**Documento Final de Plan de Mejoramiento**

Wílmer Edilson León Díaz

Corporación Unificada Nacional de Educación Superior

51551/Bloque único/24V04: Práctica 2

Ing.ro Marlon José Cárdenas Castellanos

5 de noviembre de 2024

Tabla de Contenido

[1. Planteamiento del Problema 4](#_Toc179977739)

[1.1. Título 4](#_Toc179977740)

[1.2. Contexto 4](#_Toc179977741)

[1.3. Definición del Problema 4](#_Toc179977742)

[2. Justificación 5](#_Toc179977743)

[3. Objetivo General 6](#_Toc179977744)

[4. Objetivos Específicos 6](#_Toc179977745)

[5. Metodología 7](#_Toc179977746)

[5.1. Fase de Diagnóstico 7](#_Toc179977747)

[5.1.1. Objetivo 7](#_Toc179977748)

[5.2. Fase de Diseño del Aplicativo 8](#_Toc179977749)

[5.2.1. Objetivo 8](#_Toc179977750)

[5.3. Fase de Desarrollo 9](#_Toc179977751)

[5.3.1. Objetivo 9](#_Toc179977752)

[5.4. Fase de Implementación y Capacitación 9](#_Toc179977753)

[5.4.1. Objetivo 9](#_Toc179977754)

[5.5. Fase de Evaluación y Mejora Continua 10](#_Toc179977755)

[5.5.1. Objetivo 10](#_Toc179977756)

[5.6. Especificación de Detalle Técnico de Montaje para la Metodología 11](#_Toc179977757)

[5.6.1. Herramientas y Tecnologías Utilizadas 11](#_Toc179977758)

[5.7. Fases del Proyecto 15](#_Toc179977759)

[5.7.1. Planificación 15](#_Toc179977760)

[5.7.2. Diseño 16](#_Toc179977761)

[5.7.3. Desarrollo: 17](#_Toc179977762)

[5.7.4. Pruebas: 17](#_Toc179977763)

[5.7.5. Implementación: 17](#_Toc179977764)

[5.7.6. Evaluación y Optimización: 18](#_Toc179977765)

[5.7.7. Anexo Gráfico: 18](#_Toc179977766)

[5.8. Consideraciones Finales 20](#_Toc179977767)

[6. Conclusiones 21](#_Toc179977768)

[7. Bibliografía 22](#_Toc179977769)

[8. Aval 23](#_Toc179977770)

# Planteamiento del Problema

## Título

Desarrollo de un Aplicativo para la Integración de Datos entre Excel y AVID Maestro TX en la Producción de Contenidos Deportivos

## Contexto

En el ámbito de la producción de contenidos deportivos, la gestión eficiente de datos es crucial para garantizar la calidad y la precisión de la información presentada. Actualmente, muchos equipos de producción utilizan hojas de cálculo de Excel para organizar y gestionar datos, que luego deben ser transferidos manualmente a software especializado como AVID Maestro. Este proceso manual no solo es propenso a errores, sino que también consume un tiempo valioso que podría ser utilizado en actividades más creativas y productivas.

## Definición del Problema

La necesidad de un sistema que automatice la transferencia de datos entre Excel y AVID Maestro TX se ha vuelto evidente. La falta de integración entre estas herramientas genera ineficiencias operativas, errores en la información y retrasa la producción de contenidos. Los problemas específicos incluyen:

* + 1. Errores humanos: la inserción manual de datos es susceptible a errores, lo que puede llevar a la difusión de información incorrecta y afectar la credibilidad de la producción.
    2. Ineficiencia en el proceso: la transferencia manual de datos consume tiempo y recursos que podrían ser mejor utilizados en la creación y edición de contenido.
    3. Falta de sincronización: la ausencia de un sistema automatizado impide que los datos estén actualizados en tiempo real, lo que puede resultar en inconsistencias entre la información presentada y la realidad del evento deportivo.
    4. Dificultades en la escalabilidad: a medida que la producción de contenidos crece, la carga de trabajo asociada a la gestión manual de datos se vuelve insostenible, limitando la capacidad del equipo para adaptarse a nuevas demandas.

