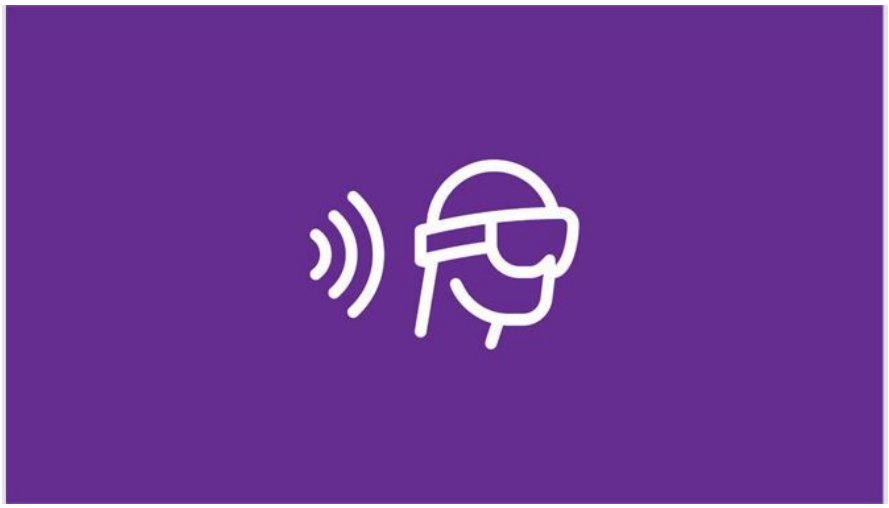
Setting up a basic Unity project to deploy to Hololens

- Set up the camera skybox
    - With the Main Camera still selected in the Hierarchy panel, find the Camera component in the Inspector panel and change the Clear Flags dropdown from Skybox to Solid Color. Set the color to black (rbga = 0,0,0,0).
- Settings for Hololens deployment
    - Select Edit > Project Settings > Quality
    - Select the dropdown under the Windows Store logo and select Very Low. You'll know the setting is applied correctly when the box in the Windows Store column and Very Low row is green.
    - Go to Edit > Project Settings > Player.
    - In the Inspector Panel for Player Settings, select the Windows Store icon.
    - Expand the XR Settings group.
    - In the Rendering section, check the Virtual Reality Supported checkbox to add a new Virtual Reality SDKs list.
    - Verify that Windows Mixed Reality appears in the list. If not, select the + button at the bottom of the list and choose Windows Mixed Reality.
- Switch the build platform
    - Open File > Build Settings window.
    - Click Add Open Scenes to add the scene.
    - Change Platform to Universal Windows Platform and click Switch Platform..
    - For Target device, switch to HoloLens.
    - UWP Build Type should be D3D.
    - UWP SDK could be left at Latest installed.
    - Check Unity C# Projects under Debugging.
- Build for Hololens
    - Click Build.
    - In the file explorer, click New Folder and name the folder "App," and select that folder.
    - When Unity is done building, a Windows File Explorer window will appear.
    - Open the App folder
    - Open the generated Visual Studio solution (*.sln). This will open Visual Studio.

- Deploying to Hololens
    - On the Hololens, open the Holographic Remoting Player. The app icon will look like this.



    - An IP will be displayed.This is the Hololens' IP address.
    - Back in Unity, Open Window > XR > Holographic Emulation.
    - Change Emulation Mode from None to Remote to Device.
    - In Remote Machine, enter the IP address of your HoloLens noted earlier.
    - Click Connect.
    - Ensure the Connection Status changes to green Connected.
    - In Visual Studio, using the top toolbar, change the target from Debug to Release and from ARM to X86.
    - Click on the arrow next to the Local Machine button, and change the deployment target to Remote Machine.
    - Enter the IP address of the Hololens. Click Debug > Start without debugging.
    - Assuming there are no errors, the Holographic Remoting Player will go black, and you will see the "Made with Unity" logo.
    - Your Unity project should now be displaying on the Hololens.
- Alternate Options
    - Alternatively, you can deploy your app to the Hololens Emulator if you do not have a Hololens present.
    - Ignore the steps involving the Hololens, and entering its IP.
    - Open the .sln file.
    - Click on the arrow next to the Device button, and from the drop down select HoloLens Emulator.

- Click Debug > Start without debugging.