




< [Back to Unity](#)

# Set up Unity for VR development



Updated: Apr 28, 2026

Learn how to set up and configure your Unity 3D development environment to build VR apps for Meta Quest headsets. This tutorial provides setup instructions for the following components:

- **XR provider:** Enables Unity to interface with Meta Quest devices and manages device tracking, input, and rendering.
- **Meta XR Core SDK:** Provides tools and components to optimize and enhance VR development for Meta devices.
- **Meta XR Platform SDK:** Enables identity management, entitlement checks, and platform features such as matchmaking, in-app purchases, achievements, and social interactions. Requires an App ID from the [Meta Horizon Developer Dashboard](#) .
- **Project Setup Tool:** Manages settings and dependencies for your Unity project.

If you're new to Unity development on Quest, check out the [Hello World guide](#) to create your first VR app.

## Prerequisites

Before starting this tutorial, ensure you have the following:

## Hardware requirements

- Development machine running one of the following:
  - Windows 10+ (64-bit)
  - macOS 10.10+ (x86 or ARM)

## Account requirements [↗](#)

- Unity ID: [Create or log in to your Unity account](#) [↗](#)

## Step 1: Add the Android Build Support module [↗](#)

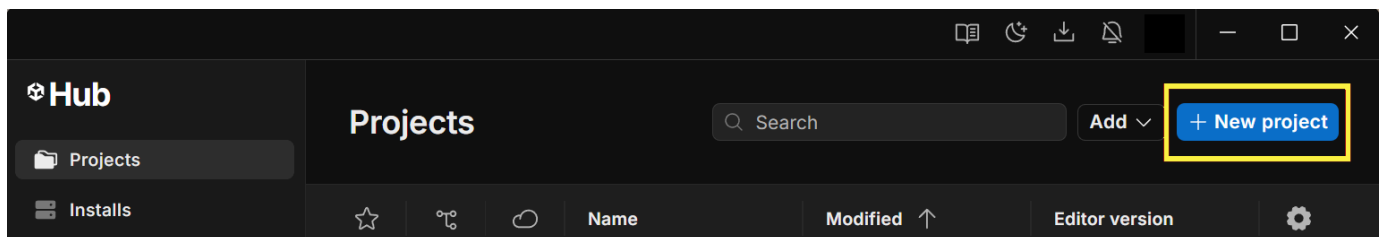
Meta Quest devices run on Android OS and require the Unity **Android Build Support** module to build and deploy applications. Follow the installation steps below to add the module and required components:

1. Open Unity Hub.
2. Click **Installs** in the left navigation bar, which displays your Unity Editor versions.
3. If the pane is empty, click **Install Editor** and choose a compatible version. Meta Quest devices require 2022.3.15f1 or later (6.1 or later recommended). Click **Install** to start the installation process.
4. When complete, click **Installs** from the left navigation bar to confirm.
5. Next, select **Manage > Add modules** from the menu next to the Unity installation.
6. Select the following components and then click **Install**:
  - Android Build Support
  - OpenJDK
  - Android SDK & NDK Tools

## Step 2: Create a Unity project [↗](#)

Follow the steps below to create a new project in Unity Hub:

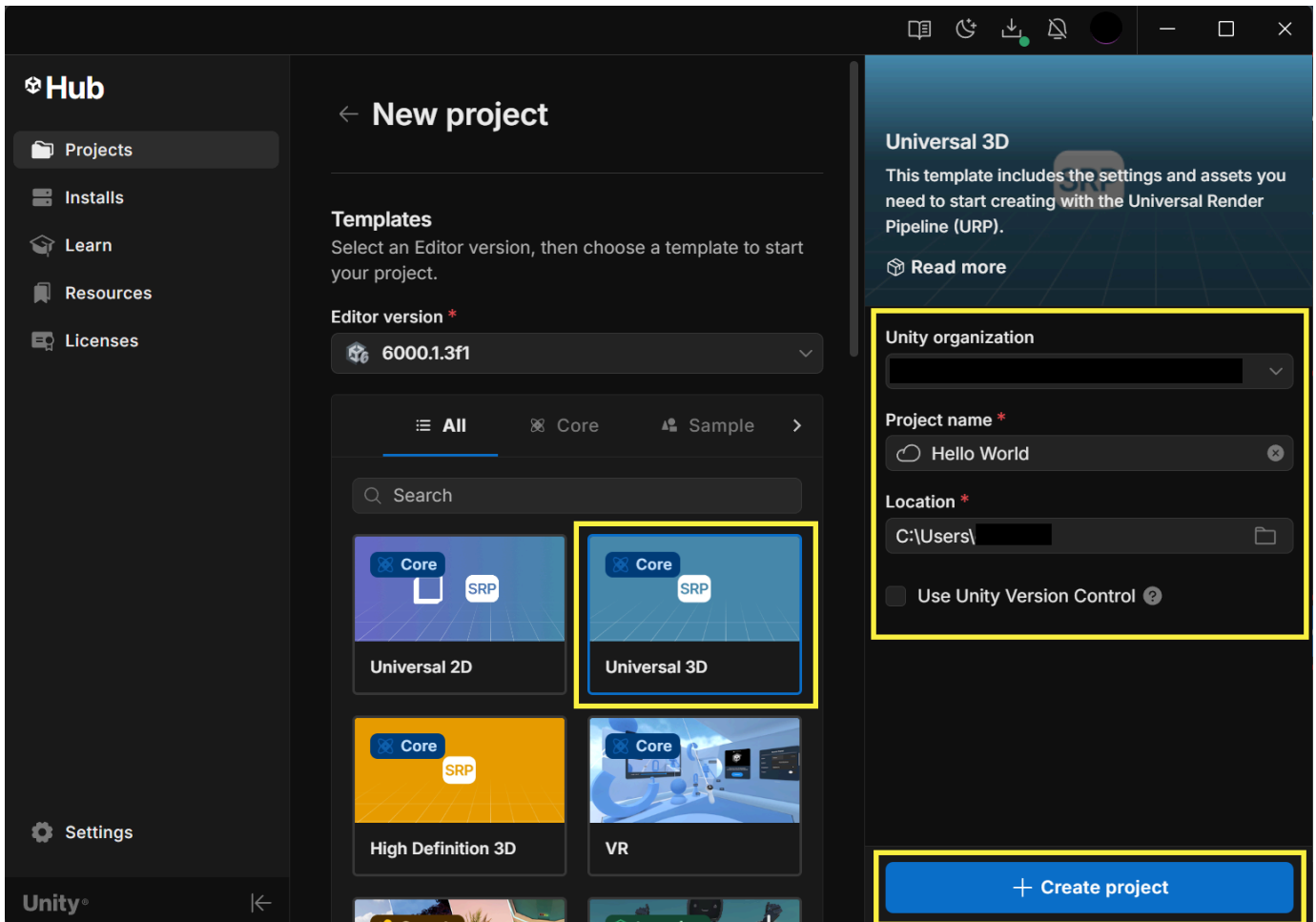
1. Select **Projects** in the left navigation bar, then click **New project**.



2. Select Unity Editor version **6.1** or later.

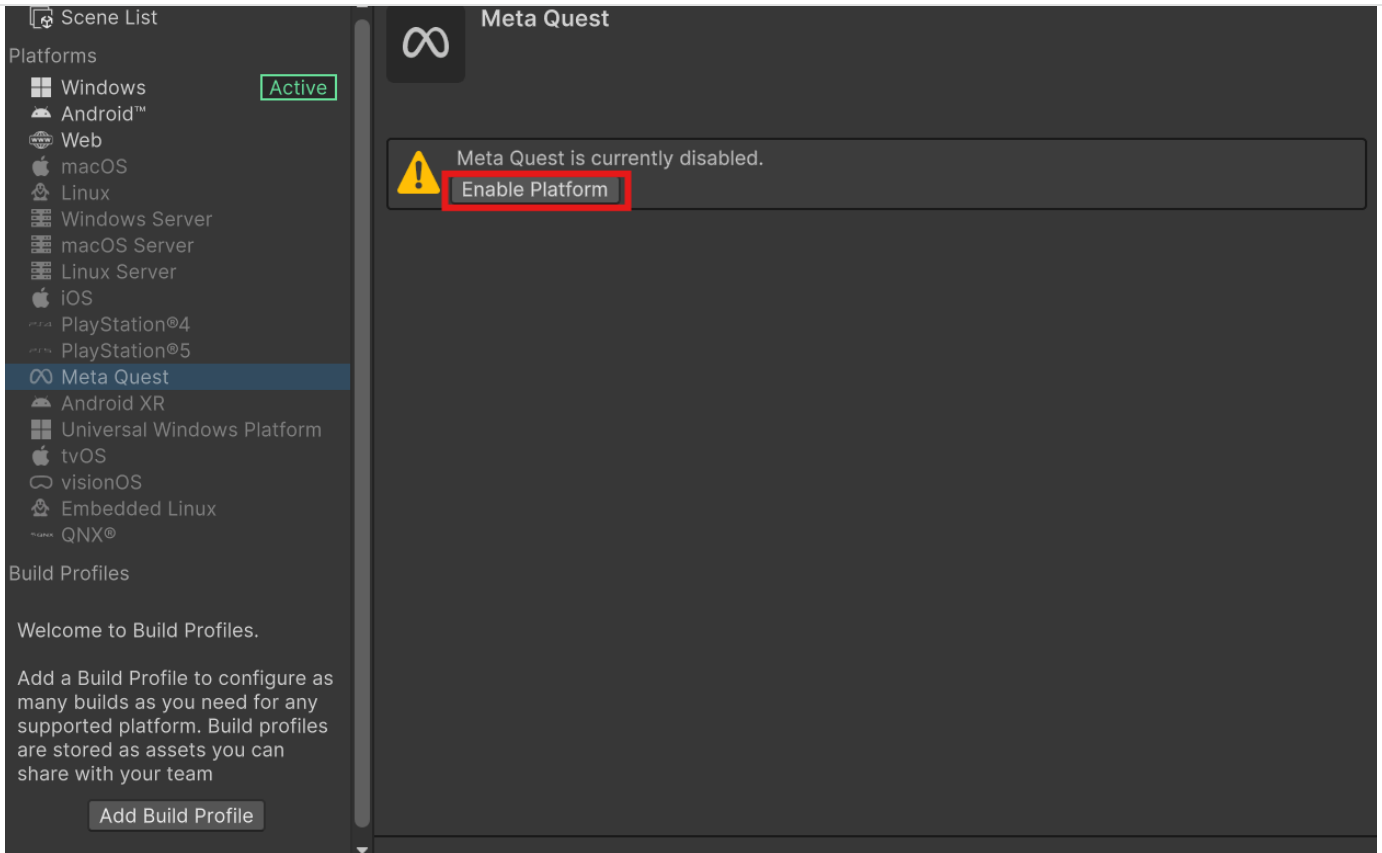
4. Enter your project name, a save location, and a Unity organization.