# Justificación

La creación de este aplicativo no solo abordará los problemas actuales de ineficiencia y errores en la gestión de datos, sino que también proporcionará una solución innovadora que puede ser replicada en otros contextos de producción de contenidos. Al mejorar la calidad y la rapidez de la producción, se espera que el equipo pueda ofrecer un contenido más atractivo y relevante para la audiencia, lo que a su vez puede contribuir al éxito general de la organización en el competitivo mundo de los medios deportivo.

La mejora en la orquestación de la inserción de textos en las piezas gráficas permitirá llevar un control más centralizado de la información que se gestiona en la continuidad; por lo tanto, por medio de una escaleta que llevará el control de la cada uno de los procesos de gestión de una producción deportiva, se dejará de lado la manipulación directa de la hoja de cálculo Excel.

Se espera que, con el montaje del aplicativo, se mejore la previsualización de las piezas gráficas con texto y que permita capturar los errores tipográficos con más facilidad.

La portabilidad es la principal característica que tendrá el aplicativo, por lo que la instalación en cada una de las terminales de trabajo permitirá que los ejecutables, resultado de la creación y edición de la escaleta, sean de fácil manejo lo que evitará el deterioro de la hoja de cálculo Excel y mitigará los fallos de fórmulas en su programación en VBA.

# Objetivo General

Desarrollar un aplicativo que permita la integración automática de datos entre Excel y AVID Maestro TX, mejorando así la eficiencia y la precisión en la producción de contenidos deportivos

# Objetivos Específicos

Este aplicativo buscará:

* 1. Automatizar la transferencia de datos, reduciendo la posibilidad de errores humanos.
  2. Optimizar el tiempo de producción al eliminar la necesidad de inserciones manuales en una hoja de cálculo segmentada por pestañas.
  3. Garantizar que la información esté siempre actualizada y sincronizada entre las diferentes plataformas.
  4. Facilitar la escalabilidad del proceso de producción, permitiendo al equipo adaptarse a un volumen creciente de datos y contenidos.

# Metodología

La metodología de esta investigación se estructurará en varias fases, cada una de las cuales se centrará en un aspecto específico del desarrollo del aplicativo para la integración de datos entre Excel y AVID Maestro TX. A continuación, se detallan las estrategias, métodos y técnicas que se utilizarán para abordar el problema planteado y facilitar el desarrollo del trabajo de investigación.

## Fase de Diagnóstico

### Objetivo

Identificar y analizar las causas del problema actual en la gestión de datos entre Excel y AVID Maestro TX.

#### Recolección de Datos:

Se llevarán a cabo entrevistas y encuestas con los miembros del equipo de producción para comprender sus experiencias y desafíos en el manejo de datos. Se utilizarán cuestionarios estructurados para obtener información cuantitativa sobre la frecuencia de errores y el tiempo dedicado a la transferencia de datos.

#### Análisis de Procesos Actuales:

Se realizará un mapeo de los procesos actuales de gestión de datos, documentando cada paso desde la entrada de datos en Excel hasta su uso en AVID Maestro TX. Esto permitirá identificar cuellos de botella y áreas de mejora.

#### Identificación de Causas

Utilizando técnicas de análisis de causa raíz, como el diagrama de Ishikawa (o espina de pescado), se identificarán las causas subyacentes de los problemas observados (Ir a Anexos).

## Fase de Diseño del Aplicativo

### Objetivo

Definir las especificaciones y características del aplicativo que se desarrollará.

#### Revisión de Literatura

Se llevará a cabo una revisión de estudios y aplicaciones existentes que aborden la integración de datos entre diferentes plataformas. Esto ayudará a identificar mejores prácticas y funcionalidades deseables.

#### Definición de Requisitos

Se organizarán sesiones de lluvia de ideas con el equipo de producción para definir los requisitos funcionales y no funcionales del aplicativo. Se utilizarán técnicas de prototipado rápido para crear maquetas iniciales del sistema.

#### Especificaciones Técnicas

Se elaborará un documento de especificaciones técnicas que detalle la arquitectura del sistema, las tecnologías a utilizar (por ejemplo, lenguajes de programación, bases de datos, las API) y los criterios de éxito.

