**INFORME 3.1: CREAR UNA APLICACION MOVIL PRUEBAS DE UNA APP**

**Indicaciones:** Mejorar la aplicación de los menús contextuales (banca móvil).

1. **Front-End (Móvil - Flutter), funciones a implementar:**

* **Pantalla de Inicio**
* Mostrar información del usuario (nombre, correo, saldo disponible).
* Accesos directos a pagos, historial y tarjetas.
* **Manejo de Tarjetas**
* **Lista de tarjetas** guardadas con opción para agregar/eliminar.
* **Congelar tarjeta** en caso de pérdida.
* **Pagos y Transferencias**
* Formulario para ingresar monto y número de tarjeta.
* Confirmación de pago con una alerta modal.
* **Historial de Transacciones**
* Lista con pagos recientes, filtrado por fecha y tipo.
* Posibilidad de descargar un **resumen en PDF**.
* **Autenticación**
* **Inicio de sesión con correo/contraseña** (Firebase Auth es una opción sencilla). Ó (NO USAR)**Autenticación con Google/Apple** para mejorar la experiencia del usuario.
* **Notificaciones y Alertas**
* Mostrar alertas en pantalla cuando se complete un pago o haya una transacción sospechosa.

1. **Base de datos Modelo Entidad-Relación**

El modelo incluirá las siguientes **entidades principales**:

1. **Usuarios (users)**
2. **Tarjetas (cards)**
3. **Pagos (payments)**
4. **Transacciones (transactions)**
5. **Notificaciones (notifications)**

* **Relaciones**

1. **users →** (Un usuario puede tener varias tarjetas).
2. **users → payments** (Un usuario puede hacer múltiples pagos).
3. **payments → transactions** (Cada pago genera una transacción).
4. **users → notifications** (Un usuario puede recibir múltiples notificaciones).

# ****Funcionalidades Cubiertas****

* **Usuarios pueden registrarse e iniciar sesión (con email y password).**
* **Usuarios pueden agregar múltiples tarjetas (con opción de congelarlas).**
* **Pueden realizar pagos y estos quedan registrados.**
* **Cada pago genera una transacción con detalles.**
* **Los usuarios reciben notificaciones de pagos y alertas.**

1. **Desarrollo del Backend**

El backend soportará una aplicación móvil desarrollada en **Flutter**, con funcionalidades relacionadas con **usuarios, tarjetas, pagos, historial de transacciones y notificaciones**.

**Tecnologías a Utilizar**

* Lenguajes: Java (Spring Boot) y Python (Flask).
* Base de datos: PostgreSQL (Datos estructurados) y MongoDB (Logs y auditoría).

### ****Pruebas Unitarias (JUnit para Java y Pytest para Python)****

**Spring Boot (Usuarios y Tarjetas)**

* + Probar **registro de usuario** con credenciales válidas.
  + Probar **inicio de sesión** con credenciales correctas e incorrectas.
  + Probar **agregar tarjeta** con datos válidos y inválidos.
  + Probar **congelar una tarjeta** y verificar el estado is\_frozen.

**Flask (Pagos y Transacciones)**

* + Probar **procesamiento de pagos** con montos válidos e inválidos.
  + Probar **generación de historial de transacciones** en formato JSON.
  + Probar **descarga de PDF** del historial.