

Team 26 Deployment Manual

ZIP File Content

- The project folder named "Popcorn Game"
 - All the scripts are under PopcornGame\Assets\PopcornGame\Scripts
- A single apk file named "PopcornGame.apk"

Hardware requirements

- A Windows Computer
- An ARCore supported device
- A USB cable to connect your device to your development machine

Part 1: Run the game using the apk file

Install app from PC

1. Open your Android's **Settings**. Scroll down and tap **Security**.
2. Allow app installation from external sources. (The process might be different in different devices).
3. Connect the Android device to the PC using a USB cable.
4. Tap the **USB for...** notification on your Android and choose **Transfer files**.
5. On your PC, navigate to the "**PopcornGame.apk**", right click the file, click **Send to** and select your Android device.
6. Open your Android's **file manager**, locate the APK file and install it by clicking.

Reference: <https://developer.android.com/training/basics/firstapp/running-app.html>

How to play

The game can be played with a single player or two players. Follow the instructions below:

Single-Player Version

1. Click **Create Room**, type in a personal **nickname**, type in 1 in **Maximum Player** field, type in a **room name** and click **Create**.
2. Click **Start** after you are inside the room.
3. Now the game will start, move the phone to detect planes, scan surroundings for at least 15 seconds, and then place the popcorn machine in an appropriate position. Use the slider to change the size of the popcorn machine to better fit.
4. Once find the location to place the popcorn machine, click **Place**. Wait until the notification change to "WaitingForHostedReferencePoint" and then click **READY**.
5. The popcorn machine will then start to spawn popcorns. Popcorns which are close to you will have a yellow halo, click to collect them.
6. After 20 seconds, the game will end and you will see your score.

Two-Player Version

1. The first player should follow the same instructions as the single-player version to **Create Room**, except change the maximum player number to 2 and wait for the second player to join in the room.
2. The second player can click **Join Room** and enter the room name to join.
3. After both players are inside the room, the first player can click **Start**.
4. Now the game starts, the first player should follow the instructions above to place the popcorn machine. The second player should move the phone to detect surroundings for at least 15 seconds and wait for the popcorn machine to be placed.
5. The Game will start after both players click **READY**.

