Informe Proyecto GM Express – Sistema de Distribución de Pedidos

Eduardo Aravena

Cristopher Araya

Gabriel Muñoz

Proyecto de Integración

D-IEI-N6-P1-C4

Jorge Cortés

*10 de septiembre del 2025*

# 1. Introducción

GM Express es una empresa de distribución que actualmente enfrenta dificultades para gestionar pedidos, asignar choferes/vehículos y mantener un control eficiente de las entregas. Este proyecto propone el desarrollo de un sistema web que permita automatizar y optimizar el proceso de distribución, asegurando mayor trazabilidad, control y satisfacción de los clientes.

# 2. Metodología

***Lenguaje de programación***  
Se utilizará Python con el framework Django, dado que permite un desarrollo ágil de aplicaciones web con estructura Modelo-Vista-Controlador (MVC), integración nativa con bases de datos, seguridad incorporada y escalabilidad.

***Base de datos***Se empleará una base de datos relacional (MySQL), ya que el sistema requiere manejar información estructurada (pedidos, choferes, clientes, vehículos, estados). El modelo relacional facilita integridad referencial y consultas complejas para reportes.

***Servidor***  
La aplicación será desplegada en Amazon Web Services (AWS), específicamente en una instancia EC2 con Ubuntu Server, lo que garantiza disponibilidad y escalabilidad. Además:

* RDS (Relational Database Service) para alojar la base de datos MySQL en la nube.
* S3 (Simple Storage Service) para almacenamiento de archivos, como evidencia de entrega (fotos o firmas digitales).

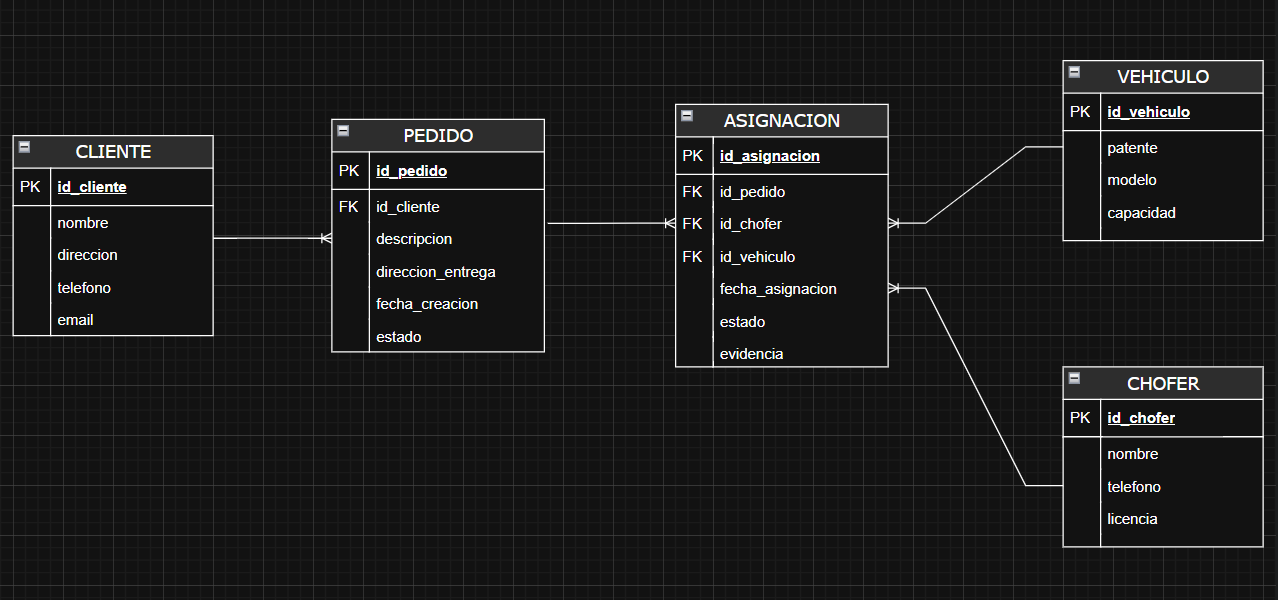
***Arquitectura / Infraestructura***  
La arquitectura propuesta es cliente-servidor en la nube bajo un esquema de 3 capas:

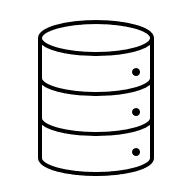
1. Capa de presentación: interfaz web y versión móvil responsive para clientes, choferes y administradores.
2. Capa de aplicación: servidor Django en AWS EC2, donde se gestionan las reglas de negocio (pedidos, asignaciones, estados, notificaciones).
3. Capa de datos: MySQL en AWS RDS y archivos en S3.

