

**Universidad de Guadalajara**  
**Centro Universitario de Ciencias Exactas**  
**e Ingenierías**  
**Ingeniería de Computación**



Análisis de Algoritmos (D01)

Proyecto final - Objetivos y Entregables

Carlos Andres Chico Aguayo

Gael Emiliano Anaya Garcia

Naylea Danae Silva Penaloza

Carlos Humberto Avila Sanchez

Nahomi Itzel Luna Ornelas

JORGE ERNESTO LOPEZ ARCE DELGADO

11/NOV/2023

## **Problema: Optimización de Rutas para un Servicio de Entrega Local**

### **Descripción del Problema:**

Su tarea es crear un algoritmo de optimización de rutas eficiente para un servicio de entrega local. La compañía cuenta con una flota de vehículos de entrega y una lista de puntos de entrega (ubicaciones de clientes) que deben visitarse. El objetivo es encontrar las rutas más cortas para que los vehículos entreguen paquetes a todos los clientes y regresen al almacén central minimizando la distancia total recorrida.

### **Roles:**

- Carlos Andres Chico Aguayo - Project Manager
- Gael Emiliano Anaya Garcia - FrontEnd
- Naylea Danae Silva Penaloza – FrontEnd2
- Carlos Humberto Avila Sanchez - BackEnd
- Nahomi Itzel Luna Ornelas - Tester

### **Requisitos del Proyecto:**

#### **Generación de Datos de Entrada (Tester):**

Cada equipo debe crear un conjunto de datos de entrada para el problema. Esto incluye especificar las coordenadas del almacén (depósito), las ubicaciones de los clientes y la capacidad de cada vehículo. Los datos deben ser presentados en un formato estructurado.

#### **Modelado del Problema (Todos):**

Deben modelar el problema como un grafo, donde el almacén y las ubicaciones de los clientes se representan como nodos y las distancias entre ellos como aristas.

Definir las restricciones, como los límites de capacidad de los vehículos y cualquier requisito específico para cada punto de entrega.

#### **Selección de Algoritmo (BackEnd):**

Cada equipo debe elegir un algoritmo (o una combinación de algoritmos) para resolver el problema de optimización. Pueden considerar:

El Problema del Viajante de Comercio (TSP) para encontrar la ruta más corta que visite a todos los clientes exactamente una vez.

El Problema de Enrutamiento de Vehículos (VRP) para optimizar las rutas teniendo en cuenta la capacidad de los vehículos y las restricciones.

Algoritmos metaheurísticos como algoritmos genéticos, recocido simulado u optimización por colonias de hormigas para soluciones más complejas.

#### **Implementación (BackEnd, FrontEnd y Tester):**

Implementar el algoritmo seleccionado en Python.

### Interfaz de Usuario (FrontEnd):

Desarrollar una interfaz de usuario amigable que permita a los usuarios cargar los datos de entrada, visualizar las rutas optimizadas y obtener las soluciones.

### Estrategias de Optimización (Todos):

Implementar estrategias de optimización para mejorar las rutas generadas por el algoritmo. Asegurarse de que las entregas cumplan con las restricciones de capacidad de los vehículos.

### Pruebas y Validación (Tester):

Probar el algoritmo con diferentes conjuntos de datos, incluyendo instancias pequeñas y grandes.

Verificar que el algoritmo produzca soluciones válidas y óptimas o cercanas a las óptimas. Comparar los resultados con soluciones de referencia (si están disponibles).

### Evaluación del Rendimiento (Tester):

Medir el rendimiento del algoritmo en términos de factores como la distancia total recorrida, el equilibrio de carga de los vehículos y el tiempo de ejecución.

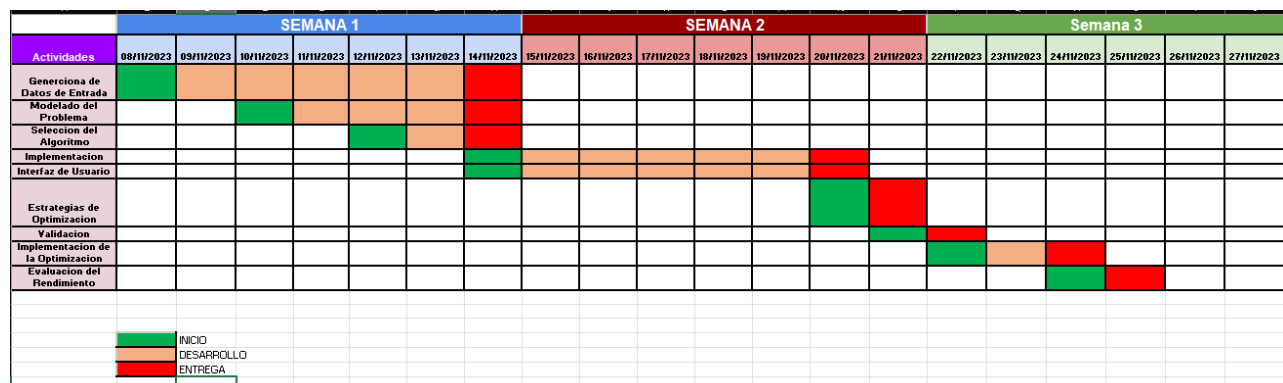
### Documentación (BackEnd y FrontEnd):

Preparar una documentación detallada que explique el problema, los algoritmos utilizados, los resultados obtenidos y el análisis del rendimiento. Esto debe incluir una descripción clara de cómo los equipos generaron los datos de entrada.

### Presentación: (ProjectManager)

Realizar una presentación final que resuma los objetivos del proyecto, la metodología, los hallazgos y las conclusiones.

Considerando las consideraciones previas, he determinado que el cronograma más propicio se estructuraría de la siguiente manera, contemplando la asignación de dos días hábiles de reserva para abordar posibles contratiempos o contingencias. Asimismo, se ha diseñado con una notable flexibilidad en la entrega durante la primera semana, con el objetivo de brindar un margen amplio que permita una preparación óptima por parte de los involucrados.



