



**Asignatura:**

Desarrollo de Software para Móviles DSM941 G01T

**Docente:**

Ing. Alexander Alberto Sigüenza Campos

**Tema:**

“Desarrollo de una Aplicación de Control de Gastos Personales con Firebase y Kotlin en Android Studio”

**Integrantes:**

Moisés Alberto Prieto Torres- PT200508

Ludwin Fuentes Beltrán FB232274,

Rodrigo Henríquez López HL211477,

Douglas Mejía Novoa MN142088,

Moisés Prieto Torres PT200508,

Ronald Renderos Ramos RR150535

Oscar Portillo Barahona PB161930

## Índice

Desarrollo de una Aplicación de Control de Gastos Personales con Firebase y Kotlin en Android Studio .....	3
Introducción.....	3
1. Investigación sobre Autenticación en Firebase.....	3
2. Documentación de la Implementación de Autenticación.....	4
3. Desarrollo de la Aplicación de Control de Gastos .....	4
4. Anexos.....	5
I. Crear un Sitio en Firabase: .....	5
II. Creamos y Registramos App en Firabase .....	6
III. Creación de Autenticación desde Firebase.....	8
5. Capturas de Pantalla Aplicacion (Referenciales).....	9
6. Conclusiones del Grupo .....	11

# Desarrollo de una Aplicación de Control de Gastos Personales con Firebase y Kotlin en Android Studio

## Introducción

El presente documento describe el desarrollo de una aplicación móvil en Android Studio utilizando Kotlin, diseñada para permitir que los usuarios gestionen sus gastos personales mediante autenticación con Firebase Authentication y almacenamiento de datos en Cloud Firestore. El objetivo principal es aplicar los conocimientos sobre autenticación, base de datos en la nube y diseño de interfaces XML, logrando un sistema funcional y visualmente claro que permita registrar, visualizar y calcular gastos de forma segura y dinámica.

## 1. Investigación sobre Autenticación en Firebase

Firebase Authentication ofrece múltiples métodos de autenticación para que los usuarios puedan iniciar sesión de manera segura en las aplicaciones. Los principales métodos incluyen autenticación por correo electrónico y contraseña, cuentas de Google, autenticación por teléfono, Facebook, Twitter, y proveedores personalizados.

A continuación, se describen los más comunes:

- Autenticación por correo electrónico y contraseña: Es el método más simple y ampliamente utilizado. Permite registrar usuarios con un correo y contraseña únicos, gestionados directamente desde la consola de Firebase. Ventajas: fácil de implementar y sin dependencias externas. Limitaciones: requiere mayor manejo de validaciones y recuperación de contraseñas.
- Autenticación con Google: Permite a los usuarios iniciar sesión usando su cuenta de Google. Ventajas: rápida, segura y sin necesidad de crear nuevas credenciales. Limitaciones: depende de los servicios de Google y requiere la configuración del OAuth 2.0 Client ID.
- Autenticación por teléfono: Envía un código de verificación al número del usuario. Ventajas: fácil para usuarios sin correos activos. Limitaciones: puede generar costos por SMS y requiere conexión estable.

En general, Firebase Authentication facilita la administración de sesiones y la protección de datos mediante tokens de acceso seguros. Se integra fácilmente con Firestore y otras herramientas del ecosistema Firebase.

## 2. Documentación de la Implementación de Autenticación

Para implementar la autenticación con Firebase (correo electrónico y Google), se siguieron los siguientes pasos:

### 1. Paso 1. Configuración del proyecto en Firebase Console:

- Se creó un nuevo proyecto en <https://console.firebase.google.com>
- Se habilitaron los métodos de autenticación: correo electrónico y Google.
- Se configuró el nombre del paquete de la aplicación (por ejemplo: com.example.gastosapp).

### 2. Paso 2. Integración del SDK en Android Studio:

- Se añadió la dependencia de Firebase Authentication y Firestore en el archivo build.gradle (Module: app).  
implementation 'com.google.firebase:firebase-auth:22.3.0'  
implementation 'com.google.firebase:firebase-firestore:24.9.1'
- Se inicializó Firebase en la clase principal usando `FirebaseApp.initializeApp(context)`.

### 3. Paso 3. Implementación del flujo de login:

- Se creó una actividad XML para el login con campos EditText (correo y contraseña) y un botón de acceso.
- En Kotlin, se implementó el método `signInWithEmailAndPassword` y el flujo con `GoogleSignInClient` para el inicio con Google.
- Se manejaron los errores de autenticación con try/catch y mensajes Toast.

### 4. Paso 4. Validaciones y manejo de errores:

- Se verificó que los campos de texto no estuvieran vacíos.
- Se mostró retroalimentación al usuario mediante mensajes de error.
- En caso de éxito, el usuario fue redirigido a la pantalla principal (MainActivity).