5. Click Create project.



## Set the build platform [🔗](#)

1. In the Unity Editor menu, select **File > Build Profiles**.
2. Under **Platforms**, select **Meta Quest** and click **Enable Platform**. If you already installed the profile, click the **Switch Platform** button.



3. If prompted to install `com.unity.xr.openxr`, click **Install**.

## Unity versions prior to 6.1

If the Meta Quest platform is absent from your version of Unity, select the **Android** build platform.

For more information about Unity build profiles, see [Build Configuration Overview](#).

## Step 3: Install an XR provider plugin [🔗](#)

The XR provider plugin lets Unity support XR devices, including headsets. Choose the provider that supports your project's requirements.

The following provider plugins support Meta Quest devices:

- Requires Unity 6 or later, Meta XR SDKs v74 or later.
- [Oculus XR Plugin](#) [↗](#) (deprecated)
  - Recommended Version: 4.5.1
  - Requires Unity 2022 or later, Meta XR SDKs v73 or earlier.

## Oculus XR Plugin is deprecated

The Oculus XR Plugin is deprecated and scheduled for removal. Use the [Unity OpenXR Plugin](#) instead.

For more information about XR vendor plugins, see [XR Plugin Management for Meta Quest](#).

### Option A: Install the Unity OpenXR Plugin [↗](#)

All new projects should install the Unity OpenXR Plugin to get the benefit of the latest features and optimizations.

To install the OpenXR provider plugin:

Project Settings window.

3. Select **XR Plug-in Management** in the **Project Settings** window.
4. If **XR Plug-in Management** is not installed, select **Install XR Plugin Management**.
5. In the **Windows, Mac, Linux settings** tab, select **OpenXR**.
6. In the **Android, Meta Quest settings** tab, select **OpenXR**.
7. Close the **Project Settings** window.
8. From the top menu of the Unity Editor, navigate to **Window > Package Management > Package Manager** to open the Unity Package Manager window.
9. On the left-hand navigation menu, make sure **In Project** is highlighted. Underneath **Search in Project**, expand the **Packages - Unity** dropdown.
10. In the list of packages in the left menu, verify that there is a check mark beside **OpenXR Plugin**, indicating that the package is installed.

For more information on how to setup the OpenXR Plugin, see the [Unity OpenXR Plugin Settings documentation](#).

### Option B: Install the Oculus XR Plugin (deprecated)

For projects built on Unity versions < 6 that use Meta XR SDKs < v74, install the Oculus XR Plugin.