Los objetivos que se perseguirán en cada actividad se delinean de la siguiente manera:

<b>Actividad</b>	<b>Objetivo/Entregables</b>	<b>Duración</b>
Generación de Datos de Entrada	Crear un conjunto estructurado de datos de entrada, incluyendo coordenadas del almacén, ubicaciones de clientes y capacidad de vehículos.	7 días
Modelado del Problema	Representar el problema como un grafo. Definir restricciones. Definir el testplan. Hacer el diagrama de flujo. Hacer el boceto de la interfaz. Hacer un diagrama caso de uso.	7 días
Selección del Algoritmo	Elegir un algoritmo que se va a aplicar al código.	7 días
Implementación	Desarrollar e implementar el algoritmo seleccionado en Python.	6 días
Interfaz de Usuario	Crear una interfaz de usuario amigable para cargar datos, visualizar rutas optimizadas y obtener soluciones.	6 días
Estrategias de Optimización	Implementar estrategias para mejorar las rutas generadas, asegurando cumplimiento de restricciones de capacidad.	2 días
Pruebas y Validación	Probar el algoritmo con diferentes conjuntos de datos, incluyendo instancias pequeñas y grandes. Verificar la validez y cercanía a la optimalidad.	2 días
Implementación de la Optimización	Implementar las estrategias de optimización	3 días
Evaluación del Rendimiento	Medir rendimiento en términos de distancia total recorrida, equilibrio de carga y tiempo de ejecución.	2 días

Con fundamento en lo anteriormente expuesto, he elaborado un plan estratégico, o "RoadMap", destinado a cada rol, distribuido a lo largo de las tres semanas que abarcará el proyecto, presentando la siguiente estructura:

## **Semana 1: Planificación y Preparación**

**Fecha máxima de entrega: 16 de noviembre**

### **Todos los Roles:**

- Revisar y comprender completamente los requisitos del proyecto.
- Modelar la solución para el problema

### **FrontEnd - Gael Emiliano Anaya García:**

- Comenzar el diagrama de flujo junto con el BackEnd.
- Analizar junto con el BackEnd como se relacionará con la lógica del algoritmo
- Investigar y evaluar tecnologías específicas para implementar la interfaz
- Revisar el boceto.

### **FrontEnd 2 - Naylea Danae Silva Penaloza:**

- Investigar y analizar sobre las tendencias de diseño.
- Desarrollar estrategias para la interacción del usuario
- Diseñar diagrama caso de uso
- Diseñar el boceto de la interfaz

### **BackEnd - Carlos Humberto Avila Sanchez:**

- Comenzar el diagrama de flujo junto con el FrontEnd.
- Investigar y seleccionar el algoritmo adecuado para el problema.
- Analizar junto con el FrontEnd como se relacionará con la interfaz del algoritmo
- Iniciar el diseño y la planificación de la implementación del algoritmo en Python.

### **Tester - Nahomi Itzel Luna Ornelas:**

- Preparar conjuntos de datos de prueba para validar el algoritmo.
- Preparar el testplan para la fase de prueba y validación.
- Investigar sobre el uso de GitHub y Git.
- Training a compañeros para aprender sobre el uso.

Esta fase es crucial para establecer una base sólida para el proyecto. Incluye la comprensión completa de los requisitos, la planificación detallada y la asignación de responsabilidades. La comunicación efectiva y la definición clara de roles y objetivos son fundamentales para el éxito del proyecto.

## **Semana 2: Implementación Inicial y Diseño de la Interfaz**

**Fecha máxima de entrega: 20 de noviembre**

### **FrontEnd - Gael Emiliano Anaya García:**

- Continuar el diseño de la interfaz de usuario.
- Desarrollar la parte inicial de la interfaz para cargar datos.
- Colaborar con el BackEnd para una buena integración con el algoritmo.

### **FrontEnd 2 - Naylea Danae Silva Penaloza:**

- Implementar posibles mejoras de diseño.
- Desarrollar la parte final de la interfaz para mostrar los datos.
- Colaborar con el BackEnd para una buena integración con el algoritmo.

### **BackEnd - Carlos Humberto Avila Sanchez:**

- Comenzar la implementación del algoritmo en Python.
- Integrar el algoritmo con el sistema general.
- Colaborar con ambos FrontEnd para la buena integración con la interfaz.

### **Tester - Nahomi Itzel Luna Ornelas:**

- Ejecutar pruebas preliminares con conjuntos de datos pequeños.
- Validar la carga de datos y la salida del algoritmo en una etapa temprana.
- Pruebas de Usabilidad de etapas tempranas de la interfaz.

Durante esta fase, se inicia el desarrollo del algoritmo de optimización y el diseño de la interfaz de usuario. Esta división permite que ambos equipos trabajen simultáneamente, aprovechando las fortalezas de cada uno. Además, el diseño de la interfaz puede evolucionar de manera iterativa, permitiendo ajustes continuos mientras el BackEnd avanza.

## **Semana 3: Optimización y Pruebas Finales**

**Fecha máxima de entrega: 25 de noviembre**

### **Todos los Roles:**

- Revisar y ajustar el plan según el progreso actual.
- Resolver cualquier problema de comunicación o bloqueo.
- Preparación para la presentación.

### **FrontEnd - Gael Emiliano Anaya García:**

- Implementar la visualización de rutas optimizadas y soluciones.
- Documentación de la interfaz

### **FrontEnd 2 - Naylea Danae Silva Penaloza:**

- Implementar posibles mejoras de diseño.
- Desarrollar la parte final de la interfaz para mostrar los datos.

### **BackEnd - Carlos Humberto Avila Sanchez:**

- Optimizar el código del algoritmo y realizar pruebas de rendimiento.
- Implementar estrategias de optimización y asegurarse de que se integren correctamente.
- Documentación del algoritmo.

### **Tester - Nahomi Itzel Luna Ornelas:**

- Ejecutar pruebas exhaustivas con conjuntos de datos variados.
- Verificar la validez y la cercanía a la optimalidad de las soluciones.
- Validar el rendimiento y documentarlo.

Esta fase se centra en la optimización final y las pruebas exhaustivas. Después de implementar el algoritmo y la interfaz, es esencial optimizar el rendimiento y la usabilidad. Las pruebas finales garantizan que el sistema funcione según lo esperado y cumpla con los requisitos del usuario.

