

Temas selectos III

(Realidad virtual y Realidad amentada)

GRUPO: 01

Práctica No. 1
Configuración inicial

Mendoza Anaya Aldair Israel

PROFESOR:
Ing. Arturo Pérez De La Cruz

20 de febrero de 2024

Manual para la Creación de Proyectos en Unity

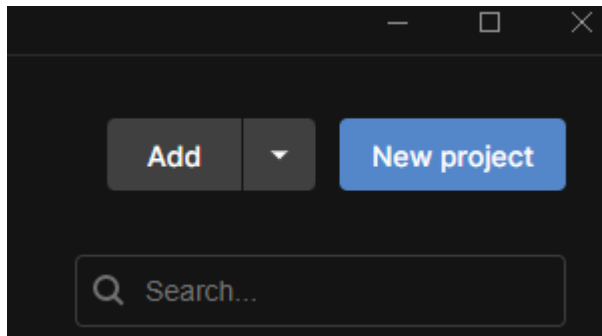
Paso 1: Crear un Nuevo Proyecto

Abre el software Unity Hub.

Haz clic en el botón "Nuevo" en la esquina superior derecha para iniciar la creación de un nuevo proyecto.

Selecciona la pestaña "3D" en la parte superior del panel de proyectos para elegir una plantilla adecuada para tu proyecto.

Elige la plantilla "3D Core" para comenzar con una configuración básica para proyectos en 3D.



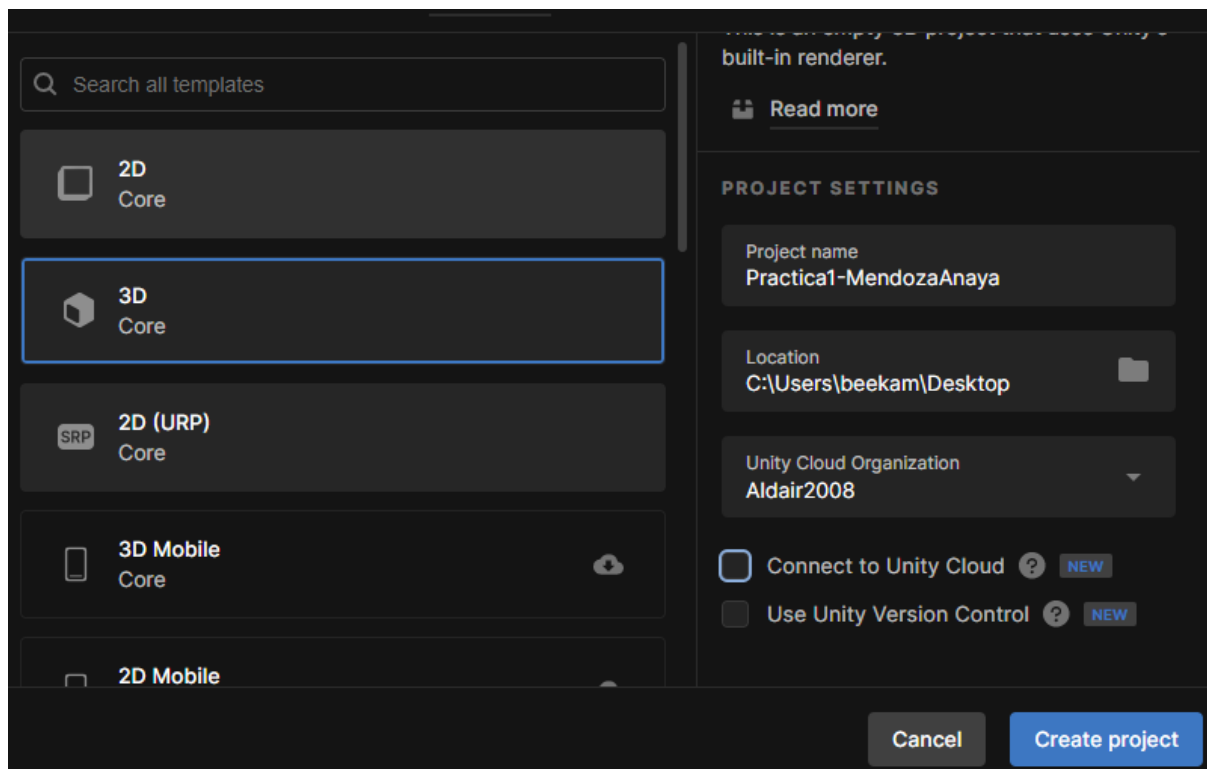
Paso 2: Configuración del Proyecto

Asigna un nombre descriptivo a tu proyecto en el campo "Nombre del Proyecto".

Desmarca la opción "Conectar a Unity Cloud" si no deseas utilizar los servicios en la nube de Unity.

Selecciona la ubicación en tu disco duro donde deseas almacenar el proyecto.

Haz clic en el botón "Crear" para iniciar el proceso de creación del proyecto.

