








-  [AE2-ABP1 Gestionar Permisos Agenda de Contactos](#)
 -  [Características](#)
 -  [Tecnologías utilizadas](#)
 -  [Capturas de pantalla](#)
 - [Guía de Ejecución del Proyecto](#)
 - [Instalación y Configuración](#)
 - [Contribuciones \(Things-To-Do\)](#)
 - [Licencia](#)

AE2-ABP1 Gestionar Permisos Agenda de Contactos

Este mini proyecto, permite demostrar el uso de permisos sensibles mediante una aplicación de agenda de contactos en Android Studio, desarrollado en Kotlin, utilizando Fragments y RecyclerView, solicitando permiso para leer los contactos, y accediendo directamente al objeto "Contacts" para agregar nuevos usuarios sin afectar las directivas de protección de datos sensibles. La interfaz muestra, inicialmente, la lista de contactos (siempre que existan datos previos, en su defecto muestra un mensaje de lista vacía), usa un FAB ("+" flotante) para agregar un nuevo contacto mostrando un formulario de entrada, al completar la operación, un listener, actualiza la pantalla y muestra la nueva lista actualizada.

Características

-  **Lista de Contactos** : Se muestra en un **RecyclerView** dentro de la **MainActivity**.
-  **Añadir Contactos** : Botón flotante para agregar un nuevo contacto.
-  **Formulario Nuevo Contacto** : Muestra un formulario con los campos nombre, apellido y teléfono de tipo texto.