Asimismo, he aplicado la metodología OKR para definir objetivos claros y resultados clave específicos para cada rol en el proyecto. Esto ha ampliado de manera detallada las responsabilidades asignadas a cada miembro del equipo, promoviendo una comprensión profunda y una alineación efectiva con los objetivos estratégicos a lo largo de las tres semanas del proyecto. Además, este enfoque también ha sido fundamental para estructurar de manera eficiente el diagrama de flujo tanto de cada rol individual como del proyecto en su conjunto.

## FrontEnd - Gael Emiliano Anaya García

### Objetivo 1: Planificar y Preparar la interfaz

Objetivo	Resultados Clave
Completar el diseño del diagrama de flujo con el BackEnd.	- Diagrama de flujo completado.
Definir la relación clara entre el FrontEnd y la lógica del algoritmo.	- Documentación de la relación establecida.
Investigar y evaluar tecnologías específicas para la implementación de la interfaz.	- Lista de tecnologías evaluadas.
Revisar y ajustar el boceto de la interfaz.	- Boceto revisado y aprobado.

### Objetivo 2: Implementar y Diseñar de la Interfaz

Objetivo	Resultados Clave
Continuar el diseño de la interfaz de usuario.	- Avances documentados en el diseño de la interfaz.
Desarrollar la parte inicial de la interfaz para cargar datos.	- Módulo de carga de datos implementado.
Colaborar con el BackEnd para una integración eficiente con el algoritmo.	- Integración exitosa con el BackEnd.

### Objetivo 3: Optimizar la interfaz

Objetivo	Resultados Clave
Implementar la visualización de rutas optimizadas y soluciones.	- Visualización de rutas funcionando correctamente.
Documentar la interfaz.	- Documentación completa y actualizada.



## FrontEnd2 - Naylea Danae Silva Penaloza:

### Objetivo 1: Diseñar la interfaz

Objetivo	Resultados Clave
Investigar y analizar las tendencias de diseño actuales.	- Resumen de las tendencias de diseño identificadas.
Desarrollar estrategias para la interacción del usuario.	- Estrategias documentadas para la interacción del usuario.
Diseñar un diagrama de caso de uso.	- Diagrama de caso de uso completo y validado.
Diseñar el boceto de la interfaz.	- Boceto de interfaz revisado y aprobado.

### Objetivo 2: Implementar la Interfaz

Objetivo	Resultados Clave
Implementar posibles mejoras de diseño en la interfaz.	- Mejoras de diseño implementadas y evaluadas.
Desarrollar la parte final de la interfaz para mostrar los datos.	- Módulo de visualización de datos implementado.
Colaborar con el BackEnd para una buena integración con el algoritmo.	- Integración efectiva con el BackEnd.

### Objetivo 3: Optimizar y adaptar la interfaz

Objetivo	Resultados Clave
Implementar optimizaciones visuales en la interfaz.	- Optimizaciones visuales implementadas y evaluadas.
Documentar la interfaz.	- Documentación completa y detallada de la interfaz.

## BackEnd - Carlos Humberto Avila Sanchez

### Objetivo 1: Planificar y modelar el algoritmo

Objetivo	Resultados Clave
Comenzar el diagrama de flujo junto con el FrontEnd.	- Diagrama de flujo colaborativo FrontEnd-BackEnd iniciado.
Investigar y seleccionar el algoritmo adecuado para el problema.	- Algoritmo seleccionado y justificación documentada.
Analizar junto con el FrontEnd cómo se relacionará con la interfaz del algoritmo.	- Documentación de la relación entre el algoritmo y la interfaz.
Iniciar el diseño y la planificación de la implementación del algoritmo en Python.	- Plan de implementación del algoritmo en Python creado.

### Objetivo 2: Implementar el algoritmo

Objetivo	Resultados Clave
Comenzar la implementación del algoritmo en Python.	- Código del algoritmo en Python iniciado.
Integrar el algoritmo con el sistema general.	- Integración exitosa del algoritmo con el sistema.
Colaborar con ambos FrontEnd para la buena integración con la interfaz.	- Integración efectiva entre el FrontEnd y el algoritmo.

### Objetivo 3: Optimizar el algoritmo

Objetivo	Resultados Clave
Optimizar el código del algoritmo y realizar pruebas de rendimiento.	- Código del algoritmo optimizado y pruebas de rendimiento realizadas.
Implementar estrategias de optimización y asegurarse de que se integren correctamente.	- Estrategias de optimización implementadas y evaluadas.
Documentación del algoritmo.	- Documentación completa y detallada del algoritmo.

## Tester - Nahomi Itzel Luna Ornelas

### Objetivo 1: Preparar al equipo y generar los datos

Objetivo	Resultados Clave
Preparar conjuntos de datos de prueba para validar el algoritmo.	- Conjuntos de datos de prueba creados y documentados.
Preparar el testplan para la fase de prueba y validación.	- Testplan detallado y aprobado.
Investigar sobre el uso de GitHub y Git.	- Informe de investigación sobre el uso de GitHub y Git.
Training a compañeros para aprender sobre el uso.	- Sesiones de capacitación realizadas y asimiladas por los compañeros.

