se considera en un 89%, que si volverían al módulo web, la cual está orientada a la gestión de Documentos Institucionales. De tal manera que se observa gran acogida por parte de los usuarios, por otra parte, existe un porcentaje del 11% que no volverían a usar el módulo web.

CAPÍTULO V

5. CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

- La documentación que se levantó para el desarrollo del sistema de gestión Documental, ha sido fundamental. En cuanto a los requerimientos funcionales se pudo constatar que el levantamiento de información con el personal del centro de doctrina militar permitió agilizar de manera concreta para cumplir con el objetivo propuesto en el proyecto.
- La parte del sistema web se ha enfocado en crear una interfaz amigable para el usuario, empleando herramientas de desarrollo de software actuales, y la metodología Scrum mismo que ha permitido que se plasme por medio de la programación todas las ideas recopiladas por parte del personal militar del Centro de Doctrina Militar y del Centro de Desarrollo de Software.
- Las pruebas de todo el sistema en su totalidad están realizadas de acuerdo a los requisitos levantados. De la misma manera la seguridad implementada se maneja con políticas, protocolos y normas establecidas de acuerdo a lo que estipula los reglamentos de manejo de datos informáticos. Por otra parte, el software que se ha desarrollado está listo y en condiciones de ser manipulado por el personal del Centro de Doctrina Militar

5.2 Recomendaciones

- Se recomienda el correcto uso del levantamiento de información, será de gran importancia ya que permitirá el fácil levantamiento de requerimientos para el desarrollo del módulo web.
- Se recomienda la implementación del presente modulo, considerando sus características principales, en cuanto a la escalabilidad, flexibilidad, seguridad, y disponibilidad, los cuales pueden mejorar la Gestión de Documentos Institucionales en el comando de educación y doctrina del Ejército

- Se recomienda que las pruebas de funcionalidad se realicen en usuarios finales que no tengan conocimientos de uso de software, para comprobar si el sistema de gestión documental es amigable e incluso fácil para su manejo.

REFERENCIAS BIBLIOGRÁFICAS

- Apache Netbeans. (2019). The Apache Software Foundation. Retrieved from The Apache Software Foundation: <https://netbeans.apache.org/download/nb120/index.html>
- Arteaga , L. M. (2019, Septiembre 15). gnu.org. Retrieved from El Sistema Operativo GNU: <https://www.gnu.org/philosophy/free-sw.es.html>
- Brocca, J. C., & Camasaqui, R. (2005). Las Licencias de Software Desde las Perspectivas del Usuario Final. Pilquen-Sección Ciencias Sociales(7), 1-10.
- Cabello Garcia, J. M. (2015). Almacenamiento de la información e introducción a los SGBD. Andalucía, España: IC Editorial. Retrieved from <https://elibro.net/es/ereader/espe/44147?page=45>
- Casado Iglesias, C. (2015). Entornos de Desarrollo. Madrid, España: RA-MA.SA Editorial. Retrieved from <https://elibro.net/es/ereader/espe/62495?page=45>
- Cassá, J. M. (2012). Programación web en Java. España: Ministerio de Educación.
- Center, I. k. (2006). MODULO WEB. Retrieved from IBM : https://www.ibm.com/support/knowledgecenter/es/SSEQTP_9.0.5/com.ibm.websphere.base.doc/ae/cweb_aov3.html
- Codina, L. (2013, Enero 16). El profesional de la Información. Retrieved from www.elprofesionaldelainformación.com
- Cristóbal Pareja, Á. A. (1994). Introducción a la Informática.
- De la Hoz Correa, E. M.-L.-H.-F. (2009.). Linux ubuntu server. Barranquilla: – Editorial Corporación Universitaria de la Costa.
- EAE Business School. (2020). Eaeprogramas.es. Retrieved 08 11, 2020, from Eaeprogramas.es: <https://www.eaeprogramas.es/blog/que-es-el-software-propietario-0>
- Free Software Foundation. (2019, 09). Gnu.org. Retrieved 08 10, 2020, from Gnu.org: <https://www.gnu.org/philosophy/free-sw.es.html>

- Garrido, J. S. (2004). Arquitectura y diseño de sistemas web modernos. Murcia(españa): Ilustre Colegio de Ingenieros en Informática de Murcia.
- Ginestá, M. G. (2012). Bases de datos en PostgreSQL. California: PostgreSQL Global Development Group.
- Gómez de León, F. C. (1998). Tecnología del Mantenimiento Industrial. Murcia: Servicio de Publicaciones, Universidad de Murcia.
- Gonzalo Soto & Nuria. (2018). Desarrollo de una API para datos abiertos. la laguna: Departamento de Ingeniería Informática y de Sistemas de la Universidad.
- Hattori, H. M. (2007).
- Letelier & Penades. (2006). Metodologías Ágiles.
- Luján Mora , S. (2002). Programación de aplicaciones web: historia, principios básicos y clientes web. España: Editorial Club Universitario.
- Maldonado, J. A. (2015). Fundamentos de la calidad total.
- Manzaba, G. (2014). Extreme Programming xp.
- Mísfuf Talón, E. (2013.). Apache. España: Ministerio de Educación de España. Retrieved from <https://elibro.net/es/ereader/espe/49359?page=18>
- Miller, F. V. (2010). Apache Maven. California: Digital Library.
- Moreno Pérez, J. C. (2015). Programación. RA-MA Editorial. Retrieved from <https://elibro.net/es/ereader/espe/62476?page=12>
- Moreno, J. C., & Ramos, A. F. (2014). Administración Software de un Sistema Informático. Madrid: RA-MA. SA Editorial y Publicaciones.
- Navarro Bustos, J. R. (2012). Implementación del Sistema de Gestión Documental Alfresco para la Facultad de Ingeniería en Geología, Minas, Petróleos y Ambiental de la Universidad Central del Ecuador. Quito: Quito: UCE.
- OpenWebinars. (n.d.). OpenWebinars. Retrieved from OpenWebinars: <https://openwebinars.net>

- Ordax Cassá, J. M. (2013). Programación web en Java. Madrid, España: Gobierno de España.
- Pachano, M. I. (2010). El desarrollo de la gestión documental que permita su fácil acceso en la dirección de planificación del H. Gobierno Provincial de Tungurahua en el Primer semestre del año 2010. Ambato: Universidad Técnica de Ambato.
- Pressman, R. (2010). Ingeniería del Software Un Enfoque Práctico. México: McGraw-Hill.
- Silberschatz, A., Korth, H. F., & Sudarshan, F. (2002). Fundamentos de bases de datos. Madrid: McGraw-Hill/Interamericana de España.
- Sutherland, S. y. (2016). WORKFLOW PARA LA INSTITUCIONALIZACIÓN. Universidad Andres Bello.
- Texier, J. D. (2013). DSpace como herramienta para un repositorio de documentos administrativos en la Universidad Nacional Experimental del Táchira. Medellín: Revista Interamericana de Bibliotecología.
- Tridibesh Satpathy. (2013). CONOCIMIENTO DE SCRUM. USA.
- TutotialsPoint . (2019). TutorialsPoint. Retrieved from TutorialsPoint: www.tutorialspoint.com
- Zapata. (2012). Gestión Documental. México.

ANEXOS