## 3. Desarrollo de la Aplicación de Control de Gastos

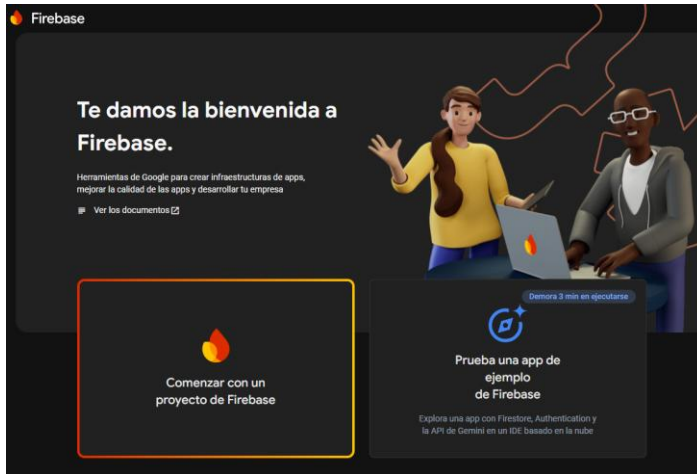
Una vez implementada la autenticación, se procedió con el desarrollo del módulo de control de gastos. Este módulo permite al usuario registrar nuevos gastos, visualizarlos y calcular un total mensual.

Funciones principales:

- Registrar nuevos gastos (nombre, monto, categoría, fecha).
- Guardar los datos en Firebase Firestore asociados al usuario autenticado.
- Visualizar los gastos en un RecyclerView.
- Calcular el total mensual y mostrarlo en pantalla.
- Filtrar los gastos por categoría o fecha (opcional).

## 4.Anexos

### I. Crear un Sitio en Firabase:



Seleccionamos comenzar Proyecto :

× Crear un proyecto

Comencemos con el nombre de tu proyecto<sup>?</sup>

Ingresa el nombre de tu proyecto

my-awesome-project-id

☐ Acepto las [Condiciones de Firebase](#)

☐ Confirmando que usaré Firebase exclusivamente para fines relacionados con mi trabajo, empresa, oficio o profesión.

Le damos un nombre al Proyecto :

Comencemos con el nombre de tu proyecto<sup>?</sup>

Nombre del proyecto

Login-Auntentificacion

login-auntentificacion

Copiamos el Identificador Único : login-aumentificacion

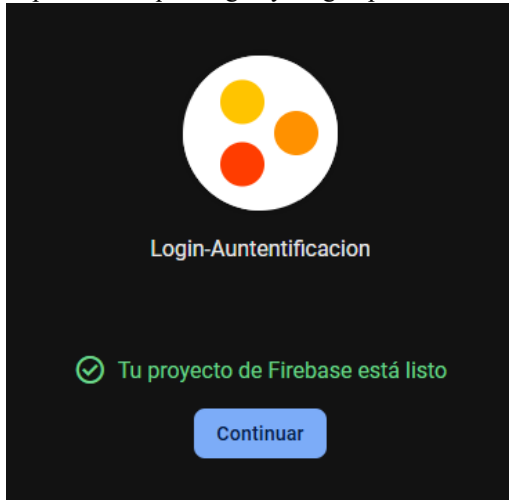
Apareceren Opciones Como:

Habilitar IA Gemini para Firabase\_\_\_\_\_ON

Habilitar Google Analytics para Firabase\_\_OFF

Entre Otros: “Habilitar si se desean Ocupar”

Esperamos a que cargue y luego aparecerá esto :



Le damos Continuar :

Aparecerá lo siguiente:



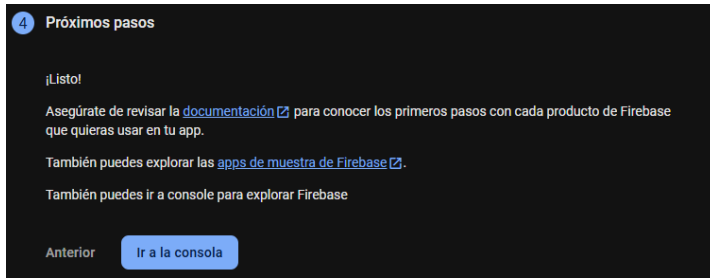
Nota: Entre Otros menús que se lograran Visualizar navegando y curioseando entre los menús de Firebase

## II. Creamos y Registramos App en Firabase

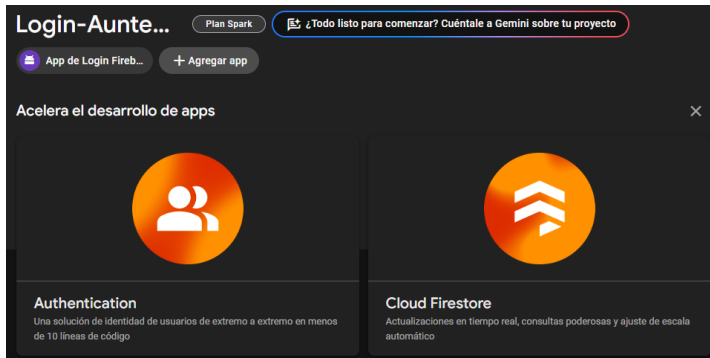
Seleccionamos el Apartado de Android en este caso ( A modo de Ejemplo )