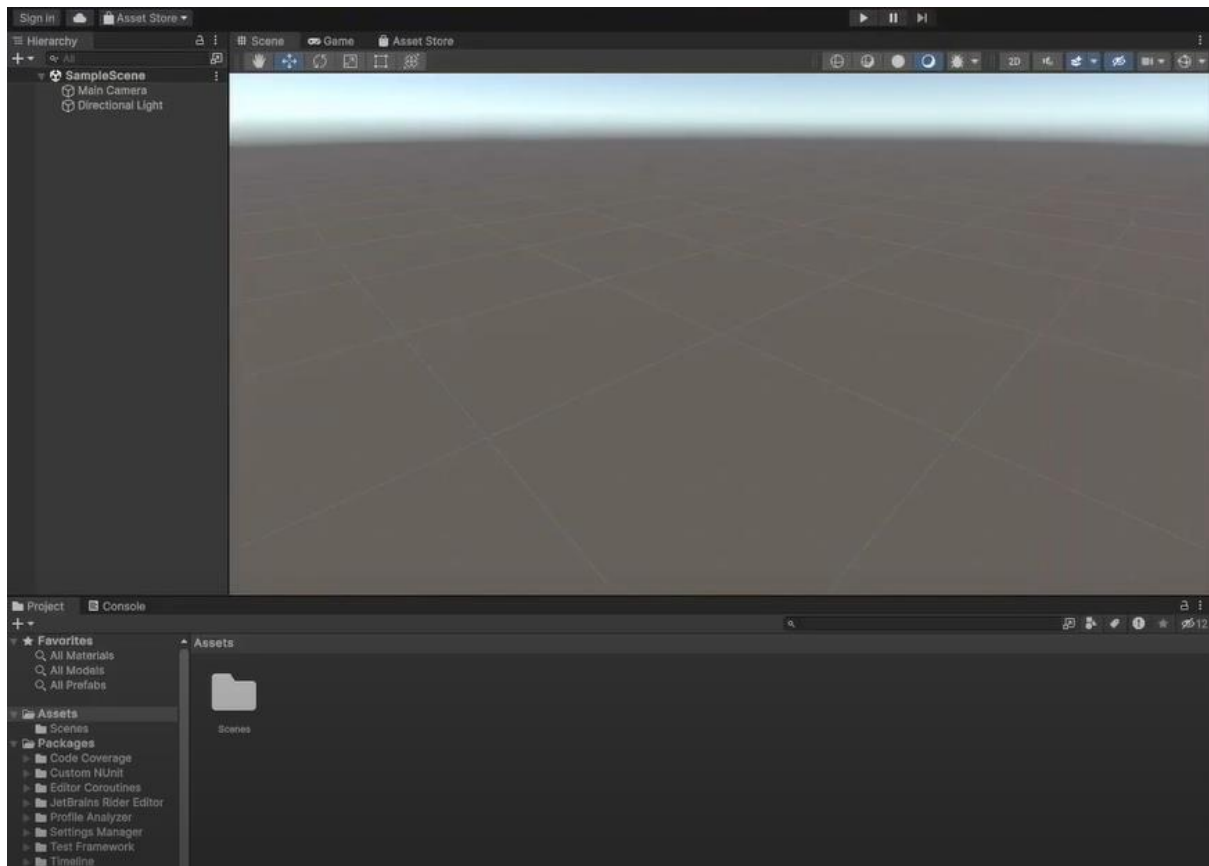


Paso 3: Esperar a que se Cree el Espacio de Trabajo

Unity comenzará a crear el espacio de trabajo según la configuración seleccionada.

Dependiendo de la velocidad de tu computadora y la complejidad del proyecto, este proceso puede llevar algunos momentos.

Una vez completada la creación del proyecto, Unity abrirá automáticamente el espacio de trabajo para que comiences a trabajar en tu proyecto.



Paso 4: Abrir el Package Manager y Agregar Google Cardboard

En la pestaña de "Windows" en la barra de menú de Unity, selecciona "Package Manager".

Se abrirá una colección de paquetes. Haz clic en "Add package from disk..." para abrir el explorador de archivos.

Busca la carpeta "cardboard-xr-plugin-master" en tu sistema y selecciona el archivo "package.json".

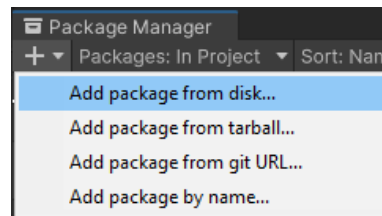
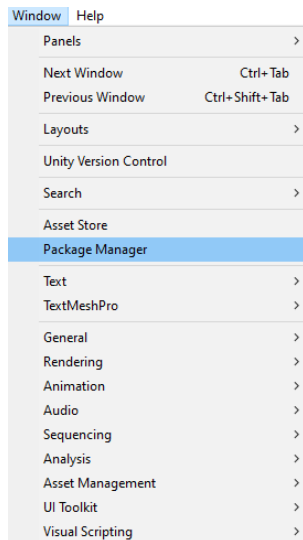
Una vez agregado el paquete, busca "Google Cardboard" en la lista y haz clic en él para seleccionarlo.

Ve a la pestaña "Samples" y haz clic en "Import" para importar los archivos de muestra.

Observa que varios elementos se agregarán a tu proyecto, lo que te permitirá integrar Google Cardboard en tu aplicación o juego.

