Carlos Guasp López

APP UGR

E.T.S. de Ingenierías Informática y de Telecomunicación

-

Colegio San Rafael, Granada

ÍNDICE

Pantalla principal - Boceto

En esta primera pantalla principal se muestran dos botones. Uno correspondiente a la pantalla de “Administrador” y otro correspondiente al login de “Alumno”.

Funcionalidades

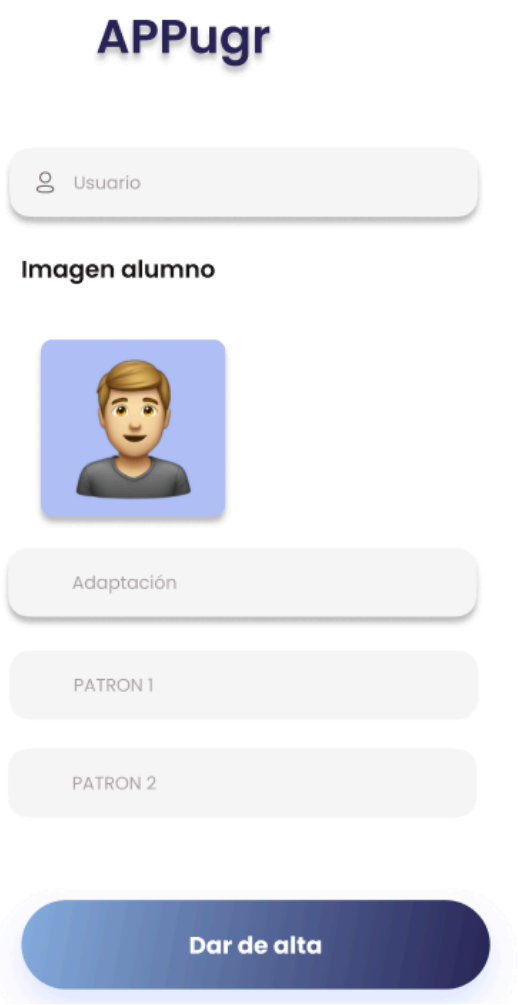
1. Administrador

Historia de usuario: "Como administrador, quiero dar de alta a usuarios con perfiles diferentes para gestionar mejor los permisos y accesos."

Diseño

- **Bocetos**

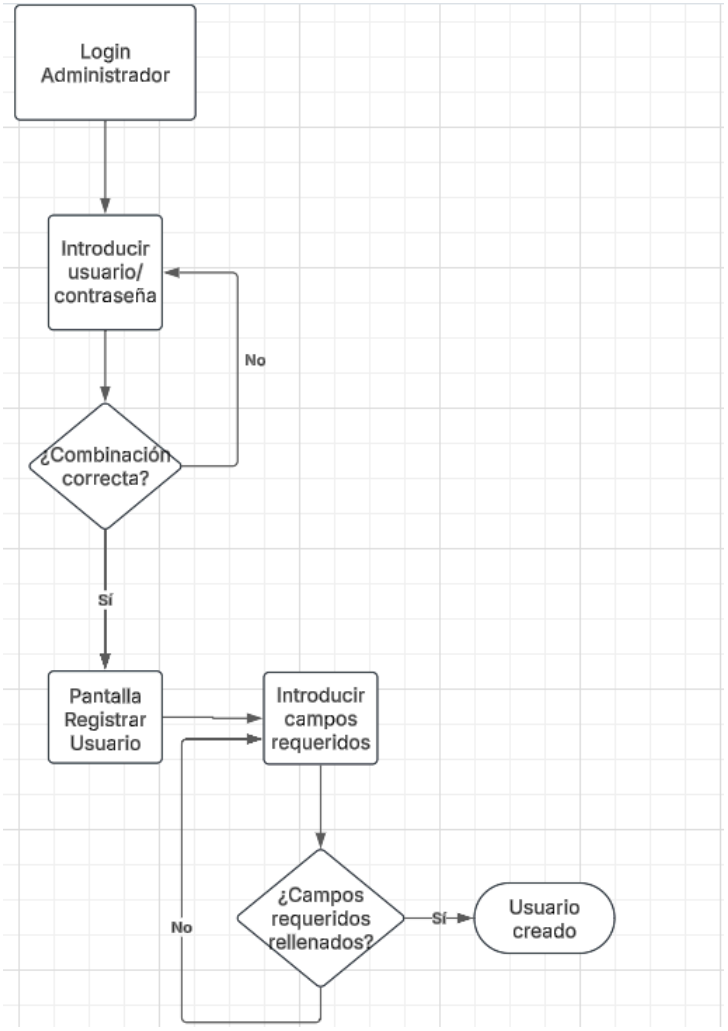
  Creación de usuarios



En esta pantalla se pueden observar distintos campos para rellenar a un usuario. En primer lugar, deberemos insertar un “nombre”. A continuación, deberemos seleccionar una imagen del alumno, el tipo de adaptación que tendrá y el patrón 1 y 2, es decir, su contraseña.

