DEVCOM
ARMY RESEARCH LABORATORY

# AXR Unity Package Builder Documentation

BY TIFFANY R. RABER, THERON T. TROUT, AND MARK S. DENNISON

# PROTOBUF LICENSE

Copyright 2008 google inc.  All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* neither the name of google Inc. Nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

This software is provided by the copyright holders and contributors "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall the copyright owner or contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

Code generated by the protocol buffer compiler is owned by the owner of the input file used when generating it.  This code is not standalone and requires a support library to be linked with it.  This support library is itself covered by the above license.

# CONTENTS


PROTOBUF LICENSE ..... 1

Overview: ..... 3

Background ..... 3

Getting Started ..... 3

Demo Project Setup ..... 3

Setting Up AURORA_XR ..... 4

Unity Login Scene Setup ..... 4

Unity Main Scene Setup ..... 5

Unity Movement Testing Scene Setup ..... 7

Return to Unity Login Scene Setup ..... 8

Creating a Test Build (Movement Test) ..... 9

Creating a Spawnable Object ..... 9

Creating a Test Build (SpawnableObject Test) ..... 11

SUPPORT ..... 12

Submitting a Ticket ..... 12

# OVERVIEW:

## Background

AURORA stands for Accelerated User Reasoning for Operations, Research, and Analysis and is a research platform consisting of server (AURORA-Server), network (AURORA-Net) and interface (AURORA-XR) modules. AURORA was designed to enable researchers to conduct controlled experimentation on the ingestion of multi-domain battlefield data on a configurable network, to its visualization, analysis, and actuation by collaborative teams of humans and intelligent agents that are both co-located and distributed over geographical space.

AURORA-XR is the interface module of the AURORA ecosystem and serves to provide users with a means of perceiving and interacting with battlefield information from heterogeneous sources, as well as interacting with other co-located and remote users. Data sources can either be locally generated (as in simulation) or, more optimally, taken from data feeds on the AURORA network that are subscribed to by applications running within an AURORA-XR-enabled device.

AURORA-XR interfaces are built on an API for the Unity engine by Unity Technologies. Any display technology that is supported by the Unity engine may be developed into an AURORA-enabled system. Currently, the system supports a Windows PC with an Oculus Rift S or Oculus Rift (CV1) head-mounted display (HMD). Future development to support the Oculus Quest and Microsoft HoloLens are planned.

## Getting Started

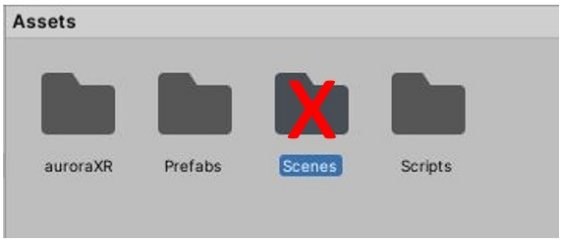
You will need to have:

1. Unity 2019.3.12.f1 downloaded
2. auroraXR_Unity_1.0.0.1.unitypackage downloaded

## Demo Project Setup

1. Open Unity 2019.3.12.f1 → Create new project
2. Navigate to **Assets** → right-click → **Import Package** → **Custom Package...** → Locate auroraXR_Unity_1.0.0.1.unitypackage and open/import into project