## Fase de Desarrollo

### Objetivo

Implementar el aplicativo basado en las especificaciones definidas.

#### Desarrollo Ágil

Se adoptará un enfoque ágil para el desarrollo del aplicativo, dividiendo el trabajo en *sprints* cortos. Cada sprint incluirá la planificación, el desarrollo, las pruebas y la revisión.

#### Integración de Herramientas

Se utilizarán herramientas de desarrollo de software que faciliten la integración entre Excel y AVID Maestro TX, como API y bibliotecas de programación específicas.

#### Pruebas de Funcionalidad

Al final de cada *sprint*, se realizarán pruebas de funcionalidad para asegurar que el aplicativo cumpla con los requisitos establecidos. Se utilizarán pruebas unitarias y pruebas de integración.

## Fase de Implementación y Capacitación

### Objetivo

Desplegar el aplicativo en el entorno de producción y capacitar al equipo en su uso.

#### Despliegue del Sistema

Se llevará a cabo la implementación del aplicativo en el entorno de producción, asegurando que todos los datos existentes se migren correctamente (por estipulaciones de tiempo, se entregará un prototipo con funcionalidades básicas y con interacción local).

#### Capacitación del Personal

Se organizarán sesiones de capacitación para el equipo de producción, donde se les enseñará a utilizar el nuevo sistema y se abordarán posibles dudas o inquietudes, para la fase de pruebas del prototipo.

#### Documentación

Se elaborará un manual de usuario que incluya instrucciones detalladas sobre el uso del aplicativo, así como un documento de preguntas frecuentes (FAQ).

## Fase de Evaluación y Mejora Continua

### 5.5.1. Objetivo

Evaluar el impacto del aplicativo y realizar mejoras basadas en la retroalimentación.

#### Monitoreo de Resultados

Se establecerán indicadores clave de rendimiento (los KPI) para medir la eficiencia del nuevo sistema, como la reducción de errores en la transferencia de datos y el tiempo ahorrado en el proceso.

#### Recopilación de Retroalimentación

Se llevarán a cabo encuestas y entrevistas postimplementación para recoger la opinión del equipo sobre el funcionamiento del aplicativo y su impacto en la producción.

#### Iteración y Mejora

Basándose en la retroalimentación recibida, se realizarán ajustes y mejoras al aplicativo. Se planificarán ciclos de revisión periódicos para asegurar que el sistema siga siendo relevante y eficiente.

## Especificación de Detalle Técnico de Montaje para la Metodología

### Herramientas y Tecnologías Utilizadas

#### 5.6.1.1. Gestión de Proyectos

**Primavera 6 Oracle**: Utilizado para crear el diagrama de Gantt y estimar los tiempos de cada fase del proyecto. Se establecerán hitos y dependencias entre tareas para asegurar un seguimiento efectivo del progreso.

**Process Dashboard**: Implementado para llevar un registro detallado de los tiempos de cada fase del proyecto, permitiendo la identificación de cuellos de botella y la optimización en ciclos futuros.

#### 5.6.1.2. Metodología de Desarrollo

**Metodología de Procesos por Prototipos**: Se utilizará para iterar sobre el desarrollo del producto, permitiendo ajustes basados en la retroalimentación continua.

**Metodología Ágil Scrum-XP**: Se aplicará para gestionar el desarrollo de manera flexible y adaptativa, facilitando la colaboración y la entrega incremental de funcionalidades.

#### 5.6.1.3. Control de Versiones y Tareas

**Git**: Utilizado para gestionar el código fuente y llevar un tablero Kanban que asigne tareas basadas en los requisitos funcionales obtenidos de las historias de usuario.

**Backlog (GitHub):** Utilizado para el control y gestión de las historias de usuario generadas de las entrevistas y la solicitud del cliente en el enunciado.

#### 5.6.1.4. Arquitectura y Diseño de Software

**Lucid**: Utilizado para la creación de diagramas de objetos, de clases, de secuencia, de paquetes, de comunicación, entre otras. Para la arquitectura de *software*, es vital para la creación de diagramas de bloques (arquitectura de capas) y la relación entre los conjuntos de componentes en un diagrama de integración de componentes. Para el diseño de *software*, la herramienta se empleará para los distintos diagramas estáticos y dinámicos; por lo tanto, se apelará a una diagramación de ingeniería directa en la fase de diseño.