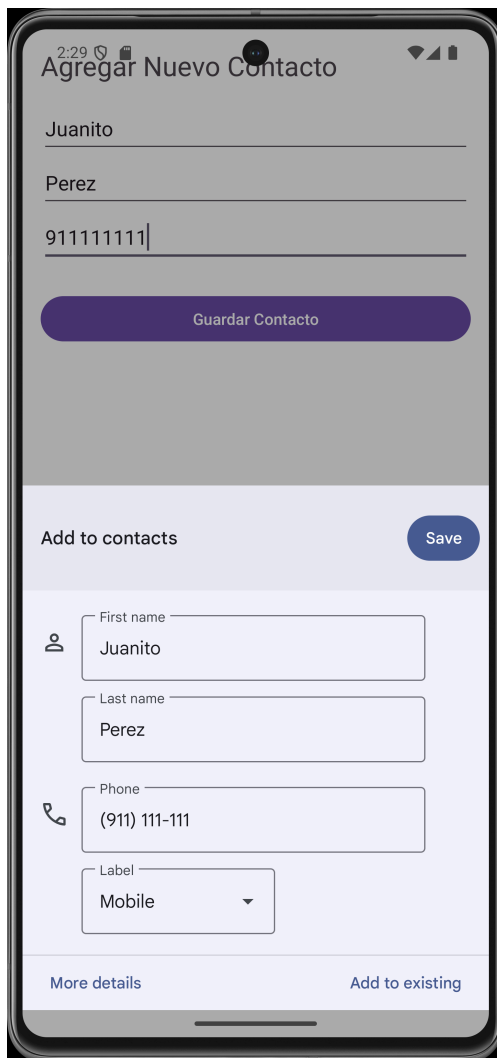
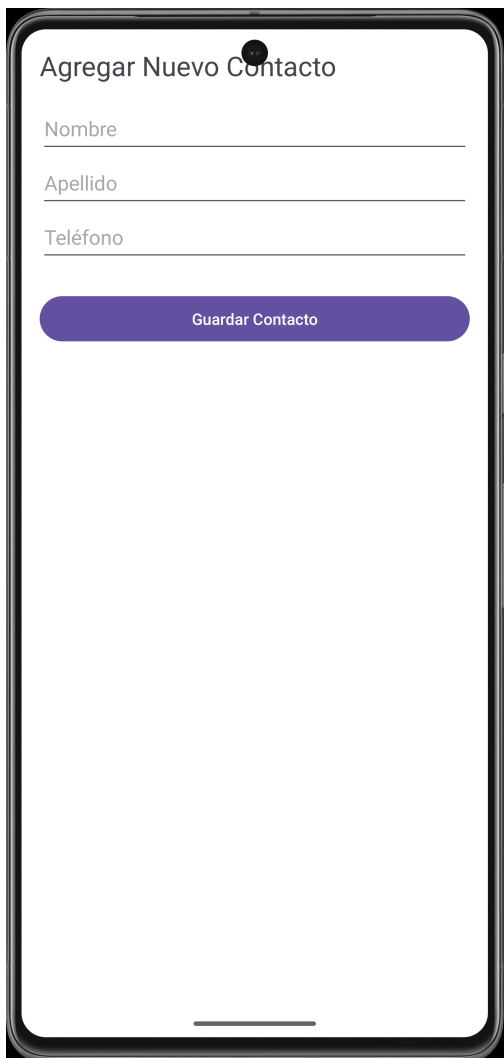
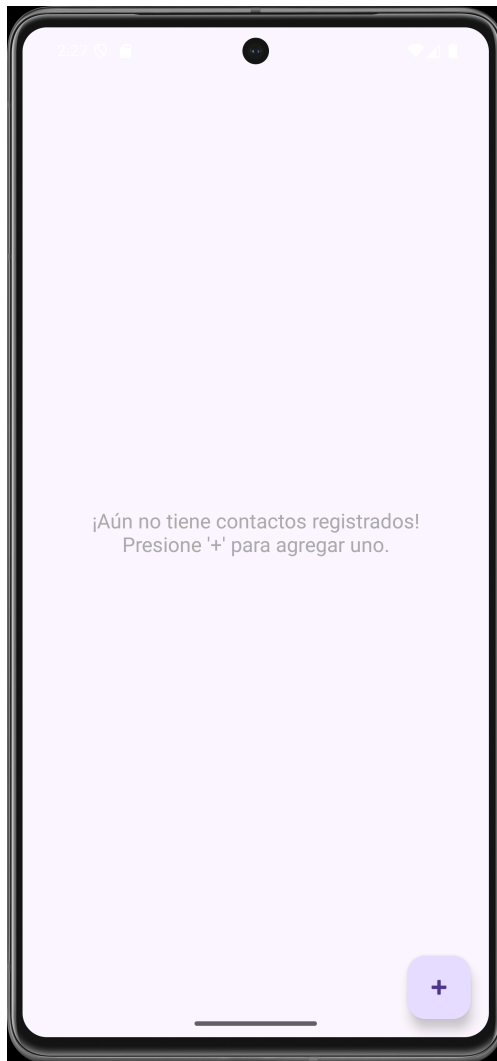
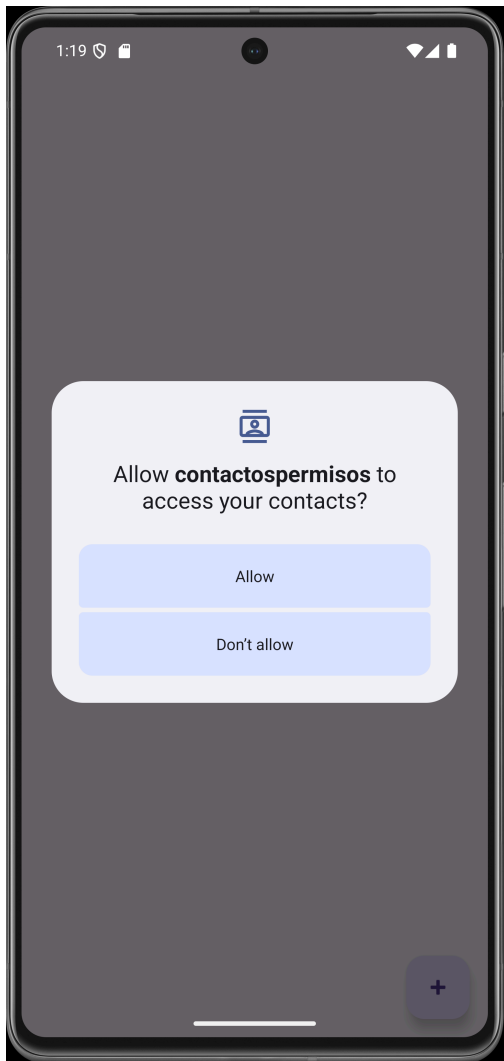
Tecnologías utilizadas

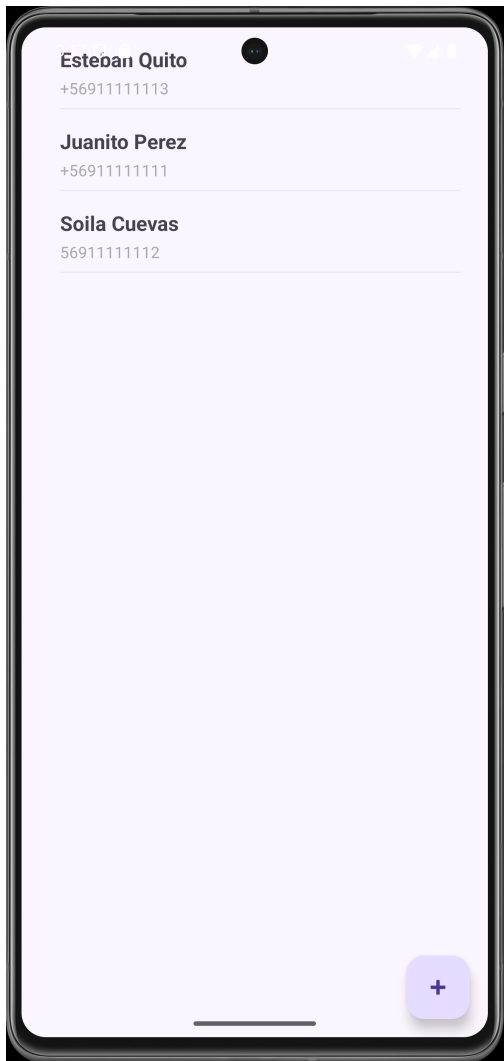
- **Kotlin** : Lenguaje de desarrollo.
- **RecyclerView**: Para mostrar la lista de contactos.