La infraestructura contempla:

* Balanceo de carga y escalabilidad automática en AWS.
* Seguridad con HTTPS y control de acceso por roles (cliente, chofer, administrador).

# 3. Modelo de Datos

El modelo entidad–relación define las principales entidades del sistema: Cliente, Pedido, Chofer, Vehículo, Estado, Evidencia. Se incluye un script SQL para la creación de tablas y relaciones que garantizan integridad y consistencia.



[script. sql](../Desktop/Project/extra%20(proyecto%20integracion)/Modelo%20de%20Datos/script.sql)

CREATE DATABASE gmexpress;

USE gmexpress;

CREATE TABLE clientes (

id\_cliente INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(100) NOT NULL,

direccion VARCHAR(255),

telefono VARCHAR(20),

email VARCHAR(100) UNIQUE

);

CREATE TABLE choferes (

id\_chofer INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(100) NOT NULL,

telefono VARCHAR(20),

licencia VARCHAR(50)

);

CREATE TABLE vehiculos (

id\_vehiculo INT AUTO\_INCREMENT PRIMARY KEY,

patente VARCHAR(20) UNIQUE NOT NULL,

modelo VARCHAR(100),

capacidad INT

);

CREATE TABLE pedidos (

id\_pedido INT AUTO\_INCREMENT PRIMARY KEY,

id\_cliente INT,

descripcion TEXT NOT NULL,

direccion\_entrega VARCHAR(255) NOT NULL,

fecha\_creacion TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

estado ENUM('pendiente','en\_camino','entregado') DEFAULT 'pendiente',

FOREIGN KEY (id\_cliente) REFERENCES clientes(id\_cliente)

);

CREATE TABLE asignaciones (

id\_asignacion INT AUTO\_INCREMENT PRIMARY KEY,

id\_pedido INT,

id\_chofer INT,

id\_vehiculo INT,

fecha\_asignacion TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

estado ENUM('pendiente','en\_camino','entregado') DEFAULT 'pendiente',

evidencia VARCHAR(255),

FOREIGN KEY (id\_pedido) REFERENCES pedidos(id\_pedido),

FOREIGN KEY (id\_chofer) REFERENCES choferes(id\_chofer),

FOREIGN KEY (id\_vehiculo) REFERENCES vehiculos(id\_vehiculo)

);

# 4. Diagramas

## 4.1 Diagramas de Casos de Uso

Los casos de uso reflejan las interacciones entre los actores (Cliente, Chofer, Administrador) y el sistema.

Insertar aquí los 12 casos de uso.

## 4.2 Diagramas de Clases

Texto

El contenido generado por IA puede ser incorrecto.Los diagramas de clases muestran la estructura del sistema y las relaciones entre objetos.

Texto

El contenido generado por IA puede ser incorrecto.

Pantalla de computadora con letras

El contenido generado por IA puede ser incorrecto.

## Diagrama El contenido generado por IA puede ser incorrecto.

## 4.3 Diagramas de Secuencia

Los diagramas de secuencia describen el flujo de eventos desde el pedido hasta la entrega final.

Insertar aquí los 4 diagramas de secuencia.

# 5. Mockups

Los mockups representan la interfaz gráfica de las diferentes vistas del sistema:  
- Cliente (formulario de pedidos, mis pedidos).  
- Chofer (pedidos asignados, actualización de estado, subir evidencia).  
- Administrador (gestión de pedidos, reportes, incidencias, métricas).

Interfaz de usuario gráfica, Aplicación, Tabla

El contenido generado por IA puede ser incorrecto.  
Vista admin – Dashboard.

Vista cliente – Mis pedidos.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Realizar una orden.

Vista chofer – Ver pedidos asignados.

Interfaz de usuario gráfica, Tabla

El contenido generado por IA puede ser incorrecto.

Actualizar estado del pedido.

Diagrama

El contenido generado por IA puede ser incorrecto.

# 6. Conclusiones

El sistema propuesto permitirá:  
- Optimizar la gestión de distribución de GM Express.  
- Mejorar la trazabilidad de los pedidos y la transparencia hacia los clientes.  
- Generar métricas y reportes para apoyar la toma de decisiones estratégicas.  
- Facilitar la asignación y seguimiento de recursos en tiempo real.  
  
Como trabajo futuro, se plantea la integración con APIs de mapas y GPS en tiempo real, además del desarrollo de una aplicación móvil nativa.