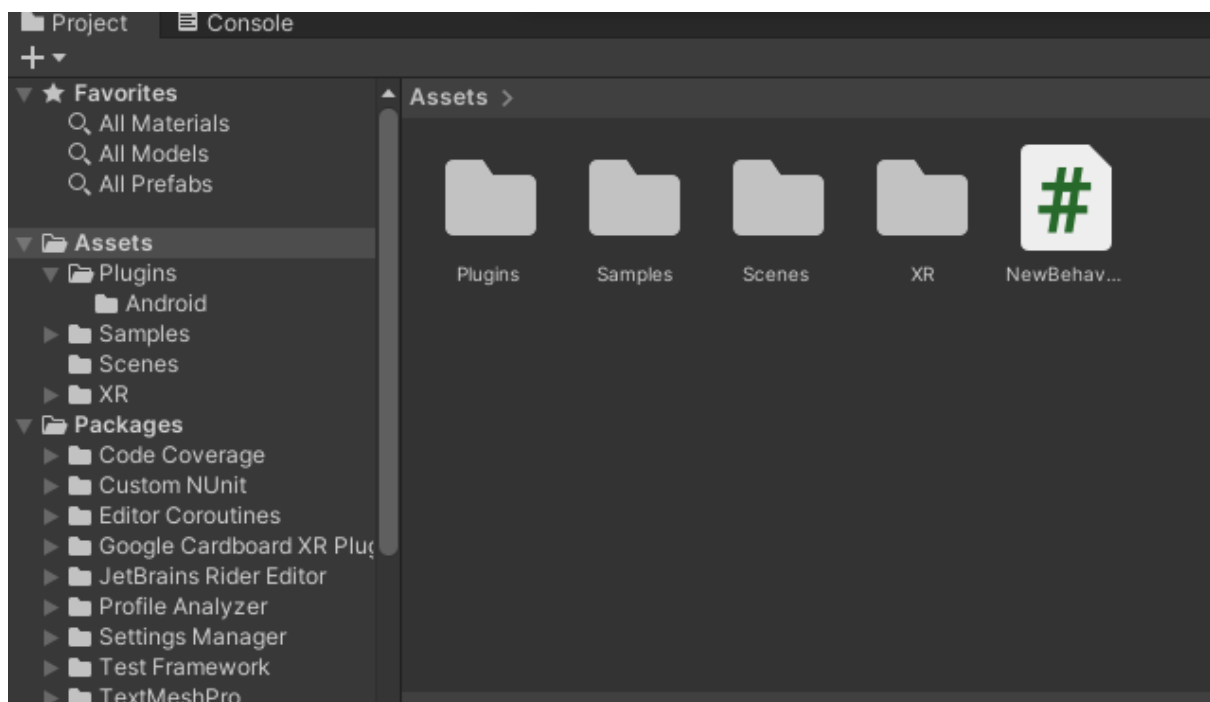
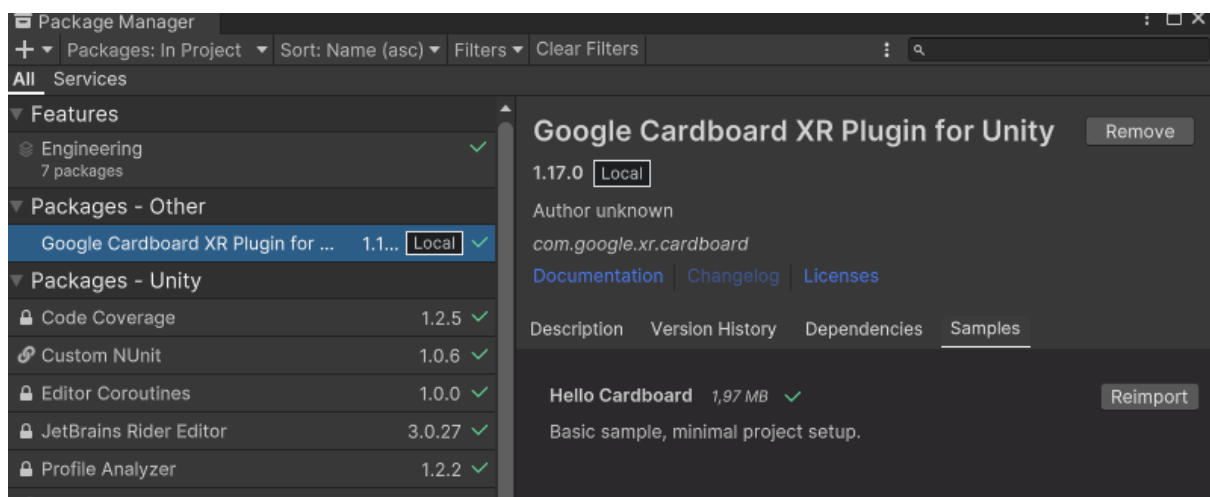
Practica 1
20/2/2024

Mendoza Anaya Aldair Israel
317222049



Este equipo > Escritorio > cardboard-xr-plugin-master >

ra carpeta	
Nombre	Fecha de modificación
.github	9/8/2022 18:15
Editor	9/8/2022 18:15
Resources	9/8/2022 18:15
Runtime	9/8/2022 18:15
Samples~	9/8/2022 18:15
package.json	9/8/2022 18:15



