Universidad Don Bosco

Ingeniería en Ciencias de la Computación

Desarrollo de Software para Móviles (DSM941)

Foro II

Presentado por

Ardon Martinez, Marlin José	AM120254
David Ernesto Mejía Villalobos	MV191973
Velasco Crespín, Alejandro Ernesto	VC161941
Ramírez Guardado, Ronald Gerardo	RG110604

Introducción

La implementación de una sólida autenticación de usuarios en aplicaciones Android desarrolladas en Kotlin se ha vuelto más accesible y versátil gracias a las múltiples opciones ofrecidas por Firebase. Desde la autenticación por correo electrónico y contraseña hasta la integración de proveedores de identidad social como Google, Facebook y Twitter, Firebase Authentication simplifica el proceso con una serie de soluciones listas para usar.

Además, la combinación entre Firebase y Kotlin no solo garantiza la seguridad de la autenticación, sino que también reduce significativamente la cantidad de código necesario.

Esta combinación ofrece una amplia gama de funcionalidades, desde la gestión de sesiones hasta la implementación de reglas de seguridad en la base de datos, lo que convierte a Firebase en una herramienta integral para gestionar la complejidad del backend en el desarrollo de aplicaciones Android.

Objetivos:

General

Evaluar la eficacia y la idoneidad de las opciones de autenticación proporcionadas por Firebase para el desarrollo de aplicaciones Android en Kotlin, analizando su impacto en la seguridad, la usabilidad y la eficiencia del proceso de autenticación de usuarios.

Específicos

- Comparativa de Métodos de Autenticación: Analizar y comparar la efectividad, la seguridad y la facilidad de implementación de distintos métodos de autenticación disponibles en Firebase (correo electrónico y contraseña, autenticación social como Google Sign-In y Facebook Login, entre otros) para determinar cuál se adapta mejor a diferentes contextos de desarrollo de aplicaciones Android.

- Experiencia del Usuario y Usabilidad: Evaluar la experiencia del usuario al utilizar diferentes métodos de autenticación en aplicaciones Android. Esto incluiría la facilidad de registro, inicio de sesión y la percepción general de seguridad y comodidad por parte de los usuarios.

- Rendimiento y Eficiencia: Analizar el rendimiento y la eficiencia de los distintos métodos de autenticación en términos de tiempo de respuesta, consumo de recursos del dispositivo y eficacia en entornos con conectividad variable o limitada.

Opciones existentes de autenticación Android Kotlin con Firebase

Existen varias opciones para implementar autenticación de usuarios en aplicaciones Android desarrolladas en Kotlin utilizando Firebase. Firebase Authentication proporciona una backend de autenticación lista para usar y soporta autenticación utilizando contraseña, número de teléfono, populares proveedores de identidad social como Google, Facebook y Twitter, y más.

La opción más básica es la autenticación por correo electrónico y contraseña. Sólo requiere unas pocas líneas de código para registrar un nuevo usuario con correo electrónico y contraseña o iniciar sesión con uno existente. Firebase se encarga de almacenar de forma segura las credenciales de usuario. También se puede habilitar la autenticación multifactor por SMS agregando un número de teléfono al perfil de usuario.

Para integrar Google Sign In, se necesita agregar la dependencia de Google Play Services auth a la app, configurar un proyecto OAuth 2.0 en la Google API Console, y luego usar el objeto GoogleSignInClient de la API de Google para desencadenar el flujo de autenticación y obtener los detalles del usuario.

De manera similar, Facebook Login puede implementarse mediante la Biblioteca de Facebook SDK para Android y el objeto LoginManager. Esta opción permite a los usuarios registrarse e iniciar sesión rápidamente con sus cuentas de Facebook.

FirebaseUI Auth proporciona una implementación lista para usar de las experiencias de autenticación completa, incluyendo una pantalla de inicio/registro estandarizada que soporta proveedores sociales e inicia sesión por correo electrónico y contraseña. Sólo debes configurar FirebaseUI con el proveedor deseado y luego comenzar el flujo desde tu actividad.

Una vez que el usuario inicia sesión a través de cualquier proveedor, puedes acceder al objeto Firebase User para obtener información como UID, correo electrónico y tokens de acceso para realizar operaciones de backend seguras.

También puedes configurar reglas de autorización en tiempo real en la base de datos Firebase para asegurar el acceso a los datos. Por ejemplo, solo los usuarios autenticados pueden leer/escribir documentos, y los campos de documento específicos solo son accesibles para ciertos usuarios.

Algunas prácticas recomendadas incluyen:

- Manejar correctamente varios estados de autenticación y sesiones expiradas
- Actualizar credenciales y tokens de acceso expirados
- Habilitar Email/Password Sign-in junto con social login para permitir más opciones
- Almacenar tokens de acceso en el Keychain seguro para evitar pérdidas
- Habilitar verificación de correo electrónico para mayor seguridad

Conclusiones

Podemos decier que Firebase y Kotlin proporcionan una excelente combinación para implementar autenticación de usuarios segura en Android de manera rápida y con muy poco código. Con múltiples opciones como autenticación social, multifactor y reglas de seguridad en la base de datos, Firebase maneja mucho de complejidad de backend.

En resumen, la combinación de Firebase y Kotlin ofrece a los desarrolladores una poderosa herramienta para implementar autenticación segura y versátil en aplicaciones Android, reduciendo la complejidad del backend y proporcionando una experiencia de usuario mejorada. La facilidad de implementación, las prácticas de seguridad y la variedad de opciones hacen de Firebase una opción robusta para la autenticación de usuarios en aplicaciones Android desarrolladas en Kotlin.