Part 2: Open project in Unity

1. Preparing Android device

Run on a real device

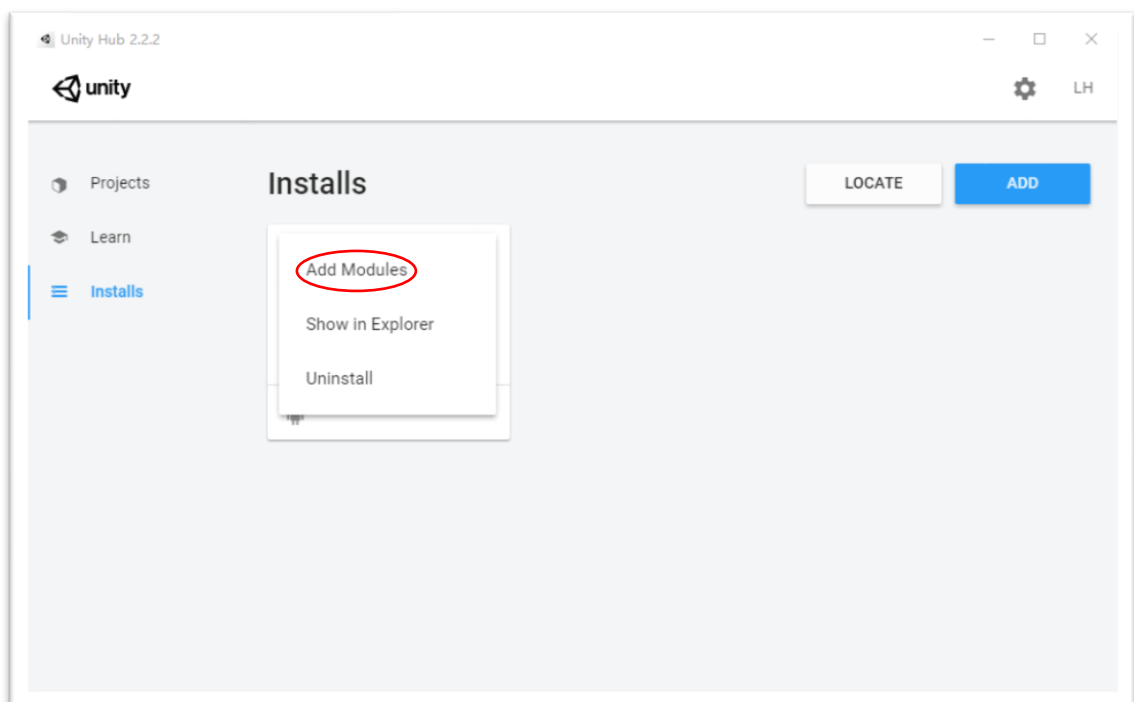
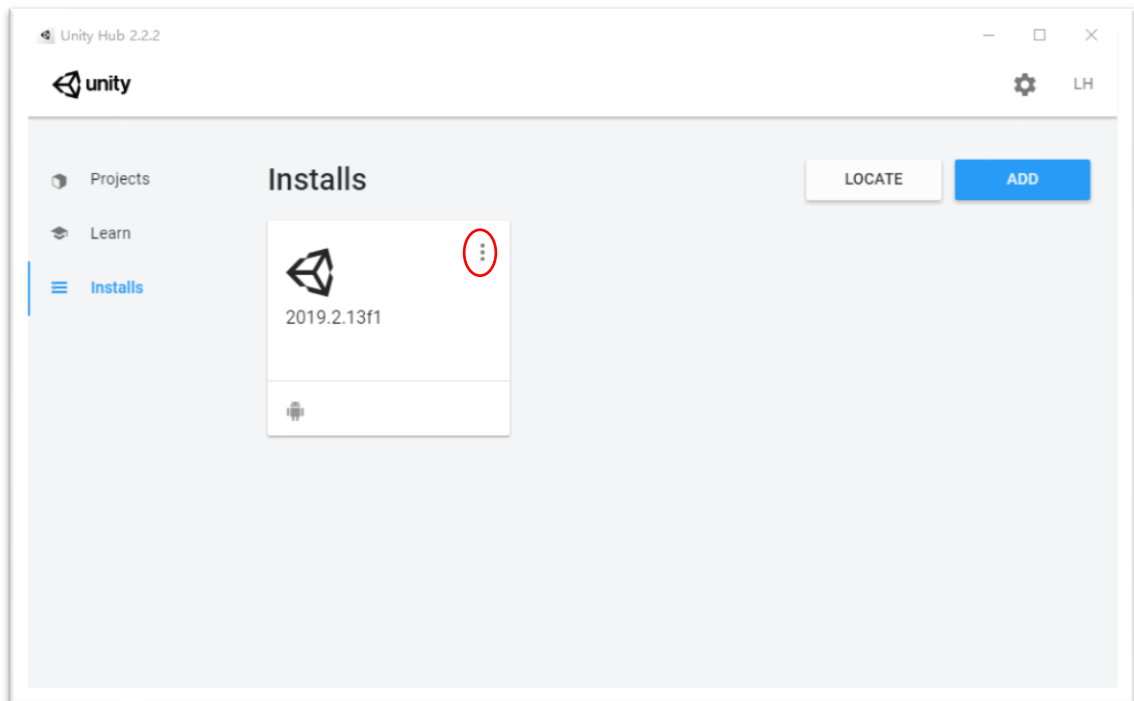
Set up your device as follows:

