

**ANÁLISIS Y ESTRUCTURACIÓN PARA LA IMPLEMENTACIÓN DE UNA APLICACIÓN WEB S.I.A.GE.M.I.**

**MEMORIA DE ESTADÍA PROFESIONAL**

**R E P O R T E T É C N I C O**

PARA OBTENER EL TÍTULO DE

**INGENIERÍA EN DESARROLLO Y GESTIÓN DE SOFTWARE**

*P R E S E N T A*

**HORACIO SOSA HERNÁNDEZ**

ASESOR DE LA ORGANIZACIÓN: Tte. F.A.E.E.A. ALLAN RICARDO RUIZ GÓMEZ

ASESORA ACADÉMICA: DRA. ARACELI GARCÍA AGUILAR

ORGANIZACIÓN: “COMPLEJO LOGÍSTICO DE LA FUERZA AÉREA MEXICANA (C.I.D.T.F.A.)”  
GENERACIÓN: ENERO 2023-AGOSTO 2024

CUATRIMESTRE DE TÉRMINO: MAYO-AGOSTO 2024

**CARTA DE AUTORIZACIÓN DE DIGITALIZACIÓN DE REPORTE TÉCNICO PARA ARCHIVO ELECTRONICO DEL REPORTE TÉCNICO**

**AGRADECIMIENTOS Y DEDICATORIA**

**ÍNDICE**

[RESUMEN 5](#_Toc170389982)

[ABSTRACT 6](#_Toc170389983)

[INTRODUCCIÓN 7](#_Toc170389984)

[OBJETIVOS 9](#_Toc170389985)

[PROGRAMA Y CRONOGRAMA 10](#_Toc170389986)

[MARCO TEÓRICO 15](#_Toc170389987)

[METODOLOGÍA 19](#_Toc170389988)

[CAPITULO 1. LEVANTAMIENTO DE INFORMACIÓN Y ANALISIS DE REQUERIMIENTOS 21](#_Toc170389989)

[1.1 Historia de la Fuerza Aérea Mexicana 21](#_Toc170389990)

[1.2 Atribuciones y responsabilidades 23](#_Toc170389991)

[1.3 Identificación de necesidades del cliente 24](#_Toc170389992)

[1.3.1 Definición de requerimientos funcionales 25](#_Toc170389993)

[1.3.2. Definición de requerimientos no funcionales 26](#_Toc170389994)

[1.3 Casos de uso 27](#_Toc170389995)

[1.4 Diagrama de secuencia 31](#_Toc170389996)

[1.5 Diagrama de clase 32](#_Toc170389997)

[CAPITULO 2. DISEÑO Y CREACIÓN DE LA BASE DE DATOS. 36](#_Toc170389998)

[2.1 Diagrama Entidad-Relación 36](#_Toc170389999)

[2.2 Modelo Relacional 37](#_Toc170390000)

[2.3 Diccionario de datos 38](#_Toc170390001)

[CAPITULO 3. DISEÑO DE INTERFACES 46](#_Toc170390002)

[CONCLUSIONES 47](#_Toc170390003)

[LISTADO DE SIGLAS O ACRÓNIMOS 48](#_Toc170390004)

[GLOSARIO 51](#_Toc170390005)

[REFERENCIAS 54](#_Toc170390006)

# RESUMEN

# ABSTRACT

# INTRODUCCIÓN

El Centro de Investigación y Desarrollo Tecnológico de la Fuerza Aérea (C.I.D.T.F.A.) es una institución esencial para el soporte operativo y tecnológico de la Fuerza Aérea Mexicana. Su responsabilidad incluye asegurar el óptimo funcionamiento de infraestructuras críticas como redes eléctricas, hidráulicas e hidrosanitarias que sustentan las operaciones diarias.

Sin embargo, se ha identificado una oportunidad de mejora en la gestión y calendarización de los mantenimientos preventivos y correctivos de estas infraestructuras. La falta de una planificación precisa en los tiempos de mantenimiento puede llevar a interrupciones no planificadas, incrementos en los costos operativos y una eficiencia subóptima.

Para abordar estas áreas de mejora, se propone el desarrollo de una aplicación de software especializada que optimice la gestión del mantenimiento tanto preventivo como correctivo. La gestión adecuada del mantenimiento es fundamental para garantizar que las infraestructuras funcionen de manera eficiente y segura.

El mantenimiento preventivo anticipa posibles fallos mediante intervenciones programadas, mientras que el mantenimiento correctivo responde a fallos imprevistos, requiriendo respuestas rápidas para minimizar el impacto en las operaciones.

El Sistema Integral de Administración y Gestión de Mantenimiento de Instalaciones (S.I.A.GE.M.I.) surge como una solución para enfrentar la falta de calendarización en los mantenimientos del C.I.D.T.F.A. Utilizando Java y PostgreSQL, este sistema busca optimizar la gestión de mantenimiento de las redes críticas, mejorando la eficiencia operativa y reduciendo costos. La implementación de esta solución tecnológica es esencial para garantizar una operatividad más eficiente y fiable, asegurando que la Fuerza Aérea Mexicana pueda cumplir su misión de manera efectiva y segura.

En el capítulo I se muestra el análisis realizado a la información proporcionada por la empresa.

En el capítulo II se desarrolla el diseño de la base de datos para el sistema.

Y, por último, en el capítulo III se presenta el diseño de las interfaces de la aplicación web, su codificación, y las pruebas realizadas para garantizar el cumplimiento de los requerimientos funcionales.

# OBJETIVOS

Objetivo general

Analizar y estructurar una aplicación web para la gestión del mantenimiento de las redes eléctricas, hidráulicas e hidrosanitarias en el (C.I.D.T.F.A.), con el fin de optimizar el control de los recursos y reducir tiempos.

Objetivos específicos

* Seleccionar la metodología de desarrollo para la construcción del software.
* Determinar los requerimientos funcionales y no funcionales para dimensionar el sistema.
* Definir la estructura de la base de datos para el uso de la aplicación web.

# PROGRAMA Y CRONOGRAMA

|  |
| --- |
| ***UNIVERSIDAD TECNOLÓGICA DE TECÁMAC*** |
| ***DIVISIÓN TIC*** |
| ***PROGRAMA DE ESTADÍAS PROFESIONALES*** |

|  |  |  |
| --- | --- | --- |
| ***PROGRAMA DE TRABAJO*** |  | |
|  | | *FECHA: 02/05/2024* |

*DATOS DEL ESTUDIANTE*

|  |  |
| --- | --- |
| NOMBRE: | Horacio Sosa Hernández |
| DIVISIÓN: | Tecnologías de la Información y Comunicación |
| CARRERA: | Ingeniería en desarrollo y gestión de software |
| MATRÍCULA: | 2521160068 |
| GENERACIÓN: | Enero 2023 – Agosto 2024 |

*ASESORA ACADÉMICA*

|  |  |
| --- | --- |
| NOMBRE: | Araceli García Aguilar |
| CARGO: | Profesora de Tiempo Completo Asociada C |

*DATOS DE LA ORGANIZACIÓN*

|  |  |
| --- | --- |
| NOMBRE DE LA ORGANIZACIÓN: | Complejo Logístico de la Fuerza Aérea Mexicana (C.I.D.T.F.A.) |
| DEPARTAMENTO: | Centro de Investigación y Desarrollo Tecnológico de la Fuerza Aérea (C.I.D.T.F.A.) |
| ÁREA: | Departamento de Electrónica Aviación |
| DIRECCIÓN: | Carretera Federal México - Pachuca s/n, Santa Lucia, Zumpango, Edo. de México C.P. 55640 |
| TELÉFONO: | 5969241295 |
| E-MAIL: | complog.dn19@sedena.gob.mx |

*ASESOR DE LA ORGANIZACIÓN*

|  |  |
| --- | --- |
| NOMBRE: | Allan Ricardo Ruiz Gómez |
| CARGO: | Tte. F.A.E.E.A. |

*PERÍODO*

|  |  |
| --- | --- |
| DURACIÓN: | 15 semanas |
| FECHA DE INICIO: | 02 de mayo de 2024 |
| FECHA DE TERMINACIÓN: | 09 de agosto de 2024 |
| HORARIO: | 8:00 am – 4:00 pm |