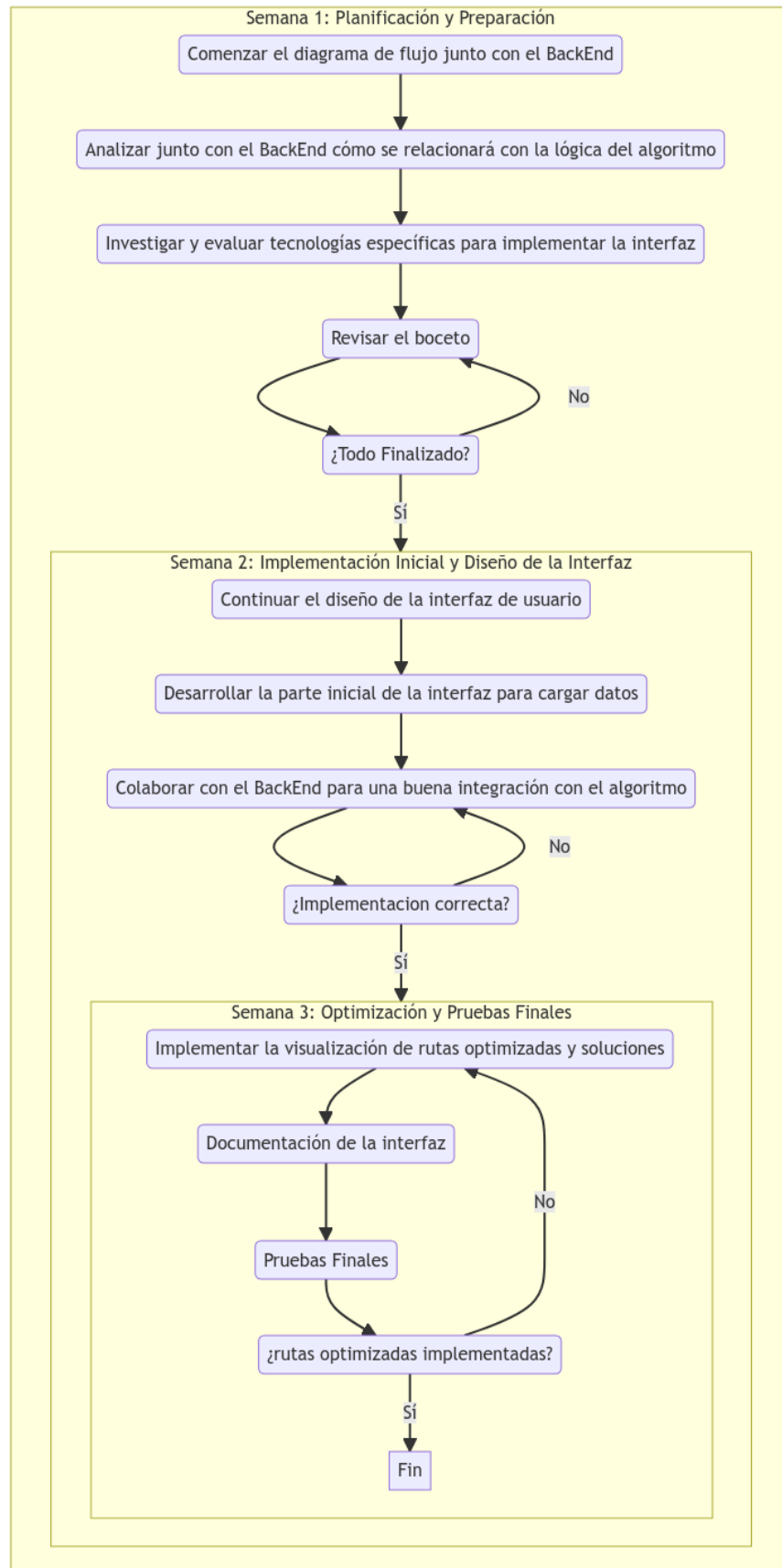
### Objetivo 2: Ejecutar pruebas y encontrar errores en etapas tempranas

Objetivo	Resultados Clave
Ejecutar pruebas preliminares con conjuntos de datos pequeños.	- Resultados de pruebas preliminares documentados.
Validar la carga de datos y la salida del algoritmo en una etapa temprana.	- Validación exitosa de la carga y salida del algoritmo.
Pruebas de Usabilidad de etapas tempranas de la interfaz.	- Retroalimentación de usabilidad recopilada y evaluada.

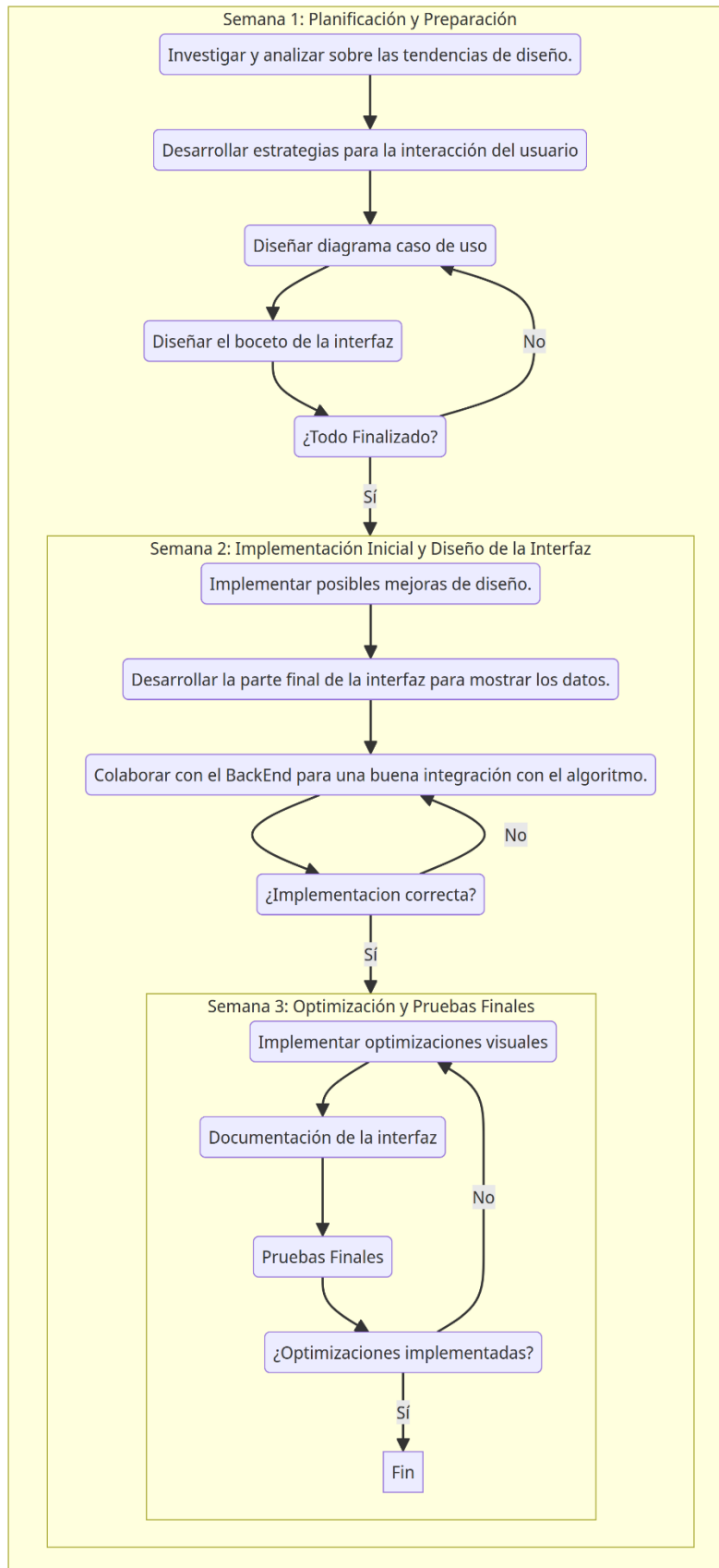
### Objetivo 3: Optimizar y medir el rendimiento del programa