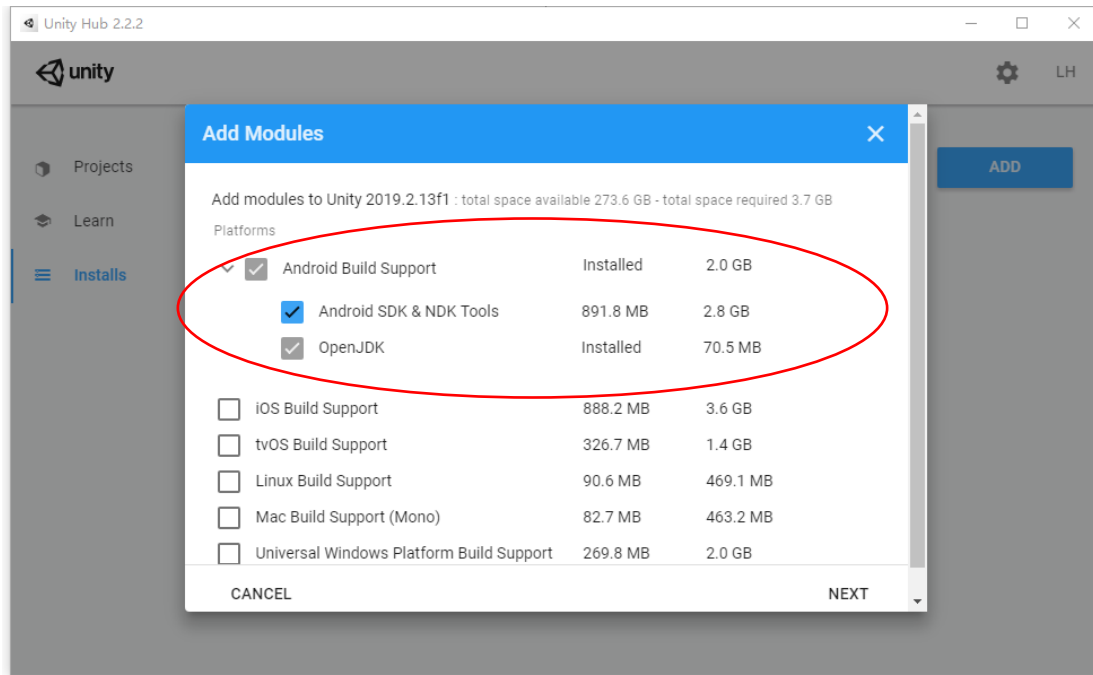
1. Connect your device to your development machine with a USB cable. If you developed on Windows, you might need to [install the appropriate USB driver](#) for your device.
2. Perform the following steps to enable **USB debugging** in the **Developer options** window:
 - a. Open the **Settings** app.
 - b. If your device uses Android v8.0 or higher, select **System**. Otherwise, proceed to the next step.
 - c. Scroll to the bottom and select **About phone**.
 - d. Scroll to the bottom and tap **Build number** seven times.
 - e. Return to the previous screen, scroll to the bottom, and tap **Developer options**.
 - f. In the **Developer options** window, scroll down to find and enable **USB debugging**.

Reference: <https://developer.android.com/training/basics/firstapp/running-app.html>

2. Preparing Unity for Android development

- 1) Download Unity hub in the Unity website.
- 2) Install Unity under **Installs** tab in Unity Hub by clicking **ADD**: The unity version our project is using is 2019.2.13f1. (Might need to visit Unity download archives to download this version)
- 3) Add modules for Android development



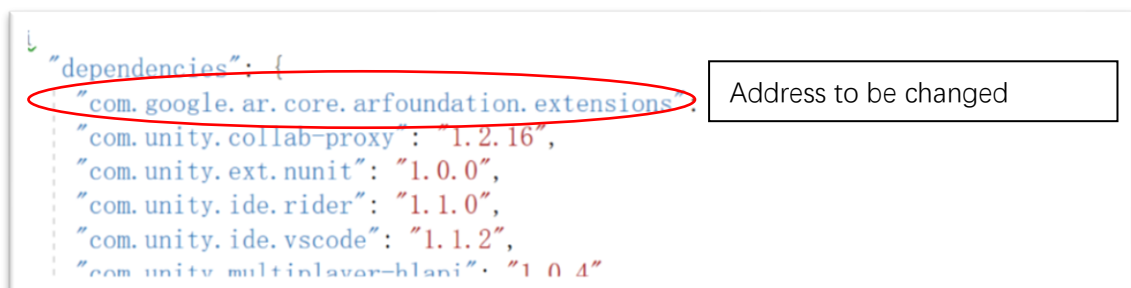


Reference: <https://developer.android.com/games/develop/build-in-unity>

- 4) Change the google AR foundation extensions address in the **manifest.json** file under folder \PopcornGame\Packages.