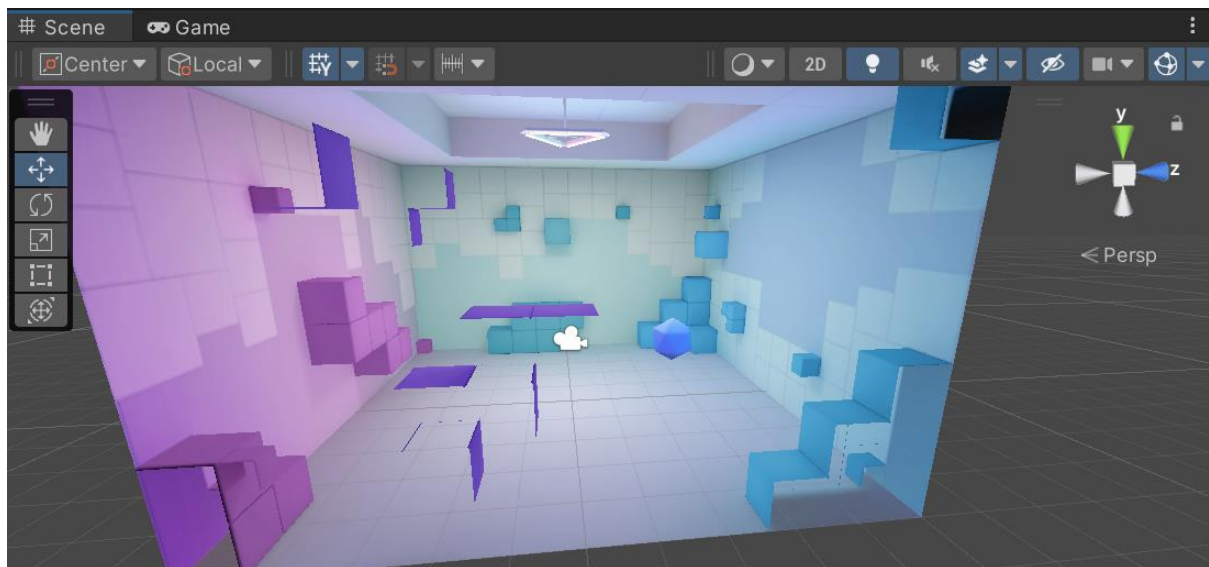
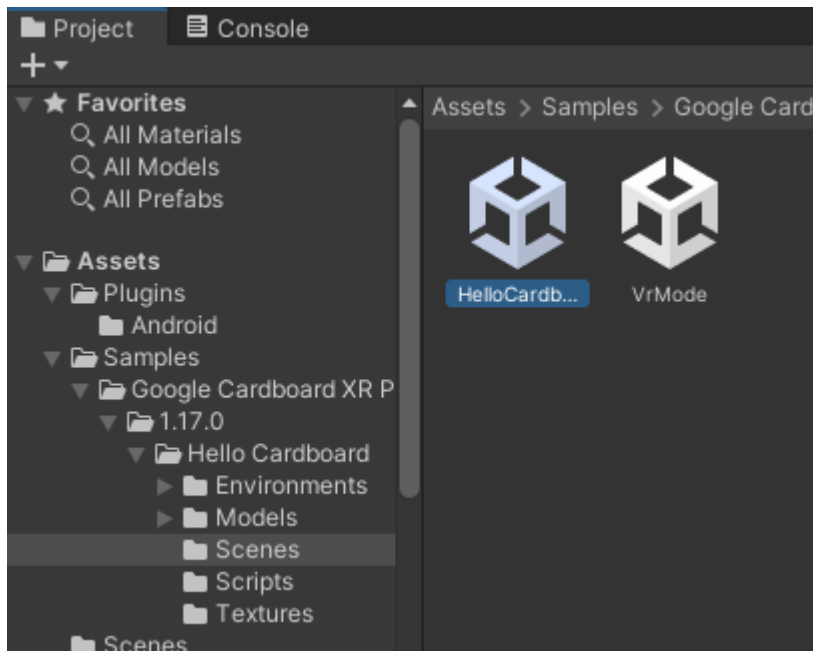
Paso 5: Abrir la Escena Prediseñada

En el panel de "Proyectos" de Unity, navega a la siguiente dirección: Assets\Samples\Google Cardboard XR Plugin for Unity\1.17.0\Hello Cardboard\Scenes.

Localiza el archivo "HelloCardboard.unity" dentro de la carpeta de escenas.

Haz doble clic en "HelloCardboard.unity" desde la consola de Unity para abrir esta escena prediseñada en el editor.

Al abrir la escena, podrás explorar el contenido prediseñado y comenzar a trabajar en ella para adaptarla a tus necesidades o utilizarla como punto de partida para tu proyecto de realidad virtual con Google Cardboard.



Paso 6: Configurar la Plataforma de Destino y Escena

En la barra de menú de Unity, haz clic en "File" (Archivo).

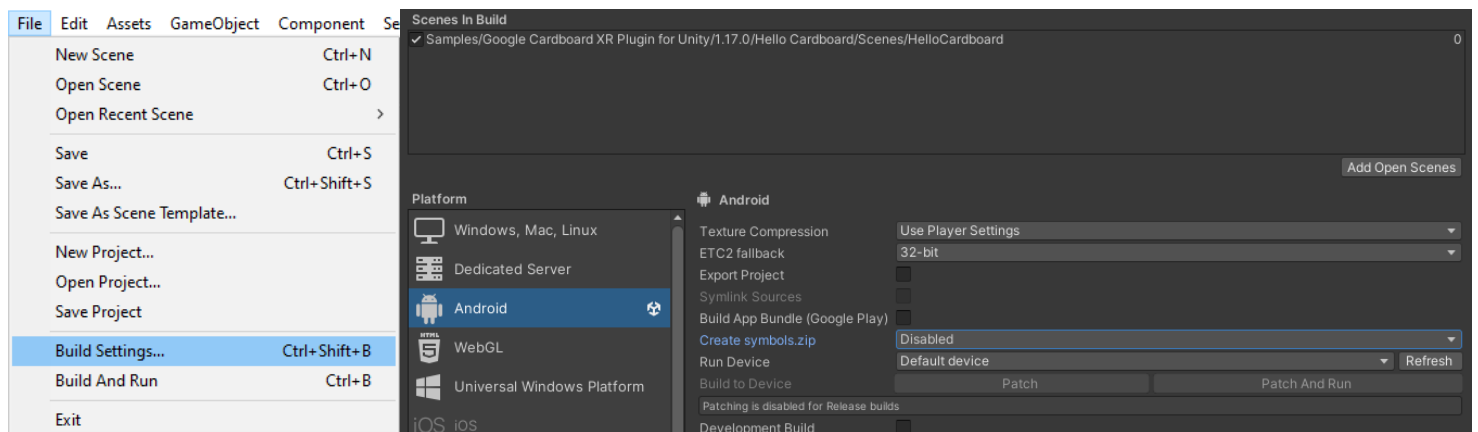
Selecciona "Build Settings" (Configuración de Compilación) en el menú desplegable.

Se abrirá una ventana de configuración. En la sección "Platform", elige "Android" como la plataforma de destino para tu aplicación.

