



Instituto Politécnico Nacional

Escuela Superior De Computo

Desarrollo De Aplicaciones Móviles Nativas



Tarea 6

Desarrollo de una Aplicación de Realidad Aumentada

Nombre Del Alumno:

García Quiroz Gustavo Ivan | 2022630278

Grupo: 7CV3

Nombre del profesor: Hurtado Avilés Gabriel

Fecha de entrega: 15/06/2025

Índice

SystemBooks - Sistema de Gestión de Libros con Firebase.....	1
Características Principales.....	1
Sistema de Autenticación Dual	1
Gestión de Libros.....	1
Sistema de Amigos	1
Sistema de Notificaciones Push (FCM)	2
Dashboard Administrativo	2
Tecnologías Utilizadas.....	2
Estructura del Proyecto.....	3
Configuración e Instalación	4
Prerrequisitos	4
Pasos de Instalación	4
Configurar Firebase	4
Configurar Firestore	4
Estructura de Datos en Firebase	5
Funcionalidades Principales.....	6
Autenticación	6
Sistema de Amigos	6
Notificaciones.....	7
Dashboard Administrativo	7
Seguridad y Permisos	7
Reglas de Firestore.....	7
Autenticación Segura	7
Arquitectura de la Aplicación	8

Patrón Repository	8
Gestión de Estado.....	8
Comunicación de Datos	8
Documentación Adicional.....	8
Licencia	9
Capturas de pantalla	10
Conclusión.....	12
Referencias	13

SystemBooks - Sistema de Gestión de Libros con Firebase

SystemBooks es una aplicación móvil Android desarrollada en Java que permite realizar búsquedas de libros y autores utilizando la Open Library API. La aplicación implementa un sistema completo de autenticación dual que incluye tanto un servidor REST tradicional como Firebase Authentication, proporcionando a los usuarios múltiples opciones de registro e inicio de sesión.

Características Principales

Sistema de Autenticación Dual

- **Autenticación tradicional:** Servidor REST con operaciones CRUD
- **Firebase Authentication:** Registro e inicio de sesión con email/contraseña
- **Gestión de roles:** ROLE_ADMIN y ROLE_USER
- **Contraseña maestra** para administradores: Admin1234!
- **Persistencia de sesión** segura

Gestión de Libros

- Búsqueda de libros y autores desde Open Library API
- Interfaz intuitiva para explorar contenido literario
- Integración con sistema de autenticación

Sistema de Amigos

- Búsqueda de usuarios por nombre o email
- Envío y gestión de solicitudes de amistad
- Lista de amigos y solicitudes pendientes
- Notificaciones automáticas para eventos de amistad

Sistema de Notificaciones Push (FCM)

- **Notificaciones administrativas:** Los administradores pueden enviar notificaciones
- **Notificaciones automáticas:** Sistema de amigos y actividades
- **Notificaciones locales:** Alertas para solicitudes de amistad pendientes
- **Segmentación:** Envío a usuarios específicos o difusión general

Dashboard Administrativo

- **Monitoreo en tiempo real** de actividades de usuarios
- **Filtros por tipo de actividad:** Login, registro, búsquedas, etc.
- **Gestión de usuarios:** CRUD completo para administradores
- **Historial de actividades** con timestamps

Tecnologías Utilizadas

- **Frontend (Android)**
- **Java** para lógica de aplicación
- **XML** para diseño de interfaces
- **Material Design** para UI/UX
- **Retrofit** para comunicación con APIs
- **Glide** para carga de imágenes
- **RecyclerView** para listas dinámicas
- **Backend Firebase**
- **Firebase Authentication** - Autenticación de usuarios
- **Cloud Firestore** - Base de datos NoSQL
- **Firebase Cloud Messaging (FCM)** - Notificaciones push

- **Firestore** - Almacenamiento de datos
- **Firebase Functions** - Lógica del servidor (opcional)
- **APIs Externas**
- **Open Library API** - Búsqueda de libros y autores
- **ImgBB API** - Almacenamiento de imágenes de perfil

Estructura del Proyecto

SystemBooks/

```

├── app/
│   ├── src/main/java/com/example/systembooks/
│   │   ├── firebase/                # Clases de Firebase
│   │   │   ├── FirebaseManager.java
│   │   │   ├── FirebaseAuthRepository.java
│   │   │   ├── FirebaseUser.java
│   │   │   ├── NotificationHelper.java
│   │   │   └── FCMService.java
│   │   ├── fragment/                # Fragmentos de la aplicación
│   │   ├── adapter/                 # Adaptadores para RecyclerViews
│   │   ├── models/                  # Modelos de datos
│   │   ├── util/                    # Utilidades y helpers
│   │   └── MainActivity.java
│   └── src/main/res/
│       ├── layout/                  # Archivos de diseño XML
│       ├── menu/                    # Menús de navegación
│       └── values/                   # Strings y recursos