*PROYECTO*

|  |  |
| --- | --- |
| NOMBRE: | Análisis y Estructuración para la Implementación de una Aplicación Web S.I.A.GE.M.I. |
| DESCRIPCIÓN: | Desarrollar una aplicación web para el departamento del C.I.D.T.F.A., con el propósito de llevar el control de mantenimiento de las tres redes existentes: eléctrica, hidráulica e hidrosanitaria. Esta aplicación proporcionará una solución integral para la gestión eficiente de los mantenimientos preventivos y correctivos, permitiendo la programación de actividades, registro de incidencias, seguimiento de tareas y generación de reportes. |
| OBJETIVO GENERAL: | Analizar y estructurar una aplicación web para la gestión del mantenimiento de las redes eléctricas, hidráulicas e hidrosanitarias en el (C.I.D.T.F.A.), con el fin de optimizar el control de los recursos y reducir tiempos. |
| OBJETIVOS ESPECÍFICOS: | Seleccionar la metodología de desarrollo para la construcción del software.  Determinar los requerimientos funcionales y no funcionales para dimensionar el sistema.  Definir la estructura de la base de datos para el uso de la aplicación web. |
| ALCANCE(S): | El Sistema Integral de Administración y Gestión de Mantenimiento de Instalaciones (S.I.A.GE.M.I.) se desarrollará en un periodo de cuatro meses, cubriendo las siguientes actividades clave:  **Documentación del Proyecto**   * **Recolección de Requisitos**: Entrevistas y encuestas con el personal técnico y administrativo para identificar necesidades y problemas. * **Especificación de Requisitos del Sistema**: Definición de requisitos funcionales y no funcionales, y creación de diagramas de casos de uso.   **Diseño de la Base de Datos**   * **Modelado de Datos**: Creación del diagrama entidad-relación (ER). * **Desarrollo del Diagrama Relacional**: Definición de tablas y relaciones en PostgreSQL. * **Elaboración del Diccionario de Datos**: Documentación de atributos, tipos de datos y restricciones de cada tabla.   **Construcción de Interfaces de Usuario**   * **Diseño de Interfaz de Usuario (UI)**: Creación de wireframes y prototipos interactivos. * **Desarrollo del Frontend**: Implementación de interfaces de usuario usando HTML, CSS, JavaScript. |
| META(S): | Desarrollar una aplicación web que permita al (C.I.T.F.A.) gestionar de manera eficiente el mantenimiento de sus redes eléctricas, hidráulicas e hidrosanitarias, mejorando la planificación, optimizando recursos y garantizando la seguridad y disponibilidad de las instalaciones. |
| RECURSOS: | Hardware:   * Procesador Intel® CoreTM i3-7100U * Tarjeta gráfica integrada: Intel® HD Graphics 620 * Resolución de FHD (1920x1080) * RAM 8 GB * SSD 256 GB / HDD 1 TB   Software:   * Sistema Operativo Windows 10 o superior, 64 bits * Netbeans IDE 12.5 * Navegador Google Chrome y Firebox * PosgrestSQL   Lenguaje de programación Java |

*PLAN DE TRABAJO*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ACTIVIDAD | | DESCRIPCIÓN | SEMANA | | FECHAS | |
| INICIO | TÉRMINO | INICIO | TÉRMINO |
| 1 |  | Recolección y refinamiento de requisitos | 1 | 5 | 02/05/2024 | 30/05/2024 |
|  | 1.1 | Historia de la Fuerza Aérea | 1 | 3 |  |  |
|  | 1.2 | Identificación de las necesidades del cliente | 1 | 3 |  |  |
|  | 1.2.1 | Definición de requerimientos funcionales | 1 | 3 |  |  |
|  | 1.2.2 | Definición de requerimientos no funcionales | 1 | 3 |  |  |
|  | 1.3 | Casos de uso | 3 | 4 |  |  |
|  | 1.4 | Diagrama de secuencia | 3 | 5 |  |  |
|  | 1.5 | Diagrama de clases | 3 | 5 |  |  |
| 2 |  | Diseño y Creación de la base de datos | 6 | 9 | 03/6/2024 | 05/07/2024 |
|  | 2.1 | Diseño Entidad-Relación | 6 | 8 |  |  |
|  | 2.2 | Modelo Relacional | 7 | 8 |  |  |
|  | 2.3 | Diccionario de datos | 7 | 8 |  |  |
|  | 2.4 | Mapa de navegación | 8 | 9 |  |  |
| 3 |  | Diseño de Interfaces y Codificación | 10 | 14 | 08/07/2024 | 07/08/2024 |
|  | 3.1 | Diseño de interfaz de Login | 10 | 14 |  |  |
|  | 3.2 | Diseño de las interfaces según el rol del usuario | 10 | 14 |  |  |
|  | 3.3 | Codificación | 10 | 14 |  |  |

*CRONOGRAMA DE ACTIVIDADES (PROGRAMA)*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| # | ACTIVIDADES | CONTROL | Mes 1 | | | | | Mes 2 | | | | Mes 3 | | | | Mes 4 | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 1 | Análisis de los requerimientos del cliente | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Elaboración de Especificaciones de Requisitos del Software y Diagramas | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Diseño y creación de la base de datos | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Diseño de interfaces | PROG. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| REAL |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

*FIRMAS*

|  |  |
| --- | --- |
| Tte. F.A.E.E.A. Allan Ricardo Ruiz Gómez  *ASESOR DE LA ORGANIZACIÓN* | |
| Horacio Sosa Hernández  *ESTUDIANTE* | Dra. Araceli García Aguilar  *ASESORA ACADÉMICA* |

# MARCO TEÓRICO

En este marco teórico se encuentran la terminología que se usó para la construcción y desarrollo del proyecto, con la finalidad de comprender los procesos y fases por los que pasó el mismo.

JAVA.

Java es un lenguaje de programación de propósito general, altamente popular y ampliamente utilizado desde su creación en 1995. Destaca por su portabilidad, seguridad y robustez. Java es orientado a objetos y ofrece un amplio conjunto de características que lo hacen adecuado para una variedad de aplicaciones, desde desarrollo web hasta aplicaciones móviles. Su capacidad de escribir una vez y ejecutar en cualquier lugar, junto con su comunidad activa y evolución constante, lo convierten en una opción atractiva para desarrolladores y empresas en todo el mundo.

PosgrestSQL.

PostgreSQL es un sistema de gestión de bases de datos relacional de código abierto, altamente valorado por su robustez, escalabilidad y extensibilidad. Desde su creación en 1986, ha sido desarrollado por una comunidad global. Sus características clave incluyen su capacidad relacional estándar, código abierto, extensibilidad, escalabilidad, confiabilidad y seguridad, así como su soporte para tipos de datos avanzados.

PostgreSQL es ampliamente utilizado en una variedad de aplicaciones, desde pequeñas empresas hasta grandes sistemas empresariales, debido a su combinación de características avanzadas y rendimiento sólido.

Modelo vista controlador (MVC).

Es un patrón de arquitectura de software basado en la reutilización de códigos y separación de conceptos mediante el cual se busca separar los datos, la lógica de negocio y la interacción del usuario, de forma tal que los componentes que manejan la información sean distintos a los que interactúa con el usuario de la aplicación.

El MVC está compuesto por 3 elementos como se puede observar en la Ilustración 3:

* Modelo: Representa la información e implementa los distintos métodos de acceso para consultarla o modificarla e indicando privilegios y condiciones según la lógica de negocio.
* Vista: Representa la información del modelo en un formato entendible por el usuario.
* Controlador: Interactúa sobre el modelo a través de los métodos de acceso del mismo y envía una respuesta a la vista según las peticiones del usuario.

JDK.

Java Development Kit (JDK) es un software para los desarrolladores de Java. Incluye el intérprete Java, clases Java y herramientas de desarrollo Java (JDT): compilador, depurador, desensamblador, visor de applets, generador de archivos de apéndice y generador de documentación.

El JDK le permite escribir aplicaciones que se desarrollan una sola vez y se ejecutan en cualquier lugar de cualquier máquina virtual Java. Las aplicaciones Java desarrolladas con el JDK en un sistema se pueden usar en otro sistema sin tener que cambiar ni recompilar el código. Los archivos de clase Java son portables a cualquier máquina virtual Java estándar

Java SE.

