

# Entregables Semana 1

Reporte de avances en backend. Semana 1

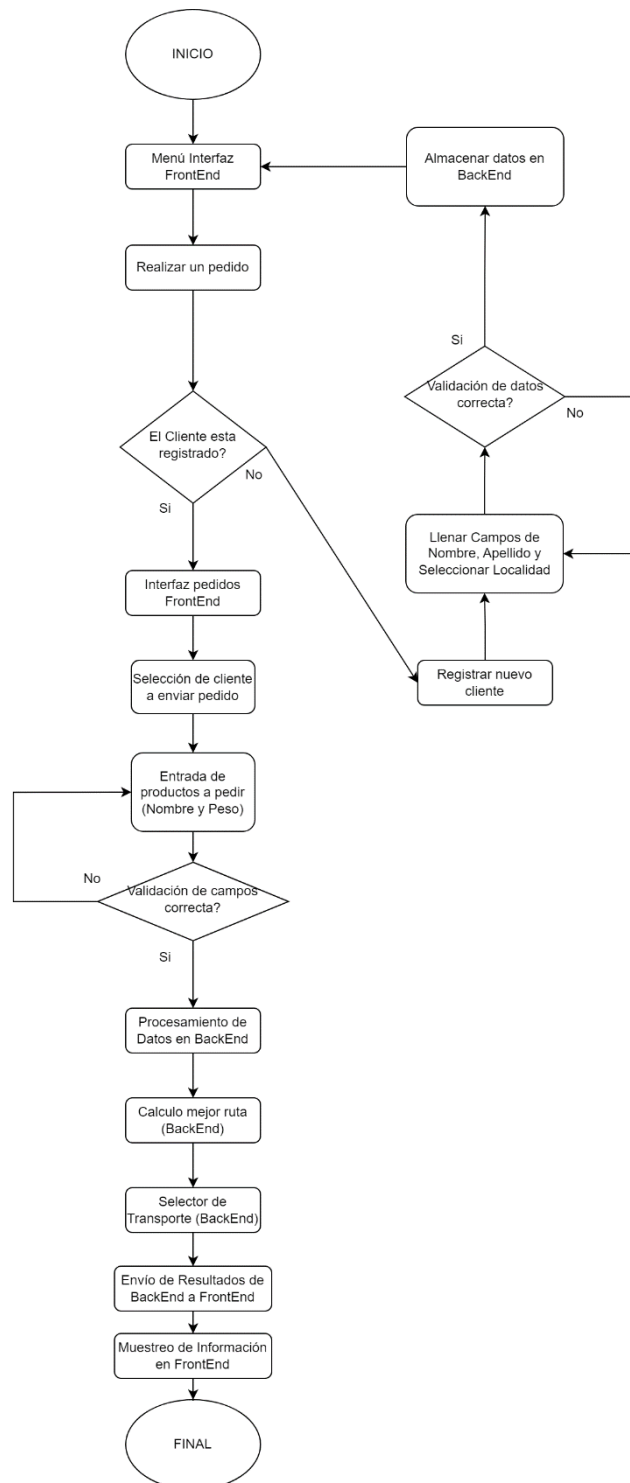
En esta primera semana se analizó el problema y se plantearon primeras soluciones.

# Entregables Semana 1

Reporte de avances en backend. Semana 1

- ✓ Comenzar con el diagrama de flujo junto con el Front-end.
- ✓ Investigar y seleccionar el algoritmo adecuado.
- ✓ Analizar con el Front-end la integración de la interfaz con el algoritmo.
- ✓ Iniciar el diseño y la planificación del algoritmo en Python.

## Diagrama de flujo junto con el Front-end.



## Investigar y seleccionar el algoritmo adecuado.

En la fase de planificación logística, se llevó a cabo una investigación para seleccionar los algoritmos más apropiados que optimizarán la distribución de productos en los vehículos. Para esta tarea específica, se optó por implementar el algoritmo de la mochila, reconocido por su eficiencia en problemas de asignación de recursos limitados. Este algoritmo demostrará ser fundamental para maximizar la capacidad de carga de cada vehículo, garantizando una distribución eficiente y equitativa de los productos.

Para mejorar aún más la eficiencia en la ejecución de las rutas, se ha decidido emplear un algoritmo de divide y vencerás. Este enfoque permitirá ordenar las rutas de manera estratégica, considerando las distancias entre los puntos de entrega. Al descomponer el problema en subproblemas más pequeños y abordarlos de manera independiente, este algoritmo contribuirá significativamente a minimizar las distancias totales recorridas, optimizando así la logística de transporte.

En cuanto al trazado de rutas específicas, se ha seleccionado el algoritmo de Dijkstra. Este algoritmo de búsqueda de rutas más cortas es reconocido por su eficacia en grafos ponderados, y será instrumental para determinar las rutas más eficientes y directas entre los puntos de entrega. La implementación de Dijkstra garantizará una planificación de rutas precisa, teniendo en cuenta las distancias y minimizando los tiempos de transporte, lo que resultará en una distribución de productos más rápida y eficiente en general.

Cabe recalcar que esta selección no será la final, ya que no descartamos cambiar de opinión dependiendo de las circunstancias.

## Analizar con el Front-end la integración de la interfaz con el algoritmo.

Durante la fase de análisis de la integración de la interfaz con el algoritmo, se ha tomado la decisión estratégica de implementar clases que permitan una fácil interoperabilidad entre el front-end y el back-end. Esta medida está diseñada para mejorar significativamente la experiencia de desarrollo, al proporcionar una arquitectura modular y reutilizable que facilitará la integración de la lógica del algoritmo con la interfaz de usuario.

La implementación de clases compartidas entre el front-end y el back-end permitirá una gestión coherente de la entrada y salida de datos, mejorando la cohesión y el control sobre los flujos de información. Además, esta práctica facilitará la realización de pruebas, ya que las clases compartidas podrán ser sometidas a pruebas unitarias tanto en el entorno del front-end como del back-end, garantizando así la consistencia y calidad del código.

La adopción de esta estrategia modular brindará una mayor flexibilidad y mantenibilidad al proyecto, ya que cualquier actualización en la lógica del algoritmo se reflejará automáticamente en la interfaz, proporcionando una experiencia de usuario más consistente y eficiente. Esta sinergia entre el front-end y el back-end contribuirá a la creación de un sistema robusto y fácil de mantener en el tiempo.

## Iniciar el diseño y la planificación del algoritmo en Python.

Se ha llevado a cabo la implementación de clases esenciales, dotadas de todos los atributos necesarios para garantizar la robustez y flexibilidad del sistema. Estas clases han sido concebidas para facilitar la integración fluida entre el front-end y el back-end, permitiendo un control coherente de la entrada y salida de datos.

En paralelo, se ha iniciado la ideación y planificación de la implementación del algoritmo. Se están explorando y evaluando diferentes enfoques para asegurar la eficiencia y precisión del algoritmo diseñado. Este proceso involucra la identificación de los componentes clave del problema y la definición de estrategias para abordarlos de manera efectiva.