

**OBLIGATORIO**  
**CARRERA ANALISTA DE SISTEMAS**  
**APLICACIONES MÓVILES CON FLUTTER**  
**NOVIEMBRE 2023**

## GENERALIDADES

- La entrega deberá realizarse el día **martes 26 de diciembre hasta las 22:00 hs.**
- Se recomienda la verificación del Modelo Conceptual con el docente hasta el día **lunes 20 de noviembre.**
- Se deberá realizar la entrega mediante correo electrónico, del software junto con toda la documentación exigida en la sección **Requerimientos de Entrega.**
  - **Asunto:** Entrega Obligatorio FLUTTER 2023
  - **Destinatarios:** [terceroanalista@bios.edu.uy](mailto:terceroanalista@bios.edu.uy) y docente del curso (OBLIGATORIO)
  - **Adjuntos (mediante Google Drive):** documentación y archivo comprimido de la solución
  - **Cuerpo:** nombre y cédula de los integrantes del grupo que realiza la entrega
- De ser necesario un comprobante de entrega, **sólo está habilitada Bedelía** a realizarlo.
- El valor de este obligatorio es de **50 por ciento** del total del curso.
- El obligatorio deberá realizarse en **grupos de hasta 3 personas.**

## IDEA GENERAL

Una inmobiliaria le ha encomendado a su equipo, el desarrollo de una app móvil multiplataforma (Android y iOS) para gestionar las propiedades que sus agentes administran, sus clientes (que ofrecen y/o buscan propiedades), y una agenda de visitas que los clientes interesados pueden solicitarle al agente para ver las propiedades personalmente.

## DESCRIPCIÓN DEL PROBLEMA

Observación: Los campos marcados en **negrita** son obligatorios, mientras que los demás son opcionales.

La aplicación debe permitir registrar los clientes con los que el agente trabaja. De éstos se debe registrar un **id** autogenerado, su **nombre completo**, y un **teléfono de contacto**. Será posible agregarlos, modificarlos y eliminarlos (sólo si no tienen propiedades vinculadas o visitas agendadas). Los clientes se podrán listar, y buscar por su nombre.

También se podrán agregar, modificar y eliminar las propiedades manejadas por el agente (si la propiedad tiene visitas agendadas, éstas también deberán eliminarse al eliminar la propiedad, mostrando previamente una advertencia).

Cada propiedad tiene **número de padrón** que la identifica, **tipo** (casa, apartamento, terreno, etc.), **zona**, **dirección**, superficie, **tipo de oferta** (alquiler / venta o ambas), precio en dólares, el **cliente** que la ofrece. Las propiedades se podrán listar y buscar por zona y/o tipo de oferta.

Los tipos de propiedad (**id** autogenerado, y **nombre**) también deben poder agregarse, modificarse y eliminarse (con baja lógica si tienen propiedades vinculadas).

Los clientes pueden pedirle al agente visitar alguna propiedad, por lo que la aplicación deberá disponer de una agenda de visitas. Cada visita se identifica con un **id** autogenerado, y debe indicar el **cliente** interesado, la **propiedad** que desea visitar, una **fecha**, y una **hora**. Las visitas podrán ser modificadas (sólo su fecha y/u hora) o marcadas como canceladas sin perder su información.

La aplicación debe comenzar con una pantalla que muestre las visitas agendadas para la fecha actual (por defecto) o las de cualquier otra fecha (seleccionándola mediante un calendario). Desde esta pantalla inicial se podrá navegar hacia las demás funcionalidades de la aplicación de manera intuitiva a través de un menú de opciones. Desde cualquier pantalla debe ser posible volver a la pantalla inicial con un solo toque.



## REQUERIMIENTOS DE IMPLEMENTACIÓN

- Diseñar e implementar el sistema requerido, que automatice todo el proceso descrito en la sección anterior, utilizando para ello las tecnologías móviles de **Lenguaje Dart** y **Flutter** estudiadas en el curso.
- El obligatorio deberá compilar y ejecutar en las versiones de software y emuladores utilizados en clase.
- La información deberá almacenarse en una base de datos local **SQLite**, que será accedida mediante el package **sqflite**.
- El sistema deberá presentar una interfaz de usuario prolija, bien organizada, intuitiva y fácil de utilizar, minimizando la posibilidad de ingresar datos erróneos.
- Se deberán aplicar todas las validaciones que correspondan, mostrando mensajes amigables tanto en el éxito como en el fracaso de las operaciones.
- La aplicación entregada deberá contener datos de prueba precargados.

## REQUERIMIENTOS DE ENTREGA

Se deberá entregar **de forma digital, vía mail**:

- Archivo comprimido en formato **.zip** conteniendo la carpeta del proyecto con el código fuente y todos los recursos utilizados por el mismo (**eliminar la carpeta build** para reducir el tamaño del proyecto).
- Archivo en formato **.pdf** con toda la documentación exigida. **Un sólo archivo** (prolijamente presentado, incluyendo carátula, índice, encabezados y pies de página adecuados con numeración de páginas, etc.) que contenga las siguientes secciones:
  - Modelo Conceptual
  - Modelo Entidad - Relación (MER)
  - Pasaje a Tablas (MR)
  - Diagrama de Clases de toda la solución implementada