3. Once imported, delete top level **Scenes** folder

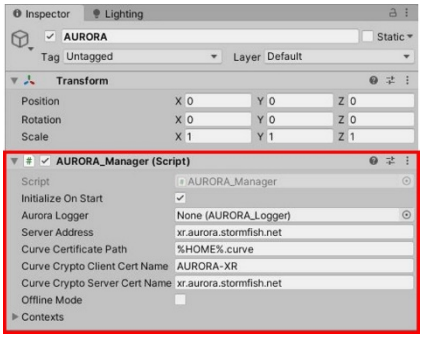



# SETTING UP AURORA_XR

## Unity Login Scene Setup

1. Navigate to **Assets** → **auroraXR** → **Scenes,** create Scene named *auroraXR_Login*

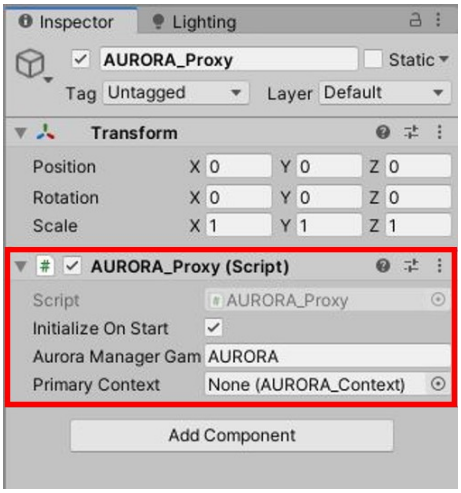
2. Create an Empty GameObject in the **Hierarchy** named *AURORA*
   a. Add *AURORA_Manager* Script Component with the following completed:
      i. Initialize On Start = Check
      ii. Aurora Logger = AURORA (AURORA_Logger)
      iii. Server Address = xr.aurora.stormfish.net
      iv. Curve Certificate Path = %HOME%.curve
      v. Curve Crypto Client Cert Name = AURORA-XR
      vi. Curve Crypto Server Cert Name = xr.aurora.stormfish.net
      vii. Offline Mode = Not Checked



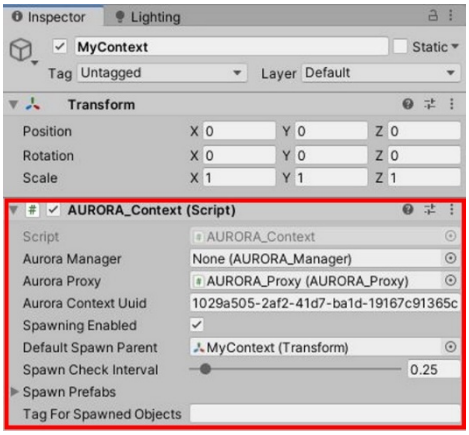
# Unity Main Scene Setup

1. Navigate to **Assets → auroraXR → Scenes,** create Scene named *auroraXR_Main*

2. Create an Empty GameObject in the **Hierarchy** named *AURORA_Proxy*
    a. Add *AURORA_Proxy* Script Component with the following completed:
        i. Initialize On Start = Check
        ii. Aurora Manager Game object = AURORA
        iii. Primary Context = None (AURORA_Context)

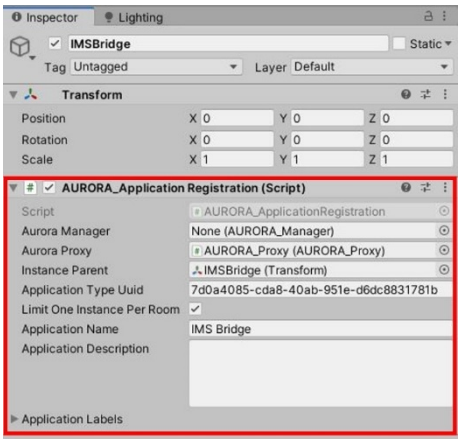



3. Create an Empty GameObject in the **Hierarchy** named *MyContext*
    a. Add *AURORA_Context* Script Component with the following completed:
        i. Aurora Manager = None (AURORA_Manager)
        ii. Aurora Proxy = AURORA_Proxy (AUORORA_Proxy)
            1. If not automatically populated, complete this step by dragging *AURORA_Proxy* **GameObject** into Aurora Proxy field
        iii. Aurora Context Uuid = "Will populate with a randomly generated value"
        iv. Spawning Enabled = Checked
        v. Default Spawn Parent = MyContext (Transform)
        vi. Spawn Check Interval = 0.25




4. Create an Empty GameObject in the **Hierarchy** named *ApplicationRegistrations*
    a. Create a Child GameObject of *ApplicationRegistrations* named *IMSBridge*

b. Add *AURORA_ApplicationRegistration* Script Component to the Child GameObject *(IMSBridge)* with the following completed:




i. Aurora Manager = None (AURORA_Manager)

ii. Aurora Proxy = AURORA_Proxy (AURORA_Proxy)