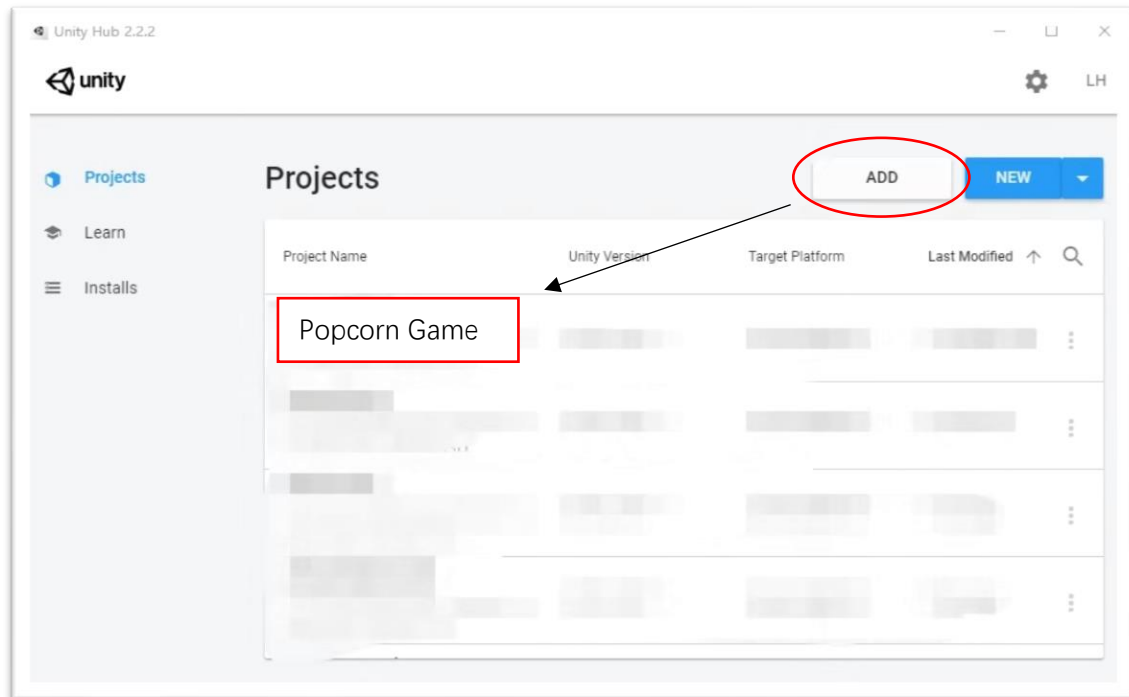
Change the address to

"file:<Your own address>/PopcornGame/Packages/arcore-unity-extensions-1.12.0/package"



- 5) Open the game project file

Open Unity Hub and go to Projects, click **ADD** and navigate to the location of the project folder named "Popcorn Game" and click **select folder**. Then click the project name to open it.

