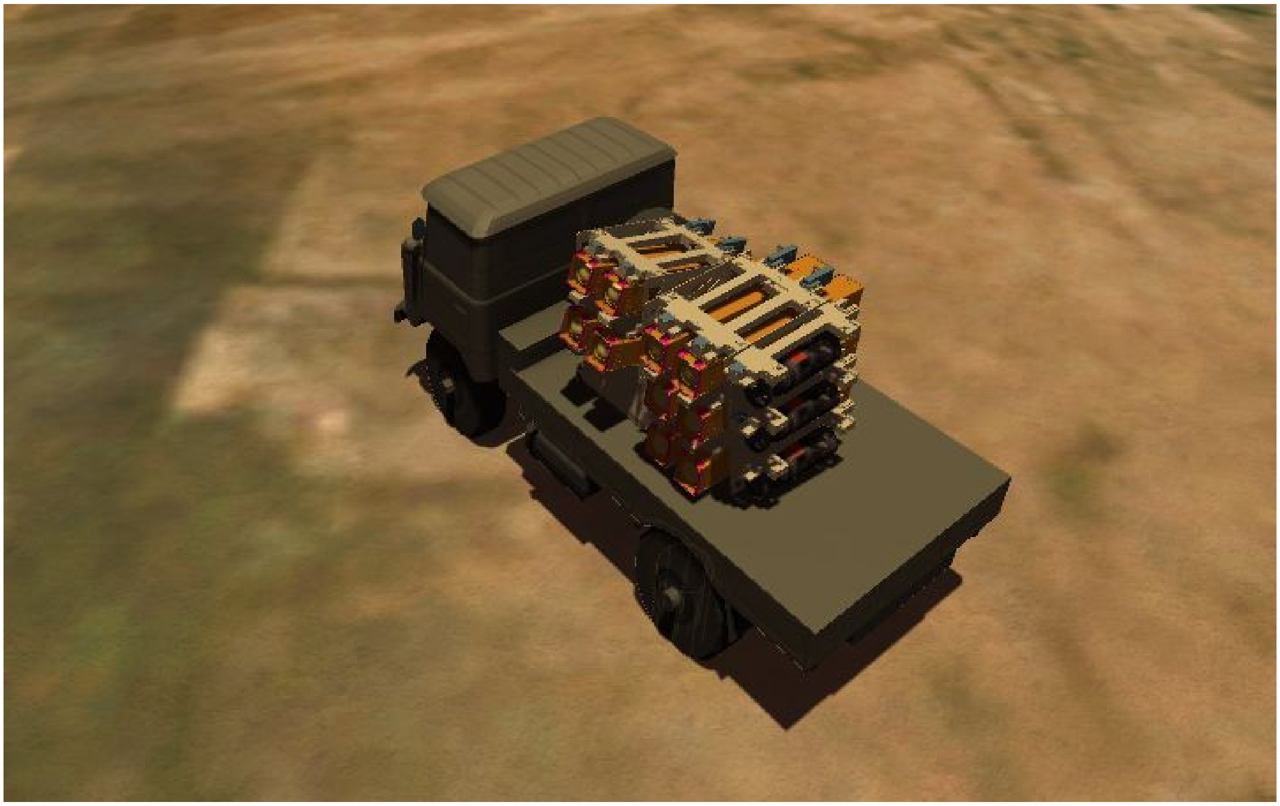
War Room Hololens

Avi Kotlicky, Dima Trushin, Natan Yellin

[GIP] | [234329] | [10.09.17]

Project idea

We developed a 3D Augmented Reality Multi-User application. Our application allows all users to simultaneously see the same scene at different angles. The Users are able to cooperatively manipulate objects and plan events in the scene. Objects can be manipulated or viewed in depth with controlled gestures and voice recognition.



Trailer Video
Application Website
Download Game here

TODO: CHANGE ABOVE LINKS

Introduction

We developed Hololens application, in Unity 5 - graphic engine, scripted with C# in Visual Studio 2015 and MonoDevelop environment.

Unity is a cross-platform game engine that is used to develop video games for PC, consoles, mobile devices and websites.

Microsoft HoloLens is a pair of mixed reality smart-glasses.

Hardware:
The HoloLens features an inertial measurement unit (IMU) (which includes an accelerometer, gyroscope, and a magnetometer) four "environment understanding" sensors (two on each side), an energy-efficient depth-camera with a 120°×120° angle of view, a 2.4-megapixel photographic video camera, a four-microphone array, and an ambient light sensor.

The HoloLens contains an internal rechargeable battery, with average life rated at 2–3 hours of active use, or 2 weeks of standby time. The HoloLens can be operated while charging.

The headset uses Bluetooth LE to pair with the included Clicker, a thumb-sized finger-operating input device that can be used
for interface scrolling and selecting.