1. If not automatically populated, complete this step by dragging *AURORA_Proxy* **GameObject** into Aurora Proxy field

iii. Instance Parent = IMSBridge (Transform)

iv. Application Type Uuid = "We are currently developing an application discovery service for the near future. In the meantime, generate your unique identifier through an external service (ex. https://www.uuidgenerator.net/)"

v. Limit One Instance Per Room = Checked

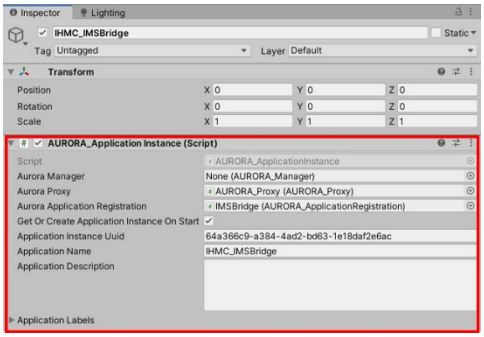
vi. Application Name = IMS Bridge

vii. Application Description = "Can leave blank"

5. Create an Empty GameObject in the **Hierarchy** named *ApplicationInstances*

a. Create a Child GameObject of *ApplicationInstances* named *IHMC_IMSBridge*

b. Add *AURORA_ApplicationInstances* Script Component to the Child GameObjcet (*IHMC_IMSBridge*) with the following completed:




i. Aurora Manager = None (AURORA_Manager)

ii. Aurora Proxy = AURORA_Proxy (AURORA_Proxy)

1. If not automatically populated, complete this step by dragging *AURORA_Proxy* **GameObject** into Aurora Proxy field

iii. Aurora Application Registration = IMSBridge (AURORA_ApplicationRegistration)

1. Complete this step by dragging IMSBridge **GameObject** into field

iv. Get or Create Application Instance on Start = Checked

v. Application Instance Uuid = "We are currently developing an application discovery service for the near future. In the meantime, generate your unique identifier through an external service (ex. https://www.uuidgenerator.net/)"

vi. Application Name = IHMC_IMSBridge

vii. Application Description = "Can be left blank"

c. Create a Child GameObject of IHMC_IMSBridge named Topics

d. Create a Child GameObject of Topics named CoT

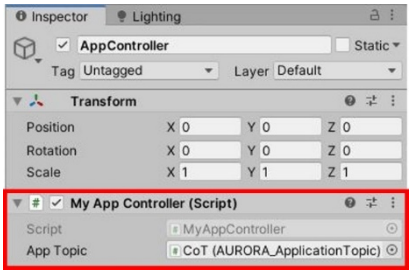
e. Add AURORA_ApplicationTopic Script Component to the Child GameObject (CoT) with the following completed:

i. Aurora Application Instance = IHMC_IMSBridge (AURORA_ApplicationInstance)

1. Complete this step by dragging *IHMC_IMSBridge* **GameObject** into field

ii. Application Topic = cot

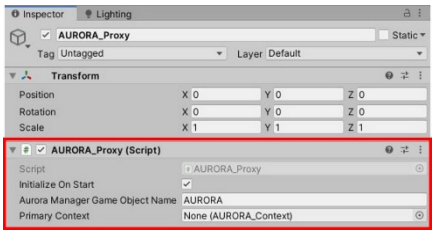
6. Create an Empty GameObject in the Hierarchy named *AppController*
    a. Add *MyAppController* Script Component with the following completed:
        i. App Topic = CoT (AURORA_ApplicationTopic)
            1. Complete this step by dragging *CoT* **GameObject** into field



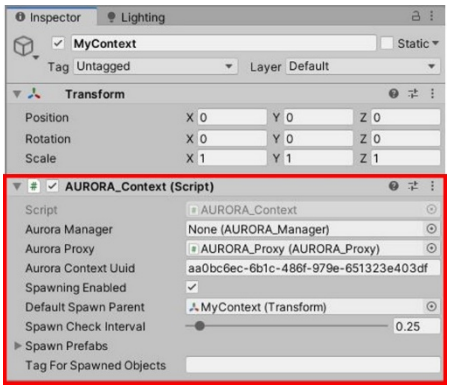

## Unity Movement Testing Scene Setup

1. Navigate to **Assets → auroraXR → Scenes,** create Scene named *auroraXR_MovementTesting*

2. Create an Empty GameObject in the **Hierarchy** named *AURORA_Proxy*
    a. Add *AURORA_Proxy* Script Component with the following completed:
        i. Initialize On Start = Checked
        ii. Aurora Manager Game Object Name = AURORA