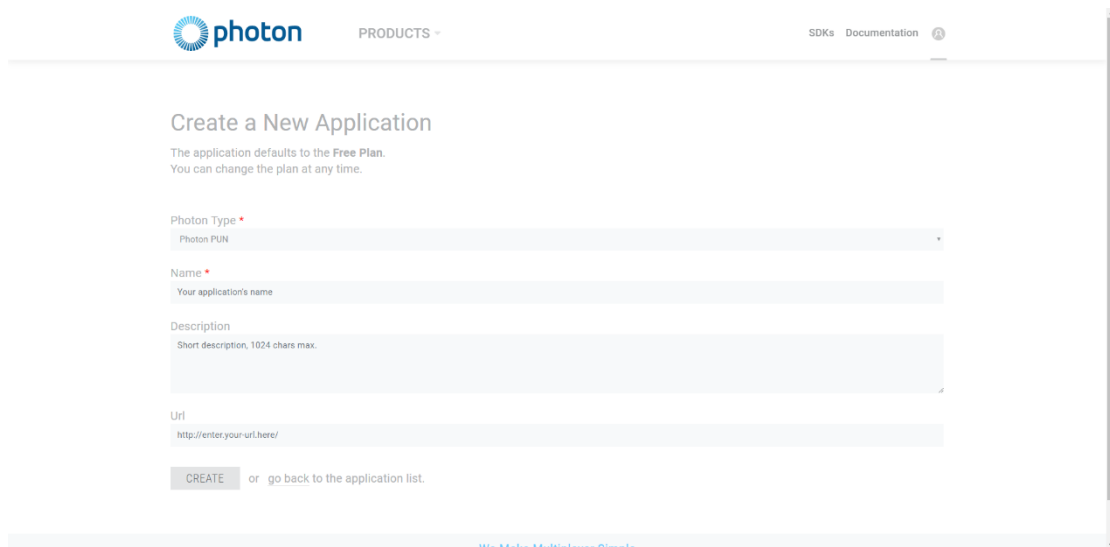


- 6) Once the project is opened, connect Android device, select **build and run**, store the apk file in an appropriate address, it will then be sent to the device.

Part 3: Customizing your AR Core and Photon configurations

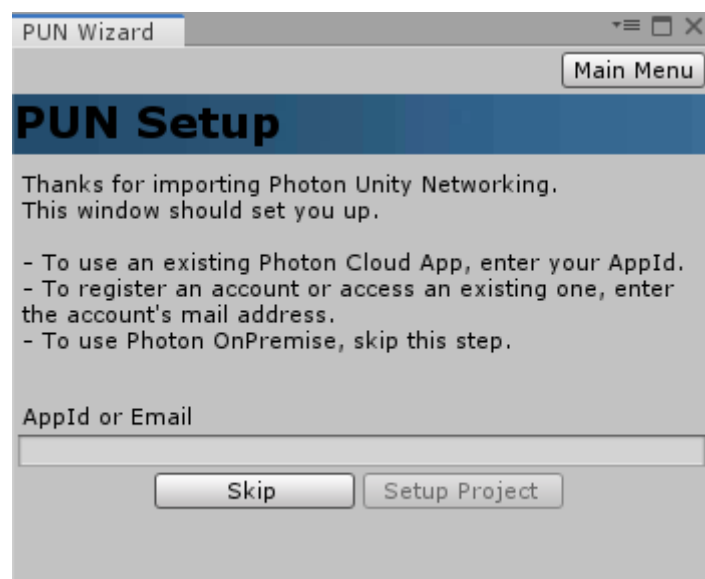
Photon Network Set Up

1. Register With Photon
2. In Photon Network Dashboard, create a new application and copy the App Id.



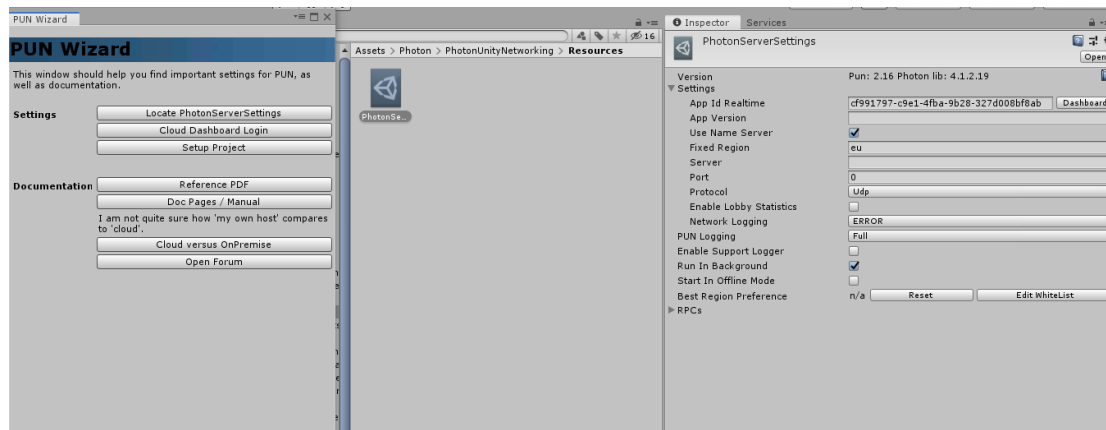
The screenshot shows the Photon Network Dashboard's 'Create a New Application' page. The page has a header with the Photon logo, 'PRODUCTS', and links for 'SDKs', 'Documentation', and a user profile icon. The main heading is 'Create a New Application', followed by a note: 'The application defaults to the Free Plan. You can change the plan at any time.' Below this are several input fields: 'Photon Type' (a dropdown menu currently showing 'Photon PUN'), 'Name' (with a subtext 'Your application's name'), 'Description' (with a subtext 'Short description, 1024 chars max.'), and 'Url' (with a subtext 'http://enter.your-url.here/'). At the bottom, there is a 'CREATE' button and a link 'or go back to the application list.'