Objetivo	Resultados Clave
Ejecutar pruebas exhaustivas con conjuntos de datos variados.	- Resultados de pruebas exhaustivas documentados y analizados.
Verificar la validez y la cercanía a la optimalidad de las soluciones.	- Validación de soluciones y cercanía a la optimalidad verificadas.
Validar el rendimiento y documentarlo.	- Validación de rendimiento completa y documentada.

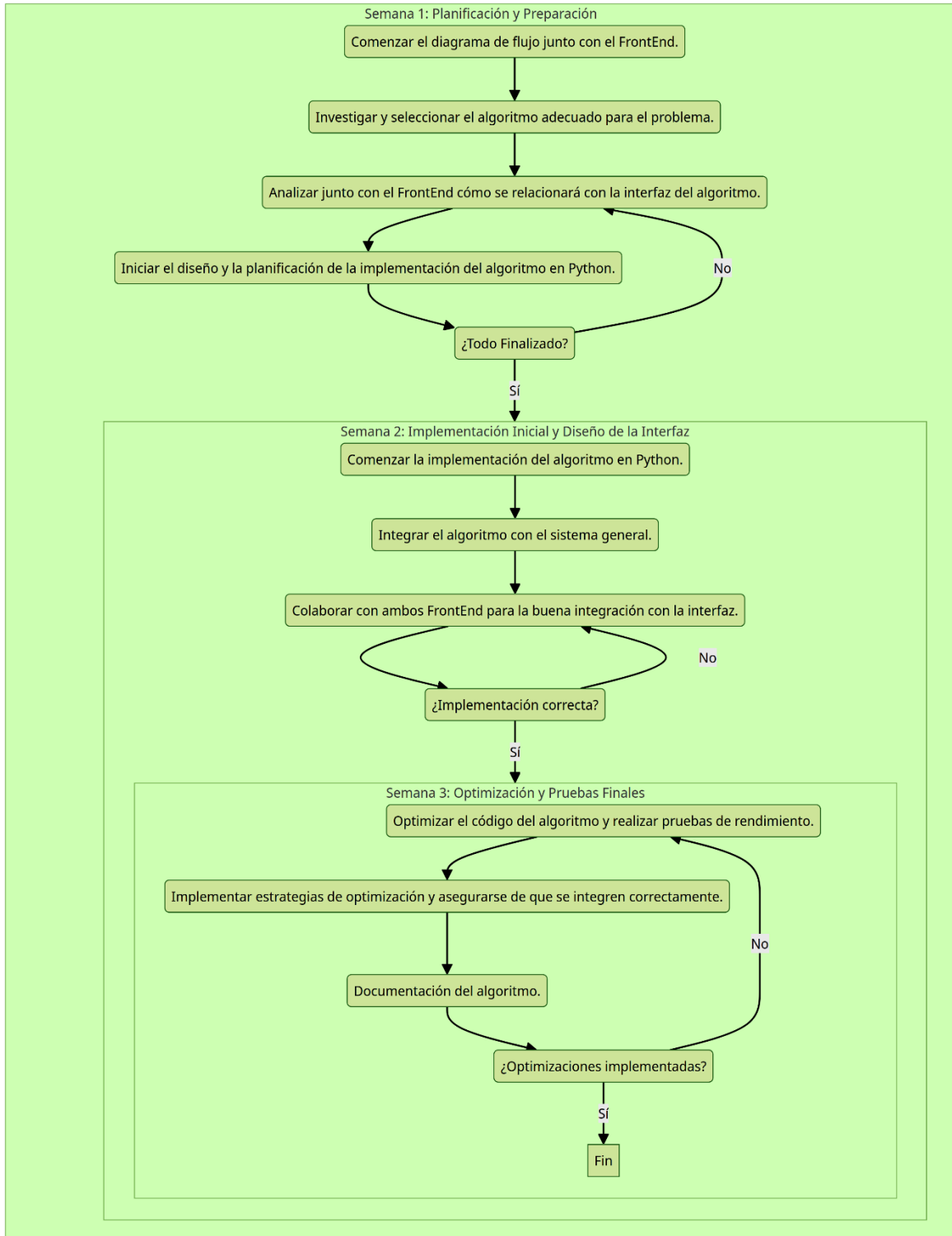
## FrontEnd - Gael Emiliano Anaya García