**PlantUML**: Utilizado para los bocetos de los diagramas para ingeniería inversa en la fase de codificación, gracias a su acople con VSC y Copilot en el diagnóstico y mantenimiento de los parámetros de los patrones de diseño GRAPS y SOLID que se implementarán al momento de abordar cada una de las acciones del aplicativo, en pro de optimizar el rendimiento (latencia y acoplamiento óptimo).

#### 5.6.1.5. Diseño de Interfaz y Documentación

**Figma**: Herramienta para el diseño de la interfaz de usuario (UI), permitiendo la creación de prototipos interactivos.

**GitBook**: Utilizado para la documentación del proyecto del esquema de diseño, facilitando la colaboración, el acceso a la información y la especificación del desarrollo de la interfaz gráfica.

Lucid: Para la creación de diagramas UML que representen la relación entre la interfaz del diseño web con el usuario del sistema y sus componentes.

**StoryboardThat**: Empleado para el estudio de caso y la compilación de eventos, ayudando a visualizar flujos de trabajo y escenarios de usuario.

#### 5.6.1.6. Desarrollo y Base de Datos:

**IDE Visual Studio Code**: Entorno de desarrollo integrado para escribir y depurar el código cualquier lenguaje; para este caso, en JavaScript.

**JavaScript**: Lenguaje de programación utilizado, siguiendo la arquitectura de patrones por capas, patrones GRAPS, patrones SOLID y Programación Orientada a Objetos (POO).

**Oracle APEX**: Gestor de bases de datos utilizado para la creación y gestión de la base de datos (en fases más allá del prototipo, se gestionará la BD en Nube por medio de este GBD).

**Oracle Data Modeler**: Herramienta para la diagramación de modelos de base de datos, facilitando la visualización y diseño de la estructura de datos (usada para el montaje de la BD en Históricos).

**MongoDB**: Utilizado para el almacenamiento y procesamiento de NoSQL de las imágenes en el componente que preprocesamiento encargado de esta tarea (contemplado para otras fases de desarrollo).

**Servidor**: Oraq, para las fases de implementación total (en caso de desarrollarse todos los componentes del aplicativo). Aclaración: el prototipo funcionará con despliegue local en las terminales; en caso de ser necesario, se usará un sistema de conexión a servidor virtual en la nube para gestionar algunos procesos referentes a la interacción con BD del componente Históricos, del aplicativo.

**Servidor en nube**: Aún se debe definir el servicio de servidor en nube (probablemente GCP).

**Terraform**: Utilizado para la infraestructura del código en la fase de despliegue en la nube.

**Docker**: Utilizado como contenedor de las distintas aplicaciones anexas al aplicativo.

**Testcontainers**: Utilizado para las pruebas de caja negra y caja blanca en cada uno de los componentes creados para el aplicativo en la integración de contenedores Docker.

#### 5.6.1.7. Creación del Prototipo

**npm**: Utilizado para la creación del ejecutable del prototipo, gestionando dependencias y scripts necesarios para la ejecución del proyecto, por medio de la creación de ejecutable de instalación con Electron.

## Fases del Proyecto

### Planificación

* Definición de objetivos y alcance del proyecto.
* Creación de gestión de proyecto en Primavera P6 Oracle.

