


Requisitos de entrega del prototipo en Flutter

- ❖ Se tendrá tiempo hasta el **Miércoles 14 de Diciembre para la entrega.** (Puede entregarse antes)
- ❖ El código deberá ser subido a GitHub, de manera pública o privada y estar correctamente documentado a través del archivo README.md. En caso de publicarse de manera privada deberá ser compartido con el docente a través de la casilla: sganan81@gmail.com
- ❖ NO es necesario que el prototipo realice peticiones http. Puede mostrarse información estática o incluir conjuntos de valores en el código (List, Map, etc.) para luego ser consultados o iterados.
- ❖ El proyecto se puede desarrollar de manera individual o en grupo hasta un máximo de 3 personas.
- ❖ La aplicación debe contar con 4 pantallas.
 1. Screen Home
 2. Screen Lista de registros
 3. Screen Visualización individual de un registro (Formulario: TextFormField, Switch, Checkbox, Radio, Dropdown, etc.)
 4. Screen con Widget reutilizable (Ej: Card Basic, Card Image, Card Swiper, clase privada, etc.)
 - Contemplar el envío de parámetros al constructor requeridos y opcionales.
- ❖ Widgets que se recomiendan contemplar:
 - Row, Column, Container, Textbox, Sizebox, etc.
 - StatelessWidget y StatefulWidget
 - Contemplar alguna pantalla que utilice el setState
 - Scaffold (pueden utilizarse diferentes modelos de navegación. EJ: drawer, bottomNavigationBar, actions con IconButton's en AppBar, FloatingActionButton, etc.)
 - ListView. (En alguna de sus variantes. ListView.builder, ListView.separated o crear una lista personalizada)
 - Imágenes. Pueden utilizarse diferentes widgets para mostrarlas
 - FadeInImage, "BoxDecoration -> DecorationImage", Image.network, etc.



❖ Estructura

- Assets. Incluir desde pubspec.yaml recursos estáticos (Imágenes, gif, fuentes. etc.)
- Routes (Rutas): Dentro del archivo main deben configurarse las rutas para navegar y el initialRoute y también pueden contemplarse en el drawer o cualquier estructura de navegación que utilicen.
- Theme (Temas). Los estilos de la aplicación deben separarse del archivo main y colocarse en un archivo y carpeta separada. (Solo es necesario contemplar un solo modo. Ej.: light)
- Utilizar archivos unificadores de exportación (Barrel files) dentro la carpeta donde almacenen sus pantallas (/screen, /pages,etc)
- Pueden utilizar paquetes externos para cubrir algunas funcionalidades.
- Material a compartir con el docente:
 - Repositorio en GitHub
 - N° de API y Dimensiones con las que fue probado el prototipo a entregar.