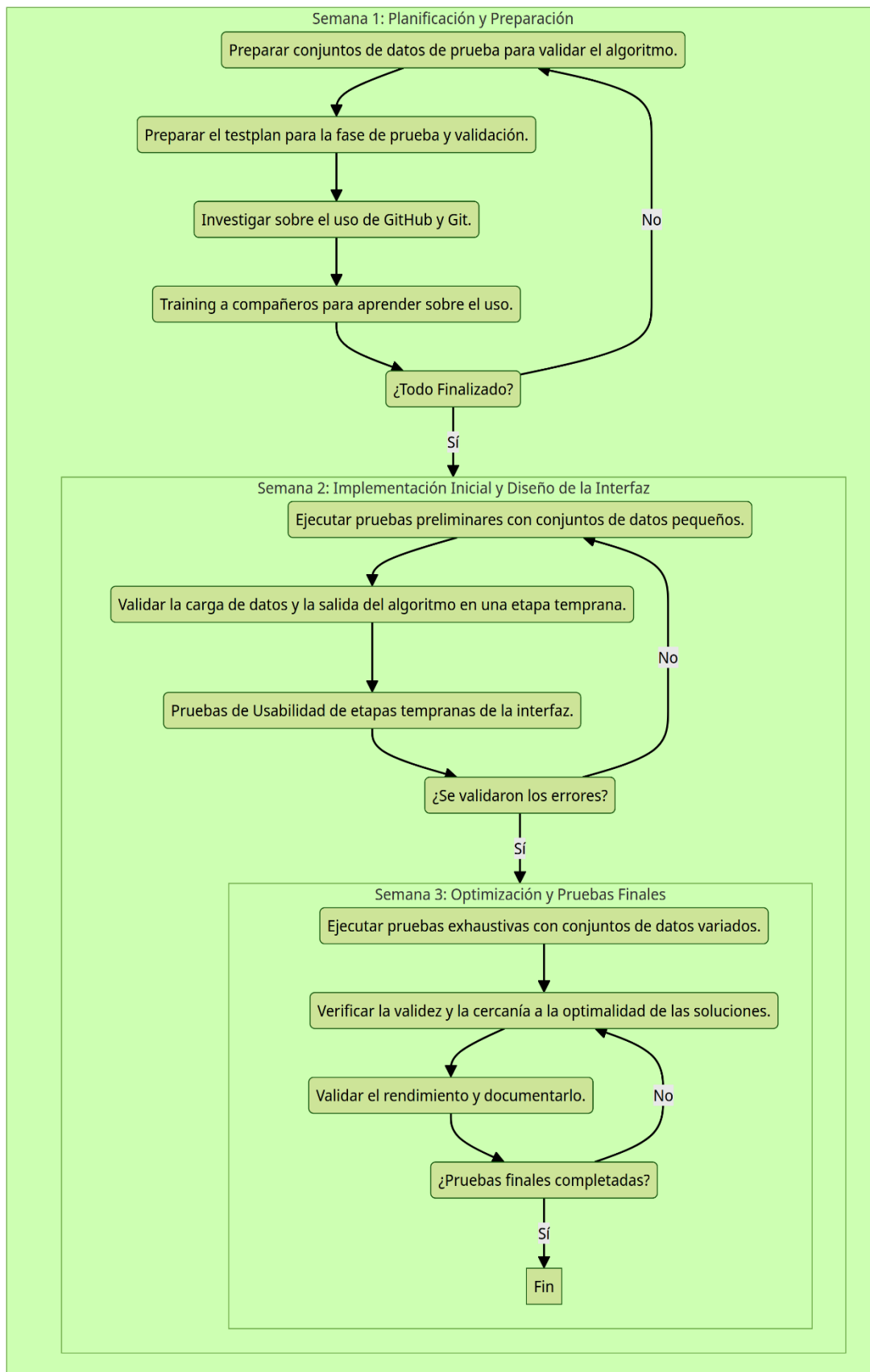
## FrontEnd2 - Naylea Danae Silva Penalzoa



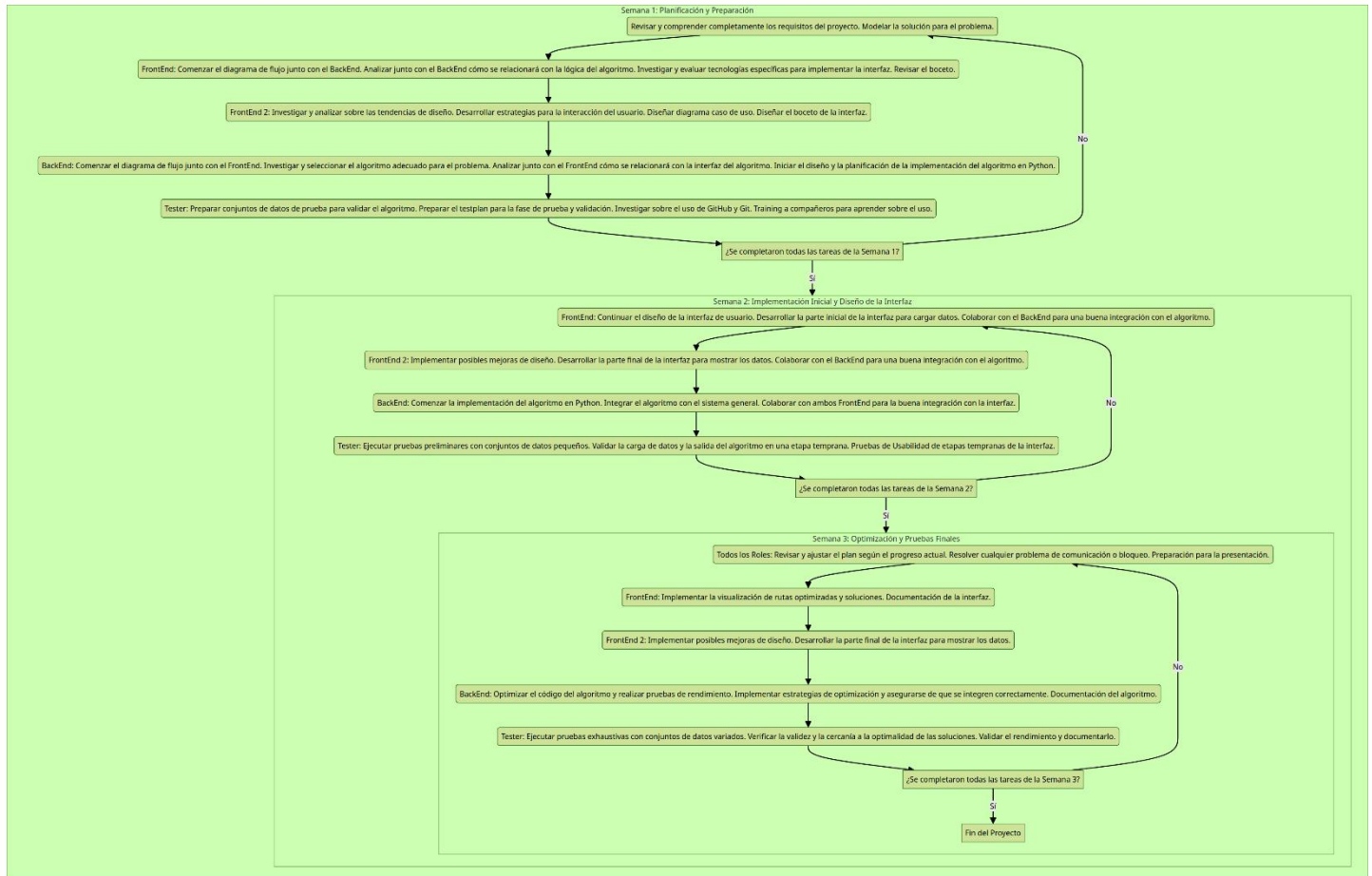
## BackEnd - Carlos Humberto Avila Sanchez



