

Tecnologías y Desarrollo en Dispositivos Móviles

Apartado 13a: **Firestore**

Autores:

Miguel Ángel García Cumberras
Víctor M. Rivas Santos

¿Qué es Firebase?

- Firebase es una plataforma de desarrollo móvil en la nube de Google, disponible para diferentes plataformas (Android, iOS, web)

<https://firebase.google.com/?hl=es-419>

¿Qué servicios ofrece?

- **Base de datos Realtime NoSQL:** con datos almacenados en formato JSON
- **Autenticación:** servicio simplificado de inicio de sesión y gestión de usuarios
- **Almacenamiento:** para aplicaciones que requieran guardar archivos del usuario
- **Hosting:** un servicio básico de hosting, con SSL y HTTP2
- **Laboratorio de tests para Android:** un servicio que permite probar tests de aplicación en el entorno que queramos configurar

¿Qué servicios ofrece?

- **Informes sobre fallos:** para detectar errores de aplicaciones Android o iOS
- **Monitorización de rendimiento:** para detectar problemas con tiempos de carga en las apps
- **Notificaciones:** gestión de notificaciones programadas
- **Analytics,** etc.

Lenguajes compatibles

- JavaScript (tanto en el Frontend como con Node)
- iOS (Swift y Objective C)
- Android (Java y Kotlin)
- Python y aplicaciones Java
- C++ y Unity

Firestore Realtime Database

Base de datos NoSQL que almacena los datos en formato JSON.

Estos datos se sincronizan con cada uno de los clientes que se encuentran conectados en ese momento.

La idea es que, cuando un equipo de desarrollo crea una aplicación con los SDKs de Android, iOS o JavaScript de Firestore, todos esos clientes comparten una misma instancia dentro de esa base de datos NoSQL y las peticiones que hacen los usuarios reciben siempre como respuesta los datos actualizados para su petición en tiempo real.

Firestore Realtime Database

Ejemplo Android:

En build.gradle:

```
implementation 'com.google.firebase:firebase-database:16.0.1'
```

Creo una instancia de DatabaseReference

```
private DatabaseReference mDatabase;
```

```
// ...
```

```
mDatabase = FirebaseDatabase.getInstance().getReference();
```

Firestore Realtime Database

Ejemplo Android:

Operaciones básicas de escritura:

Creo una nueva clase que gestiona los datos

```
public class User {  
    public String username;  
    public String email;  
    public User() {  
        // Default constructor required for calls to DataSnapshot.getValue(User.class)  
    }  
    public User(String username, String email) {  
        this.username = username;  
        this.email = email;  
    }  
}
```


Firestore Realtime Database

Ejemplo Android:

Operaciones básicas de escritura:

Añado un nuevo usuario

```
private void writeNewUser(String userId, String name, String email) {  
    User user = new User(name, email);  
    FirebaseDatabase.getInstance().getReference().child("users").child(userId).setValue(user);  
}
```

Firestore Realtime Database

Ejemplo Android:

Detecto eventos en valores con `addValueEventListener()` o `addListenerForSingleValueEvent()`

```
ValueEventListener postListener = new ValueEventListener() {  
    @Override  
    public void onDataChange(DataSnapshot dataSnapshot) {  
        // Get Post object and use the values to update the UI  
        Post post = dataSnapshot.getValue(Post.class);  
        // ...  
    }  
    @Override  
    public void onCancelled(DatabaseError databaseError) {  
        // Getting Post failed, log a message  
        Log.w(TAG, "loadPost:onCancelled", databaseError.toException());  
        // ...  
    }  
};  
mPostReference.addValueEventListener(postListener);
```