Augmented Reality Setup

Minimum Requirements for computers

- 64-bit Windows 10 (Not home Edition)-(for SDK)
- 64-bit CPU - (for SDK)
- 8 GB of Ram- (for SDK)
- Visual Studio 2017 or later -(for UWP)
- Bios Features required and enabled: - (for SDK)
    - Hardware-assisted virtualization
    - Second Level Address Translation (SLAT)
    - Hardware-based Data Execution Prevention (DEP)
- GPU**
    - Directx 11.0 or later - (for SDK)
    - WDDM 1.2 Driver or later - (for SDK)
    - Feature level 9.3 capabilities - (for Unity)

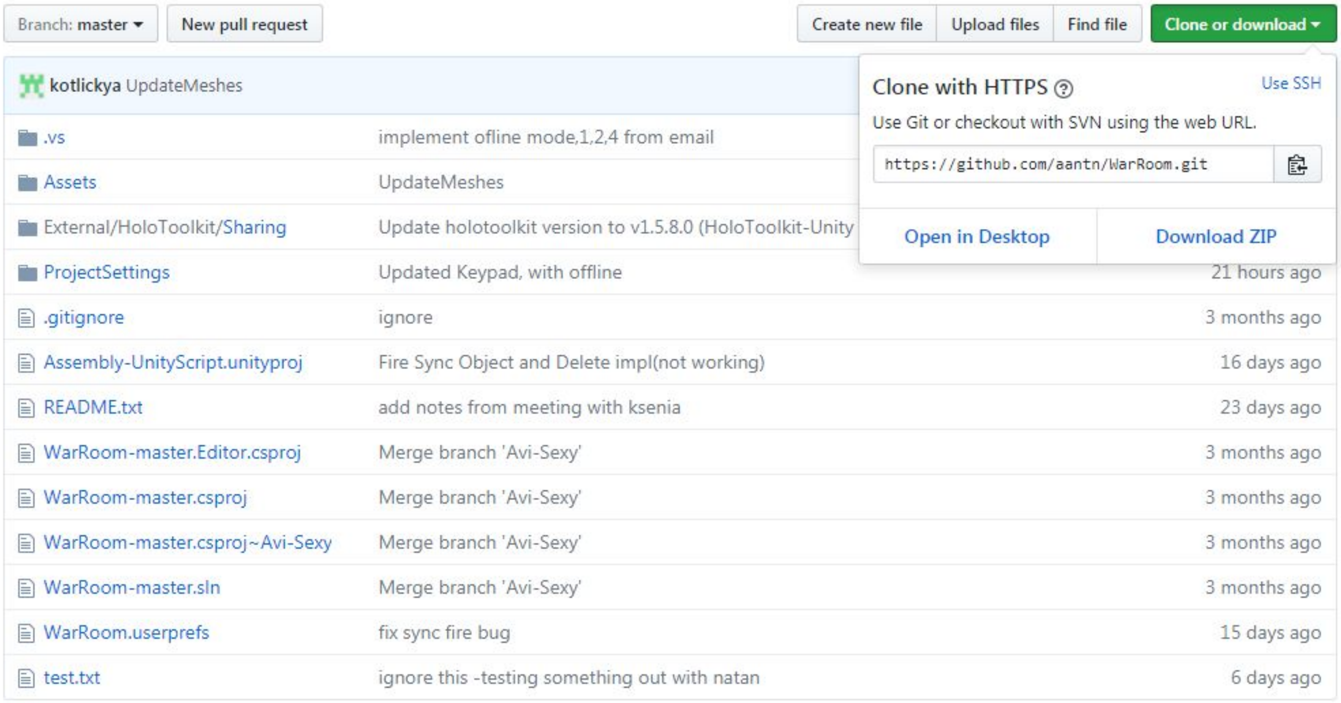
***The software could possibly work on lower quality GPU's but the speed and performance will be impacted.*

More Details can be found here:
https://developer.microsoft.com/en-us/windows/mixed-reality/install_the_tools#system_requirements

Setup -> Build -> Deploy

- Download the required software
    - Visual studio 2017
    - Hololens Emulator
    - Unity 5.6

        Links to download locations can be found here:
        https://developer.microsoft.com/en-us/windows/mixed-reality/install_the_tools
- **Download the "WarRoom" Unity Project: https://github.com/aantn/WarRoom.git**



Clone with HTTPS

Use SSH

Use Git or checkout with SVN using the web URL.

