

## 2do examen parcial de sistemas de información II – 14/10/2025

El presente documento es una breve descripción que se debe tomar base inicial para el desarrollo del proyecto, sin embargo, cada alumno como parte del examen deberá realizar una interpretación y compresión de la temática que aborda el presente proyecto

### 1. Nombre del Proyecto

“SmartSales365”

Sistema Inteligente de Gestión Comercial y Reportes Dinámicos

### 2. Descripción General del Proyecto

Desarrollar un sistema de gestión comercial inteligente denominado “SmartSales365”, el cual combine funcionalidades tradicionales de gestión de ventas con componentes de Inteligencia Artificial aplicada.

El sistema deberá permitir: **3 módulos**

- Gestionar información comercial (clientes, productos, ventas)
- Generar reportes dinámicos a través de prompts de texto o comandos de voz
- Incluir un dashboard con predicciones de ventas utilizando un modelo de aprendizaje automático (Machine learning : Random Forest Regressor )

Complementar la versión web con una **aplicación móvil Flutter**, la cual deberá incluir funcionalidades estratégicamente seleccionadas que aprovechen capacidades propias del entorno móvil (por ejemplo, notificaciones push).

Este proyecto busca integrar conceptos de desarrollo de sistemas de gestión empresarial avanzado, tecnologías full stack modernas y aplicaciones reales de IA en un caso práctico, simulando escenarios empresariales reales.

### 3. Stack Tecnológico Requerido

Para garantizar uniformidad en la evaluación y asegurar que se desarrolle competencias reales, deberán utilizar obligatoriamente el siguiente stack:

- **Backend**
  - Python (Django REST Framework)
  - Generación de reportes (PDF, Excel)

- Modelo de IA integrado para predicción de ventas
- API REST documentada (Swagger)
- **Frontend Web**
  - React
  - Aplicación web progresiva (PWA), Responsive y otras orientada a facilitar la experiencia del usuario
- **Aplicación Móvil**
  - Flutter
- **Base de Datos**
  - PostgreSQL
- **Despliegue**
  - AWS, Azure o Google cloude

#### 4. Funcionalidades Mínimas Obligatorias

##### a. Gestión Comercial Básica

El sistema debe cubrir, como mínimo:

- Gestión de **productos** (categorías, precios, stock).
- Gestión de **clientes**
- Gestionar **carrito de compra** online que permite a los usuarios seleccionar productos y comprarlos (Texto y Voz)
- Gestionar métodos de pago en línea (PayPal, Stripe)
- Gestión de **ventas**
- Emisión de comprobantes (notas de venta)
- Listado histórico de ventas con filtros básicos.

##### b. Generación Dinámica de Reportes (Texto o Voz)

Los usuarios deberán poder generar reportes a demanda mediante:

- Prompt de texto (en la interfaz web y/o móvil).
- Comando de voz, usando reconocimiento de voz en el frontend y enviando el texto resultante al backend para su interpretación.

Ejemplos de prompts válidos:

- “Quiero un reporte de ventas del mes de septiembre, agrupado por producto, en PDF.”

- “Quiero un reporte en Excel que muestre las ventas del periodo del 01/10/2024 al 01/01/2025. Debe mostrar el nombre del cliente, la cantidad de compras que realizó, el monto total que pagó y el rango de fechas en las que hizo la compra.”

El backend deberá:

1. Interpretar el prompt recibido.
2. Construir dinámicamente la consulta SQL (o su equivalente en un ORM) necesaria.
3. Generar el reporte en el formato solicitado (pantalla, PDF o Excel).
4. Retornar el resultado para su visualización o descarga.

Para la interpretación de comandos de texto, pueden usar reglas, expresiones regulares o un parser propio simple. No se exige NLP avanzado.

Para voz, se permite el uso de la Web Speech API (navegadores) o librerías de Flutter como speech\_to\_text.

## Dashboard de Predicción de Ventas

El sistema deberá incluir un dashboard que muestre:

- Ventas históricas (por período, producto, cliente).
- Predicciones de ventas futuras (por categoría o total mensual).
- Gráficas dinámicas (líneas, barras, comparativas).

Modelo de IA requerido:

- Random Forest Regressor (de scikit-learn), entrenado con datos históricos de ventas (initialmente dataset para entrenamiento pueden ser datos sintéticos)

Este modelo fue seleccionado por su facilidad de implementación, buena capacidad de generalización y porque no requiere un dataset extenso ni preprocesamiento complejo.

Se espera que el modelo:

- Se entrene con los datos de la base de datos periódicamente
- Se serialice (guarde) para reutilización.
- Genere predicciones reales que se visualicen en el dashboard.

Frontend puede usar librerías como Recharts o Chart.js u otras para la visualización.

### c. Aplicación Móvil (Flutter)

La app móvil no debe replicar todas las funcionalidades de la web. En cambio, cada equipo debe identificar y justificar qué funcionalidades trasladar al entorno móvil.

Se recomienda priorizar:

- Visualización de reportes o dashboards resumidos.
- Acciones rápidas.
- Notificaciones push

La selección de funcionalidades será parte de la evaluación. Se valorará el uso estratégico del entorno móvil.

## 5. Proceso de Desarrollo

El desarrollo debe seguir el Proceso Unificado de Desarrollo de Software, utilizando UML 2.5+ como lenguaje de modelado.

## 6. Recursos Técnicos Recomendados

- **Python / IA:**
  - pandas, numpy
  - scikit-learn (RandomForestRegressor)
  - joblib (serialización de modelo)
  - reportlab / weasyprint / openpyxl
- **Frontend React:**
  - axios, SpeechRecognition, Recharts o Chart.js
- **Flutter:**
  - speech\_to\_text, http, firebase\_messaging (para push)

## 7. Entregables

1. Documentación: (arquitectura, modelo UML, descripción IA, manual breve de uso.)
2. Repositorio en GitHub (backend / frontend / móvil) (imprimir QR en documentación)

**8. Evaluación (Segundo Examen Parcial 20 puntos)****PRESENTACION 1: MARTES 28/10/2025 23:59****PRESENTACION 2: MARTES 04/11/2025 23:59****PRESENTACION 3 : MARTES 11/11/2025 23:59****DEFENSA SEGUNDO EXAMEN PARCIAL: JUEVES 13 NOVIEMBRE****9. Consideraciones Finales**

- Trabajo en grupos de 2
- Se evaluará tanto el producto final como el dominio individual durante la defensa oral
- El uso de IA debe ser real y justificable
- Se permite el uso de librerías estándar, pero no soluciones completas pre-armadas.
- Se valorará la creatividad en la definición de prompts y funcionalidades móviles.
- Se recomienda trabajar con datos simulados pero consistentes para asegurar entrenamientos realistas.

Se penalizará la copia literal de código entre grupos

**10. DOCUMENTO****DOCUMENTO SEGUNDO PARCIAL****1) Perfil**

1.1 Introducción

1.2 Objetivo General

1.3 Objetivos Específicos

1.4 Descripción del problema

1.4 Alcance

Marco Teórico

**2) Proceso de desarrollo**

- a) Seguir los pasos descritos por el PUDS para el desarrollo de la Aplicación
- b) Generar los modelos pertinentes utilizando UML

2.1) Flujo de Trabajo: Captura de Requisitos

2.2) Flujo de Trabajo: Análisis

2.3) Flujo de Trabajo: Diseño

2.4) Flujo de Trabajo: Implementación

2.5) Flujo de Trabajo: Pruebas

Manual de Usuario

CONCLUSION

RECOMENDACIÓN

BIBLIOGRAFIA