Si aún no has añadido la escena actual al proceso de compilación, haz clic en "Add Open Scenes" (Añadir Escenas Abiertas) para incluir la escena actual en la compilación de tu proyecto.

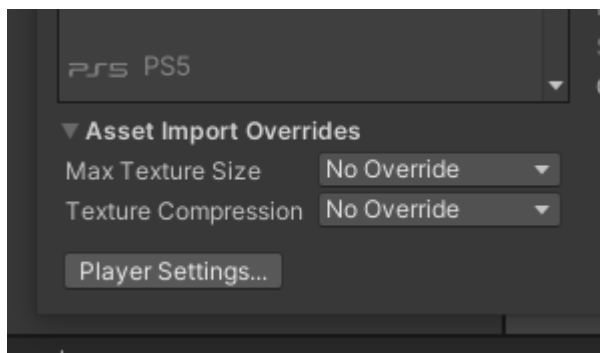
Asegúrate de que la escena "HelloCardboard.unity" esté listada en la sección "Scenes In Build" (Escenas en Compilación).

Si no está presente, puedes arrastrar y soltar la escena desde el panel de "Proyectos" hacia la lista de escenas en compilación.



Paso 7: Configuraciones Adicionales en Player Settings

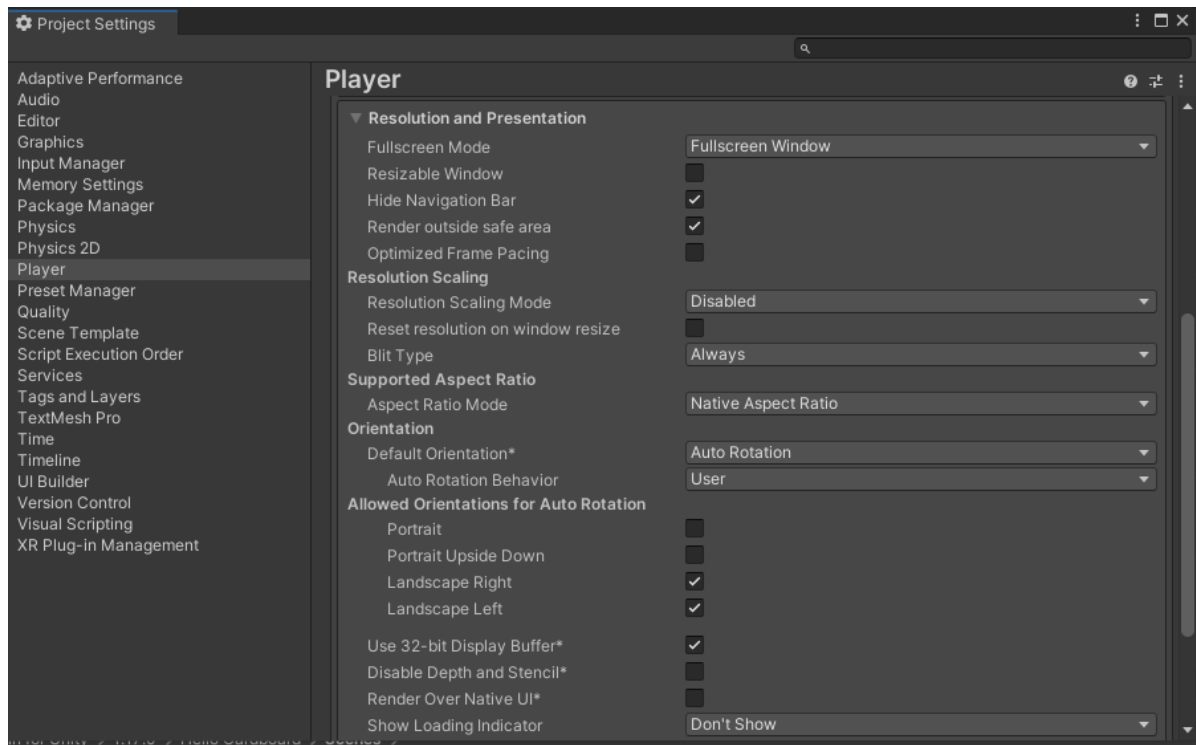
En la misma ventana de "Build Settings" (Configuración de Compilación), haz clic en el botón "Player Settings" (Configuración del Jugador) ubicado en la parte inferior izquierda. Esto abrirá las configuraciones del jugador en el panel de Inspector.



Ve a la sección "Resolution and Presentation" (Resolución y Presentación) dentro de "Project Settings > Player".

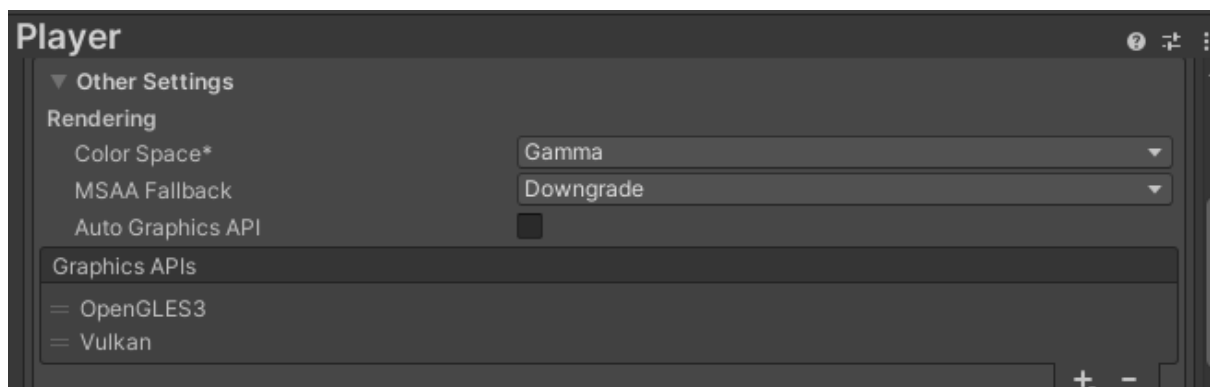
Establece la "Orientación predeterminada" en "Horizontal Left" o "Landscape Right" para garantizar que la aplicación se ejecute en modo apaisado.