## Tester - Nahomi Itzel Luna Ornelas

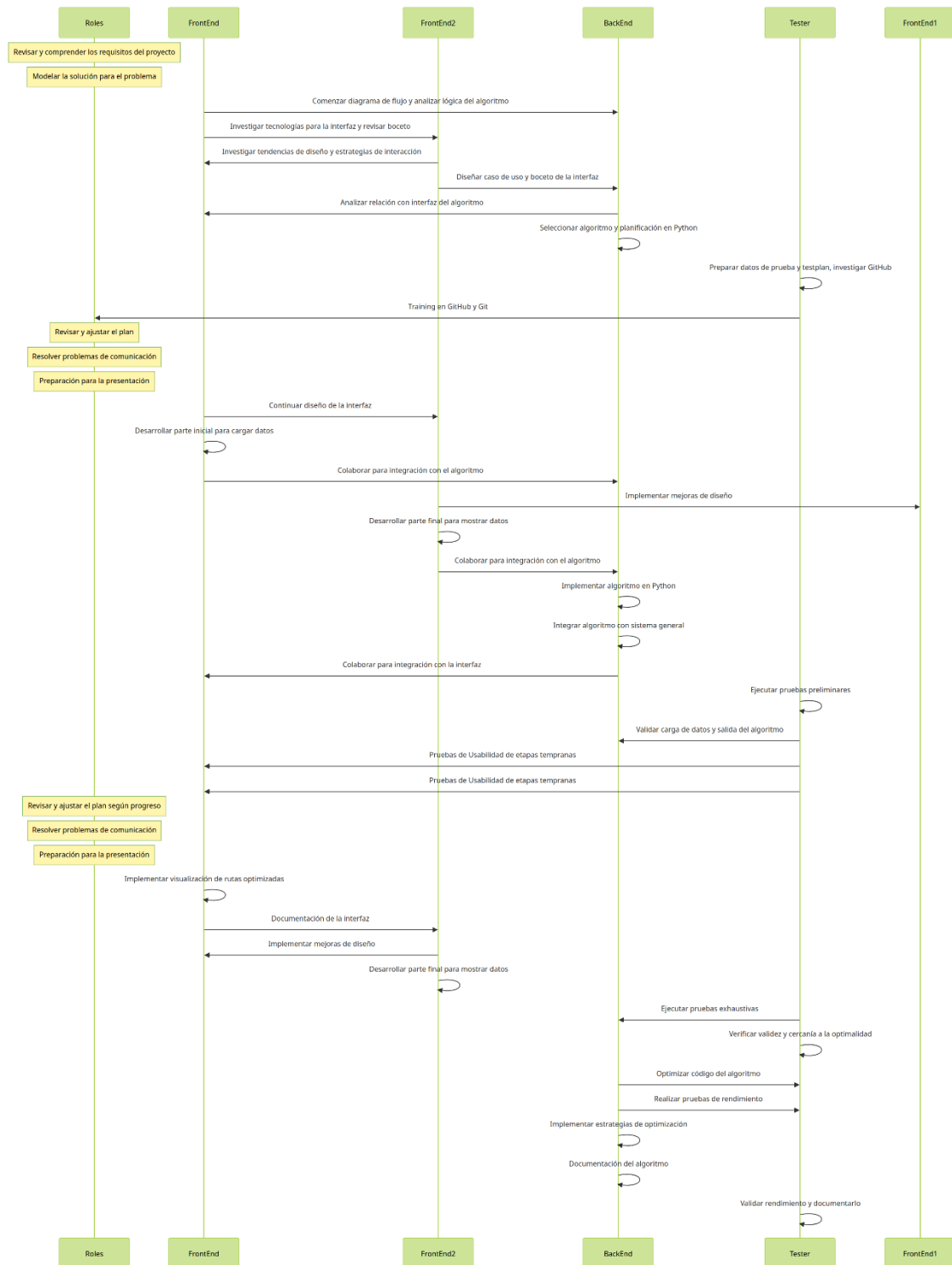


# Proyecto Completo





# Diagrama de Secuencia



Se eligió esta metodología porque el enfoque de Equipo ayuda a la distribución de tareas, como FrontEnd, BackEnd y Tester, permite un enfoque más eficiente y especializado. Cada equipo puede concentrarse en su área de experiencia, garantizando una implementación de alta calidad además de ser iterativo ya que permite ajustes y mejoras continuas a medida que el proyecto avanza. Las fases posteriores pueden beneficiarse de los aprendizajes y descubrimientos de las fases anteriores, permitiendo una adaptación constante y el hecho de dividir el proyecto en fases permite validar cada componente y funcionalidad de manera progresiva. Esto ayuda a identificar y abordar problemas temprano en el proceso, mejorando la calidad general del proyecto y además la inclusión de la documentación en varias fases garantiza que se capturen y comuniquen adecuadamente los detalles del proyecto, desde los requisitos hasta la implementación y los resultados obtenidos.

### **Project Manager - Carlos Andres Chico Aguayo**

Tras haber evaluado exhaustivamente las designaciones de roles atribuidas a los demás integrantes del proyecto, considero oportuno proporcionar una descripción más detallada de mi función como Director de Proyecto. Mi papel se centrará en orquestar y supervisar eficazmente todas las fases del proyecto, desde la concepción hasta la implementación final.

En este sentido, asumiré la responsabilidad de la planificación estratégica, la asignación eficiente de recursos y la gestión de los plazos establecidos. Mi objetivo primordial será garantizar la cohesión y coordinación entre los diversos roles, cultivando así un entorno de trabajo sinérgico y eficiente. En mi papel como líder, trabajaré en estrecha colaboración con cada miembro del equipo, adaptando mi enfoque de liderazgo según las necesidades específicas de cada fase del proyecto.

### **Informe de Estado Semanal:**

- Incluye métricas clave de rendimiento para cuantificar el progreso.
- Destaca los éxitos individuales del equipo y muestra reconocimiento.
- Proporciona una sección de lecciones aprendidas para compartir experiencias valiosas.