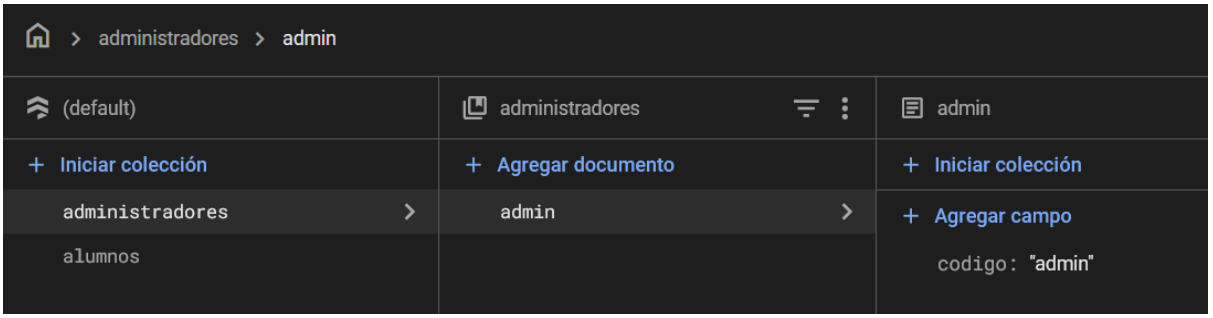
Al presionar el botón “Dar de alta”, el perfil, quedará registrado en la base de datos y el usuario podrá iniciar sesión desde el apartado “Alumno”.

Diagrama de flujo



Base de datos

El administrador, dentro de la colección “Administrador”, únicamente tendrá un código(contraseña).



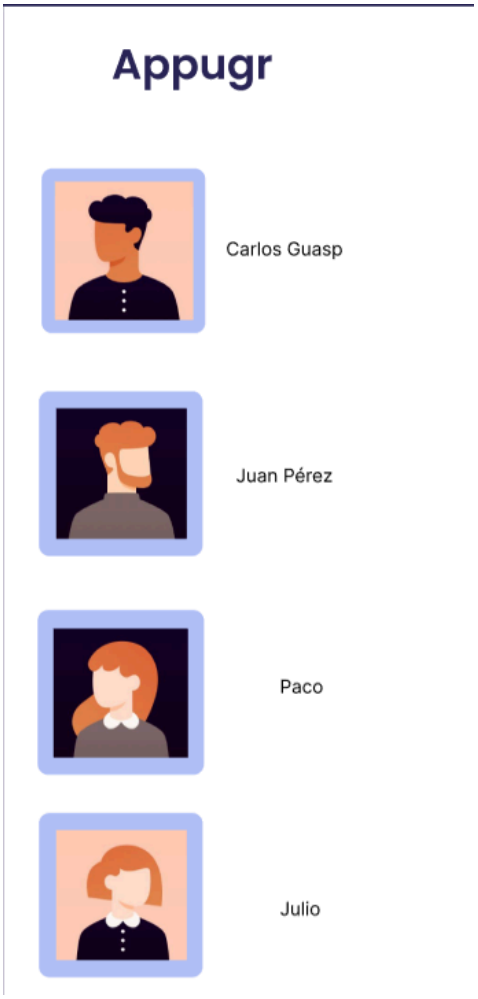
2. Alumnos

Historia de usuario: "Como estudiante, quiero un sistema de login accesible para acceder rápidamente y fácilmente a mis recursos personalizados."

Diseño

- **Bocetos**

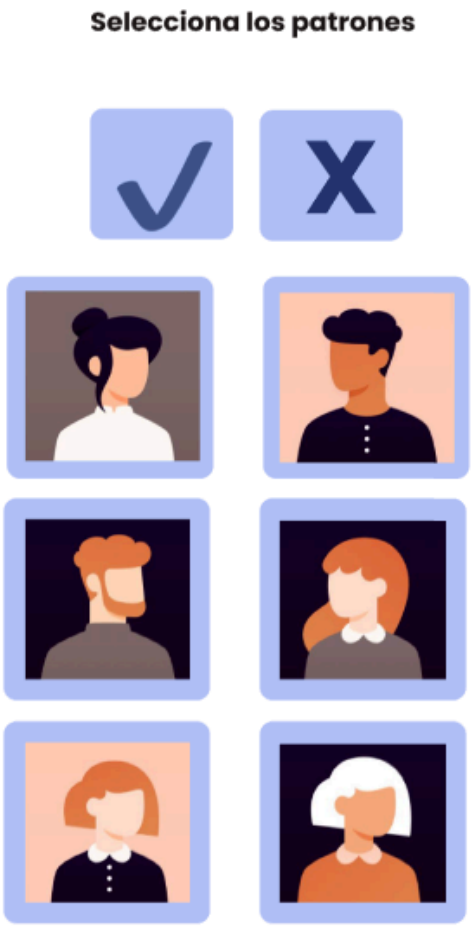
Selección de usuario



En esta primera pantalla se pueden observar los distintos usuarios que hay en el sistema (base de datos) con su respectiva imagen. Se ha adaptado con texto e imagen para que sea accesible de manera general. Las funcionalidades de audio estarían disponibles mediante software externo en el dispositivo.

Al seleccionar un perfil, nos dirigirá a otra pantalla, en la cuál se introducirá un patrón.