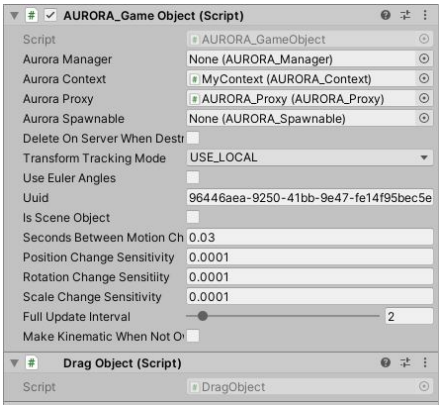



3. Create an Empty GameObject in the **Hierarchy** named *My Context*
    a. Add *AURORA_Context* Script Component with the following completed:
        i. Aurora Manager = None (AURORA_Manager)
        ii. Aurora Proxy = AURORA_Proxy (AURORA_Proxy)
            1. If not automatically populated, complete this step by dragging *AURORA_Proxy* **GameObject** into Aurora Proxy field
        iii. Aurora Context Uuid = “Will populate with a randomly generated value”
        iv. Spawning Enabled = Checked
        v. Default Spawn Parent = MyContext (Transform)
        vi. Spawn Check Interval = .25




4. Create a Child 3DObject (Cube) of AURORA_Context named Cube
    a. Add AURORA_GameObject Script Component with the following completed:
        i. Aurora Manager = None (AURORA_Manager)

ii. Aurora Context = MyContext (AURORA_Context)
    1. Complete this step by dragging *MyContext* **GameObject** into Aurora Context field
iii. Aurora Proxy = AURORA_Proxy (AURORA_Proxy)
    1. If not automatically populated, complete this step by dragging *AURORA_Proxy* **GameObject** into Aurora Proxy field
iv. Aurora Spawnable = None (AURORA_Spawnable)
v. Delete On Server When Destroyed = Not Checked
vi. Transform Tracking Mode = USE_LOCAL
vii. Use Euler Angles = Not Checked
viii. Uuid = Generate by navigating to **AURORA** in navigation bar → **AURORA Game Objects** → **Generate New UID**
ix. Is Scene Object = Checked




b. Add DragObject Script Component onto Cube
    i. NOTE: This will enable selection of the cube and dragging in the Z-axis

5. Return to AURORA_Proxy in the Hierarchy
    a. In its AUORA_Proxy Script Component, complete the following:
        i. Primary Context = MyContext (AURORA_Context)
            1. Complete this step by dragging *MyContext* **GameObject** into Primary Context field

---

## Return to Unity Login Scene Setup

1. Navigate to **Assets** → **auroraXR** → *auroraXR_Login*

2. Create an Empty GameObject in the **Hierarchy** named *SceneManager*
    a. Add *SwitchScene* Script Component and complete the following:
        i. Scene Name = auroraXR/Scenes/auroraXR_MovementTesting

3. Navigate to AURORA in the Hiearchy

i. Add *AURORA_Login* Script Component with the following completed:AURORA Manager = AURORA (AURORA_Manager)

ii. Authenticate On Start = Checked

iii. Add an Authentication Successful Event

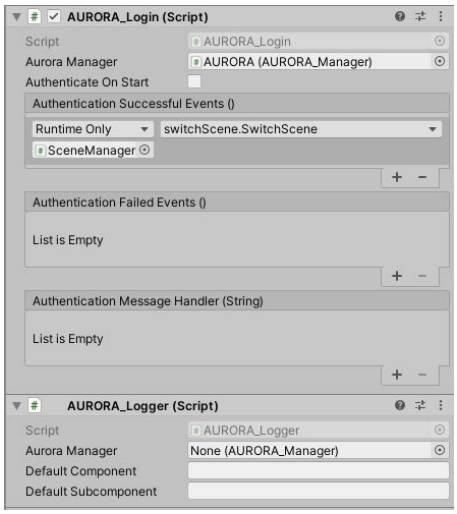
1. Drag *SceneManager* GameObject into *None(Object)*
2. Select under **Function** dropdown *SwitchScene()*

b. Add *AURORA_Logger* Script Component with the following completed:

i. Aurora Manager = AURORA (AURORA_Manager)

ii. Default Component = “blank”

iii. Default Subcomponent = “blank”

