

## *Integrantes*

*Mariana Rivera Felipe* *s23020355*

*Itzia Dayanna Rayón Illescas* *s23002499*

*Yahaira García Martínez* *s23002505*

## Diagrama general

<b>*anxiosimetro</b>	<b>*inicio</b>	<b>*registro</b>
<b>*autoabrazo</b>	<b>*login</b>	<b>*respiracion</b>
<b>*calendario</b>	<b>*pantallasEjercicios</b>	<b>*sentidos</b>
<b>*checkingScreen</b>	<b>pantallaRapida</b>	<b>*test</b>
<b>*escaneo</b>	<b>pantallaTest</b>	<b>*testDiario</b>
<b>*infoAnsiedad</b>	<b>perfil</b>	<b>*testPersonalidad</b>

## APLICACIÓN ANXIOUSLY

El flujo de acceso comienza con la clase **inicio**, la cual dirige la navegación hacia **login** o **registro**. Tras la autenticación, el **login** establece la sesión del **Usuario** (implícito) y navega hacia el **test** que nos lleva con el **anxiosimetro**

El **anxiosimetro** es la vista principal que usa las estadísticas de racha y tiempo sin ansiedad. El **anxiosimetro** tiene una asociación con **test** (el test inicial que configura el monitoreo) y una dependencia directa con **pantallasEjercicios** (la intervención de apoyo rápido).

El **calendario** es la clase de seguimiento de datos que agrega la información recolectada por el usuario que indica como se siente (las emociones y notas diarias).

El **checkingScreen** actúa como la interfaz central de recursos. **checkingScreen** establece una relación de agregación (relación débil) con todos los ejercicios y recursos disponibles:

**respiracion, sentidos, escaneo, autoabrazo, testDiario, testPersonalidad e infoAnsiedad.**

El **testDiario** (el cuestionario de chequeo diario) tiene una dependencia con **pantallaTest** (el video de intervención rápida) que se sugiere en momentos de alta ansiedad. El recurso psicoeducativo **infoAnsiedad** es introducido por la clase **pantallaRapida** (la introducción animada con progreso).

## ANÁLISIS DE ENUNCIADOS

### Oración 1: Flujo de acceso y perfil

El flujo de acceso comienza con la clase **inicio**, la cual dirige la navegación hacia **login** o **registro**. Tras la autenticación, el **login** establece la sesión del **Usuario** (implícito) y navega hacia el **test** que nos lleva con el **anxiosimetro**.

Asociación	Multiplicidad	Relación
inicio -> login (dirige a)	1 -> 1	asociación
inicio -> registro (dirige a)	1 -> 1	asociación
Login -> test (navega a)	1 -> 1	asociación
Test -> anxiosimetro (navega a)	1 -> 1	asociación

### Oración 2: Anxiosimetro y apoyo rápido

El **anxiosimetro** es la vista principal que usa las estadísticas de racha y tiempo sin ansiedad. El **anxiosimetro** tiene una asociación con **test** (el test inicial que configura el monitoreo) y una dependencia directa con **pantallasEjercicios** (la intervención de apoyo rápido).

Asociación	Multiplicidad	Relación
Anxiosimetro -> test (usa)	1 -> 1	asociación
Anxiosimetro -> pantallasEjercicios (depende de)	1 -> 1	asociación

### Oración 3: calendario y emociones

El **calendario** es la clase de seguimiento de datos que agrega la información recolectada por el usuario que indica como se siente (las emociones y notas diarias).

Asociación	Multiplicidad	Relación
Calendario -> datosEmocion (agregar registros)	1 -> *	agregación
Usuario -> datosEmocion (genera)	1 -> *	asociación

