



**UNIVERSIDAD MARIANO GÁLVEZ DE GUATEMALA CAMPUS  
JUTIAPA, JUTIAPA**

**FACULTAD DE INGENIERÍA EN SISTEMAS DE INFORMACIÓN  
CURSO: PROGRAMACION II**

**DOCENTE: INGE RULDYN AYALA**

**TRABAJO REALIZADO: PROGRAMA DE LOS HOBBYS**

**REALIZADO POR:**

**JOSTYN MANRRIQUE GODOY CHINCHILLA      0905-23-2991**

**ROBERTSON ALESSANDRO MUÑOZ PAREDES      0905-24-16433**

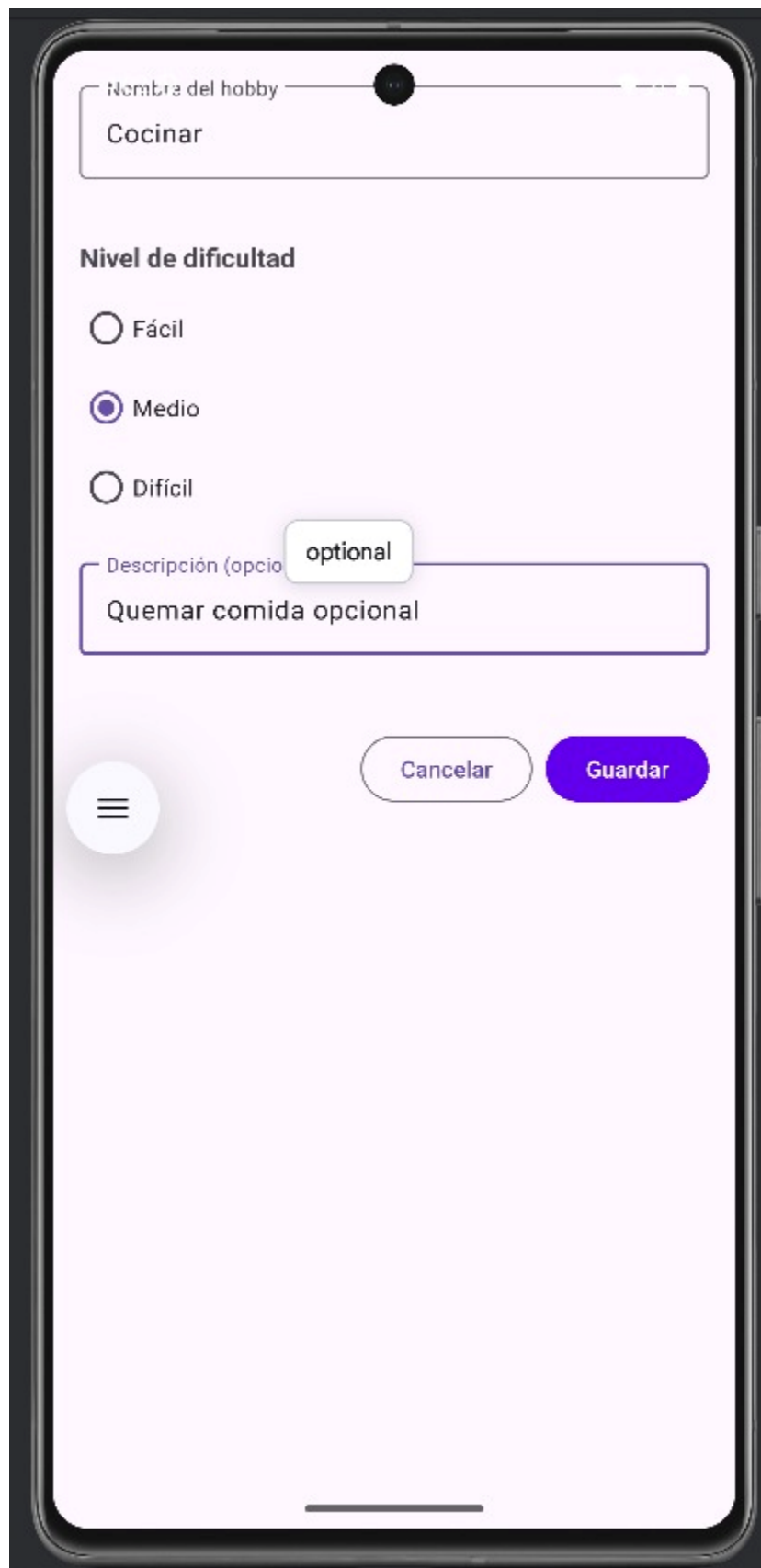
- PANTALLAZOS DE LA INTERFAZ YA EN FUNCIONAMIENTO



- **PANTALLAZO PARA AGREGAR EL HOBBY**

A screenshot of a mobile application interface for adding a hobby. The screen has a light pink background. At the top, there is a text input field with the placeholder text "Nombre del hobby". Below this, the section is titled "Nivel de dificultad" in bold. Under the title, there are three radio button options: "Fácil" (which is selected), "Medio", and "Difícil". Below the radio buttons is another text input field with the placeholder text "Descripción (opcional)". At the bottom right of the screen, there are two buttons: "Cancelar" (a white button with a grey border) and "Guardar" (a solid purple button).

- AGREGANDO UN HOBBY



A mobile application form for adding a hobby. The form is displayed on a smartphone screen with a light purple background. At the top, there is a text input field labeled "Nombre del hobby" with the text "Cocinar" entered. Below this, the section "Nivel de dificultad" (Difficulty Level) contains three radio button options: "Fácil", "Medio" (which is selected), and "Difícil". Underneath the difficulty level, there is a text input field labeled "Descripción (opcional)" with the text "Quemar comida opcional" entered. A small yellow callout box with the word "optional" points to the description field. At the bottom right, there are two buttons: "Cancelar" (Cancel) and "Guardar" (Save). On the bottom left, there is a circular menu icon with three horizontal lines. The entire form is framed by a dark border representing the phone's bezel.

Nombre del hobby

Cocinar

Nivel de dificultad

☐ Fácil

☒ Medio

☐ Difícil

Descripción (opcional)