Selección del patrón (contraseña) correspondiente



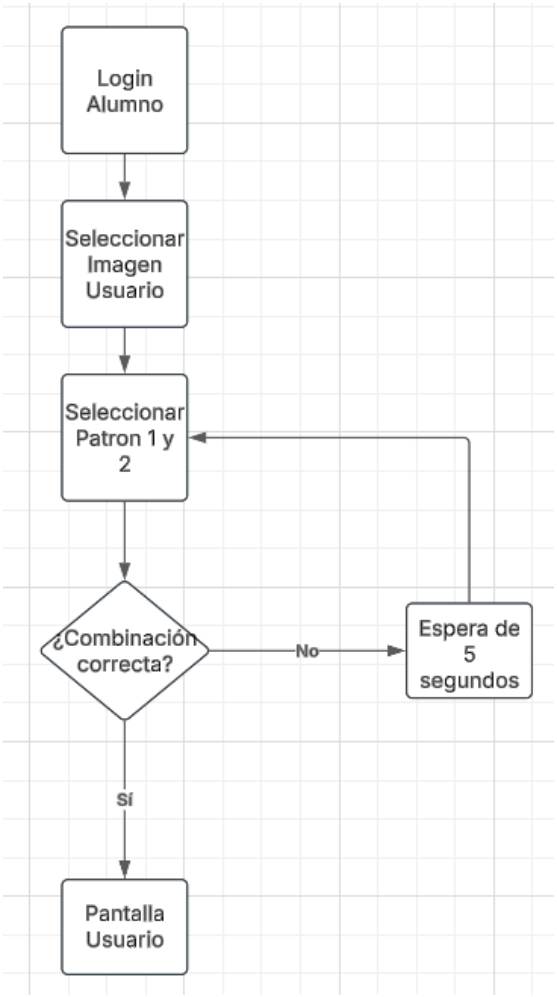
En esta segunda pantalla del login de usuario, deberemos introducir un patrón de 2 imágenes.

Si la primera imagen se corresponde con la primera imagen del patrón, nos aparecerá un “✔” en la primera casilla superior. Si la segunda imagen seleccionada es incorrecta, aparecerá una “X”. Y así en función de la casilla correspondiente.

Si se falla la combinación, deberá esperar unos segundos antes de introducir la siguiente. Si es correcta, avanzará directamente a la siguiente pantalla.

Diagrama de flujo

Proceso de inicio de sesión por parte del usuario



Base de datos

Los alumnos se guardarán dentro de una colección llamada “Alumnos” en las que el nombre del alumno será el documento. Además, tendrá una serie de atributos: accesibilidad, imagenURI(nos servirá para encontrar la imagen dentro del dispositivo), patron1, patron2 y el nombre nuevamente(para usarlo de otra manera).

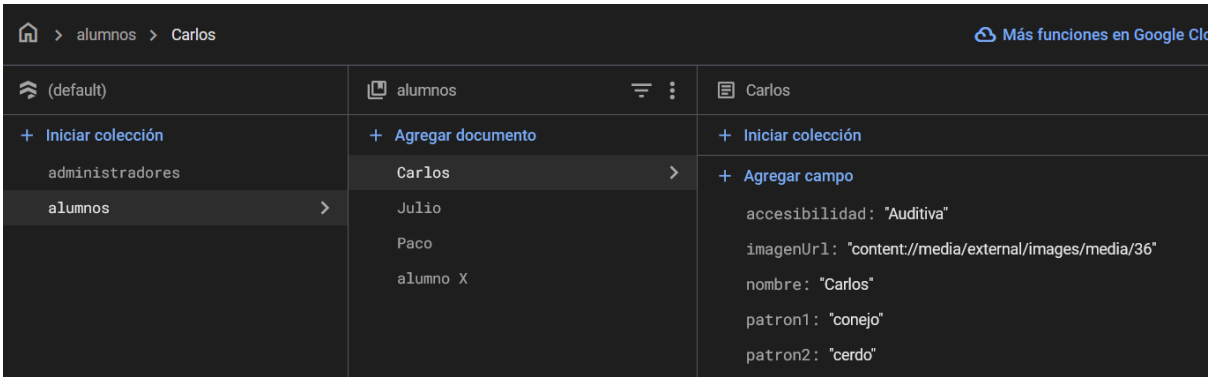
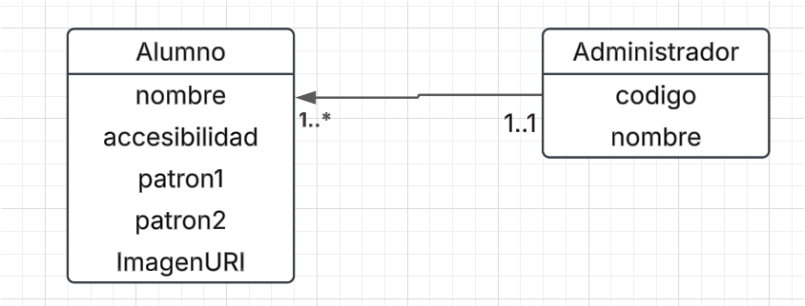


Diagrama de clases (alumno-administrador)/ Diagrama E-R



Alumno:

- nombre: Nombre del alumno.
- accesibilidad: Tipo de accesibilidad que tiene el alumno.
- patron1 y patron2: Combinación para el login.
- ImagenURI: URI de la imagen asociada al alumno.