### Oración 4: checking screen y recursos

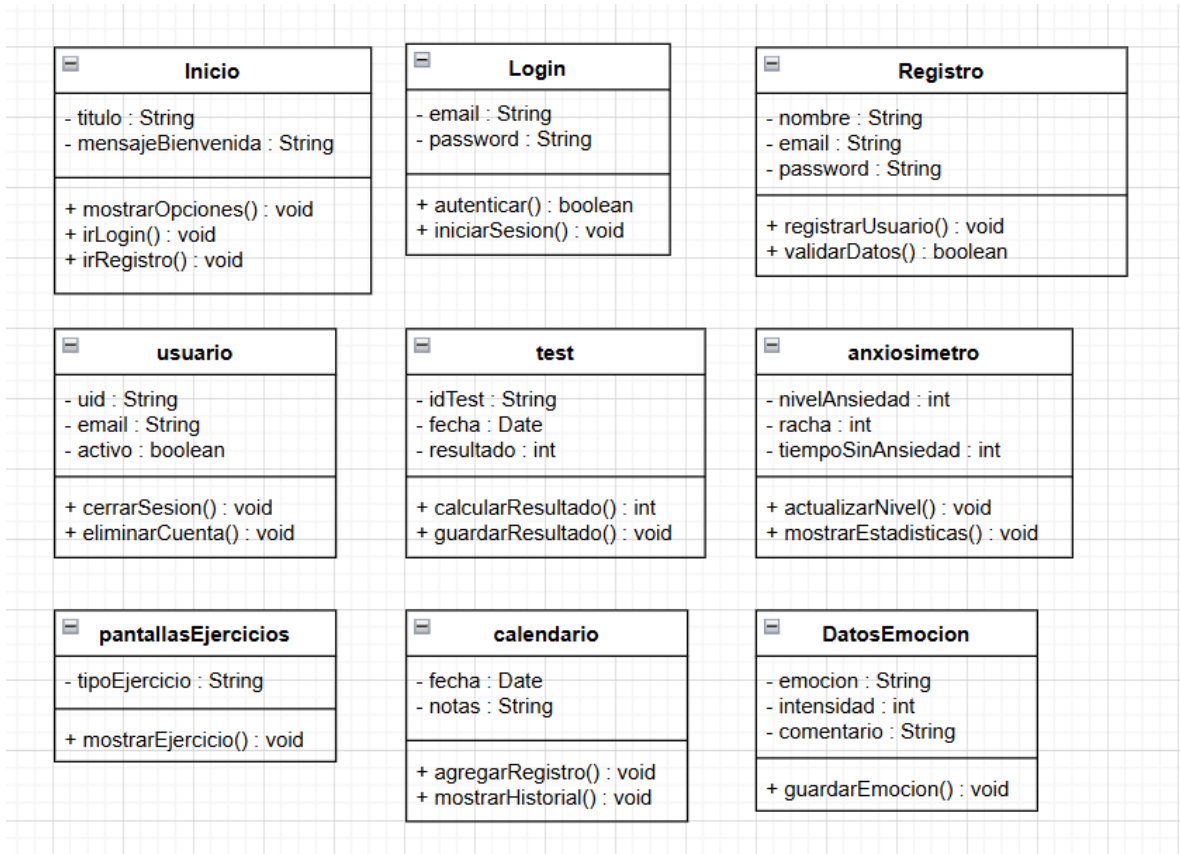
El **checkingScreen** actúa como la interfaz central de recursos. **checkingScreen** establece una relación de agregación (relación débil) con todos los ejercicios y recursos disponibles: **respiracion, sentidos, escaneo, autoabrazo, testDiario, testPersonalidad e infoAnsiedad**.


Asociación	Multiplicidad	Relación
checkingScreen -> respiracion (agrega)	1 -> *	agregación
checkingScreen -> sentidos (agrega)	1 -> *	agregación
checkingScreen -> escaneo (agrega)	1 -> *	agregación
checkingScreen -> autoabrazo (agrega)	1 -> *	agregación
checkingScreen -> testDiario (agrega)	1 -> *	agregación
checkingScreen -> testPersonalidad (agrega)	1 -> 1	agregación
checkingScreen -> infoAnsiedad (agrega)	1 -> *	agregación


### Oración 5: Intervenciones específicas y recursos educativos


El **testDiario** (el cuestionario de chequeo diario) tiene una dependencia con **pantallaTest** (el video de intervención rápida) que se sugiere en momentos de alta ansiedad. El recurso psicoeducativo **infoAnsiedad** es introducido por la clase **pantallaRapida** (la introducción animada con progreso).


