

DESARROLLO DE VIDEOJUEGOS Y REALIDAD VIRTUAL CON UNITY 3D

Tools – Firebase – Database Realtime

1



FIREBASE — REALTIME DATABASE

- Instalación: descargar el SDK y descomprimir:
 - <https://firebase.google.com/docs/unity/setup?hl=es-419>
 - Importa el package correspondiente (DOTNET3): assets - import packages - Custom package a Realtime Database

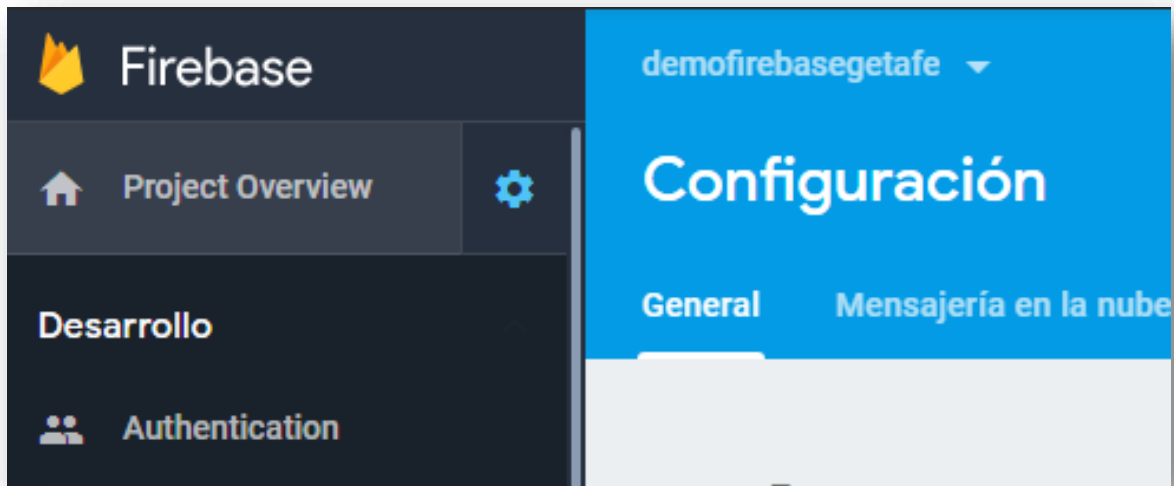


REFERENCIAS:

<https://firebase.google.com/docs/reference/js/firebase.database>

FIREBASE — REALTIME DATABASE

- Crear un proyecto Firebase
 - <https://console.firebase.google.com>
 - Agregar Firebase para Android (en Project Overview)
 - Descargar google-services.json y arrastrar al proyecto



FIREBASE — REALTIME DATABASE

- Importar los componentes:

```
using Firebase;  
using Firebase.Database;  
using Firebase.Unity.Editor;
```

- Declarar la referencia a la base de datos:

```
DatabaseReference reference;
```

- Obtener la referencia a la base de datos:

```
FirebaseApp.DefaultInstance.SetEditorDatabaseUrl("https://xxx.  
firebaseio.com/");
```

```
reference = FirebaseDatabase.DefaultInstance.RootReference;
```

FIREBASE — REALTIME DATABASE

- ESCRITURA SIMPLE:

```
reference.Child("x").SetValueAsync(x);
```

FIREBASE — REALTIME DATABASE

- LECTURA SIMPLE:
- Vincular el método de tratamiento de cambios sobre una rama del árbol (en este caso a la rama x):

```
FirebaseDatabase.DefaultInstance.GetReference("x").ValueChanged += GestionarPosicionX;
```

- Programar el método de gestión:

```
void GestionarPosicionX(object sender, ValueChangedEventArgs args)
{
    if (args.DatabaseError != null)
    {
        Debug.LogError(args.DatabaseError.Message);
        return;
    }
    print(args.Snapshot.GetValue(true));
}
```

FIREBASE — REALTIME DATABASE

- ESCRITURA CON CREACIÓN DE CLAVE:

```
private void AgregarPuntuacion(string userId, int puntos)
{
    string key = reference.Child("puntuacion").Push().Key;
    Dictionary<string, object> entrada;
    Dictionary<string, object> puntuacion;
    entrada = new Dictionary<string, object>();
    puntuacion = new Dictionary<string, object>();
    puntuacion.Add(userId, puntos);
    entrada["/puntuacion/" + key] = puntuacion;
    reference.UpdateChildrenAsync(entrada);
}
```

FIREBASE — REALTIME DATABASE

■ LECTURA DE COLECCIÓN

```
void Gestionar(object sender, ValueChangedEventArgs args)
{
    if (args.DatabaseError != null)
    {
        Debug.LogError(args.DatabaseError.Message);
        return;
    }
    string clave = args.Snapshot.Key; //Clave
    long numeroItems = args.Snapshot.ChildrenCount; //Número de hijos
    string jsonValue = args.Snapshot.GetRawJsonValue(); //Datos en formato JSON
    IEnumerable<DataSnapshot> puntuaciones = args.Snapshot.Children;
    foreach(var elemento in puntuaciones)
    {
        IEnumerable<DataSnapshot> elementos = elemento.Children;
        foreach (var puntuacion in elementos)
        {
            print(puntuacion.Key);
            print(puntuacion.Value);
        }
    }
}
```


FIREBASE — REALTIME DATABASE

- BORRARD

```
reference.Child("prueba").RemoveValueAsync();
```