**Distribución de carpetas Flutter**

Para el proyecto Flutter o móvil de la aplicación picao se sugiere realizar la utilización de una distribución de carpetas modular con el fin de que el código se manejable y tenga una alta escalabilidad a futuro, adicional se utilizara GetX como manejador de estados, gestor de dependencias y gestor de rutas.

**Diagrama.**

**Descripción.**

1. app/data/
   1. models: Aquí se definen las clases que representan los modelos de datos que utiliza la aplicación.
   2. providers: Se colocan las clases responsables de hacer peticiones HTTP, manejar conexiones a bases de datos u otras fuentes de datos.
   3. repositories: Aquí se encapsulan las operaciones que involucran el manejo de datos, como la lógica de negocio y la interacción entre modelos y proveedores.
2. app/modules/
   1. bindings: Las clases de bindings enlazan los controladores con las vistas. Se usa para la inyección de dependencias al iniciar una vista.
   2. controllers: Controladores para manejar la lógica de negocio y la gestión del estado.
   3. views: Vistas de la UI que utilizan los controladores para mostrar datos.
   4. widgets: Componentes UI específicos de cada módulo, reutilizables dentro del módulo.
3. app/routes/
   1. app\_pages.dart: Aquí se definen las rutas de la aplicación usando GetX, gestionando las transiciones entre vistas y la navegación.
4. app/global/:
   1. bindings: Binding generales que se utilizan en toda la aplicación.
   2. controllers: Controladores globales (por ejemplo, controladores de sesión o de configuración).
   3. themes: Configuración de temas y estilos globales.
   4. widgets: Componentes UI globales reutilizables en toda la aplicación.