Java Platform, Standard Edition (Java SE) es una especificación que describe una plataforma Java de resumen. Proporciona una base para crear y desplegar aplicaciones de negocio centradas en la red que van desde un ordenador de escritorio PC a un servidor de grupo de trabajo. Java SE lo implementa el kit de desarrollo de software (SDK) Java. Rule Execution Server puede ejecutar conjuntos de reglas con código Java SE 100%. Muchos casos de uso existen para la ejecución pura de Java SE como, por ejemplo, la ejecución de lotes o la ejecución de reglas desde un proveedor Java Message Service (JMS) o un Enterprise Service Bus (ESB) no Java EE.

Microsoft Visio.

Microsoft Visio es una herramienta de diagramación y visualización de datos desarrollada por Microsoft. Ofrece una amplia variedad de plantillas predefinidas para crear diagramas profesionales, como diagramas de flujo, organigramas y diagramas de red. Su interfaz intuitiva facilita la creación y edición de diagramas, mientras que su integración con otras aplicaciones de Microsoft permite incrustar diagramas en documentos de Word, hojas de cálculo de Excel o presentaciones de PowerPoint. Visio también ofrece capacidades de colaboración en tiempo real y opciones avanzadas de personalización para adaptarse a las necesidades específicas de los usuarios.

Figma.

Figma es una herramienta en la nube para diseñar interfaces digitales como aplicaciones web y móviles. Permite a los equipos crear gráficos vectoriales, prototipos interactivos y colaborar en tiempo real en proyectos de diseño. Es conocida por su facilidad de uso, capacidad de colaboración simultánea y herramientas para crear prototipos que ayudan a validar diseños antes de la implementación.

HTML.

HTML son las siglas en inglés de HyperText Markup Lenguage, que significa Lenguaje de Marcado de HiperTexto. Se llama así al lenguaje de programación empleado en la elaboración de páginas Web, y que sirve como estándar de referencia para la codificación y estructuración de las mismas, a través de un código del mismo nombre (html).  
Es un lenguaje simple y general que sirve para definir otros lenguajes que tienen que ver con el formato de los documentos. El texto se crea a partir de etiquetas también llamado tags, que permiten interconectar diversos conceptos y formatos.

JavaScript.

JavaScript es un lenguaje de programación o de secuencias de comandos que te permite implementar funciones complejas en páginas web, cada vez que una página web hace algo más que sentarse allí y mostrar información estática para que la veas, muestra oportunas actualizaciones de contenido, mapas interactivos, animación de Gráficos 2D/3D, desplazamiento de máquinas reproductoras de vídeo, etc., puedes apostar que probablemente JavaScript está involucrado. Es la tercera capa del pastel de las tecnologías web estándar, dos de las cuales (HTML y CSS) hemos cubierto con mucho más detalle en otras partes del Área de aprendizaje.

CSS.

CSS son las siglas en inglés para «hojas de estilo en cascada» (Cascading Style Sheets). Básicamente, es un lenguaje que maneja el diseño y presentación de las páginas web, es decir, cómo lucen cuando un usuario las visita. Funciona junto con el lenguaje HTML que se encarga del contenido básico de las páginas.

Se les denomina hojas de estilo «en cascada» porque puedes tener varias hojas y una de ellas con las propiedades heredadas (o «en cascada») de otras.

NetBeans.

NetBeans es un entorno de desarrollo integrado (IDE) versátil y potente, utilizado principalmente para crear aplicaciones en Java y otros lenguajes de programación. Ofrece características como completado automático de código, depuración integrada, refactorización, y soporte para control de versiones.

Además, incluye un diseñador de interfaces gráficas que facilita la creación de interfaces de usuario visualmente atractivas, especialmente para aplicaciones JavaFX. NetBeans es de código abierto y cuenta con una amplia comunidad de usuarios y desarrolladores que proporcionan recursos y soporte adicional.

Aplicación web.

Una aplicación web es un programa de software que se ejecuta en un servidor y se accede a través de un navegador web mediante Internet o una intranet. Combina la funcionalidad interactiva de las aplicaciones de escritorio con la facilidad de acceso de las aplicaciones basadas en la web. Las aplicaciones web permiten a los usuarios realizar tareas específicas (como gestionar el mantenimiento de infraestructuras) desde cualquier lugar con conexión a Internet, ofrecen actualizaciones centralizadas, integran sistemas y servicios diversos, y son escalables para manejar un número creciente de usuarios y tareas sin perder rendimiento.

# METODOLOGÍA

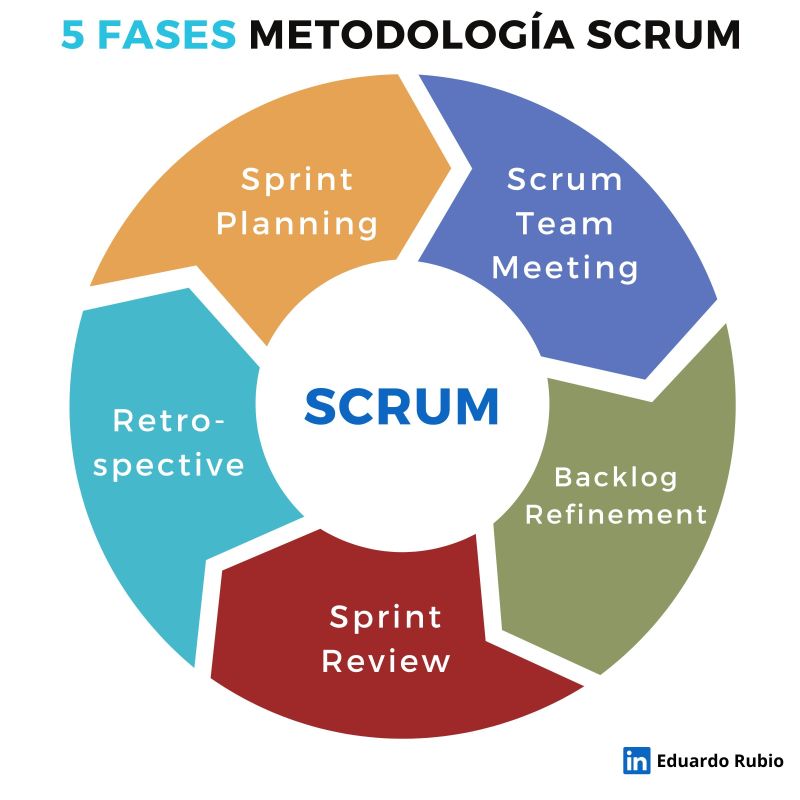
Una metodología de desarrollo es un campo de trabajo en el cual se busca analizar, planificar y ejecutar un proceso detallado de la realización de producto, “En el desarrollo de *software*, una metodología hace cierto énfasis al entorno en el cual se plantea y estructura el desarrollo de un sistema” (Ok Hosting). La implementación de una metodología en un desarrollo de *software* proporciona grandes beneficios al proyecto mejorando la calidad y productividad dentro del entorno de trabajo.

El proyecto fue desarrollado utilizando la metodología Scrum, un marco ágil que facilita la colaboración y mejora la eficiencia en el desarrollo de software. Scrum se basa en iteraciones cortas y ciclos de retroalimentación constantes, permitiendo una entrega incremental del producto y adaptaciones rápidas a los cambios.

Scrum consta de varias fases clave, que son:

1. **Planificación del Sprint**: Definición de los objetivos y tareas del sprint (un período de trabajo corto, generalmente de 2 a 4 semanas). El equipo selecciona los elementos del backlog del producto que se abordarán en el sprint.
2. **Sprint**: Período de desarrollo durante el cual el equipo trabaja en las tareas planificadas. Las tareas se dividen en incrementos manejables y se completan durante el sprint.
3. **Daily Scrum**: Reuniones diarias cortas (de 15 minutos) donde el equipo revisa el progreso, discute los desafíos y ajusta el plan de trabajo diario para asegurar que el sprint se mantenga en el camino correcto.
4. **Revisión del Sprint**: Al final de cada sprint, el equipo presenta el incremento de producto desarrollado a los stakeholders, recibiendo retroalimentación y ajustando el backlog del producto según sea necesario.
5. **Retrospectiva del Sprint**: Reflexión del equipo sobre el sprint concluido, identificando mejoras en el proceso y planificando acciones para optimizar la eficiencia y la colaboración en futuros sprints.

A continuación, se presenta una imagen que ilustra visualmente las fases de Scrum:



**Figura 1.** Etapas de la metodología Scrum

# CAPITULO 1. LEVANTAMIENTO DE INFORMACIÓN Y ANALISIS DE REQUERIMIENTOS

En este capítulo se da a conocer la parte de inicial de este proyecto, en donde se enfocará a las necesidades del cliente y un fácil entendimiento para el administrador de la aplicación web, así mismo contemplando las aplicaciones que se utilizarán para su desarrollo como lo son Java, posgrestSQL y NetBeans.