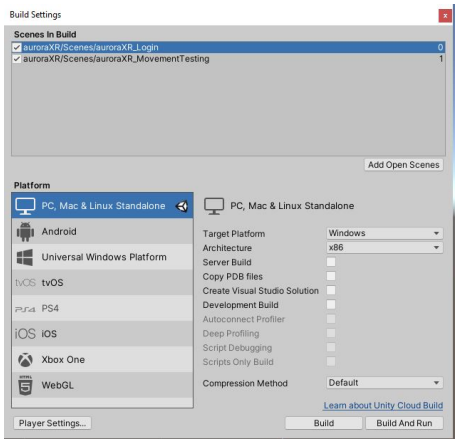



---

## Creating a Test Build (Movement Test)

1. Navigate to **File** → **Build Settings** and Add the following **Scenes**, in order
   a. auroraXR/Scenes/auroraXR_Login
   b. auroraXR/Scenes/auroraXR_MovementTesting

2. Select **Build**, Create a “Builds” folder in the top-level project folder

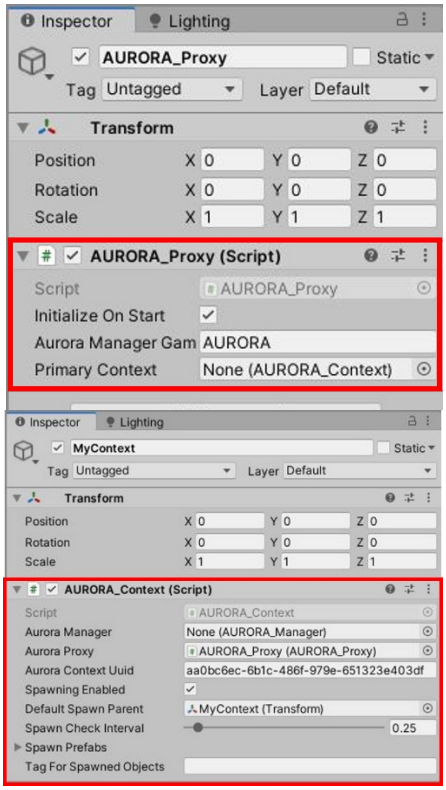



---

## Creating a Spawnable Object

1. Navigate to **Assets** → **auroraXR** → **Scenes,** create Scene named *auroraXR_SpawnableObjectTesting*

2. Create an Empty GameObject in the **Hierarchy** named *AURORA_Proxy*
    a. Add *AURORA_Proxy* Script Component with the following completed:
        i. Initialize On Start = Checked
        ii. Aurora Manager Game Object Name = AURORA

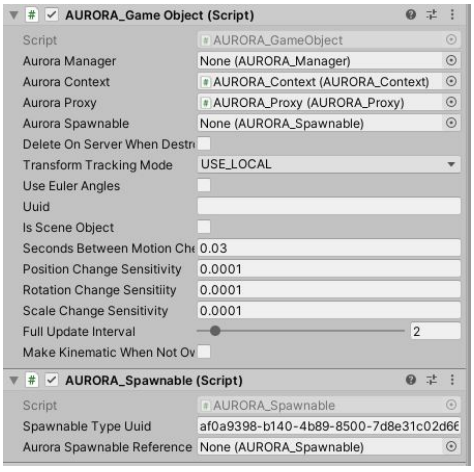



3. Create an Empty GameObject in the **Hierarchy** named *AURORA_Context*
    a. Add *AURORA_Context* Script Component with the following completed:
        i. Aurora Manager = None (AURORA_Manager)
        ii. Aurora Proxy = AURORA_Proxy (AURORA_Proxy)
            1. If not automatically populated, complete this step by dragging *AURORA_Proxy* **GameObject** into Aurora Proxy field
        iii. Aurora Context Uuid = "Will populate with a randomly generated value"
        iv. Spawning Enabled = Checked
        v. Default Spawn Parent = MyContext (Transform)
        vi. Spawn Check Interval = .25