Inhabilita "Optimized Frame Pacing" para desactivar la sincronización de fotogramas optimizada.



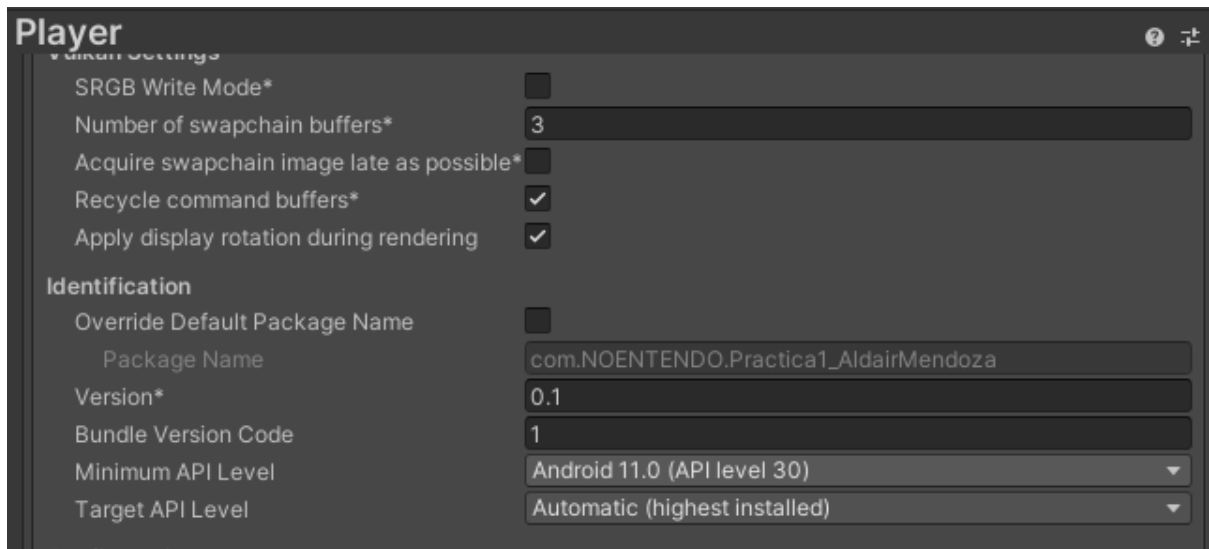
Dirígete a la sección "Other Settings" (Otras Configuraciones) en las configuraciones del jugador.

Elige "OpenGL ES3" en las "Graphics APIs" (APIs de Gráficos) para seleccionar la API gráfica adecuada.



Selecciona "Android 11.0 'Oreo' (API level 30)" como la "Minimum API Level" (Nivel Mínimo de API) requerida para tu aplicación.

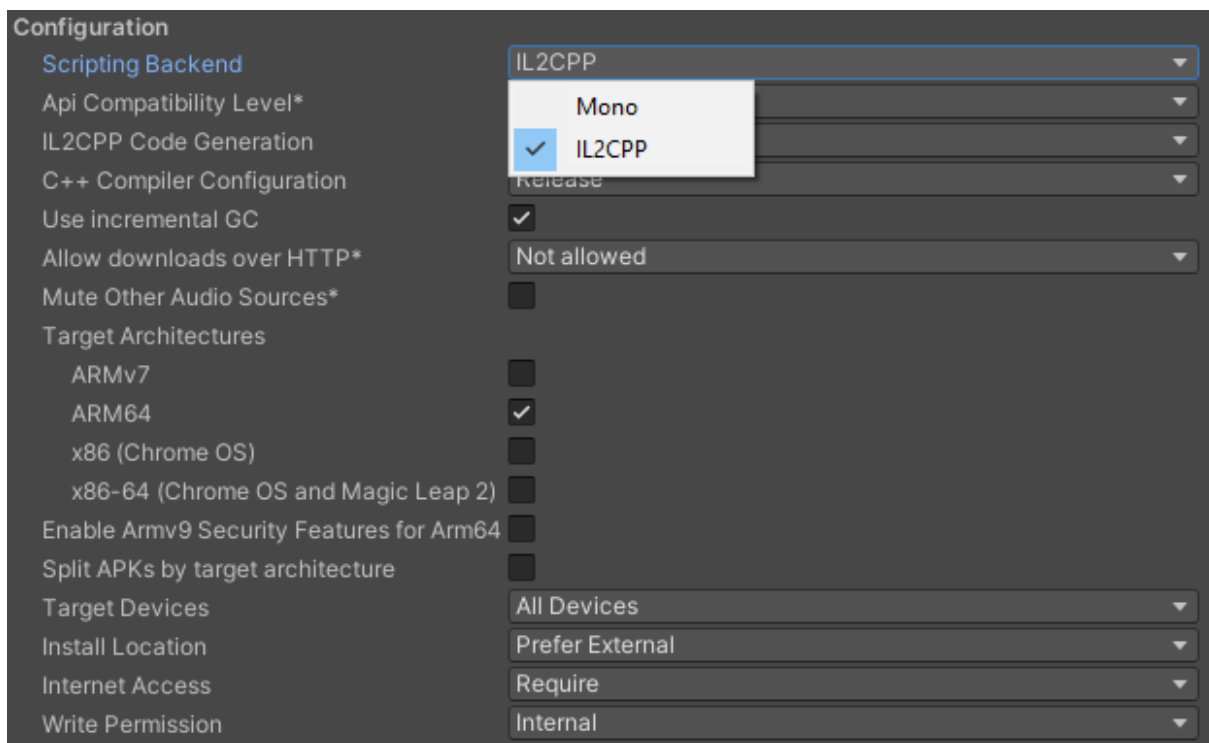
Selecciona "Automatic (highest installed)" (Automático - la más alta instalada) en "Target API Level" (Nivel de API Objetivo) para utilizar la API de Android más alta instalada en el dispositivo del usuario.