## 1.1 Historia de la Fuerza Aérea Mexicana

La Fuerza Aérea Mexicana (F.A.M.) es una institución militar encargada de la defensa del espacio aéreo mexicano, así como de llevar a cabo acciones que promuevan la seguridad nacional y el bienestar del país desde el ambiente aeronáutico. Sus antecedentes se remontan a principios del siglo XX, cuando el avance tecnológico en el campo de la aviación comenzó a tener un impacto significativo en el ámbito militar.

El interés por la aviación militar en México se manifestó de manera concreta durante la Revolución Mexicana (1910-1920), un periodo de intensos conflictos armados en el que diversos líderes políticos y militares buscaron el control del país. Durante este periodo, figuras como Venustiano Carranza y Álvaro Obregón reconocieron el potencial estratégico de la aviación y comenzaron a incorporar aviones en sus fuerzas armadas para tareas de reconocimiento, transporte y bombardeo.

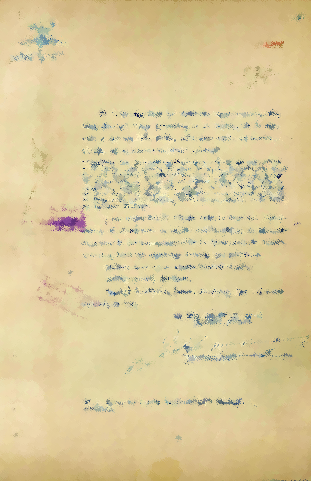
El 10 de febrero de 1915, durante la Revolución Mexicana, el piloto estadounidense Patrick Henry Scott y el teniente Miguel Lebrija realizaron el primer vuelo de reconocimiento militar en México, utilizando un avión Curtiss JN-3. Este hecho marcó un hito en la historia de la aviación militar mexicana y sentó las bases para el desarrollo futuro de la Fuerza Aérea Mexicana. Figura 1.1.



**Figura 1.1** Avión Curtiss JN-3.

El 5 de febrero de 1917, el presidente Venustiano Carranza expidió un decreto mediante el cual se creaba el Servicio Aéreo Militar de la Nación, precursor directo de la Fuerza Aérea Mexicana. Este decreto estableció las bases para la organización y operación de la aviación militar en México, aunque aún no se consolidaba como una institución independiente. Fue hasta el 10 de febrero de 1919 cuando se creó oficialmente la Fuerza Aérea Mexicana como una institución independiente, mediante un decreto del presidente Venustiano Carranza. Este decreto estableció la organización, estructura y funciones de la nueva Fuerza Aérea, que se convirtió en una parte integral de las Fuerzas Armadas Mexicanas Figura 1.2.

Así mismo, el Servicio Aéreo Militar de la Nación, sentó las bases para la organización y operación de la aviación militar en México.



**Figura1.2** Decreto para la creación del arma de Aviación Mexicana.

El escaso rendimiento obtenido con los motores y hélices importados, durante su empleo en el altiplano de México, originó la necesidad de contar con tecnología propia que permitiera el desarrollo y progreso de la aviación nacional. Por ende, el 15 de noviembre de 1915, se inauguraron oficialmente los “Talleres Nacionales de Construcciones Aeronáuticas” para la producción de aviones, motores y hélices en nuestro país, constituyendo un motivo de orgullo por emplear exclusivamente tecnología mexicana, en los “Talleres Nacionales de Construcción Aeronáutica (TNCA)”. Figura 1.3.



**Figura 1.3** Talleres Nacionales de Construcción Aeronáutica.

Los TNCA fueron pioneros en la industria aeronáutica mexicana y desempeñaron un papel crucial en el diseño, fabricación y mantenimiento de aviones para uso civil y militar.

## 1.2 Atribuciones y responsabilidades

La Fuerza Aérea Mexicana ostenta una serie de atribuciones y responsabilidades fundamentales en el contexto de la seguridad nacional y la salvaguarda del espacio aéreo mexicano. Estas atribuciones están definidas por la Constitución Política de los Estados Unidos Mexicanos, así como por diversas leyes, reglamentos y disposiciones gubernamentales. A continuación, se presentan las principales responsabilidades de la Fuerza Aérea Mexicana:

1. Defensa del Espacio Aéreo: La F.A.M. tiene el mandato de proteger y resguardar el Espacio Aéreo Mexicano, garantizando la integridad del territorio nacional frente a cualquier amenaza o incursión aérea.
2. Seguridad Nacional: Contribuye activamente a la seguridad nacional mediante la ejecución de operaciones aéreas destinadas a combatir actividades ilícitas que puedan comprometer la estabilidad del país.
3. Operaciones de Emergencia y Desastres: Participa activamente en operaciones de respuesta ante desastres naturales, proporcionando transporte, evacuación, suministros y apoyo logístico a través de medios aéreos.

Esta tercera y última implica un alto grado de disponibilidad y fiabilidad de las aeronaves de la F.A.M.; Para asegurar que las aeronaves estén listas para responder de manera efectiva en situaciones de emergencia, es esencial llevar a cabo un mantenimiento riguroso y periódico. Este mantenimiento incluye inspecciones previas a vuelos, mantenimiento preventivo, mantenimiento correctivo y, en algunos casos, mantenimiento en profundidad o mayor.

## 1.3 Identificación de necesidades del cliente

Después de haber realizado una entrevista con el jefe del departamento de electrónica, teniente Allan Ricardo Ruiz Gómez, se identificaron las necesidades de la institución, obteniendo los siguientes requisitos.

1. **Gestión de activos:** Este módulo permitiría llevar un registro detallado de los activos de las redes, incluyendo equipos eléctricos, componentes hidráulicos, y sistemas hidrosanitarios. Esto incluiría información como la ubicación, fecha de adquisición, historial de mantenimiento y vida útil esperada.
2. **Planificación de mantenimiento:** Este módulo ayudaría a programar y calendarizar los mantenimientos preventivos de manera eficiente. Permitiría establecer intervalos de mantenimiento para cada activo, generar órdenes de trabajo para el personal responsable y enviar recordatorios automáticos cuando se acerque la fecha de mantenimiento.
3. **Gestión de inventario:** Este módulo facilitaría el control del inventario de repuestos y materiales necesarios para los mantenimientos. Permitiría llevar un registro actualizado de los materiales disponibles, gestionar pedidos de reposición y optimizar el uso de los recursos disponibles.
4. **Seguimiento de mantenimiento:** Este módulo registraría todos los trabajos de mantenimiento realizados en cada activo, incluyendo detalles como la fecha, el tipo de mantenimiento realizado, las horas invertidas y cualquier problema detectado. Esto proporcionaría un historial completo del mantenimiento de cada activo.
5. **Análisis y reportes:** Este módulo permitiría analizar datos históricos de mantenimiento para identificar tendencias, realizar análisis de fallas y tomar decisiones informadas sobre estrategias de mantenimiento futuras. Además, facilitaría la generación de informes personalizados para informar a la dirección sobre el estado de las redes y el desempeño del equipo de mantenimiento.