3. Open Unity, **go to the menu -> Window -> Photon Unity Networking -> PUN Wizard -> Setup Project**. Paste the App Id.



The screenshot shows the 'PUN Wizard' window in Unity, specifically the 'Setup Project' step. The window has a title bar 'PUN Wizard' and a 'Main Menu' button. The main heading is 'PUN Setup'. Below it, a message says: 'Thanks for importing Photon Unity Networking. This window should set you up.' This is followed by three bullet points: '- To use an existing Photon Cloud App, enter your AppId.', '- To register an account or access an existing one, enter the account's mail address.', and '- To use Photon OnPremise, skip this step.' Below the instructions is a label 'AppId or Email' and a text input field. At the bottom, there are two buttons: 'Skip' and 'Setup Project'.

4. To change server settings, in PUN Wizard, click **Locate PhotonServerSettings**



Reference: <https://doc.photonengine.com/en-us/pun/v2/getting-started/initial-setup>

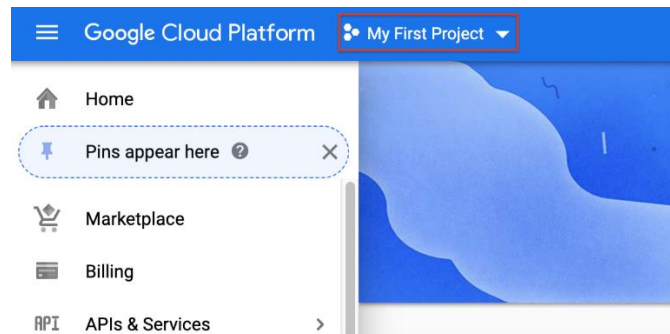
ARCore Extensions for AR Foundation: Cloud Reference Points

Configure your Cloud Anchor API key

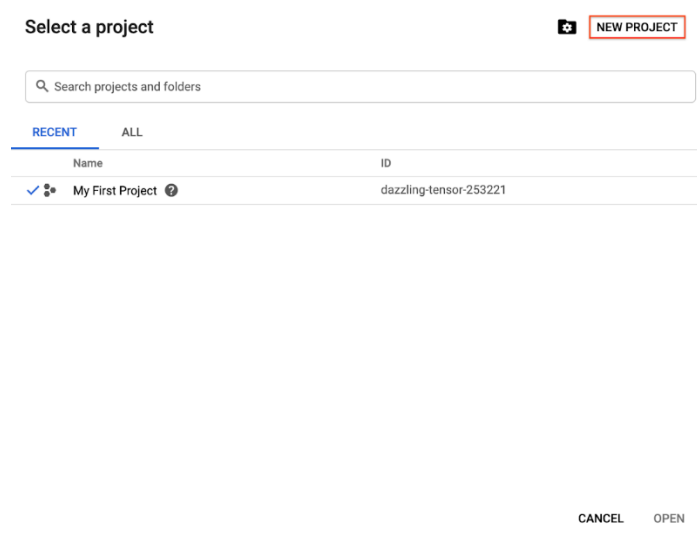
1. Enable the ARCore Cloud Anchor API within a Google Cloud Platform project. Using your Google account, enable the ARCore Cloud Anchor API in this link: <https://console.cloud.google.com/apis/library/arcorecloudanchor.googleapis.com>

The screenshot shows the Google Cloud Platform 'Welcome' page. At the top, it says 'Welcome Margaery!' and 'Create and manage your Google Cloud Platform instances, disks, networks, and other resources in one place.' Below this is the 'Terms of Service' section. It contains a checkbox that is checked, with the text 'I agree to the Google Cloud Platform Terms of Service, and the terms of service of any applicable services and APIs.' This section is highlighted with a red border. Below the terms of service is a 'Country of residence' dropdown menu, which currently shows 'United States'. At the bottom right, there is a button labeled 'AGREE AND CONTINUE', also highlighted with a red border.