- **Fragments** : Para el formulario de nuevo contacto.



Capturas de pantalla





Guía de Ejecución del Proyecto

Para ejecutar este proyecto en tu entorno de desarrollo, siga estos 'quick steps':

1. **Clonar el Repo:** Clone el proyecto en su máquina local.
2. **Abrir en Android Studio:** Abra la carpeta del proyecto con Android Studio. El IDE detectará automáticamente la configuración de Gradle.
3. **Sincronizar Gradle:** Haz clic en el botón "Sync Now" si Android Studio te lo solicita. Esto descargará todas las dependencias necesarias.
4. **Ejecutar:** Conecta un dispositivo Android físico o inicia un emulador. Luego, haz clic en el botón "Run 'app'" (el ícono de la flecha verde) para desplegar la aplicación.

Para ejecutar este proyecto en tu celular, sigue estos 'quick steps':

1. **Copiar la APK:** Compila y copia la aplicación (APK) en tu celular.
2. **Instalar:** Instala la aplicación, salta los avisos de advertencia, es normal si la aplicación no ha sido productivizada la plataforma de Android.

3. **Abrir la App:** Haz doble clic en el ícono "Agenda".
4. **Recorrer las opciones:** Cliquea en las opciones y podrás acceder al listado de eventos, editar cada evento, crear nuevos eventos, regresando a cualquier punto de la app.

Instalación y Configuración

a. Clonar el repositorio:

```
https://github.com/jcordovaj/AE2_ABP1_ContactosPermisos.git
```

b. Abrir el Proyecto en Android Studio:

b.1. Abrir Android Studio.

b.2. En la pantalla de bienvenida, seleccionar **"Open an existing Android Studio project"** (Abrir un proyecto de Android Studio existente).

b.3. Navegar a la carpeta donde se clonó el repositorio y seleccionarla. Android Studio detectará automáticamente el proyecto de Gradle y comenzará a indexar los archivos.

c. Sincronizar Gradle:

c.1. Este es el paso más importante. Después de abrir el proyecto, Android Studio intentará sincronizar la configuración de Gradle. Esto significa que descargará todas las librerías, dependencias y plugins necesarios para construir la aplicación.

Normalmente, una barra de progreso se mostrará en la parte inferior de la consola de Android Studio con un mensaje como **"Gradle Sync in progress"**.

c.2. Si no se inicia, o si el proceso falla, intente con el botón **"Sync Project with Gradle Files"** en la barra de herramientas. Es el icono con el **"elefante" de Gradle**. Eso forzará la sincronización.

c.3. Esperar que el proceso de sincronización termine. De haber errores, puede ser por problemas en la configuración de Android u otros conflictos, la aplicación debe descargar lo que requiera y poder ser ejecutada **"AS-IS"**.

d. Configurar el Dispositivo o Emulador:

Para ejecutar la aplicación, se requiere un dispositivo Android, puedes usarse el emulador virtual o un dispositivo físico.

d.1. Emulador: En la barra de herramientas, haga click en el botón del "AVD Manager" (Android Virtual Device Manager), que es el icono de un teléfono móvil con el logo de Android. Desde ahí, puedes crear un nuevo emulador con la versión de Android que prefiera (Nota: Debe considerar que cada celular emulado, puede requerir más de 1GB de espacio en disco y recursos de memoria).

d.2. Dispositivo físico: Conecte su teléfono Android a la computadora con un cable USB (también puede ser por WI-FI). Asegúrese de que las **Opciones de desarrollador y la Depuración por USB** estén habilitadas en su dispositivo. Consulte a su fabricante para activar estas opciones.

e. **Ejecutar la aplicación:**

e.1. Seleccione el dispositivo o emulador deseado en la barra de herramientas del emulador.

e.2. Haga click en el botón "Run 'app'" (el triángulo verde en la parte superior, o vaya al menu "RUN") para iniciar la compilación y el despliegue de la aplicación, puede tardar algunos minutos, dependiendo de su computador.

e.3. Si todo ha sido configurado correctamente, la aplicación se instalará en el dispositivo y se iniciará automáticamente, mostrando la pantalla de inicio.

Contribuciones (Things-To-Do)

Se puede contribuir reportando problemas o con nuevas ideas, por favor respetar el estilo de programación y no subir código basura. Puede utilizar: forking del repositorio, crear pull requests, etc. Toda contribución es bienvenida.

Licencia

Proyecto con fines educativos, Licencia MIT