### 1.3.1 Definición de requerimientos funcionales

Los requerimientos funcionales son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que este debe reaccionar a entradas particulares y de cómo se debe comportar en situaciones particulares.

|  |  |  |
| --- | --- | --- |
| ID | Modulo | Requerimientos |
| RF1 | Sistema | El sistema debe permitir el inicio de sesión de los usuarios registrados. |
| RF2 | El sistema debe permitir el acceso a los módulos según el perfil del usuario en sesión. |
| RF3 | El sistema debe permitir la modificación de la configuración de parámetros del sistema. |
| RF4 | El sistema debe permitir el registro, búsqueda, consulta, modificación y eliminación de usuarios del sistema. |
| RF6 | Mantenimiento | El sistema debe permitir la programación y gestión de mantenimientos preventivos para cada activo de las redes (eléctricas, hidráulicas e hidrosanitarias). |
| RF7 | El sistema debe permitir establecer calendarios de mantenimiento periódicos y programar recordatorios automáticos para los próximos mantenimientos. |
| RF8 | El sistema debe permitir la creación y asignación de órdenes de trabajo para la realización de los mantenimientos programados. |
| RF9 | El sistema debe permitir el seguimiento del estado y progreso de las órdenes de trabajo, incluyendo la capacidad de registrar actividades realizadas y tiempos dedicados. |
| RF8 | Inventario | El sistema debe permitir llevar un registro actualizado del inventario de repuestos y materiales necesarios para los mantenimientos. |
| RF9 | El sistema debe permitir realizar un seguimiento de las existencias disponibles, registrar entradas y salidas de inventario, y generar alertas de reabastecimiento cuando sea necesario. |
| RF10 | Reportes | El sistema debe permitir generar informes periódicos sobre el estado de las redes y la eficacia de los programas de mantenimiento. |
| RF11 | Seguridad | El sistema debe garantizar la seguridad de la información y restringir el acceso a los datos sensibles a usuarios autorizados. |
| RF14 | Notificaciones | El sistema debe permitir enviar notificaciones automáticas por correo electrónico o mensajes internos sobre eventos importantes, como vencimientos de mantenimiento o cambios en el estado de las órdenes de trabajo. |

**Tabla 1.1** Requerimientos funcionales de los módulos

### 1.3.2. Definición de requerimientos no funcionales

Requerimientos no funcionales son restricciones de los servicios o funciones ofrecidos por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo y estándares.

|  |  |  |
| --- | --- | --- |
| ID | Nivel | Requerimiento |
| RNF1 | Instalación | El sistema debe poder instalarse en cualquier ambiente requerido. |
| RNF2 | Usabilidad | El sistema debe poder ser usado de una forma intuitiva. |
| RNF3 | Accesibilidad | El sistema debe poder ser accedido desde cualquier dispositivo compatible conectado a internet. |
| RNF4 | Disponibilidad | El sistema debe poder encontrarse disponible en todo momento para los usuarios registrados. |
| RNF5 | Seguridad | El sistema debe garantizar la seguridad de la información confidencial de la empresa. |
| RNF6 | Rendimiento | El sistema debe ser capaz de manejar grandes volúmenes de datos y proporcionar respuestas rápidas a las solicitudes de los usuarios. Este requerimiento debe ser implementado utilizando Java, conforme al estándar establecido. |

**Tabla 1.2** Requerimientos no funcionales del sistema

## 1.3 Casos de uso

Un caso de uso contiene una descripción textual de todas las maneras que los actores previstos podrían trabajar con el software o el sistema.

En la figura 1.1 se muestra el diagrama de casos de uso, el cual está relacionado a la gestión de usuarios con su respectivo rol.

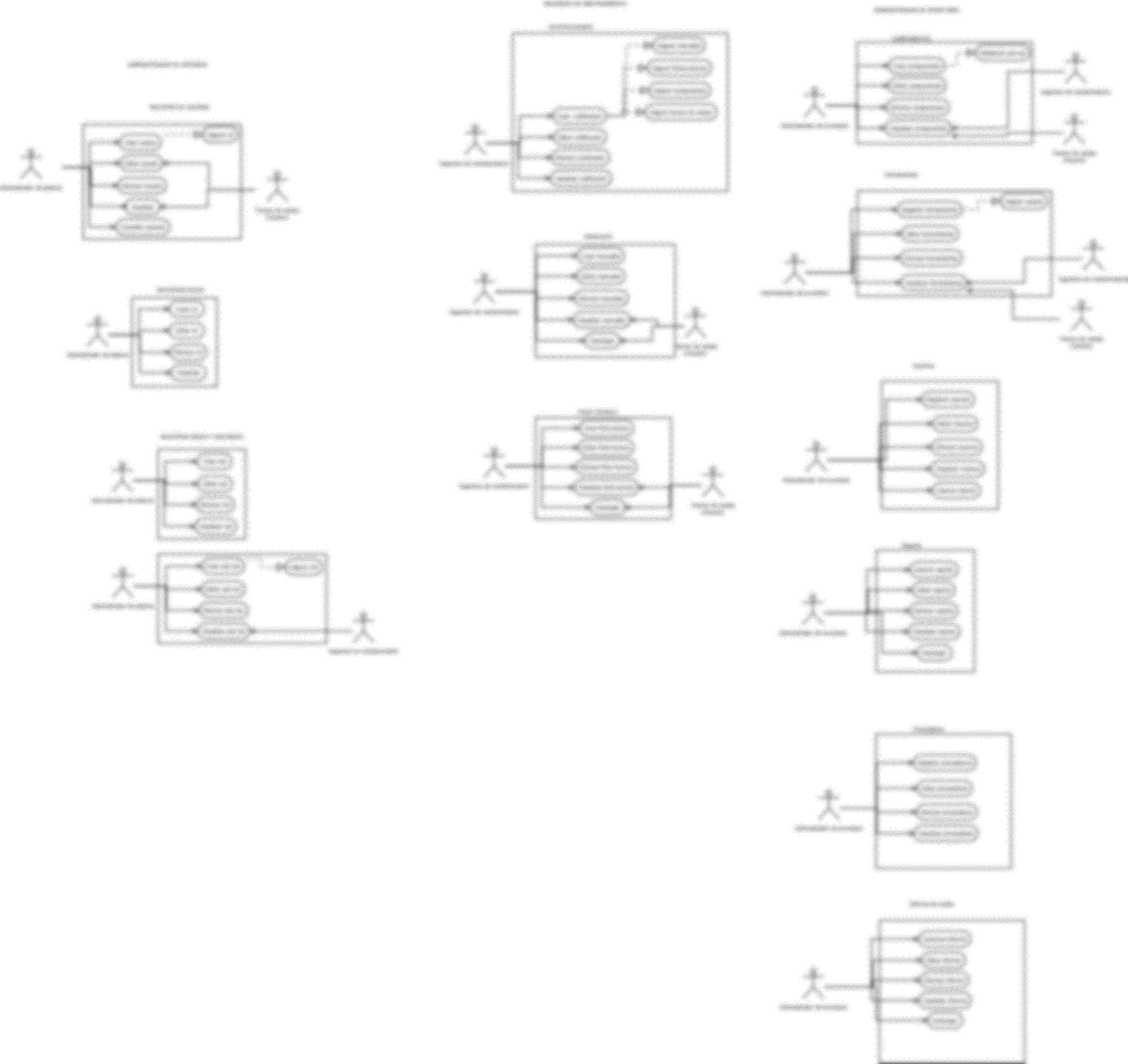
**Figura 1.1** Caso de Uso de gestión de usuario.

En la tabla 1.3 se muestra a detalle el caso de uso de gestión de usuario.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CU001 | | Usuarios | | |
| Descripción: | * Permite al administrador crear nuevos usuarios para la plataforma. * Permite al administrador modificar datos de los usuarios. * Permite al administrador eliminar usuarios de la plataforma | | Actores: | Administrador |
| Observaciones: | N/A | | | |
| Escenarios: | | | | |
| 1. Usuarios creados correctamente.  2. Error al crear usuarios.  3. Modificar el usuario.  4. Error al modificar usuario.  5. Consulta o búsqueda de usuarios.  6. Eliminar usuario.  7. Error al eliminar usuario. | | | | |

**Tabla 1.3** Detalle de gestión de usuario.

En la figura 1.2 se muestra el diagrama de casos de uso, el cual está relacionado a la gestión de insumos que tiene el C.I.D.T.F.A.



**Figura 1.2** Caso de Uso de gestión de insumos.

En la tabla 1.4 se muestra a detalle el caso de uso de gestión de insumos.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CU0010 | | Usuarios | | |
| Descripción: | * Permite al administrador de inventario crear un nuevo insumo. * Permite al administrador de inventario editar los datos ingresados en los insumos. * Permite al administrador de inventario eliminar los insumos creados. * Permite al administrador de inventario visualizar los insumos registrados. * Permite al administrador de inventario generar reporte. | | Actores: | Administrador de inventario. |
| Observaciones: | N/A | | | |
| Escenarios: | | | | |
| 1. Insumos creados correctamente.  2. Error al crear los insumos.  3. Modificar los insumos registrados.  4. Error al modificar los insumos.  5. Eliminar los insumos.  6. Error al eliminar los insumos.  7. Visualizar datos de los insumos agregados.  8. Error al visualizar los insumos.  9. Generar reporte.  10. Error al generar reporte. | | | | |

**Tabla 1.4** Detalle de gestión de insumos

Se definieron dos casos de uso ya que el Centro de Investigación y Desarrollo Tecnológico de la Fuerza Aérea (C.I.D.T.F.A.) tiene políticas de confidencialidad y de privacidad, así evitando el acceso a la misma por terceros y la mala divulgación de este.