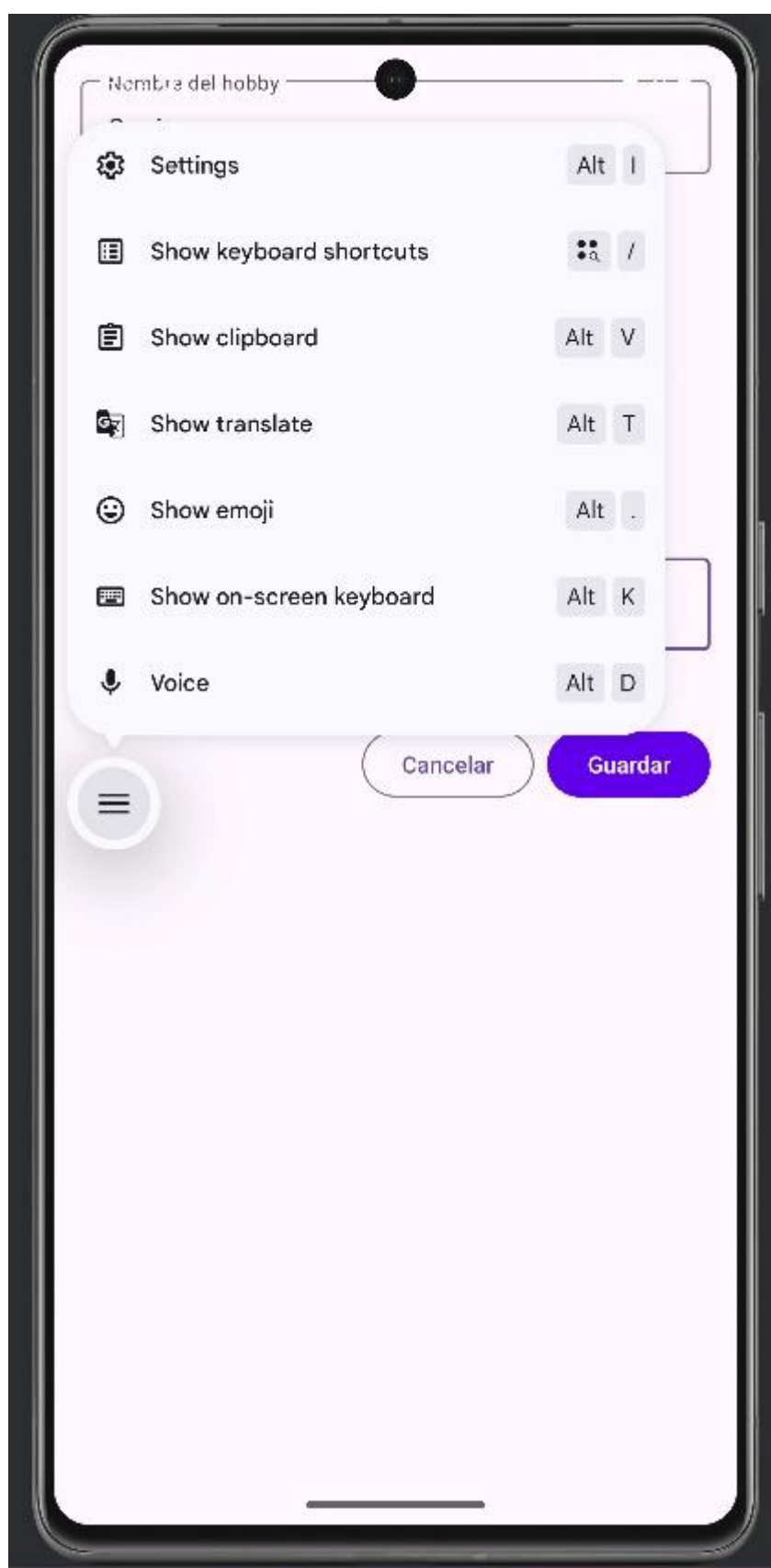
Quemar comida opcional

optional

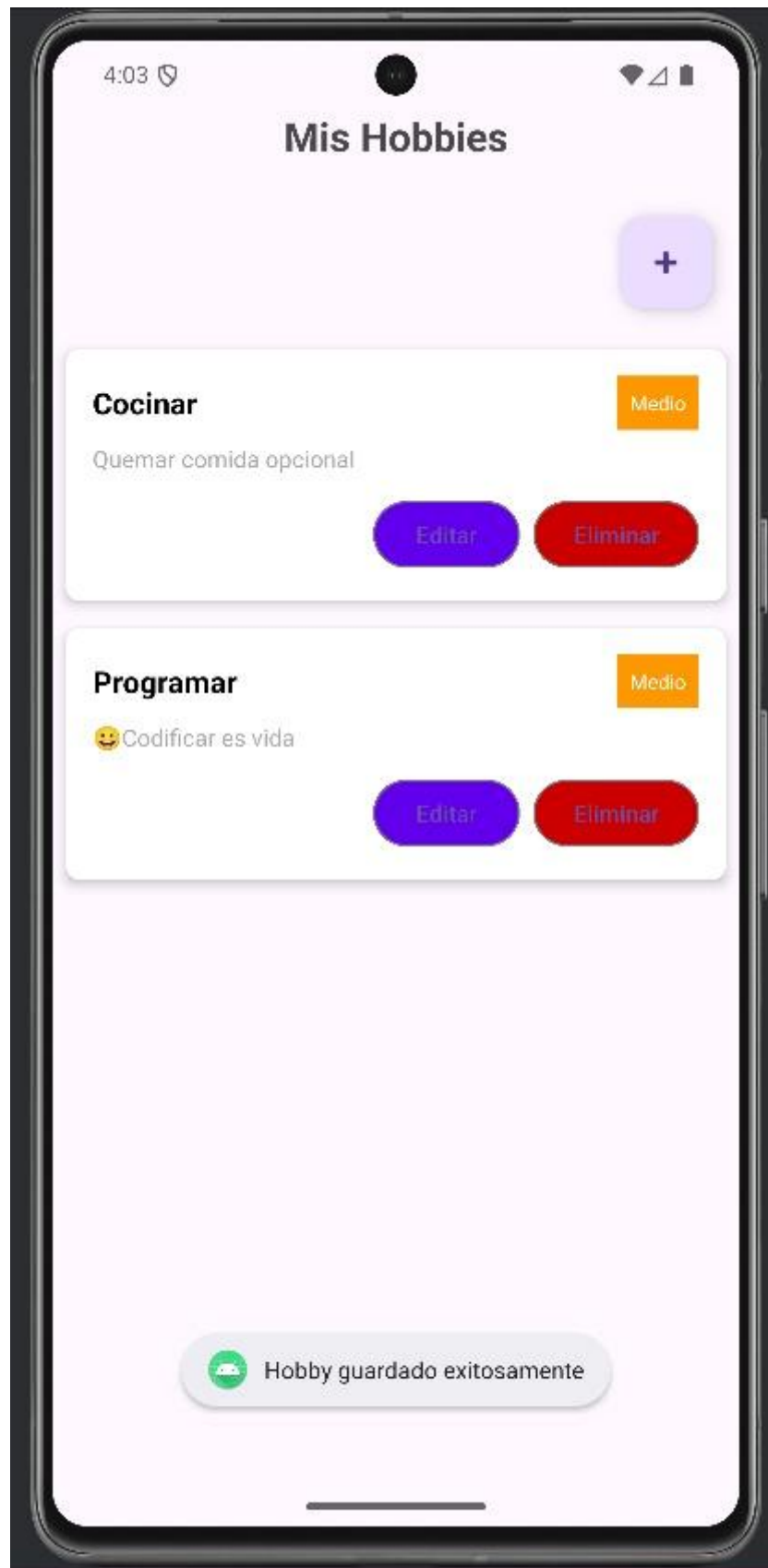
Cancelar

Guardar

- EXTRAS AGREGADOS



- LISTAS DE HOBBYS YA GUARDADOS



- DESCRIPCION DE EXPERIENCIA:

Se tuvieron muchos problemas ya que con el agente no nos funciono de manera correcta, se utilizó una herramienta externa de agenten para crear la app, la cual no nos dio problemas, la dificultad que se encontró fue entrarla para que nos funcionara como lo requeríamos.

El prompt utilizado en formato .MD fue el siguiente obtenido en clase, este prompt solo se adaptó y se mejoró con la herramienta de IA QWEN

# Instrucciones para crear la aplicación Jobicat

---

## ## Descripción del proyecto

Crear una aplicación Android en Java llamada \*Jobicat\* para administrar hobbies personales.

## ## Requisitos funcionales

### ### Funcionalidad principal

- Administrar hobbies con niveles de dificultad
- Niveles de dificultad: \*Fácil, \*\*Medio, \*\*Difícil\*