To install the Oculus XR provider plugin:

1. Open any 3D project in the Unity Editor (version 2022 or 2023).
2. From the top menu of the Unity Editor, navigate to **Edit > Project Settings** to open the **Project Settings** window.
3. Select **XR Plug-in Management** in the **Project Settings** window.
4. If **XR Plug-in Management** is not installed, select **Install XR Plugin Management**.
5. In the **Windows, Mac, Linux settings** tab, select **Oculus**.
6. In the **Android settings** tab, select **Oculus**.
7. Close the **Project Settings** window.
8. From the top menu of the Unity Editor, navigate to **Window > Package Management > Package Manager** to open the Unity Package Manager window.



plugin, indicating that the package is installed.

**Note:** Installing an XR plugin does not download and import the Meta XR SDKs in to your project. Make sure you [add the Meta XR SDKs](#) you need from the [Unity Asset Store](#) .

## Step 4: Import the Meta XR Core SDK


Meta develops and maintains several Unity SDKs for VR and extended reality (XR) app development. If you are just getting started, add the [Meta XR Core SDK](#)  from the Unity Asset Store. This SDK includes the most essential components and assets for development.

Follow these steps to add the Meta XR Core SDK to your project:


1. Go to the [Meta Quest Unity Asset Store](#) , and sign in using your Unity credentials.
2. Select or navigate to [Meta XR Core SDK](#) .
3. Click **Add to My Assets** to add the package to your Unity account's assets.
4. Click **Open in Unity** to open the **Package Manager** window in the Unity Editor.
5. Enter your Unity credentials if prompted.
6. In the **Package Manager** window, in the upper-right side of the window, click **Install** to install the SDK.
7. If prompted, click **Enable** for the Meta XR Feature Set to ensure the utilities function as expected.
8. If Unity prompts you to restart the Unity Editor, select **Restart Editor**.

For more details about Meta XR packages, see [Meta XR Packages reference page](#).

## Step 5: Import the Meta XR Platform SDK

The [Meta XR Platform SDK](#)  provides access to platform features essential for Meta Quest development, including identity management, entitlement checks, and social features. You need this SDK and an App ID to unlock features such as user authentication, matchmaking, in-app purchases, achievements, and cloud storage.

Follow these steps to add the Meta XR Platform SDK to your project:

1. Go to the [Meta XR Platform SDK](#)  page on the Unity Asset Store, and sign in using your Unity credentials.
2. Click **Add to My Assets** to add the package to your Unity account's assets.

5. In the **Package Manager** window, click **Install** to install the SDK.

6. If Unity prompts you to restart the Unity Editor, select **Restart Editor**.

After installing the SDK, you need an App ID to initialize it. To get your App ID, create an app on the [Meta Horizon Developer Dashboard](#) and copy the App ID from the API page. For detailed instructions, see [Set up a platform app for Unity development](#).

For more details about Platform Solutions features, see [Platform Solutions overview](#).

## Step 6: Confirm OpenXR features are enabled

You can enable and disable specific OpenXR features at build time for testing your app.

For this project, make sure you enable the following features to access them on your Meta Quest headset:

- Meta XR Feature
- Meta XR Foveation
- Meta XR Subsampled Layout

To locate and enable these features:

1. Navigate to **Project Settings > XR Plug-in Management > OpenXR** and select the **Android** tab.
2. Select **Meta XR** in the **OpenXR Feature Groups** section to display the list of Meta XR features.

## Step 7: Manage project dependencies using the Project Setup Tool

The **Project Setup Tool** is a Unity Editor extension that quickly configures and resolves dependency issues in Unity projects for Meta VR development. Follow the steps below to fix dependency issues in your project:

1. Select **Meta > Tools > Project Setup Tool** in the Unity Editor. Alternatively, click **Meta XR Tools** at the top of the Unity Editor, and select **Project Setup Tool** from the pop-up menu.
2. To resolve all outstanding issues for the Android platform, click **Fix All**.
3. To apply all recommended settings for the Android platform, click **Apply All**.

Your project is now ready for Quest development.

## Learn more [↗](#)

To learn more about project setup, see the following resources:

- [Set up Unity for AI-powered Meta Quest VR development](#): Build a complete VR game for Meta Quest using AI-assisted development.
- [Meta XR Packages](#)
- [The Project Setup Tool](#)
- [XR Plugin Management for Meta Quest](#)
- [Unity XR Plugin Management](#) [↗](#)
- [Project Configuration Overview](#)
- [Build Configuration Overview](#)
- [Platform Solutions overview](#)
- [Set up a platform app for Unity development](#)
- [Unity Manual](#) [↗](#)

## Next steps [↗](#)

After you set up your Unity 3D development environment and project, see [Set up your device](#) to learn how to connect your headset.

Did you find this page helpful?



Want to share more detailed feedback? [Let us know.](#)