## 1.4 Diagrama de secuencia

En la figura 1.3 se muestra el diagrama de secuencia que tendrá la gestión del usuario en el sistema

. **Figura 1.3** Diagrama de secuencia en la gestión de usuario

En la figura 1.4 se muestra el diagrama de secuencia que tendrá la gestión de las entradas de inventario al sistema.

**Figura 1.4** Diagrama de secuencia en la gestión de entradas de inventario

Se definieron dos diagramas de secuencia ya que la Centro de Investigación y Desarrollo Tecnológico de la Fuerza Aérea (C.I.D.T.F.A.) tiene políticas de confidencialidad y de privacidad, así evitando el acceso a la misma por terceros y la mala divulgación de este. Posteriormente estos dos diagramas están ligados a los dos casos de uso ya presentados.

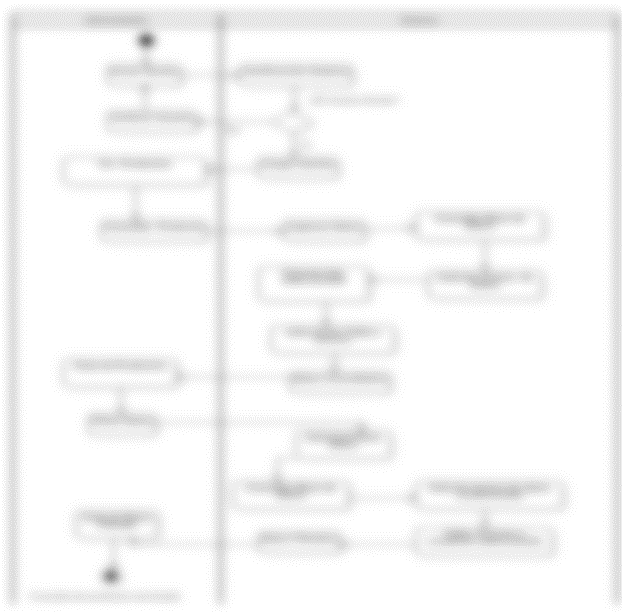
## 1.5 Diagrama de clase

El diagrama de clases representa la estructura básica del desarrollo del Sistema Integral de Gestión del Mantenimiento de Instalaciones (S.I.A.GE.M.I.) para el mantenimiento de las instalaciones del C.I.D.T.F.A. En él se reflejan las clases con sus atributos y métodos necesarios para el buen funcionamiento del sistema, como se muestra en la figura 1.5.

**Figura 1.5** Diagrama de clases para el sistema de C.I.D.T.F.A.

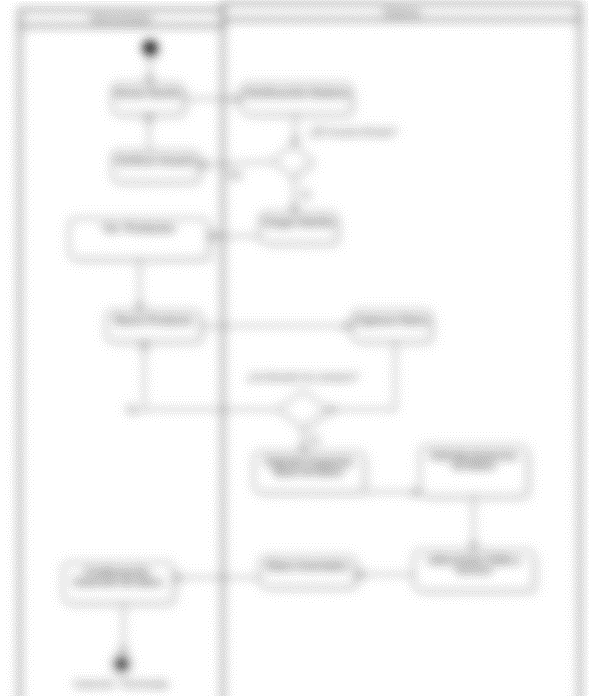
1.6 Diagrama de actividades

En la figura 1.6 se muestra el diagrama de actividad consultar insumos el cual describe las acciones que realiza el usuario administrador cuando interactúa con el sistema al consultar un insumo registrado en la base de datos.



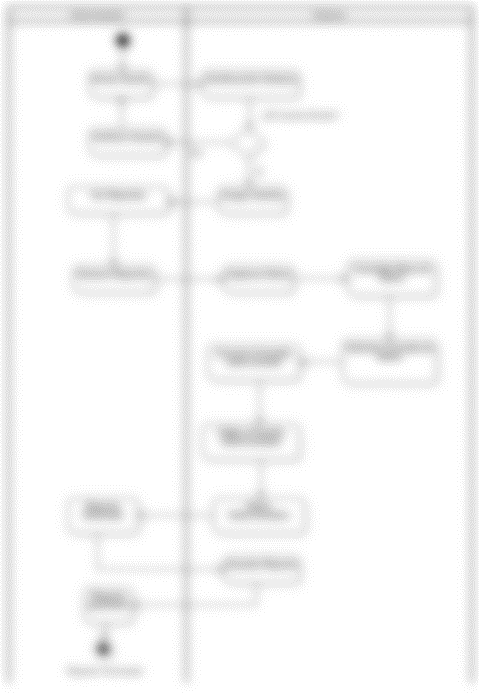
**Figura 1.6** Diagrama de actividades para consultar insumos

En la figura 1.7 se muestra el diagrama de actividad *insertar insumos* el cual describe las actividades que realiza el usuario administrador cuando interactúa con el sistema al insertar un nuevo insumo en la base de datos.



**Figura 1.7** Diagrama de actividades para insertar insumos

En la figura 1.8 se muestra el diagrama de actividad *generar reporte* el cual describe las acciones que realiza el usuario administrador cuando interactúa con el sistema al generar reporte de las compras que se han realizado en alguna fecha.

**Figura 1.8** Diagrama de actividades para generar reportes

Se crearon tres diagramas de actividades ya que el Centro de Investigación y Desarrollo Tecnológico de la Fuerza Aérea (C.I.D.T.F.A.) tiene políticas de confidencialidad y de privacidad, así evitando el acceso a la misma por terceros y la mala divulgación de este.

# CAPITULO 2. DISEÑO Y CREACIÓN DE LA BASE DE DATOS.

En este capítulo se hace muestra del diseño del diagrama entidad-relación, así como su modelo relacional y su diccionario de datos para la construcción de la base de datos.

## 2.1 Diagrama Entidad-Relación

Un diagrama entidad-relación es una herramienta gráfica para diseñar bases de datos, mostrando entidades (objetos del sistema), atributos (propiedades de las entidades) y relaciones (asociaciones entre entidades).

En la figura 2.1 se muestra el diagrama E-R, elaborado con la colaboración del equipo industrial, donde se representa la estructura global lógica de la base de datos que tendrá el sistema.



**Figura 2.1** Representación del diagrama Entidad-Relación

## 2.2 Modelo Relacional

Un modelo relacional organiza datos en tablas con filas y columnas para gestionar información de manera estructurada y eficiente en bases de datos.

Se muestra en la figura 5.2 el desarrollo del modelo relacional, donde se detalla de manera logia la base de datos donde está comprendiendo por tablas, tuplas y columnas que tendrá la misma de este modo permitirá a la base de datos tener un buen rendimiento.

**Figura 2.2** Representación del modelo relacional

## 2.3 Diccionario de datos

Un diccionario de datos es un registro detallado que describe todos los elementos de datos utilizados en un sistema o base de datos, incluyendo su significado, origen, formato y relaciones con otros datos.

En la tabla 2.1 se muestra las características que tiene el apartado de la tabla de área.

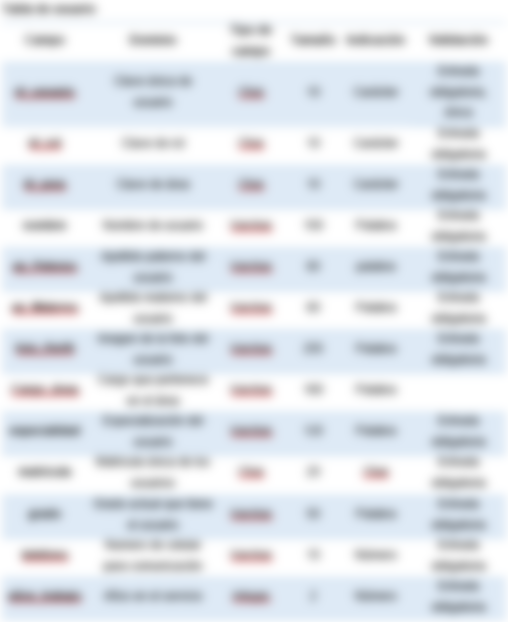
****

**Tabla 2.1** Diccionario del área.

En la tabla 2.2 se muestra las características que tiene el apartado de la tabla de rol.