Administrador:

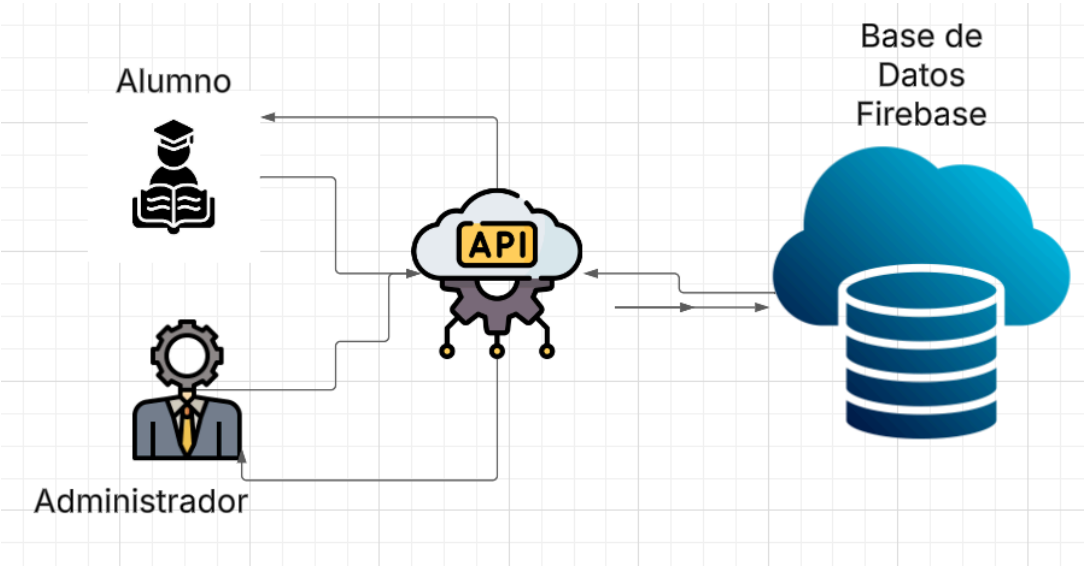
- codigo: Código único del administrador.
- nombre: Nombre del administrador.

Relación entre "Alumno" y "Administrador":

- Un "Administrador" (1..1) puede estar asociado con uno o más "Alumnos" (1..*).

En resumen, este diagrama muestra cómo un administrador puede crear múltiples alumnos y sus atributos.

Arquitectura del sistema



En esta arquitectura, hay dos tipos de usuarios: **administradores y alumnos**.

El **administrador** solo puede registrar nuevos usuarios en la base de datos. En cambio, el **alumno** inicia sesión utilizando su imagen y dos patrones como credenciales de acceso.

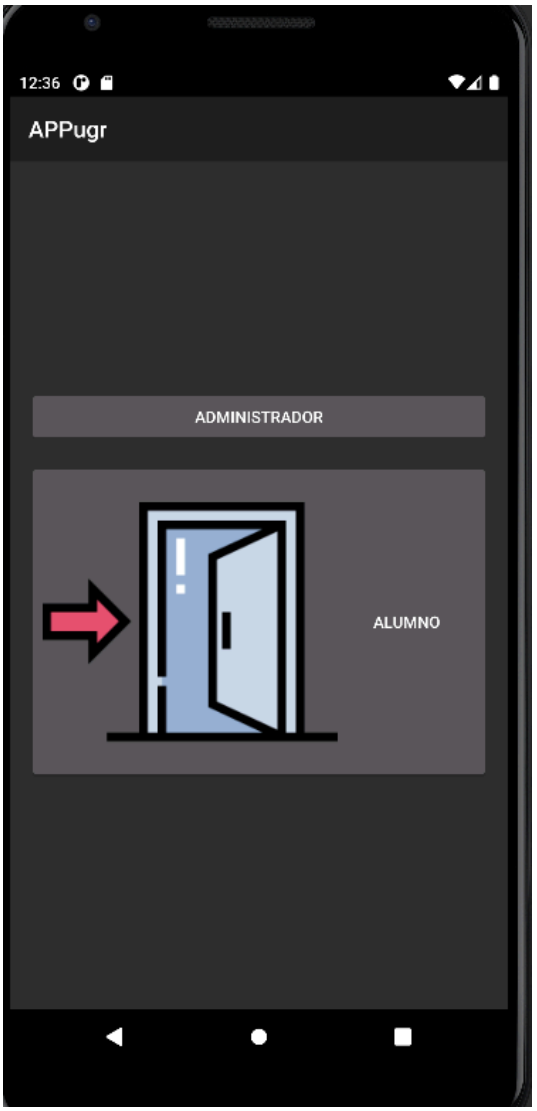
La **API** actúa como intermediaria entre la app y Firebase, procesando las solicitudes, validando credenciales y gestionando los permisos de acceso.

Cuando un administrador crea un usuario o un alumno intenta iniciar sesión, la app envía la solicitud a la API. Esta verifica los datos y se comunica con **Firebase**, donde se almacenan los usuarios y sus credenciales. Una vez confirmada la autenticación, la API responde a la app con el resultado de la operación.

De esta manera se controla el acceso a los datos de forma segura y en tiempo real desde cualquier dispositivo con acceso a internet.