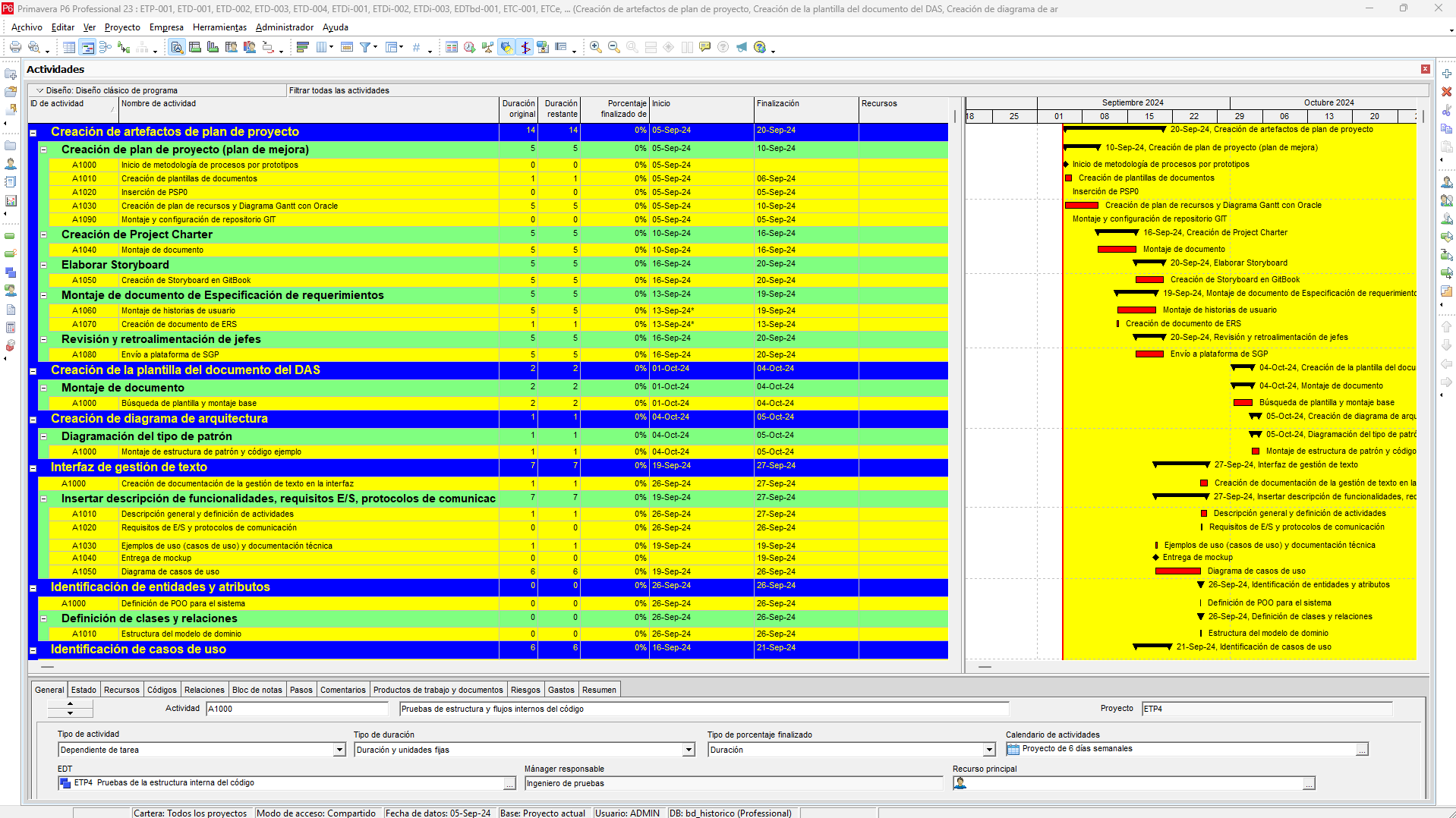


Ilustración 1: Creación en P6

* Establecimiento de roles y asignación de tareas en Git
* Montaje de Carta de proyecto (para el área administrativa de la empresa) y Plan de mejoramiento (para el área técnica/Ingeniería de la empresa).

Artefactos: [Carta de proyecto de Estimación de tiempo y recursos, Especificación de requerimientos, Formato Plan de mejoramiento 2 y Anexos gráficos codificados y en imagen.](https://drive.google.com/drive/folders/11F3rogmxQPWwKQQQILEB4to2PqnXl3l0?usp=sharing)

Backlog (GitHub): [Aplicativo de Escaleta-Teclado para GC.](https://github.com/users/wilmereleon/projects/8/views/1)

### Diseño

* Desarrollo de prototipos de UI en Figma.
* Documentación de requisitos y diseño en GitBook.
* Creación de diagramas UML en Lucid.