```

| └─ google-services.json # Configuración de Firebase
| └─ functions/ # Firebase Cloud Functions
| └─ docs/ # Documentación adicional

Configuración e Instalación

Prerrequisitos

- Android Studio Arctic Fox o superior
- SDK de Android 21 o superior
- Cuenta de Firebase (gratuita)
- Proyecto Firebase configurado

Pasos de Instalación

1. **Clonar el repositorio**
2. `git clone [URL_DEL_REPOSITORIO]`

`cd SystemBooks`

Configurar Firebase

- Crear un proyecto en [Firebase Console](#)
- Agregar aplicación Android al proyecto
- Descargar google-services.json y colocarlo en app/
- Habilitar Authentication, Firestore y Cloud Messaging

Configurar Firestore

- Crear índices compuestos necesarios
(ver FIRESTORE_INDEXES.md)
- Configurar reglas de seguridad para las colecciones

3. Compilar y ejecutar

`./gradlew assembleDebug`

Estructura de Datos en Firebase

- **Colección: users**

```
{  
  "uid": "string",  
  "username": "string",  
  "email": "string",  
  "role": "ROLE_USER | ROLE_ADMIN",  
  "photoUrl": "string",  
  "createdAt": "timestamp"  
}
```

- **Colección: friend_requests**

```
{  
  "senderId": "string",  
  "receiverId": "string",  
  "status": "PENDING | ACCEPTED | REJECTED",  
  "createdAt": "timestamp",  
  "updatedAt": "timestamp"  
}
```

- **Colección: friendships**

```
{  
  "user1Id": "string",
```



```
"user2Id": "string",  
"createdAt": "timestamp"  
}
```

- **Colección: dashboard_activities**

```
{  
  "type": "string",  
  "userId": "string",  
  "username": "string",  
  "description": "string",  
  "timestamp": "timestamp",  
  "details": "string"  
}
```

Funcionalidades Principales

Autenticación

- **Registro Firebase:** Email, contraseña y rol
- **Login Firebase:** Validación y persistencia de sesión
- **Gestión de roles:** Automática según contraseña maestra
- **Perfiles de usuario:** Edición y foto de perfil

Sistema de Amigos

- **Búsqueda de usuarios:** Por nombre o email
- **Solicitudes de amistad:** Envío, aceptación y rechazo
- **Notificaciones automáticas:** Para nuevas solicitudes y aceptaciones

- **Lista de amigos:** Visualización de conexiones

Notificaciones

- **Push notifications:** FCM para eventos importantes
- **Notificaciones locales:** Alertas para solicitudes pendientes
- **Panel administrativo:** Envío de notificaciones personalizadas
- **Segmentación:** A usuarios específicos o broadcast

Dashboard Administrativo

- **Actividades en tiempo real:** Monitoreo de acciones de usuarios
- **Filtros avanzados:** Por tipo de actividad y fecha
- **Métricas de uso:** Estadísticas de la aplicación
- **Gestión de usuarios:** CRUD para administradores

Seguridad y Permisos

- **Roles de Usuario**
- **ROLE_USER:** Acceso a búsqueda de libros, perfil y amigos
- **ROLE_ADMIN:** Acceso completo + gestión de usuarios y notificaciones

Reglas de Firestore

- Lectura/escritura basada en autenticación
- Validación de roles en operaciones administrativas
- Protección de datos sensibles

Autenticación Segura

- Tokens JWT para sesiones
- Encriptación de contraseñas

- Validación de entrada en formularios

Arquitectura de la Aplicación

Patrón Repository

- FirebaseAuthRepository: Gestión de autenticación
- FriendshipRepository: Operaciones de amistad
- DashboardActivityRepository: Registro de actividades

Gestión de Estado

- SessionManager: Persistencia de sesión de usuario
- RoleManager: Validación de permisos
- ActivityTracker: Seguimiento de actividades

Comunicación de Datos

- **Firestore Listeners**: Actualizaciones en tiempo real
- **Retrofit**: Comunicación con Open Library API
- **FCM**: Mensajería push