Asociación	Multiplicidad	Relación
testDario -> pantallaTest (depende de)	1 -> 1	asociación
pantallaRapida -> InfoAnsiedad (introduce contenido)	1 -> 1	asociación





 <b>checkingScreen</b>
- titulo : String
+ mostrarRecursos() : void


 <b>Respiracion</b>
- duracion : int
+ iniciarEjercicio() : void


 <b>Sentidos</b>
- sentidoActivo : String
+ iniciarEjercicio() : void


 <b>escaneo</b>
- zonaCuerpo : String
+ iniciarEscaneo() : void


 <b>autoabrazo</b>
- tiempo : int
+ iniciarEjercicio() : void

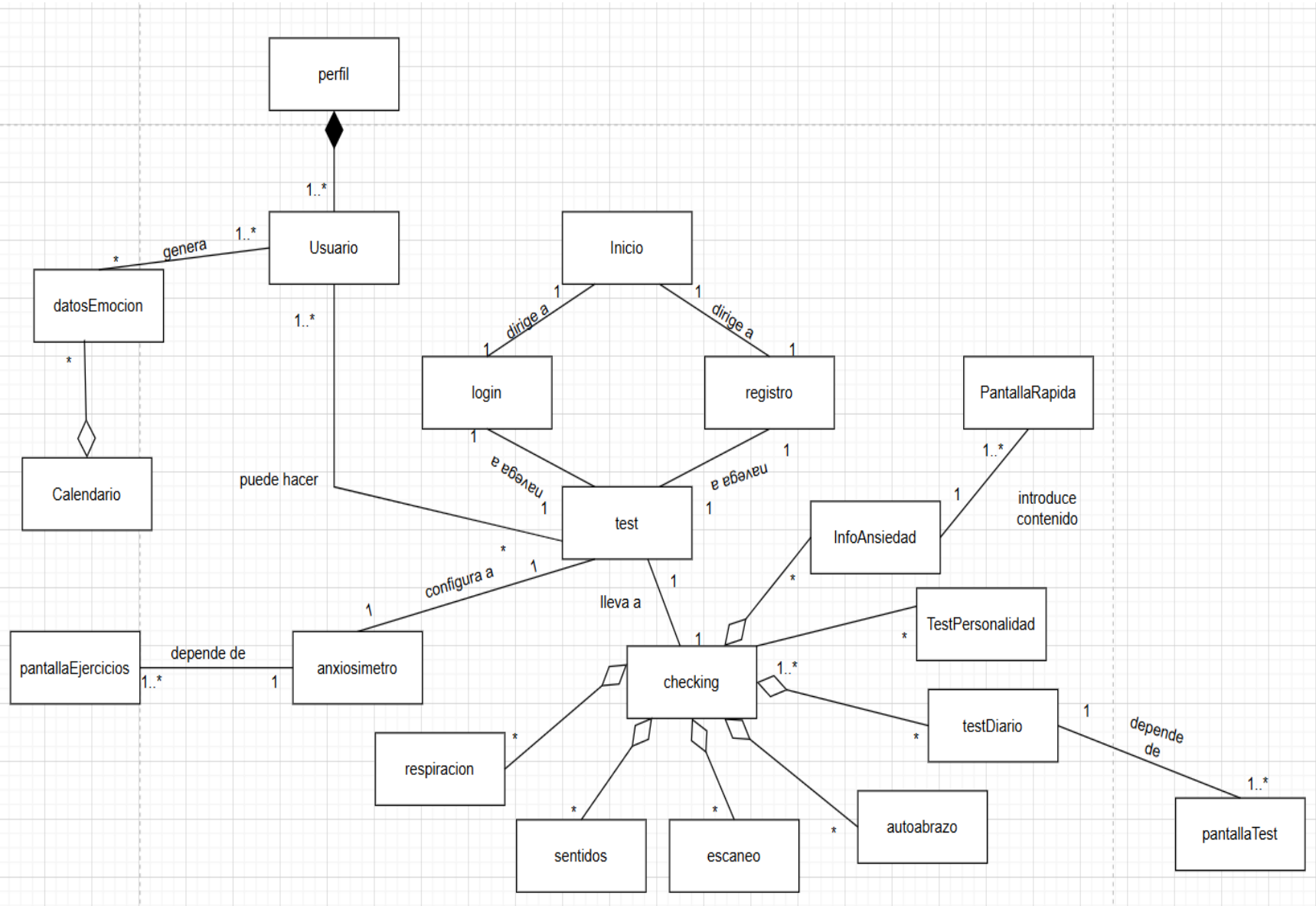
 <b>testDiario</b>
- respuestas : int[]
+ evaluarRespuestas() : int