Artefactos: [Documento de arquitectura de software, Documento de Especificación de realización de casos de uso, Documento de Especificaciones de casos de uso del sistema, Documento de Modelo de implementación de sistema, Documento de Prototipo de interfaz de usuario de sistema y carpetas (Arquitectura, Caso de uso, Diagramas de paquetes en PlantUML, Diagramas en imagen, Imágenes para prototipo, Latencia, MATERIALES DCSHA, Objetos, Plantillas, PSP y Secuencia y despliegue)](https://drive.google.com/drive/folders/11HmsITEjJG4w1ZHwngIQFDASYcUOn27I?usp=sharing)

Prototipo: [Esquema de diseños | App de escaleta y texto GC (Diseño de navegación).](https://wilmers-organization-1.gitbook.io/esquema-de-disenos-or-app-de-escaleta-y-texto-gc/)

### Desarrollo:

* Implementación del código en Visual Studio Code utilizando JavaScript.
* Gestión de versiones y control de tareas a través de Git y el tablero Kanban.
* Desarrollo de la base de datos en Oracle APEX y modelado con Oracle Data Modeler.

Artefactos: [ETgc.](https://drive.google.com/drive/folders/11LUeElffvSEkHxcdY5w-_swswdVCJ7zv?usp=sharing)

### Pruebas:

* Ejecución de pruebas unitarias y de integración.
* Revisión de prototipos y ajustes basados en la retroalimentación.

Artefactos: [Resultados.](https://drive.google.com/drive/folders/11OO1Wd3XFYDkM0rgc27huhzvOhv-2m_7?usp=sharing)

### Implementación:

* Creación del ejecutable del prototipo con npm.
* Despliegue del sistema y capacitación de usuarios.

Artefactos: [Instalador y carpeta (Manual de operación de usuario y Manual de administrador del sistema).](https://drive.google.com/drive/folders/11To-1xZUMeTzYmio9m2i-t6WkUFbFRB1?usp=sharing)

### Evaluación y Optimización:

* Registro de tiempos y análisis de resultados en Process Dashboard.
* Identificación de áreas de mejora para futuros ciclos de desarrollo.

Artefactos: [carpeta (Registro de tiempos en PSP) y Retroalimentación Postmortem.](https://drive.google.com/drive/folders/11Z5pUJWGTiqoq8FzdWmZjSoyByfpatTc?usp=sharing)

### Anexo Gráfico:

* Diagrama de Ishikawa creado en Lucid.