Documentación Adicional

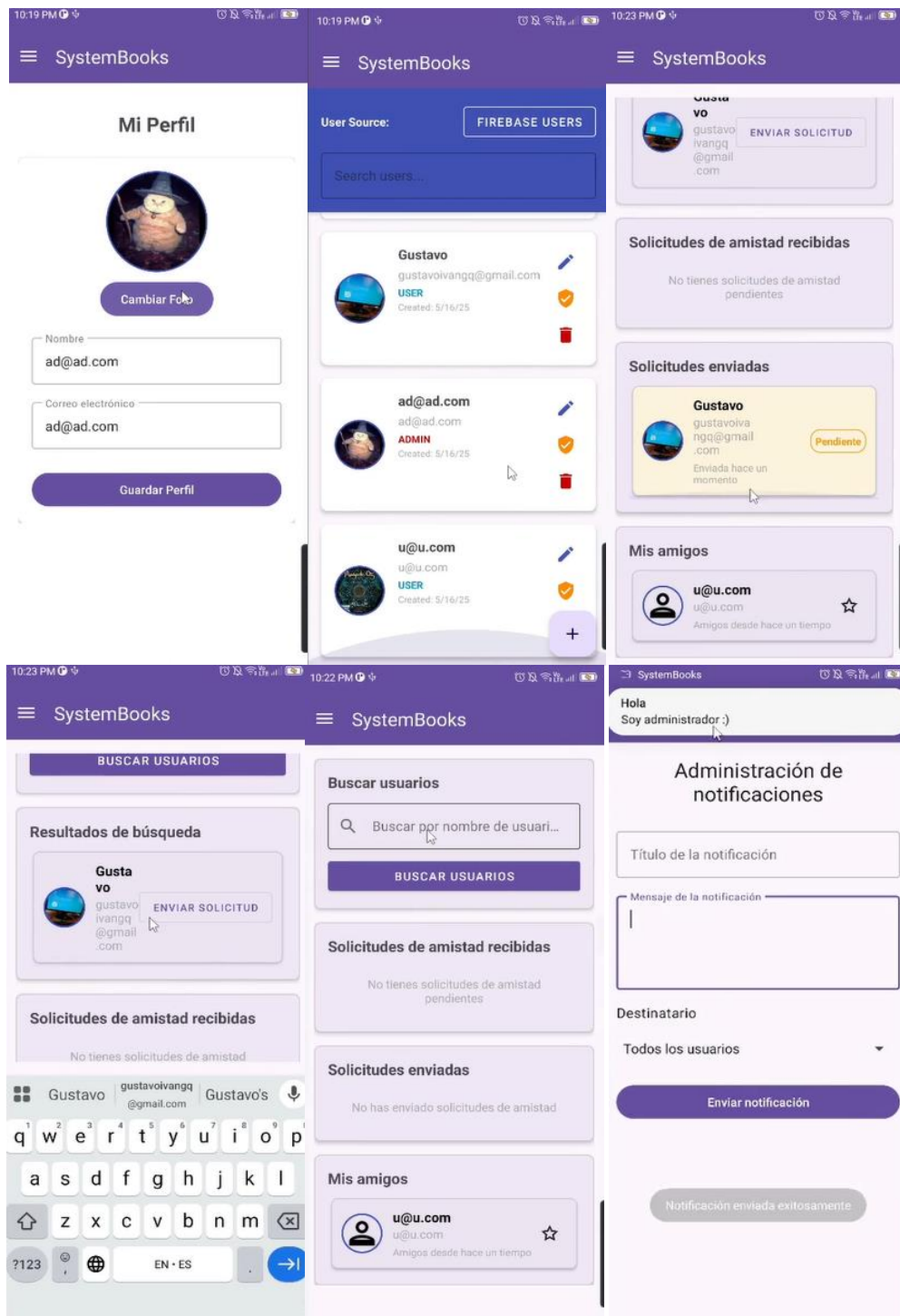
Para información más detallada, consultar los siguientes archivos:

- FIREBASE_INTEGRATION.md - Guía de integración de Firebase
- FRIENDS_SYSTEM.md - Sistema de amigos completo
- FIRESTORE_INDEXES.md - Configuración de índices
- FIREBASE_FUNCTIONS.md - Cloud Functions para FCM
- LOCAL_FRIEND_NOTIFICATIONS.md - Notificaciones locales
- DEBUG_FRIENDS_ISSUE.md - Solución de problemas comunes

Licencia

Este proyecto es desarrollado con fines educativos como parte del curso de Desarrollo de Aplicaciones Móviles Nativas.

