

## ACTIVIDAD #2

**INTRODUCCIÓN**

La aplicación permitirá a docentes y alumnos gestionar calificaciones. Por ejemplo:

- El docente puede registrarse, iniciar sesión, crear grupos, añadir alumnos, asignar calificaciones, editar o eliminar registros.
- El alumno puede ver sus calificaciones.

**Vista de Inicio / Home**

- Breve descripción del sistema ("Sistema de Gestión de Calificaciones").
- Botones: "Iniciar Sesión" / "Registrarse".
- Imágenes o íconos educativos.

Propósito: Página de bienvenida que guía al usuario a loguearse o registrarse.

**¡Úsalos para crear magia en el tablero!**

nota sugerida

será una aplicación entendible y manejable

Conectores

elementos de pantalla principal

facebook

Google

Search

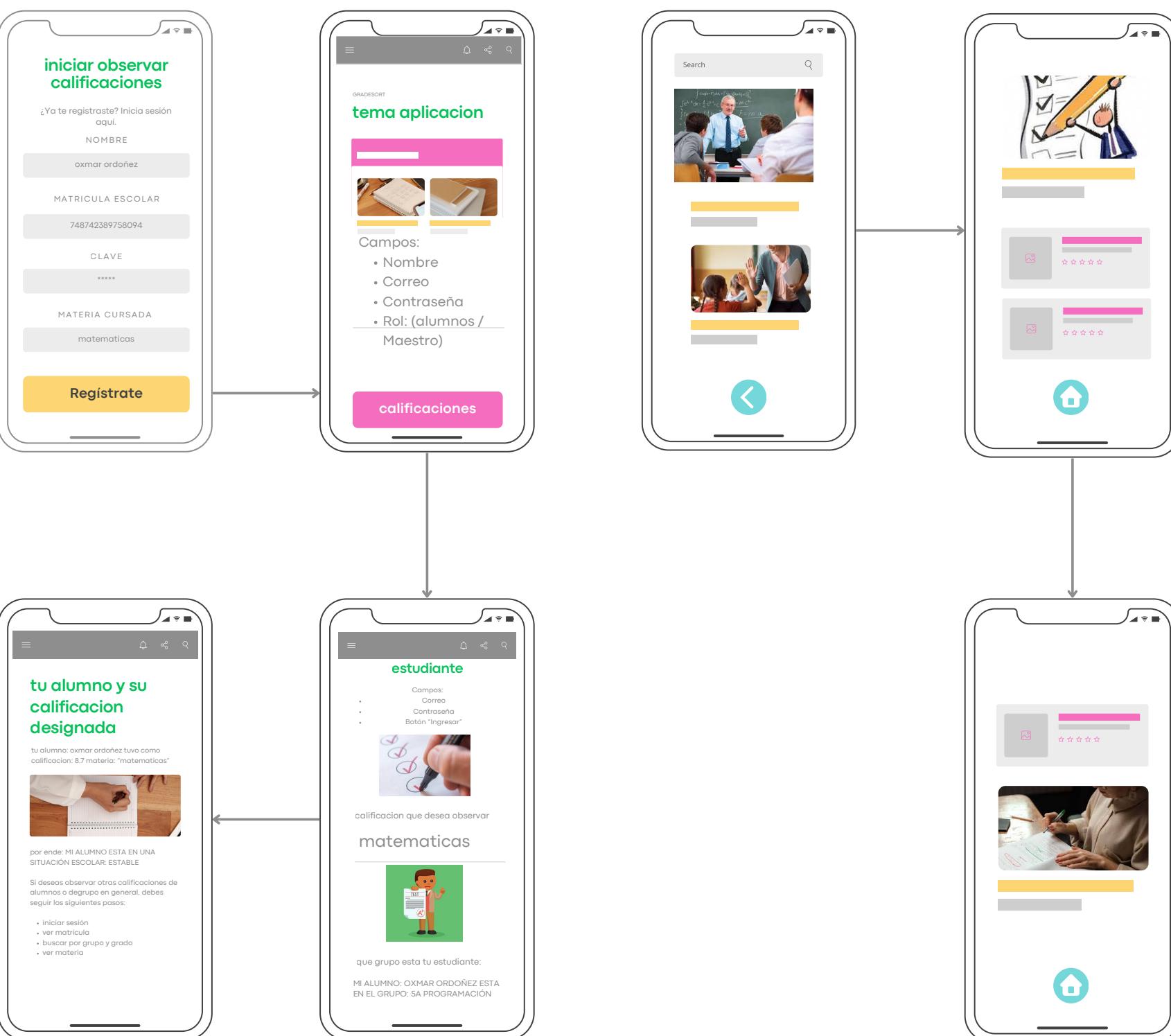
matemáticas

AQUÍ TENDRÁ BOTONES Botón de iniciar sesión

Botones:

# Flujo de pantalla

Es lo que relativamente quiero que lleve mi aplicación y la pantalla principal de mi proyecto.



## ALGUNAS IDEAS PARA LA APLICACIÓN

Elemento de acción	Persona designada
EJECUTAR LA CALIFICACION	MAESTRO
OBSERVAR LA CALIFICACION	ALUMNO/MAESTRO
MODIFICAR CALIFICACION	MAESTRO
SUBIR CALIFICACION	MAESTRO
PONER REPROBADO	MAESTRO
DAR DE BAJA MATERIA	ALUMNO/MAESTRO

### APLICACIÓN: GradeSort

#### METODO CRUD

En la aplicación, el usuario interactúa con distintas vistas que ejecutan operaciones CRUD. Cuando entra al dashboard se realiza **(R) Read**, ya que el frontend solicita los estudiantes al backend mediante una petición GET y los muestra en pantalla. Al crear un nuevo estudiante se ejecuta **(C) Create**, pues el formulario envía un POST con los datos y el backend guarda el registro. Al abrir el detalle de un estudiante ocurre nuevamente **(R) Read**, obteniendo sus calificaciones mediante otra petición GET. Para agregar una nueva calificación se usa **(C) Create**, enviando un POST que el backend almacena. Cuando el usuario decide editar una calificación se realiza **(U) Update**, enviando un PUT para modificar los datos existentes. Finalmente, al eliminar una calificación se aplica **(D) Delete**, mediante una petición DELETE que la borra de la base de datos y actualiza la interfaz.