- Almacenamiento en base de datos SQLite
- Operaciones CRUD completas (Altas, Bajas, Cambios, Consultas)

### ### Características técnicas

- Utilizar el proyecto ya existente
- Validar que no se registren valores en blanco
- Crear todas las actividades, vistas y objetos necesarios

## ## Componentes a desarrollar

### ### 1. Base de datos SQLite

- Crear tabla para almacenar hobbies
- Campos requeridos:
  - ID (autoincremental)
  - Nombre del hobby
  - Nivel de dificultad
  - Descripción (opcional)

### ### 2. Actividades principales

- \*MainActivity\*: Pantalla principal con listado de hobbies
- \*AddHobbyActivity\*: Formulario para agregar nuevos hobbies

- \*EditHobbyActivity\*: Formulario para editar hobbies existentes

### ### 3. Validaciones requeridas

- Verificar que el nombre del hobby no esté vacío
- Validar selección de nivel de dificultad
- Mostrar mensajes de error apropiados

### ### 4. Interfaz de usuario

- ListView o RecyclerView para mostrar la lista de hobbies
- Formularios con campos de texto y selección de dificultad
- Botones para acciones CRUD
- Diseño intuitivo y responsive

## ## Consideraciones importantes

- Antes de iniciar cualquier codificación, consultar cualquier duda sobre los requisitos
- Mantener la estructura del proyecto existente
- Seguir buenas prácticas de programación en Android
- Implementar manejo adecuado de la base de datos

## ## Acciones pendientes

1. Confirmar estructura de base de datos
  2. Definir diseño de interfaces
  3. Implementar funcionalidades CRUD
  4. Agregar validaciones
  5. Probar la aplicación completa
- 

- **APRENDIZAJE CON LA APP DESARROLLADA**

R// se tiene que armar un buen prompt, ya que debemos tener la lógica para obtener el resultado deseable, ya que si no sabemos que pedir no lograremos obtener ningún resultado ni lograremos entender lo que nos genere la IA lo cual no nos servirá de nada si no logramos entender bien