 <b>testPersonalidad</b>
- tipoPersonalidad : String
+ evaluarPersonalidad() : String

 <b>infoAnsiedad</b>
- titulo : String
- descripcion : String
+ mostrarInformacion() : void

 <b>pantallaTest</b>
- urlVideo : String
+ reproducirVideo() : void

 <b>pantallaRapida</b>
- progreso : int
+ mostrarIntroduccion() : void



# Tecnologías utilizadas

## 1. Desarrollo del frontend

Categoría	Tecnología	Uso específico
Framework principal	React native	Componentes base (View, Text, ScrollView, TouchableOpacity, etc) , useState, useEffect, useRef.
Plataforma de desarrollo	Expo	Paquetes para notificaciones y el entorno para permitir el uso de iconos.
Navegación/ UI	React navigation	Utilizado para navegar entre pantallas
Componentes	Ionicons	Uso de @expo/vector-icons para mostrar iconos (stats-chart-outline, flash-outline, etc.).
Estilos y apariencias	React native style sheet	Uso de stylesheet para definir estilos.
Animaciones	React native animated	Uso de Animated.Value, Animated.timing, Animated.parallel, Animated.sequence, y Animated.loop para las animaciones del botón de ayuda y las barras de progreso.
Degradados	Expo linear gradient	Utilizado para el fondo de pantalla

## 2. Backend

Categoría	Tecnología	Uso específico
Base de datos	Firebase	Importaciones de firebase/auth y firebase/firestore. Es la fuente de la verdad para datos de usuario y autenticación.
Servicio de autenticación	Firebase authentication	Uso de <code>getAuth()</code> y <code>auth.currentUser</code> para obtener el ID de usuario ( <code>uid</code> ).
Servicio de base de datos	Cloud firestore	Uso de <code>doc</code> , <code>getDoc</code> , y <code>db</code> (configuración) para obtener datos específicos del usuario (genero, nombre) de la colección "usuarios"

## 3. Servicios específicos

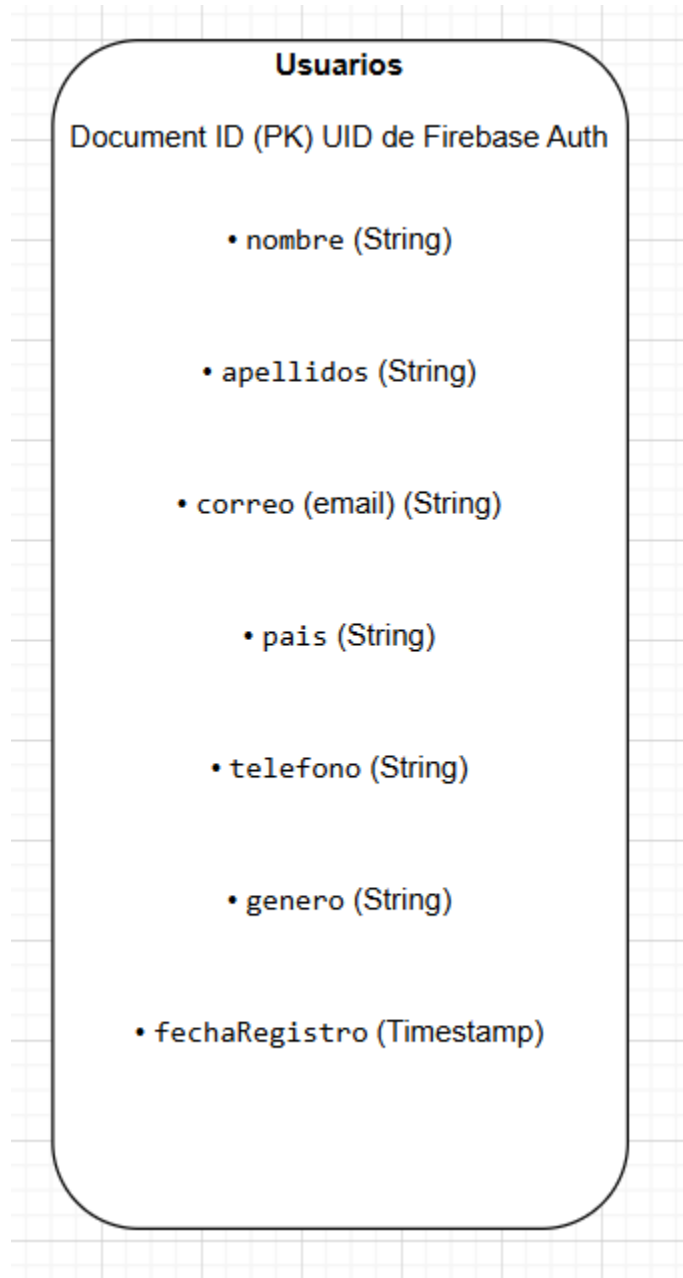
Categoría	Tecnología	Uso específico
Notificaciones	Expo notificaciones	Utilizado para <code>setNotificationHandler</code> , <code>getPermissionsAsync</code> , <code>requestPermissionsAsync</code> , <code>scheduleNotificationAsync</code> , y <code>cancelScheduledNotificationAsync</code> para programar las felicitaciones.

