UNIVERSIDAD DON BOSCO
ESCUELA DE INGENIERÍA EN COMPUTACIÓN
DESARROLLO DE SOFTWARE PARA MÓVIL

Segundo Desafío Práctico 15%

Indicaciones Generales:

- El desafío se deberá de hacer en grupos de 2-3 integrantes según el docente lo indique.
- La solución a los ejercicios se debe realizar en aplicación móvil con **Android Kotlin.**
- El desarrollo del desafío se debe compartir en aula digital en **un enlace**.
- Si hay soluciones similares, y además se detecta código de internet, automáticamente la nota signada será **"0"**.

Criterio de evaluación:

Criterio	Ponderación
Puntualidad – Entrega	5%
Creación de interfaz gráfica.	10%
Uso de valores de android (strings,dimens,colors)	5%
Todos los campos estan debidamente validados	20%
Desarrollo de ejercicio al 100% sin errores (Conexión a la base, Authenticación, Uso de MVC y RecyclerView)	60%
Total:	100%

Ejercicio:

Desarrolla una aplicación en Android Studio que permita simular un sistema de ventas.

La app debe integrar **Firebase Authentication** y **Firebase Realtime Database**, mostrando la información con **RecyclerView** y representando los datos con **data class**.

La aplicación contará con una **Activity de Login/Registro**, donde el usuario podrá autenticarse con correo y contraseña usando Firebase; una vez dentro, accederá a la **Activity Principal (Menú)** que servirá como pantalla de navegación hacia las demás secciones. Desde el menú, el usuario podrá ingresar a la **Activity de Productos**, donde gestionará su inventario mediante un RecyclerView y formularios para agregar, editar y eliminar productos; a la **Activity de Clientes**, que permitirá registrar y administrar clientes con sus datos básicos también en un RecyclerView; y finalmente a la **Activity de Ventas**, donde podrá registrar nuevas ventas seleccionando productos y clientes, visualizar el historial de transacciones y consultar el total generado. Cada Activity representa una vista dentro del patrón MVC, conectándose con los controladores que gestionan la lógica de Firebase.

Indicaciones:

La aplicación se llamará **“VentaExpress [Temática]”** y deberá permitir a cada usuario.

- Registrarse e iniciar sesión con **Firebase Authentication** (correo y contraseña, con GitHub, y con Facebook).

  Se usa **solo para los empleados de la tienda**. Esto simula que, los empleados deben registrarse o iniciar sesión para poder acceder al sistema. Esto garantiza que solo personal autorizado maneje productos, clientes y ventas.
- Cada usuario(empleado) verá solo sus propios datos (organizados por UID en la base de datos).
- La base de datos debe contener **3 colecciones principales**:
    - **Productos** → id, nombre, descripción, precio, stock.
    - **Clientes** → id, nombre, correo, teléfono.
    - **Ventas** → id, cliente, lista de productos, total, fecha.

Las temáticas serán asignadas en el foro de grupos para el desafío del aula digital.

El flujo sería el siguiente:

1. El empleado inicia sesión en la app.
2. Accede al **Menú principal**, desde donde puede ir a:
    - **Productos:** gestionar inventario (agregar, editar, eliminar, ver stock).
    - **Clientes:** registrar datos de los compradores.
    - **Ventas:** registrar una nueva venta seleccionando productos y un cliente, y generar el total de la transacción.

Cada empleado verá únicamente la información que él mismo haya ingresado, gracias a que los datos se guardan organizados por el **UID** del usuario autenticado.

Uso de las tres colecciones en Firebase Realtime Database

1. **Productos**
    - Representa el inventario disponible en la tienda.
    - Campos: id, nombre, descripción, precio, stock.
    - Ejemplo: un empleado puede registrar un nuevo producto (Ej. “Laptop Dell”, precio $800, stock 5) y luego actualizar su stock al realizar una venta.
2. **Clientes**
    - Contiene la información de los compradores que adquieren productos en la tienda.
    - Campos: id, nombre, correo, teléfono.
    - Importante: los clientes **no se autentican** en la aplicación, solo son registrados como parte del sistema para asociarlos a ventas.
    - Ejemplo: el empleado agrega un cliente “María López, maria@gmail.com, 7777-8888”.
3. **Ventas**
    - Guarda cada transacción realizada.
    - Campos: id, cliente, lista de productos (con cantidades), total, fecha.
    - Cada venta hace referencia a un cliente y a uno o varios productos.
    - Ejemplo: el empleado selecciona el cliente “María López” y registra que compró 2 laptops y 1 mouse, generando un total de $1,650.

Se deberá subir un video en el Read.me del repositorio, en el video se presentara la estructura MVC y el funcionamiento del RecyclerView, la base de datos y la autenticación y por último la funcionabilidad de la APP.