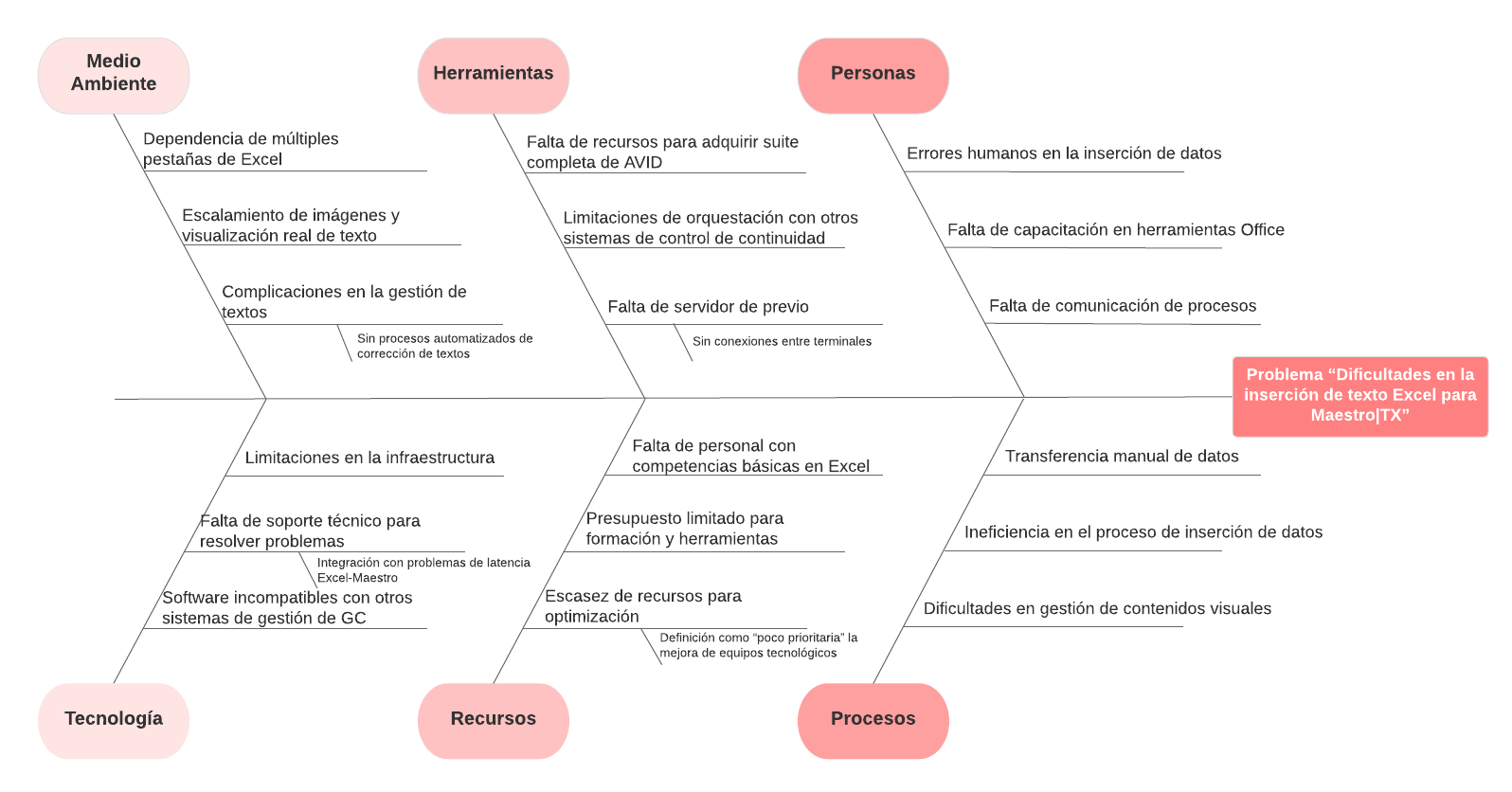


Ilustración 2: Diagrama de Ishikawa

* Diagrama de Gantt creado en Oracle Primavera P6 Visualizer.y optimizado en PlantUML.