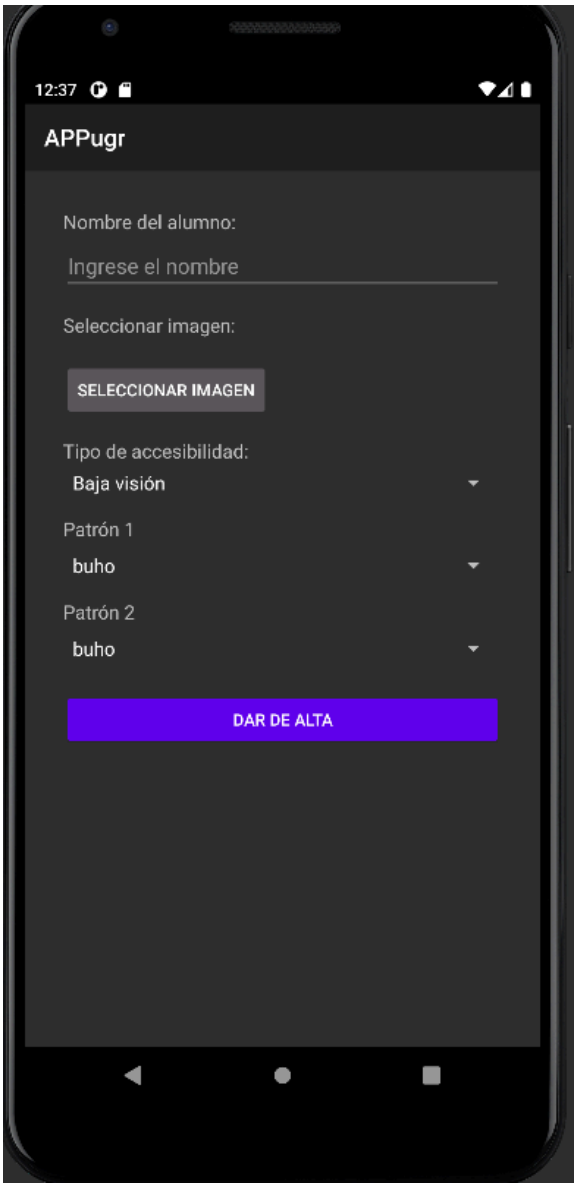
Pantallas finales

Pantalla principal

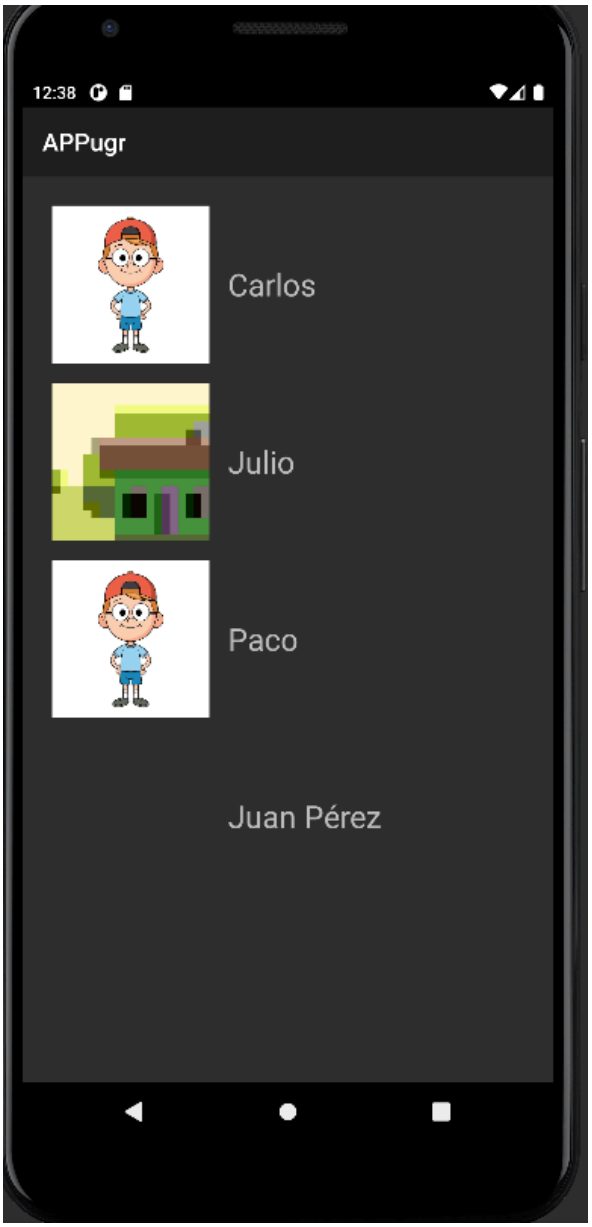


Pantalla inicio de sesión administrador

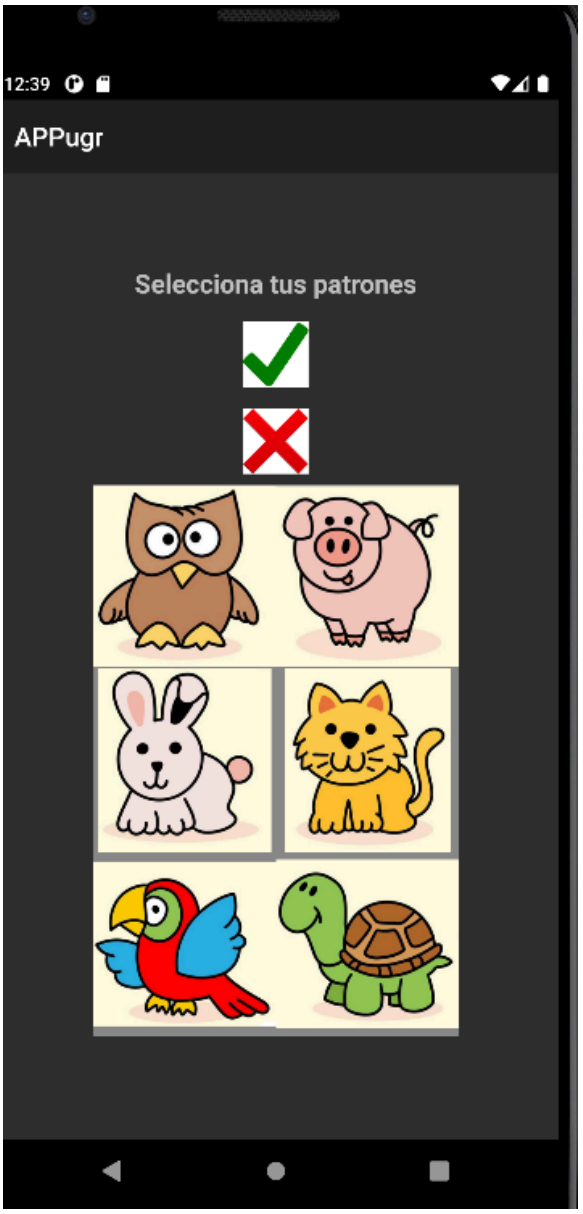
Pantalla creación alumno



Pantalla selección (alumnos)



Pantalla selección Patrones

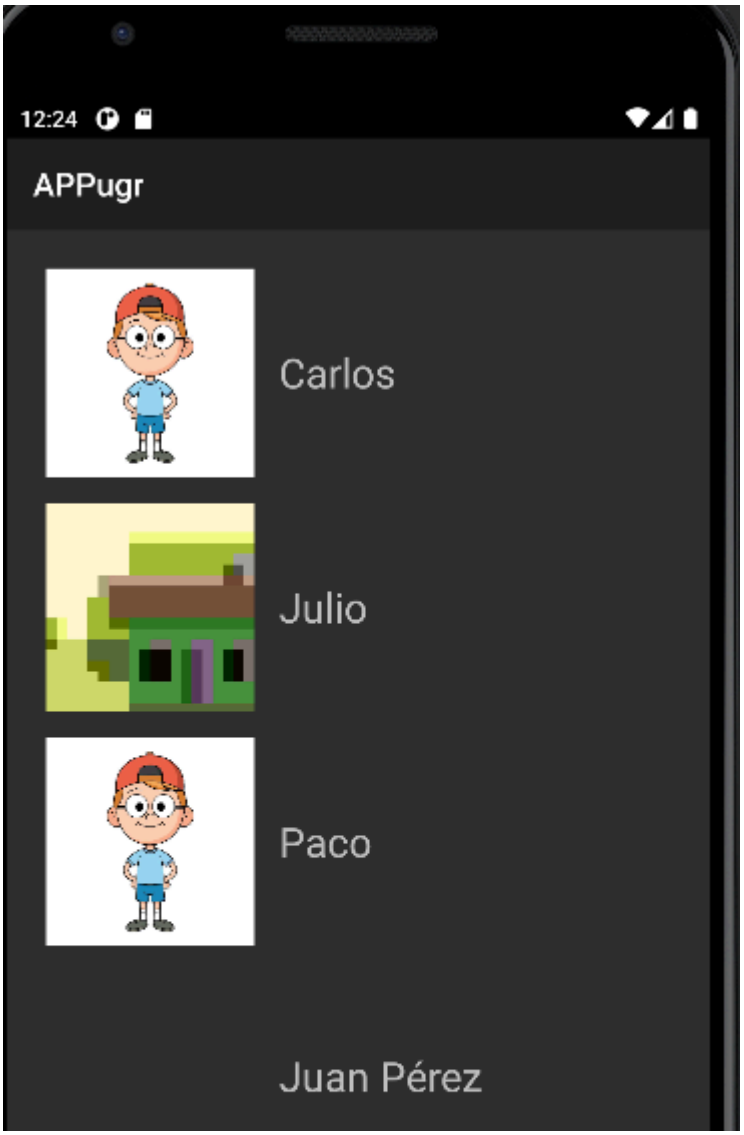


Pruebas realizadas (administrador y alumno)