Capturas de pantalla



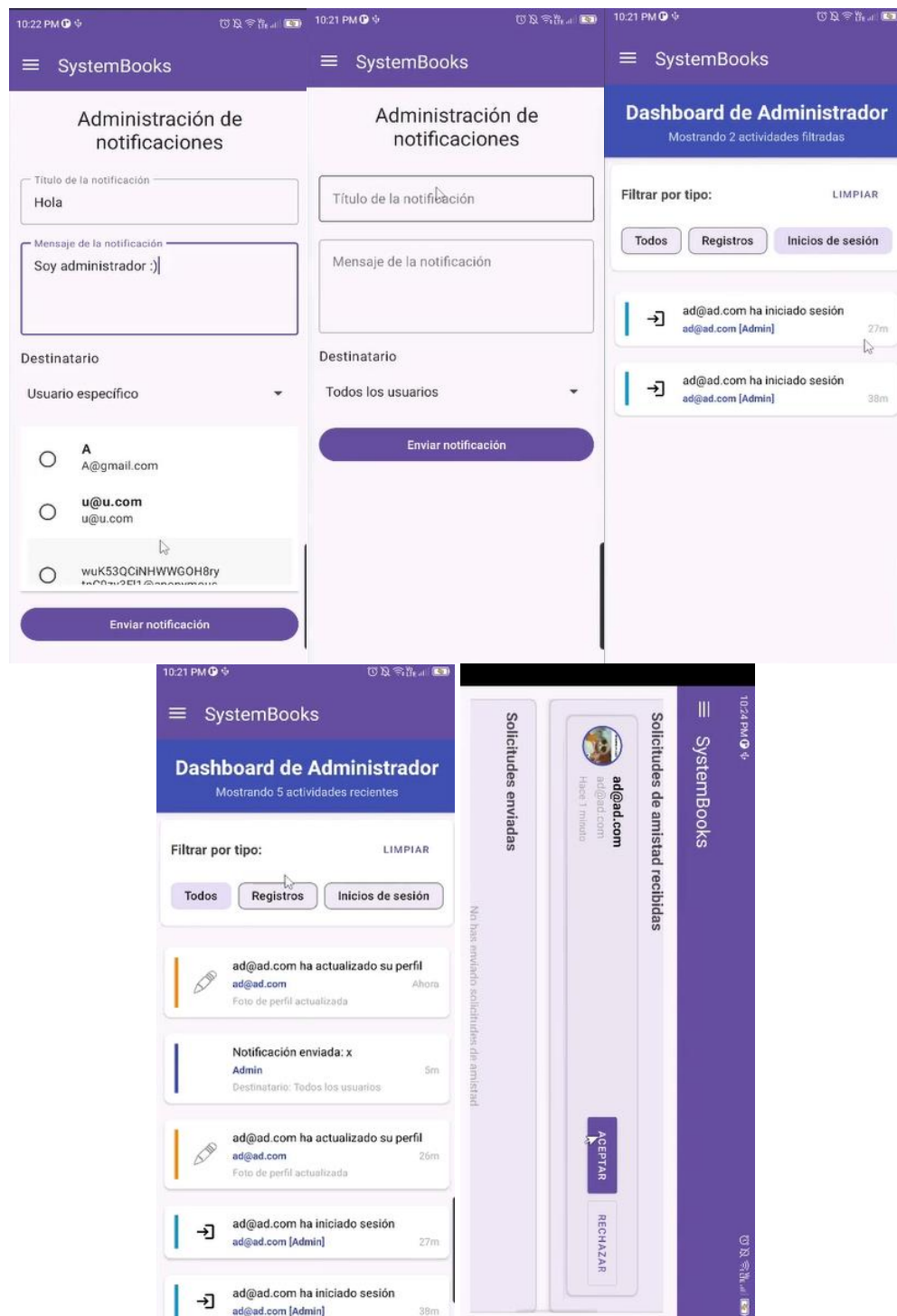


Figura 1 Dispositivo móvil

Conclusión

Este proyecto demuestra la integración exitosa de Firebase con una aplicación Android tradicional, proporcionando un sistema de autenticación dual y funcionalidades avanzadas de red social.

Referencias

- Android Developers. (2025). *Android SDK documentation*. Google.
<https://developer.android.com/>
- Firebase. (2025). *Firebase documentation*. Google.
<https://firebase.google.com/docs>
- Google. (2025). *Firebase Authentication*. Firebase.
<https://firebase.google.com/docs/auth>
- Google. (2025). *Cloud Firestore*. Firebase.
<https://firebase.google.com/docs/firestore>
- Google. (2025). *Firebase Cloud Messaging*. Firebase.
<https://firebase.google.com/docs/cloud-messaging>
- Google. (2025). *Material Design*. <https://material.io/design>
- ImgBB. (2025). *ImgBB API documentation*. <https://api.imgbb.com/>
- Internet Archive. (2025). *Open Library API*.
<https://openlibrary.org/developers/api>
- Oracle Corporation. (2025). *Java SE documentation*.
<https://docs.oracle.com/javase/>
- Square, Inc. (2025). *Retrofit*. <https://square.github.io/retrofit/>
- Bumptech. (2025). *Glide*. <https://bumptech.github.io/glide/>