****

**Tabla 2.2** Diccionario de rol.

****En la tabla 2.3 se muestra las características que tiene el apartado de la tabla de usuario.

**Tabla 2.3** Diccionario de usuario.

En la tabla 2.4 se muestra las características que tiene el apartado de la tabla de red.

****

**Tabla 2.4** Diccionario de red.

En la tabla 2.5 se muestra las características que tiene el apartado de la tabla de subred.

****

**Tabla 2.5** Diccionario de subred.

En la tabla 2.6 se muestra las características que tiene el apartado de la tabla de componentes.

****

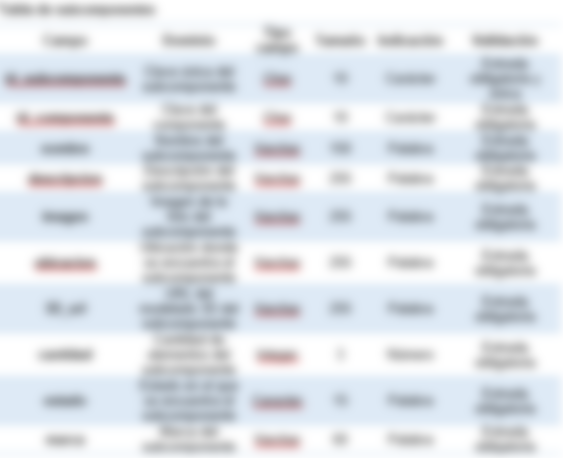
**Tabla 2.6** Diccionario de componentes.

En la tabla 2.7 se muestra las características que tiene el apartado de la tabla de tipo\_Servicio.

****

**Tabla 2.7** Diccionario de tipo\_servicio.

En la tabla 2.8 se muestra las características que tiene el apartado de la tabla de subcomponentes.

****

**Tabla 2.8** Diccionario de subcomponentes.

En la tabla 2.9 se muestra las características que tiene el apartado de la tabla de informacion\_Hidraulica.

**Tabla 2.9** Diccionario de informacion\_Hidraulica.

****En la tabla 2.10 se muestra las características que tiene el apartado de la tabla de informacion\_Electrica.

**Tabla 2.10** Diccionario de informacion\_Electrica.

En la tabla 2.11 se muestra las características que tiene el apartado de la tabla de información\_Hidrosanitaria.

****

**Tabla 2.11** Diccionario de informacion\_Hidrosanitaria.

En la tabla 2.12 se muestra las características que tiene el apartado de la tabla de fichas\_Tecnicas.

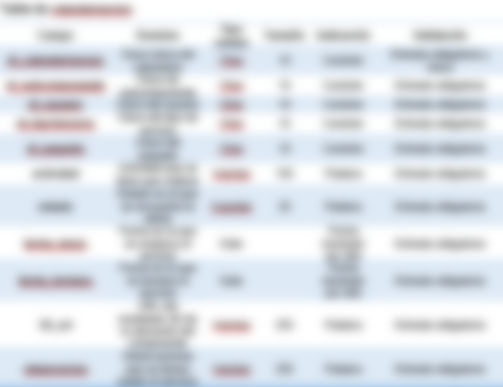
****

**Tabla 2.12** Diccionario de fichas\_Tecnicas.

En la tabla 2.13 se muestra las características que tiene el apartado de la tabla de manuales.

****

**Tabla 2.13** Diccionario de manuales.

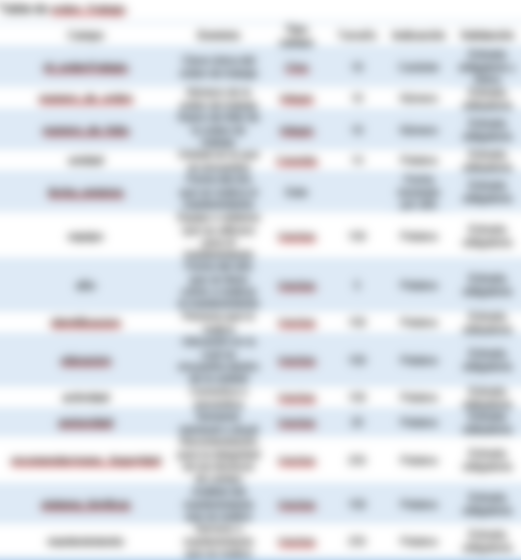
****En la tabla 2.14 se muestra las características que tiene el apartado de la tabla de calendarización.

**Tabla 2.14** Diccionario de calendarizacion.

En la tabla 2.14 se muestra las características que tiene el apartado de la tabla de notificacion.

**Tabla 2.14** Diccionario de notificacion.

En la tabla 2.15 se muestra las características que tiene el apartado de la tabla de orden\_Trabajo.

**Tabla de orden\_Trabajo**

**Tabla 2.15** Diccionario de orden\_Trabajo.

En la tabla 2.16 se muestra las características que tiene el apartado de la tabla de orden\_subcomponentes.

**Tabla 2.16** Diccionario de orden\_subcomponentes.

****En la tabla 2.17 se muestra las características que tiene el apartado de la tabla de paquete.

**Tabla 2.17** Diccionario de paquete

# CAPITULO 3. DISEÑO DE INTERFACES

En este capítulo se diseñarán las interfaces del sistema S.I.A.GE.M.I., satisfaciendo las necesidades del cliente y asegurando un manejo intuitivo para los usuarios. Se utilizará Figma para el diseño y se implementará el código con HTML, CSS, JavaScript y Java.

## Desarrollo de interfaces en Figma

EEn la

# CONCLUSIONES

# LISTADO DE SIGLAS O ACRÓNIMOS

**ASP:** otro lenguaje de programación web que se ejecuta en el servidor, con características similares al PHP, aunque es de propiedad de Microsoft, y de pago.

**BackUp:** Copia de seguridad.

**Bit:** Es la unidad mínima de información que puede almacenar y manejar un ordenador, equivalente a un 0 o un 1.

**Bitmap:** (o mapa de bits). Un tipo de imágenes para ordenador, en las que se almacena información sobre los puntos que las componen y el color de cada punto (al contrario que en las imágenes vectoriales).

**CSS:** (Cascade Style Sheet). Hojas de estilo en cascada. Es un lenguaje que permite definir el aspecto de los textos, bloques y otros contenidos de una página web. Por ejemplo, el color, el tamaño, la posición en pantalla. Mientras el HTML define los contenidos y su estructura jerárquica, el CSS determina cómo se muestran en pantalla.

**DHTML:** (HTML Dinámico): uso conjunto de HTML y JavaScript que permite aportar animación e interactividad a una página web. Puede usarse para mostrar un pase de dispositivas, paneles desplegables.

**FTP:** (siglas de File Transfer Protocol). Protocolo de transferencia de ficheros. Es el sistema mediante el que se "suben" a Internet (se publican) los archivos que componen una página web.

**HTML:** (HyperText Markup Language). Lenguaje de descripción del contenido y la estructura de una página web. Es el lenguaje básico en el que están escritas las páginas web.

**HTTP:** El protocolo usado en las páginas del WWW (HyperText Transfer Protocol).

**ISP:** (siglas de Internet Service Provider). Un proveedor de servicios de Internet es cualquier empresa que ofrece alojamiento (hosting), dominios y otros servicios relacionados.

**JS:** abreviatura de JavaScript, un lenguaje de programación que permite añadir interactividad a las páginas web (ver DHTML)

**MySQL:** sistema de gestión de base de datos muy utilizado en programación de aplicaciones web, en combinación con el lenguaje PHP.

**OSI:** Modelo para la interconexión de sistemas abiertos (Open Systems Interconnection). Es un modelo teórico de conexión de sistemas, estructurado en 7 capas (física, enlace, red, transporte, sesión, presentación y aplicación).