Administrador

Prueba de registro de usuario: Se verificó que el administrador puede dar de alta a nuevos usuarios asignando correctamente sus características (nombre, patrones, tipo de accesibilidad, etc). Estos deben aparecer en la sección usuarios.

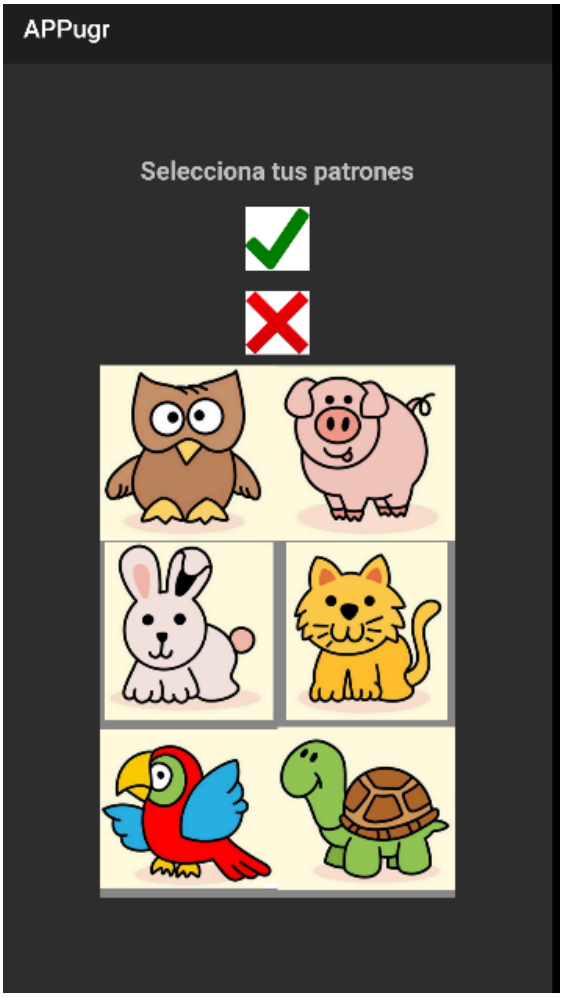
Se observa como aparecen los usuarios creados desde el apartado administrador.