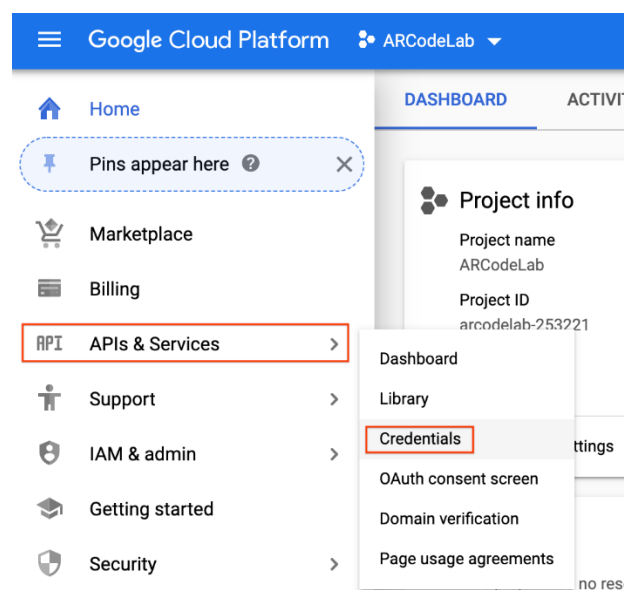
2. Select the drop down **Select a project**.



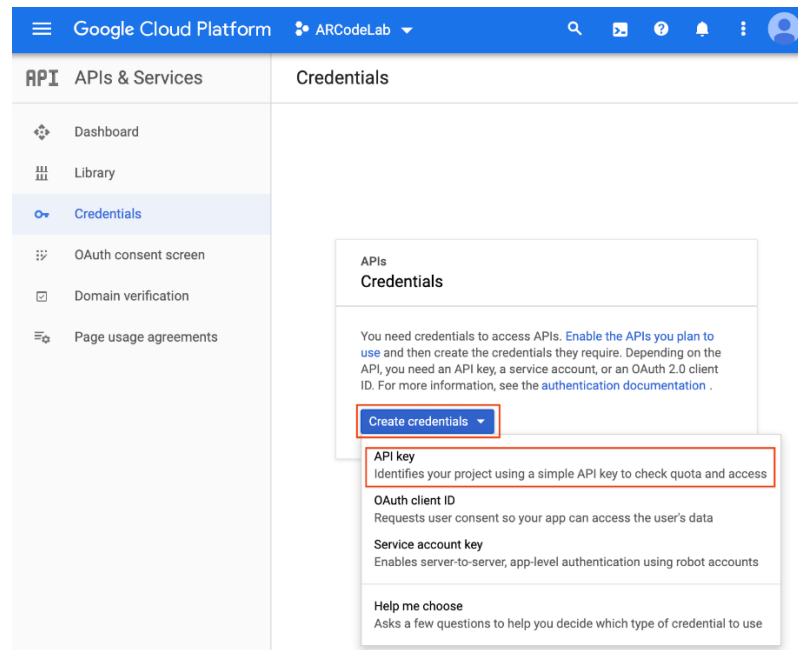
3. In the top right corner select **NEW PROJECT**.



4. After project created, Again use the **Select a project** drop down, (you may need to navigate back to the **Home** screen) this time select your new project. Use the menu icon in the top left to navigate to **APIs & Services > Credentials**.



5. After selecting your project, you will see a message informing you that you need to create credentials to access the APIs. Click Create credentials and select API key. This generates a new API key named API key 1 and an alphanumeric string. Copy the alphanumeric string to your clipboard, and return to Unity.



Adding your Cloud Anchor API Key to your project

Back in Unity, follow these steps to add your Cloud Anchor API key to your project:

1. Choose **Edit > Project settings**.
2. Expand the category **XR > ARCore Extensions**
3. Paste your API key into the **Android** field under **Cloud Anchor API Keys**.
4. Close the **Project Settings** window.

Your app is now fully configured to use the ARCore Cloud Anchor service.

Reference: <https://codelabs.developers.google.com/codelabs/arcore-extensions-cloud-anchors/#6>