En resumen, utilizamos:

- frontend: React native (Expo)
- lenguaje: JavaScript
- Base de datos: Google Firebase



# Diseño de la base de datos (ERD o modelo lógico)



Solo se maneja una entidad que es USUARIO el sistema puede registrar, guardar, iniciar sesión y eliminar cuenta de la base de datos, no almacena contraseñas y la autenticación se maneja con firebase authentication.

Debido a que Firebase es una base de datos NoSQL orientada a documentos el diseño se presenta como un modelo lógico, donde nuestra entidad USUARIO se representa mediante una colección y documentos sin relaciones.

## Estructura de módulos

### 1. Módulos de pantalla (screen)

Estos módulos gestionan la creación, autenticación y configuración personal de la cuenta.

Módulo	Responsabilidad principal
Inicio.js	Pantalla de bienvenida e inicio al flujo de navegación hacia el login o registro.
Registro.js	Crea la cuenta del usuario en una firebase y guarda los datos iniciales.
Login.js	Autentica al usuario en firebase para acceder a la aplicación.
Perfil.js	Permite al usuario ver y gestionar sus datos personales, preferencias y acciones de la cuenta.

Módulo que contiene la lógica principal de seguimiento de ansiedad

Módulo	Responsabilidad principal
Test.js	Es una comprobación inicial de que tanta ansiedad tiene el usuario cada que entra para su chequeo.
Anxiosimetro.js	Mide el tiempo sin ansiedad que ha pasado el usuario y al momento de detener el conteo se muestran ejercicios rápidos para ayudar al usuario.

pantallaEjercicios.js	Intervención rápida de apoyo activada desde el anxiosímetro.
-----------------------	--

Módulos que se enfocan en los test de seguimiento y datos emocionales.

Módulo	Responsabilidad principal
testDiario.js	Cuestionario de 12 preguntas que registran el modo del día y otros síntomas.
Calendario.js	Muestra las fechas, se puede registrar las emociones diarias, notas y grafica de proceso.
TestPersonalidad.js	Este test es único cuestiona preguntas que genera un perfil emocional y un puntaje único.

Módulos de ejercicios de calma y técnicas terapéuticas

Módulo	Responsabilidad principal
Autoabrazo.js	Ejercicios de autoabrazo con temporizador, guía de respiración y audio de fondo suave.
respiracion.js	Ejercicio de respiración guiada con audio y animación de círculo.
escaneo.js	Ejercicio de escaneo corporal guiado con audio por diferentes zonas para identificar y liberar tensión.
sentidos.js	Ejercicios de anclaje de realidad para enfocar la atención usando los cinco sentidos.

Módulo que provee información psicoeducativa y gestiona intervenciones rápidas basadas en videos.

Módulo	Responsabilidad principal
infoAnsiedad.js	Contenido psicoeducativo sobre la ansiedad y el cuerpo.
PantallaRapida.js	Pantalla de introducción animada con un temporizador de 8 segundos que lanza el artículo infoAnsiedad.
checkingScreen.js	Interfaz principal para acceder a todos los recursos y ejercicios de los módulos.
Pantallatest.js	Serie de videos y audios que se usan como intervención rápida sugerida por el testdiario en caso de ansiedad.

Módulo de la base de datos y nuestro main

Módulo	Responsabilidad principal
app.js	Define las rutas de la aplicación y especifica las pantallas.
firebaseConfig.js	Define el acceso a la firestore, que es donde almacenamos los datos de los usuarios.