Alumno

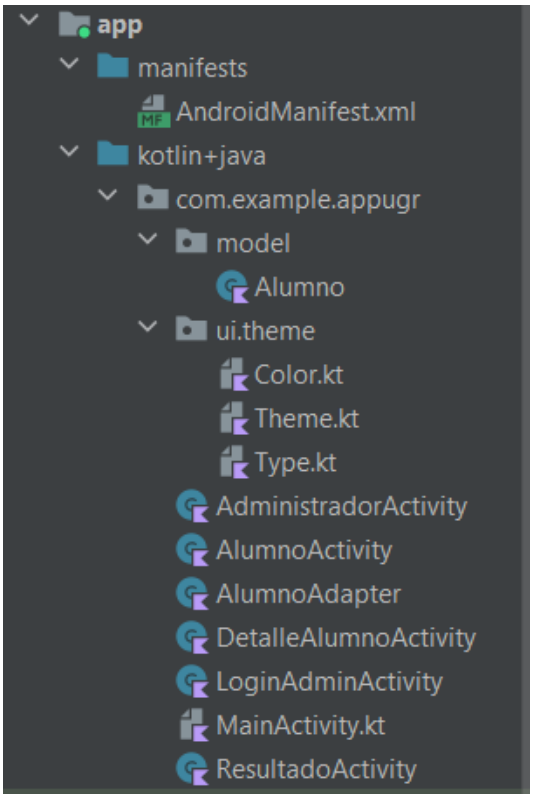
Prueba de login accesible: Se probó que el sistema de login sea fácilmente accesible, tanto en términos de diseño como de usabilidad, incluyendo la validación de que los estudiantes puedan iniciar sesión correctamente con sus credenciales (patrones).

Imágenes llamativas y fáciles de recordar. Tanto si acierta como si falla aparece en la parte superior.



Estructuras de ficheros

Estructuración backend



(Model) Alumno: Representa a un alumno con sus atributos, como nombre e identificadores.

AlumnoActivity: Muestra la interfaz donde los alumnos interactúan con la app.

AdministradorActivity: Interfaz destinada a la gestión por parte del administrador.

AlumnoAdapter: Adaptador que organiza y muestra los datos de los alumnos en listas.

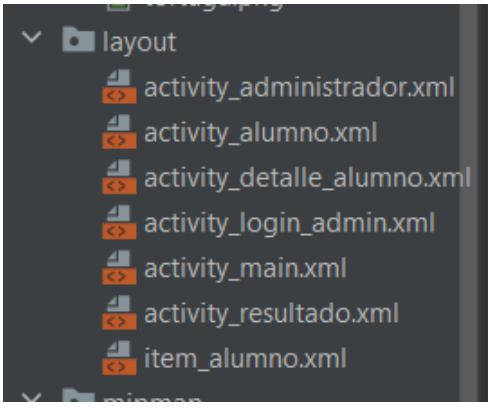
DetalleAlumnoActivity: Maneja la validación de los patrones de acceso del alumno.

LoginAdminActivity: Pantalla donde los administradores ingresan usuario y contraseña.

MainActivity: Pantalla inicial con opciones de acceso.

ResultadoActivity: Muestra el resultado de la validación del patrón sin lógica adicional.

Estructuración frontend (Diseño de interfaz)



activity_administrador.xml: Define la apariencia de la pantalla del administrador.

activity_alumno.xml: Diseño de la interfaz para los alumnos.

activity_detalle_alumno.xml: Diseño para la validación de patrones.

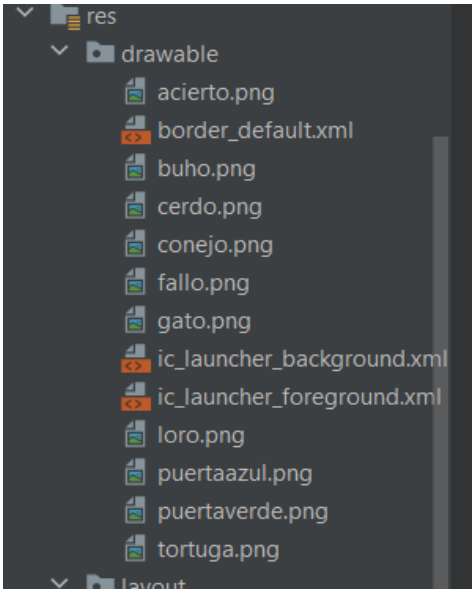
activity_login_admin.xml: Diseño de la pantalla de login del administrador.

activity_main.xml: Define la estructura visual de la pantalla principal.

activity_resultado.xml: Muestra visualmente si el patrón ingresado es correcto.