Selecciona "IL2CPP" en "Scripting Backend" (Backend de Secuencias de Comandos) para utilizar este método de compilación de scripts.

Para elegir las arquitecturas deseadas para la compilación de la aplicación, selecciona "ARMv7", "ARM64", o ambos en "Target Architectures" (Arquitecturas de Destino).

Selecciona "Require" (Requerir) en "Internet Access" (Acceso a Internet) si tu aplicación necesita acceso a internet para funcionar correctamente.

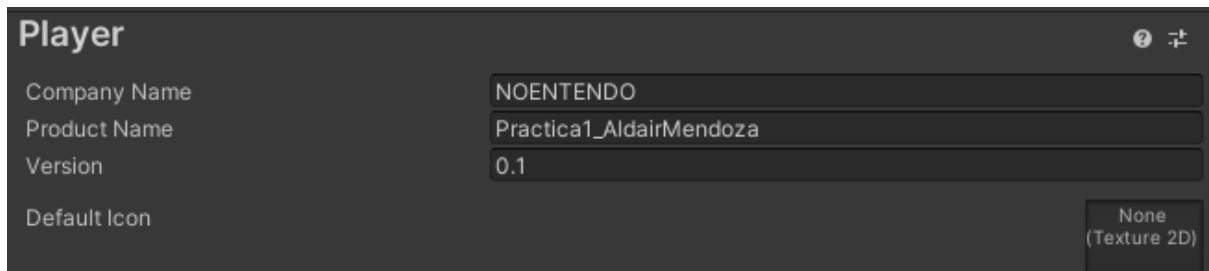


Paso 8: Especificar el Dominio de tu Empresa en el Package Name

En las configuraciones del jugador, ve a la sección "Other Settings" (Otras Configuraciones).

Especifica el dominio de tu empresa como prefijo en el nombre del paquete.

Después del dominio de tu empresa, añade el nombre de tu proyecto seguido de "_NombredelAlumnoPractica" donde "N" es el número de práctica y "NombredelAlumno" es tu nombre.



Paso 9: Agregar Dependencias y Configuraciones Gradle

En las configuraciones del jugador, ve a la sección "Build" (Compilación).

Selecciona "Custom Main Gradle Template" (Plantilla Personalizada de Gradle Principal) y "Custom Gradle Properties Template" (Plantilla Personalizada de Propiedades de Gradle).

Abre el archivo Assets/Plugins/Android/mainTemplate.gradle.

Dentro de la sección de dependencias, agrega las siguientes líneas:

`implementation 'androidx.appcompat:appcompat:1.4.2'`

`implementation 'com.google.android.gms:play-services-vision:20.1.3'`

`implementation 'com.google.android.material:material:1.6.1'`

`implementation 'com.google.protobuf:protobuf-javalite:3.19.4'`

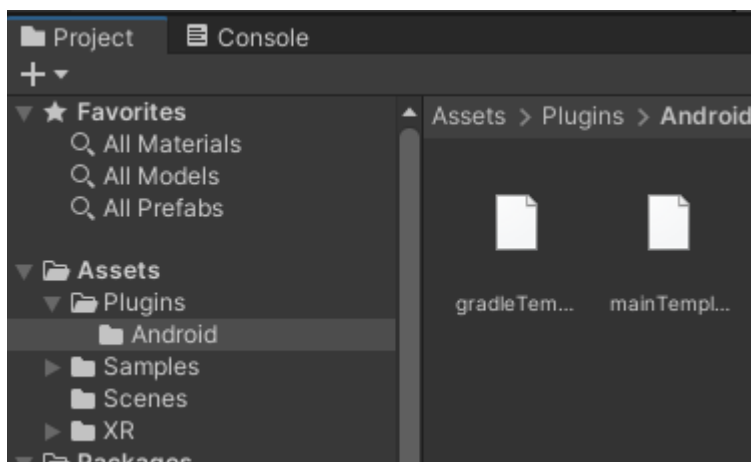
Abre el archivo Assets/Plugins/Android/gradleTemplate.properties.

Agrega las siguientes líneas al archivo:

`android.enableJetifier=true`

`android.useAndroidX=true`

Guarda los cambios realizados en ambos archivos.



```
1  org.gradle.jvmargs=-Xmx**JVM_HEAP_SIZE**M
2  org.gradle.parallel=true
3  unityStreamingAssets=**STREAMING_ASSETS**
4  android.enableJetifier=true
5  android.useAndroidX=true
6  **ADDITIONAL_PROPERTIES**

1  apply plugin: 'com.android.library'
2  **APPLY_PLUGINS**
3
4  dependencies {
5      implementation fileTree(dir: 'libs', include: ['*.jar'])
6      implementation 'androidx.appcompat:appcompat:1.4.2'
7      implementation 'com.google.android.gms:play-services-vision:20.1.3'
8      implementation 'com.google.android.material:material:1.6.1'
9      implementation 'com.google.protobuf:protobuf-javalite:3.19.4'
10  **DEPS**
11
12  android {
13      ndkPath "**NDKPATH**"
14
15      compileSdkVersion **APIVERSION**
16      buildToolsVersion '**BUILDTOOLS**'
17
18      compileOptions {
19          sourceCompatibility JavaVersion.VERSION_11
20          targetCompatibility JavaVersion.VERSION_11
21      }
22
23      defaultConfig {
24          minSdkVersion **MINSDKVERSION**
25          targetSdkVersion **TARGETSDKVERSION**
26          ndk {
27              abiFilters **ABIFILTERS**
28          }
29          versionCode **VERSIONCODE**
30          versionName '**VERSIONNAME**'
31          consumerProguardFiles 'proguard-unity.txt'**USER_PROGUARD**
32      }
33
34      lintOptions {
35          abortOnError false
36      }
37
38      aaptOptions {
39          noCompress = **BUILTIN_NOCOMPRESS** + unityStreamingAssets.tokenize(', ')
40          ignoreAssetsPattern = "!svn:!git:!ds_store:!*.scc:!CVS:!thumbs.db:!picasa.ini:!*~"
41      }**PACKAGING_OPTIONS**
42  }
43  **IL_CPP_BUILD_SETUP**
44  **SOURCE_BUILD_SETUP**
45  **EXTERNAL_SOURCES**
```

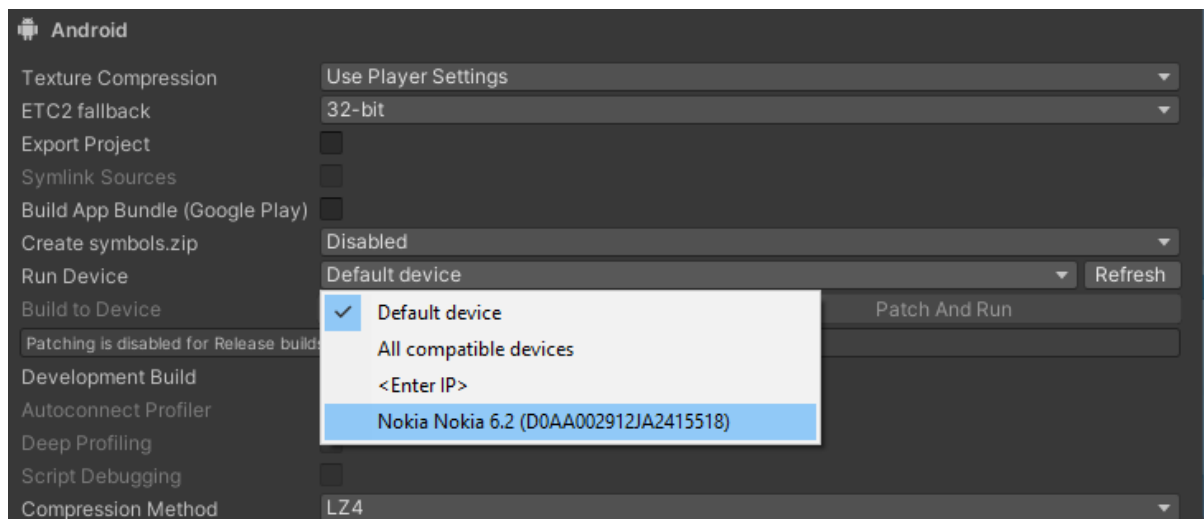
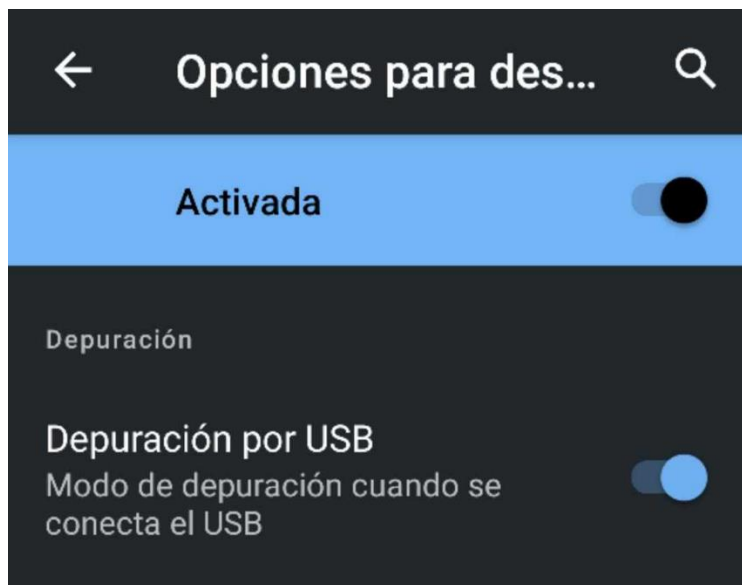
Paso 10: Configurar el Dispositivo Móvil y Ejecutar la Aplicación

Coloca tu teléfono en modo desarrollador siguiendo las instrucciones específicas para tu modelo de dispositivo y versión de Android. Por lo general, esto implica ir a la sección "Acerca del Teléfono" en la configuración del dispositivo, encontrar el número de compilación y tocarlo varias veces hasta que se habilite el modo desarrollador.

Una vez que el modo desarrollador esté habilitado, activa la opción "Depuración por USB" en la configuración de desarrollador de tu dispositivo.

Conecta tu dispositivo móvil a tu computadora utilizando un cable USB.

En "Run Device" (Ejecutar Dispositivo), selecciona tu dispositivo móvil conectado.



Haz clic en "Build and Run" (Compilar y Ejecutar) para comenzar el proceso de compilación y despliegue de la aplicación en tu dispositivo móvil.

Espera a que Unity compile la aplicación y la instale en tu dispositivo móvil. Una vez que la instalación haya finalizado, la aplicación se ejecutará automáticamente en tu dispositivo.