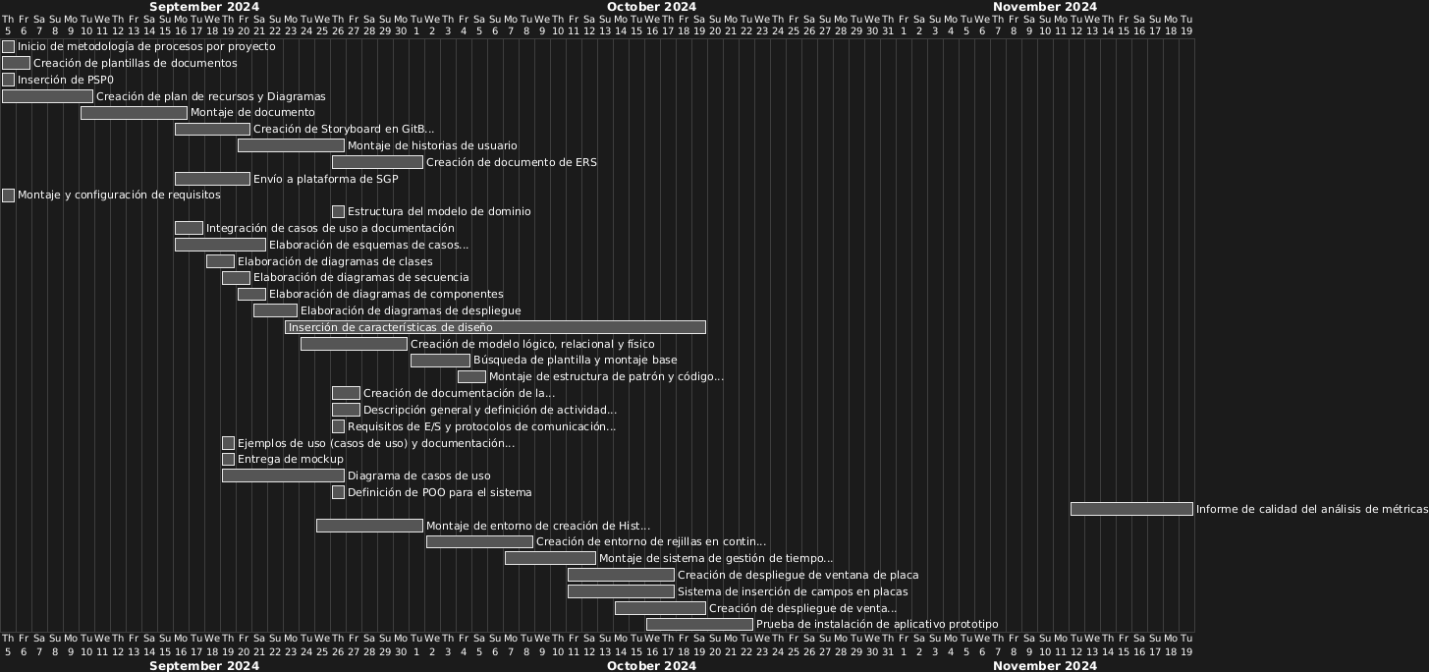


Ilustración 3: Diagrama de Gantt

* Diagrama de Pareto creado en Jupyter/Python.

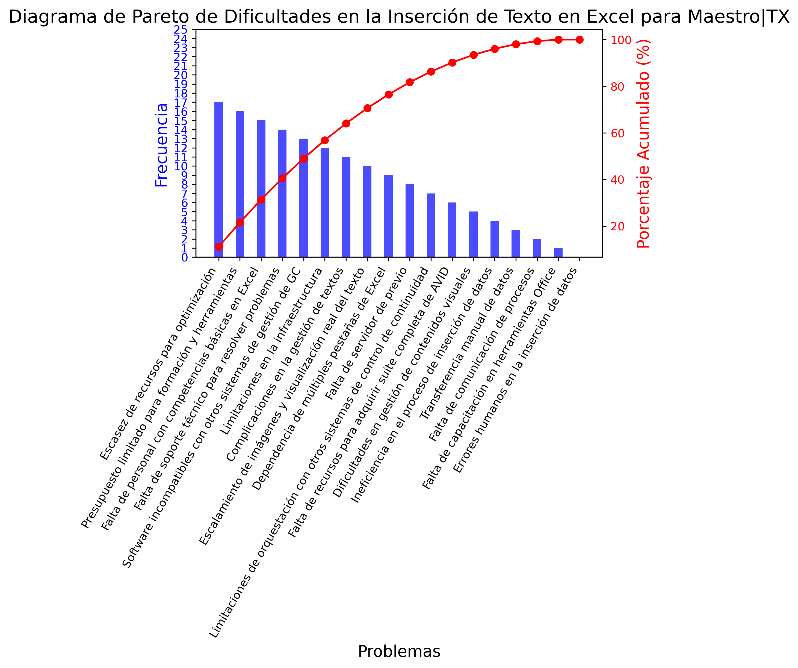


Ilustración 4: Diagrama de Pareto

Artefactos: [Diagrama de Ishikawa, Diagrama de Gantt y Diagrama de Pareto.](https://drive.google.com/drive/folders/11afxjW863J5RiDINavrzJSUDkaJD5awR?usp=sharing)

## Consideraciones Finales

**Documentación Continua**: se mantendrá una documentación actualizada a lo largo de todas las fases del proyecto, asegurando que todos los miembros del equipo tengan acceso a la información relevante.

**Revisión Periódica**: se realizarán reuniones regulares para revisar el progreso, ajustar el plan según sea necesario y asegurar la alineación con los objetivos del proyecto.

**Flexibilidad**: la metodología ágil permitirá adaptarse a cambios en los requisitos y prioridades, asegurando que el producto final cumpla con las expectativas del cliente y los usuarios finales

# Conclusiones

# Bibliografía

# Aval

Esta investigación está avalada, revisada y aprobada por:

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Firma de jefe inmediato