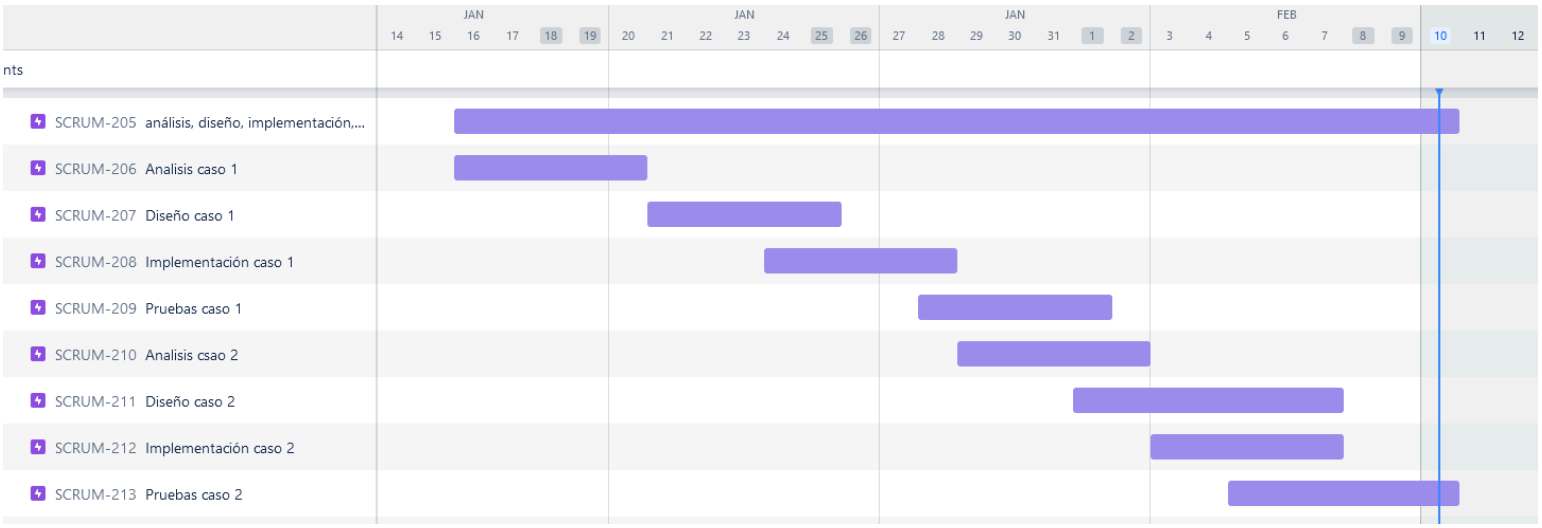
item_alumno.xml: Define cómo se ven los elementos en una lista de alumnos.

res/drawable (archivos utilizados en ejecución)



Planificación temporal

Caso 1: Historia de usuario Administrador
Caso 2: Historia de usuario Alumno



Las partes trabajadas en cada uno de los casos son las siguientes:

Análisis

Objetivo: Definir requisitos y funcionalidades clave.
Administrador: Identificación de necesidades para crear nuevos usuarios y gestionar accesos.
Alumno: Establecer el método de autenticación mediante imagen y patrón.
Tareas: Investigación de Firebase, análisis de seguridad, diseño de flujo de autenticación.

Diseño

Objetivo: Estructurar la interfaz y la arquitectura del sistema.
Administrador: Boceto de la pantalla de registro y almacenamiento de usuarios en Firebase.
Alumno: Diseño de la pantalla de login con autenticación por imagen y patrón.
Tareas: Creación de prototipos, diagramas de clases, esquema de la base de datos.

Implementación

Objetivo: Desarrollar la funcionalidad y conectar con Firebase.
Administrador: Programación de la funcionalidad para registrar nuevos usuarios.
Alumno: Implementación del sistema de login con imagen y patrón.
Tareas: Configuración de Firebase, desarrollo de la API, integración con la app.

Pruebas

Objetivo: Verificar que todo funciona correctamente.
Administrador: Pruebas de creación de usuarios y restricciones de acceso.
Alumno: Validación del login con imágenes y patrones correctos/incorrectos.
Tareas: Tests unitarios, pruebas de integración con Firebase, detección y corrección de errores.

Dedicación horaria al proyecto

https://docs.google.com/spreadsheets/d/1Lh5ViNGB8MtljvPky3pWNx5E551NzqoyGTBKPwrgUws/edit?usp=sharing

Inicio	Fin	Tarea realizada	Horas dedicadas
16/01/24	17/01/24	Análisis de requisitos	3
18/01/24	19/01/24	Diseño de la base de datos Firebase	2
20/01/24	21/01/24	Creación de estructura de la app	3
22/01/24	23/01/24	Desarrollo del login/función de alumnos	6
24/01/24	25/01/24	Desarrollo del login/función de admin	5
26/01/24	27/01/24	Integración con Firebase	6
28/01/24	29/01/24	Pruebas de login y autenticación	1
30/01/24	31/01/24	Implementación de UI mejorada	2
01/02/24	02/02/24	Ajustes en base de datos y permisos	2
03/02/24	04/02/24	Pruebas generales	1
05/02/24	06/02/24	Documentación del proyecto	7
07/02/24	08/02/24	Revisión final y correcciones	2
09/02/24	10/02/24	Preparación entrega del proyecto	1
TOTAL			41