4. Create 3D GameObject (Sphere) in the **Hierarchy** named *SpawnableSphere*

    a. NOTE: Gravity should be checked on the RigidBody Component

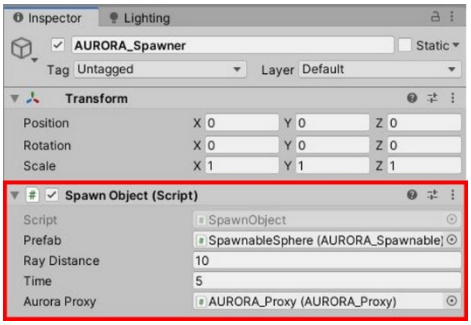
    b. Add *AURORA_GameObject* Script Component with the following completed:
        i. Aurora Manager = None (AURORA_Manager)
        ii. Aurora Context = MyContext (AURORA_Context)
            1. Complete this step by dragging *MyContext* **GameObject** into Aurora Context field
        iii. Aurora Proxy = AURORA_Proxy (AURORA_Proxy)
            1. If not automatically populated, complete this step by dragging *AURORA_Proxy* **GameObject** into Aurora Proxy field
        iv. Aurora Spawnable = None (AURORA_Spawnable)
        v. Delete On Server When Destroyed = Not Checked
        vi. Transform Tracking Mode = USE_LOCAL
        vii. Use Euler Angles = Not Checked
        viii. Uuid = Generate by navigating to **AURORA** in navigation bar → **AURORA Game Objects** → **Generate New UID**
        **ix. Is Scene Object = Not Checked**



c. Add AURORA_Spawnable Script Component with the following completed:
    i. Spawnable Type Uuid = “Will populate with a randomly generated value”
    ii. Aurora Spawnable Reference = SpawnableSphere (AURORA_Spawnable)

d. Make *SpawnableSphere* a Prefab by dragging it to the Assets/Prefabs folder (Your game object should turn blue)
    i. You can delete the SpawnableSphere out of your scene

5. Create an Empty GameObject in the **Hierarchy** named *AURORA_Spawner*
    a. Add *AURORA_Spawner* Script Component with the following completed:
        i. Pefab = SpawnableSphere (AURORA_Spawnable)
            1. Complete by dragging SpawnableSphere from Assets/Prefabs into field
        ii. Ray Distance = 10
        iii. Time = 5
        iv. Aurora Proxy = AURORA_Proxy (AURORA_Proxy)
            1. Complete by dragging *AURORA_Proxy* from **Hierarchy** into field



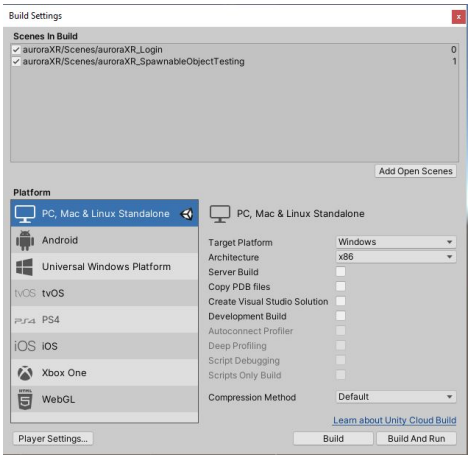

6. Navigate back to AURORA_Context → AURORA_Context Script Component and complete the following:
    a. Expand Spawn Prefabs
        i. Size = 1
        ii. Element 0 = SpawnableSphere (AURORA_Spawnable)
            1. Complete by dragging SpawnableSphere from Assets/Prefabs into field


---

## Creating a Test Build (SpawnableObject Test)

1. Navigate to **File → Build Settings** and Add the following **Scenes**, in order
    a. auroraXR/Scenes/auroraXR_Login
    b. auroraXR/Scenes/auroraXR_SpawnableObjectTesting

2. Select **Build**, Create a “Builds” folder in the top-level project folder




# SUPPORT

## Submitting a Ticket

To submit an auroraXR Support ticket, navigate to https://redmine.stormfish.net/projects and login with the provided credentials. Select the auraoraXR Unity 3D support to submit a bug, give input on a feature, or request support.

If you do not have access to the support system, please contact one of the auroraXR developers.