Primeros Pasos :

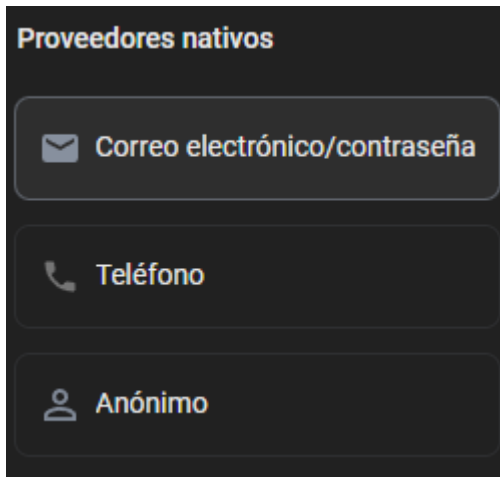




### III. Creación de Autenticación desde Firebase



Seleccionamos el Método por Autenticación  
Seleccionamos Proveedores Nativos



En la Configuración Interna debe de verse de la siguiente manera :

Proveedor	Estado
<a href="#">Correo electrónico/contraseña</a>	<span>✓</span> <b>Habilitado</b>



## 5. Capturas de Pantalla Aplicacion (Referenciales)

Pantalla de inicio de sesión: muestra el formulario de autenticación por correo y Google.

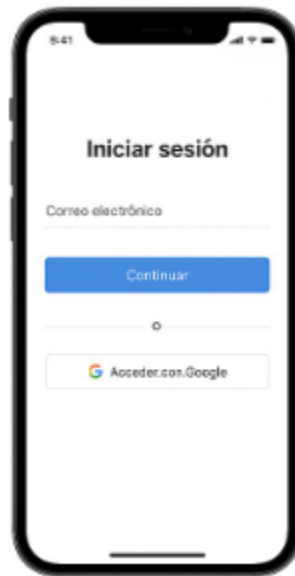


Figura 1 Pantalla de inicio de sesión

Pantalla principal: lista de gastos registrados del usuario.

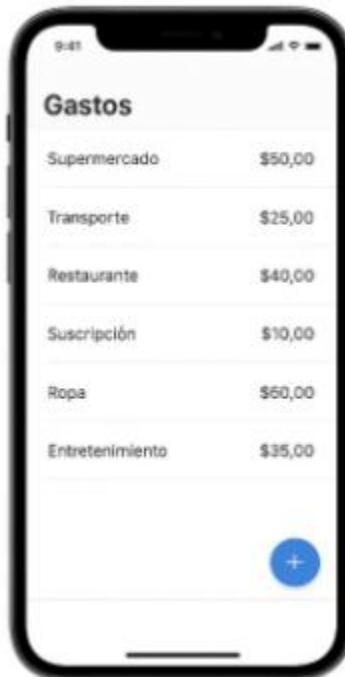


Figura 2 Pantalla principal

Formulario de nuevo gasto: campos para ingresar nombre, monto, categoría y fecha.



Nuevo gasto

Nombre

Monto

Categoría >

Fecha 22 abr, 2024

Guardar

Figura 3 Formulario de nuevo gasto

Resumen mensual: gráfico o texto con el total de gastos del mes.

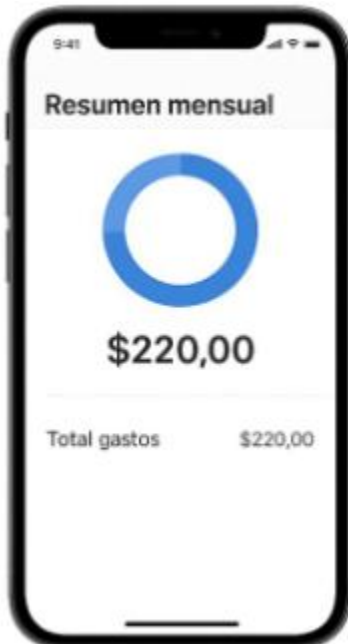


Figura 4 Resumen mensual

## 6. Conclusiones del Grupo

El desarrollo de esta aplicación permitió reforzar el conocimiento en Kotlin, Firebase y el diseño de interfaces en XML. Firebase Authentication facilitó la gestión de usuarios de manera segura, mientras que Firestore permitió almacenar y recuperar datos en tiempo real.

La experiencia demostró la importancia de integrar herramientas en la nube para crear aplicaciones escalables, seguras y funcionales. Además, la práctica ayudó a comprender los flujos de login, validaciones de formularios y la persistencia de datos en entornos móviles modernos.