**Patch:** "Parche" que se aplica a un programa, normalmente con la intención de corregir algún "bug".

**PHP:** lenguaje de programación para páginas web que se ejecuta en el servidor (en el hosting). Permite utilizar información de una base de datos o añadirla a la misma, realizar cálculos, etc, para generar una página web dinámica que se genera en el momento en que se visita dicha web.

**Pixel:** Es el elemento de menor tamaño que forma una imagen ("un punto"). Abreviatura de Picture Element.

**POO:** Programación Orientada a Objetos, en español (es más frecuente ver las siglas en inglés: OOP).

**RAM:** Memoria de acceso directo (Random Access Memory). Normalmente se usa este nombre para referirse a memorias en las que se puede leer y también escribir (RWM). En los últimos PC es habitual que se use Fast Page Ram (386 y anteriores), EDO Ram..

**RWD:** (Responsive Web Design). Diseño web adaptable, que faculta a una página web para ajustarse automáticamente a cualquier ancho de pantalla

**SDK:** Kit de desarrollo de software, un conjunto de aplicaciones para desarrollar programas en un determinado lenguaje o para un determinado entorno (Software Development Kit).

**TCP/IP:** Protocolo de comunicaciones estándar en Internet (Transmission Control Protocol/Internet Protocol).

**UI:** (User eXperience). Experiencia de usuario. Se refiere a la facilidad o dificultad de manejo de un producto o servicio, como por ejemplo una página web, y a las sensaciones que genera en el que la utiliza. Una mala experiencia de usuario provoca el abandono de una página web y, por tanto, que ésta no alcance sus objetivos.

**UI:** (User Interface). Entorno de usuario, es decir, el aspecto visual de los elementos que forman un sistema (por ejemplo, un sitio web) y la interactividad que permite a sus usuarios. Ambos factores determinan en gran medida la usabilidad web y la experiencia de usuario (ver a continuación).

**URL / URI:** (siglas de Uniform Resource Locator / Identifier). Es la dirección de un recurso en Internet, que puede ser un nombre de dominio, una página web específica, un servidor FTP, una dirección de correo electrónico, etc.

**VGA:** Matriz gráfica de video (Video Graphics Array). Tarjeta gráfica de ordenadores PC y compatibles, evolución de la MCGA, que permitía trabajar también a 16 colores con 640x480 puntos.

**W3C:** (World Wide Web Consortium). Es un organismo internacional que se ocupa de crear o adaptar estándares en diseño web, y fomentar las buenas prácticas en el sector.

**WWW:** (World Wide Web). La telaraña que abarca todo el mundo es el conjunto de sitios y páginas web publicadas en Internet, y las tecnologías que lo hacen posible.

**XAML:** Xammel del inglés *eXtensible Application Markup Language,* Lenguaje Extensible de Formato para Aplicaciones en español.

**XML:** Lenguaje de descripción de páginas de Internet (eXtensible Markup Language), diseñado con la intención de reemplazar al estándar actual HTML.

# GLOSARIO

**Aplicación:** Programa que lleva a cabo una función específica para un usuario en Internet tales como WWW, FTP, correo electrónico y Telnet.

**Applet:** pequeño programa hecho en lenguaje Java que se introduce en un archivo HTML.

**Atributo:** es una parte de un elemento que añade información adicional a ese elemento.

**Base de datos:** formato estructurado para organizar y mantener informaciones que pueden ser fácilmente recuperadas.

**Broadcast:** dirección que se emplea para mandar mensajes a todos los puestos de una red.

**Cliente**: ordenador que requiere los servicios de otro ordenador. También, programa que requiere los servicios de otros programas. Normalmente, el navegador es un cliente de un servidor de datos.

**Consultas:** una consulta es el método para acceder, modificar, borrar, obtener y agregar datos en una base de datos.

**Cookie:** procedimiento ejecutado por un servidor el cual consiste en guardar información acerca del cliente para su posterior recuperación (el proceso realizado por el Internet Explorer cuando utiliza Microsoft Network)

**Dato:** unidad mínima que compone cualquier información.

**Desarrollador de Web:** web developer – persona o empresa responsable de la programación de un sitio Web así la cual incluye, si se da el caso, plataformas de comercio electrónico.

**Desarrolladores:** Son especialistas en informática que es capaz de concebir y elaborar sistemas informáticos, así como de implementarlos y ponerlos a punto, utilizando uno o varios lenguajes de programación.

**DNS:** Domain Name System. Servidor de dominios. Base de datos distribuida a través de Internet. A partir de un nombre la máquina es capaz de devolver su número IP y viceversa.

**Formato Utf-16:** es una forma de codificación de caracteres UCS y Unicode utilizando símbolos de longitud variable. Está oficialmente definida en el Anexo C de la norma ISO/IEC 10646:2003.

**Formato Utf-8:** es un formato de codificación de caracteres Unicode e ISO 10646 que utiliza símbolos de longitud variable.

**Framework:** proporciona una manera estándar de crear aplicaciones. Consiste de muchas soluciones predefinidas para funcionalidades comunes que son utilizadas para ayudar a crear aplicaciones de software de manera más rápida y estandarizada.

**Hardware:** Se refiere a las partes físicas, tangibles, de un sistema informático, sus componentes eléctricos, electrónicos, electromecánicos.

**Hipertexto:** modelo teórico de organización de la información de forma multisecuencial. También se denomina hipertexto al programa informático o herramienta de software que permite leer y escribir hipertextos (sistema de gestión de hipertextos).

**HTML Dinámico:** extensiones del lenguaje HTML que permiten crear páginas web más animadas y expresivas.

**Java:** lenguaje de programación de alto nivel orientado a objetos, desarrollado por Sun Microsystems y que corre sobre diversas plataformas.

**Keyword** (clave de búsqueda, palabra clave): término para buscar información dentro de un buscador o para registrar una página web en un buscador.

**Login:** clave de acceso que se le asigna a un usuario con el propósito de que pueda utilizar los recursos de una computadora.

**Metadato:** dato que describe otro dato. Los metadatos se utilizan par describir recursos.

**Metodología De *Software*:** Es un enfoque, una manera de interpretar la realidad o la disciplina en cuestión, que en este caso particular correspondería a la Ingeniería de software.

**MYSQL**: Motor de base de datos de código abierto. Es multi-hilo y multi-usuario.

**Página de Servidor Java (JSP):** tipo especial de página HTML que cont iene unos pequeños programas (también llamados scripts) que son ejecutados en servidores Netscape antes de ser enviados al usuario para su visualización en forma de página HTML.

**Páginas Web:** es un documento o información electrónica capaz de contener texto, sonido, vídeo, programas, enlaces, imágenes, hipervínculos y muchas otras cosas, adaptada para la llamada *World Wide Web.*

**Plataforma:** Es un sistema que sirve como base para hacer funcionar determinados módulos de *hardware* o de *software* con los que es compatible.

**Portal:** Sitio web cuyo objetivo es ofrecer al usuario, de forma fácil e integrada, el acceso a una serie de recursos y de servicios, entre los que suelen encontrarse buscadores, foros, compra electrónica, etc.

**Protocolo:** Descripción formal de formatos de mensaje y de reglas que dos ordenadores deben seguir para intercambiar dichos mensajes.

**Raíz (Root):** Directorio inicial de un sistema de archivos mientras que en entornos UNIX también se refiere al usuario principal.

Recursos: Es un recurso utilizado por algunos modelos computacionales en la solución de problemas computacionales.

**Requerimientos funcionales**: descripción de la interacción entre el sistema y su ambiente independientemente de su implementación.

**Requerimientos no funcionales:** son requisitos que imponen restricciones en el diseño o la implementación como restricciones en el diseño o Estándares de Calidad. Son propiedades o cualidades que el producto debe tener.

**Seguimiento:** Es la acción y efecto de seguir o seguirse, en el contexto popular suele usarse como sinónimo de persecución, observación o vigilancia.

**Servidor:** Un nodo de red que proporciona servicios a PCs clientes; por ejemplo, acceso a archivos, centro de impresión o ejecución remota.

**Sistemas:** es un sistema que permite almacenar y procesar información; es el conjunto de partes interrelacionadas: *hardware, software* y personal informático.

**Software:** Conjunto de programas, documentos, procesamientos y rutinas asociadas con la operación de un sistema de computadoras, es decir, la parte intangible o lógica de una computadora.

Soporte: medio de almacenamiento de la información digital.

# REFERENCIAS