### **Actualización del Cronograma:**

- Identifica dependencias críticas y cualquier cambio en la secuencia de tareas.
- Utiliza gráficos visuales para ilustrar el progreso y los hitos alcanzados.
- Indica cualquier desviación significativa del plan original y las razones detrás de ellas.

**Registro de Problemas:**

- Clasifica los problemas en función de su gravedad y urgencia.
- Proporciona una evaluación de impacto en el cronograma y el presupuesto.
- Incluye soluciones propuestas y los responsables de implementarlas.

**Resumen de Tareas Realizadas:**

- Relaciona cada tarea con los objetivos y metas del proyecto.
- Muestra cómo las tareas completadas contribuyen al progreso general.
- Destaca cualquier mejora en la eficiencia o proceso.

**Revisión de Recursos:**

- Evalúa la carga de trabajo de cada miembro del equipo y ajusta según sea necesario.
- Considera la capacitación continua para mejorar las habilidades del equipo.
- Asegúrate de que los recursos externos estén alineados con los requisitos del proyecto.

**Revisión de Calidad:**

- Destaca las áreas de mejora y acciones correctivas implementadas.
- Utiliza datos y métricas para respaldar la evaluación de calidad.
- Proporciona ejemplos específicos de cómo se está manteniendo la calidad.

**Seguimiento de Riesgos:**

- Actualiza la probabilidad e impacto de los riesgos identificados.
- Muestra cómo se están gestionando los riesgos para mitigar su impacto.
- Destaca cualquier cambio significativo en la lista de riesgos.

**Planificación para la Próxima Semana:**

- Establece expectativas claras para los objetivos clave y los resultados esperados.
- Asigna responsabilidades específicas para cada tarea planificada.
- Considera la flexibilidad para adaptarse a cambios inesperados.

**Solicitudes de Apoyo o Recursos:**

- Proporciona datos y análisis que respalden cada solicitud.
- Muestra cómo las solicitudes de recursos se alinean con los objetivos del proyecto.
- Proporciona un plan para evaluar la efectividad de cualquier recurso adicional.

Viendo lo anterior el plan de trabajo que voy a tomar será el siguiente:

## *Semana 1: Planificación y Preparación*

### **Fecha de entrega del reporte semanal: 16 de noviembre**

**Resumen General:** La semana se enfocará en la comprensión completa de los requisitos del proyecto y la modelación de la solución. La asignación de responsabilidades será clara, estableciendo las bases para una colaboración efectiva entre los equipos.

### **Revisión y Comprensión de Requisitos:**

- Asegurar que todos los miembros del equipo comprendan completamente los requisitos del proyecto.

### **Modelado de Solución:**

- Supervisar la creación del diagrama de flujo y la relación entre FrontEnd y BackEnd.

### **Asignación de Tareas:**

- Confirmar que cada miembro del equipo tenga tareas claras y comprensibles.

### **Comunicación Efectiva:**

- Facilitar la comunicación entre los equipos para garantizar la coherencia en la planificación.

### **Definición de Roles y Objetivos:**

- Asegurarse de que cada miembro del equipo tenga objetivos claros para la fase de planificación.

En esta etapa, liderare la revisión y comprensión completa de los requisitos del proyecto con todo el equipo. Se supervisará el modelado de la solución mediante la creación de un diagrama de flujo y se asegurará de que la asignación de responsabilidades sea clara. La comunicación efectiva y la definición detallada de roles y objetivos serán mi responsabilidad para establecer una base sólida.

## *Semana 2: Implementación Inicial y Diseño de la Interfaz*

**Fecha de entrega del reporte semanal: 20 de noviembre**

**Resumen General:** La implementación inicial del algoritmo y el diseño de la interfaz comenzarán. La colaboración entre FrontEnd y BackEnd se mantendrá fluida, permitiendo avances simultáneos.

### **1. Desarrollo de la Interfaz:**

- Supervisar el progreso en el diseño de la interfaz de usuario por parte de ambos equipos FrontEnd.

### **2. Implementación del Algoritmo:**

- Coordinar con el BackEnd para garantizar una integración suave del algoritmo con la interfaz.

### **3. Iteración en el Diseño de la Interfaz:**

- Facilitar la implementación de posibles mejoras en el diseño de la interfaz.

### **4. Pruebas Preliminares:**

- Supervisar las pruebas preliminares ejecutadas por el equipo de pruebas.

Durante esta semana, coordinare las actividades de los equipos FrontEnd y BackEnd. Me asegurare de que la implementación inicial del algoritmo y el diseño de la interfaz estén alineados con las metas del proyecto. La colaboración entre los equipos será clave, y supervisare la comunicación fluida para garantizar una integración efectiva.

### *Semana 3: Optimización y Pruebas Finales*

**Fecha de entrega del reporte semanal: 25 de noviembre**

**Resumen General:** La fase final se centrará en la optimización del código del algoritmo, la implementación de mejoras en el diseño y la realización de pruebas exhaustivas. Se prepararán para la presentación y se revisará el plan según el progreso actual.

**1. Optimización del Algoritmo:**

- Coordinar con el BackEnd para optimizar el código y realizar pruebas de rendimiento.

**2. Mejoras en el Diseño:**

- Facilitar la implementación de mejoras en el diseño de la interfaz.

**3. Pruebas Exhaustivas:**

- Supervisar las pruebas exhaustivas realizadas por el equipo de pruebas con conjuntos de datos variados.

**4. Preparación para la Presentación:**

- Asegurarse de que el equipo esté listo para la presentación, revisando la documentación de la interfaz y del algoritmo.

**5. Revisión del Plan:**

- Revisar y ajustar el plan según el progreso actual y resolver cualquier problema de comunicación o bloqueo.

En la fase final, liderare la optimización del código del algoritmo y las implementaciones de mejoras en el diseño. Se coordinará con ambos equipos para garantizar una integración armoniosa. El rol de pruebas también estará bajo mi supervisión, asegurándome de que las pruebas sean exhaustivas y validen el rendimiento del sistema. Además, revisare y ajustare el plan según el progreso, preparando al equipo para la presentación final.

Cada semana se entregará un documento que abordará y desarrollará cada punto del informe de estado.