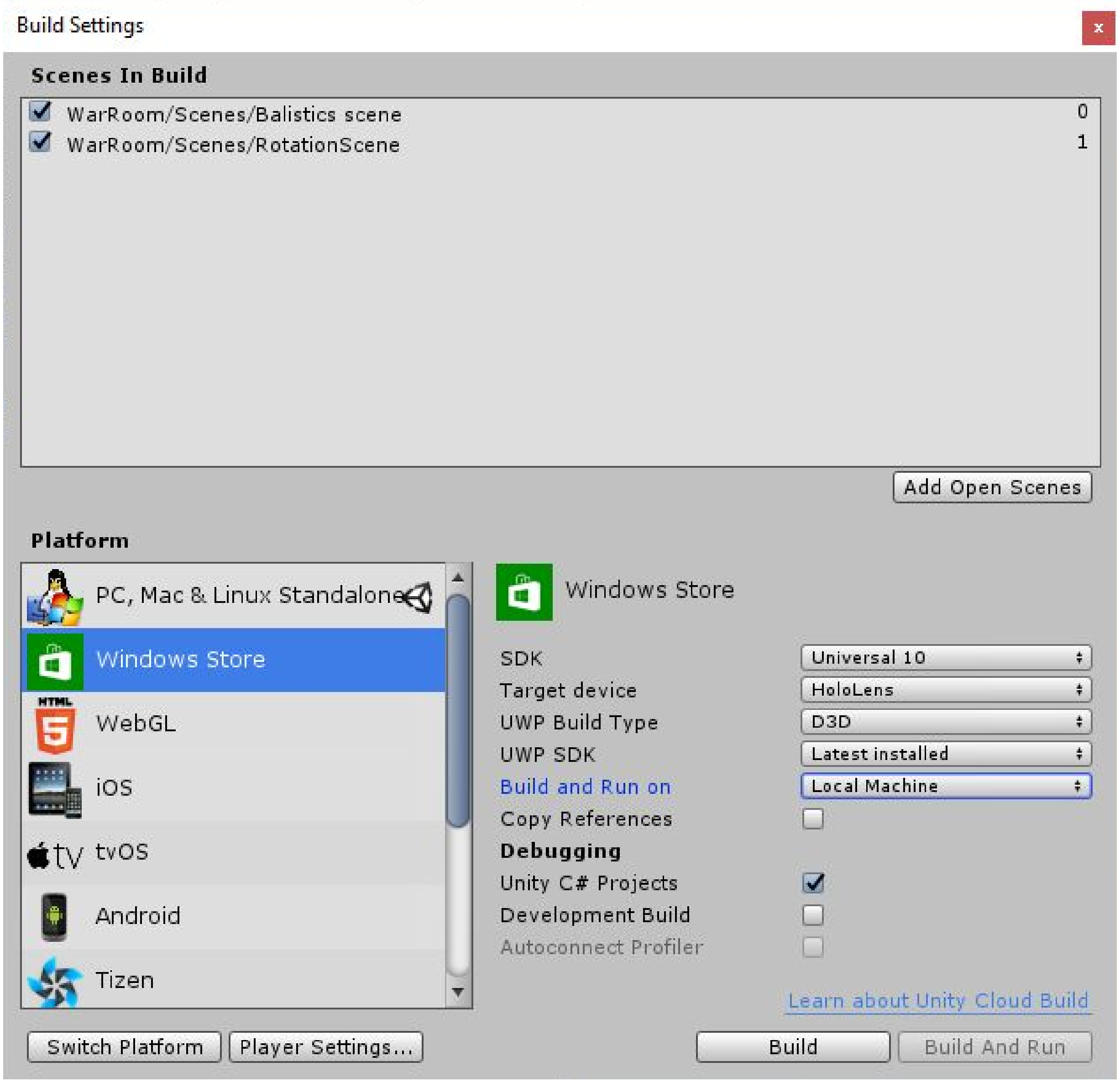
https://github.com/aantn/WarRoom.git

kotlickya UpdateMeshes

.vs	implement ofline mode,1,2,4 from email	
Assets	UpdateMeshes	
External/HoloToolkit/Sharing	Update holotoolkit version to v1.5.8.0 (HoloToolkit-Unity	
ProjectSettings	Updated Keypad, with offline	21 hours ago
.gitignore	ignore	3 months ago
Assembly-UnityScript.unityproj	Fire Sync Object and Delete impl(not working)	16 days ago
README.txt	add notes from meeting with ksenia	23 days ago
WarRoom-master.Editor.csproj	Merge branch 'Avi-Sexy'	3 months ago
WarRoom-master.csproj	Merge branch 'Avi-Sexy'	3 months ago
WarRoom-master.csproj~Avi-Sexy	Merge branch 'Avi-Sexy'	3 months ago
WarRoom-master.sln	Merge branch 'Avi-Sexy'	3 months ago
WarRoom.userprefs	fix sync fire bug	15 days ago
test.txt	ignore this -testing something out with natan	6 days ago

- Open the WarRoom project in Unity 5.6
- In the Hierarchy go to Managers->Sharing
  In the inspector under “Server Address” put the IP address of the computer that will be running the sharing service.
  ***This can be found by running the command ipconfig from command prompt of the desired computer.*

- Go to File->Build Settings and select the Windows store under Platform
  ***Install the windows store plugin for Unity if it isn’t already installed*
- Build the project for Windows with the settings listed to the right.
- Notice that you make a build with te “WarRoom/Scenes/Balistics” scene only!



Build Settings

Scenes In Build

☑	WarRoom/Scenes/Balistics scene	0
☑	WarRoom/Scenes/RotationScene	1

Platform

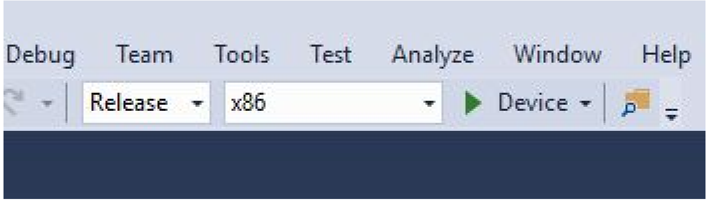
Windows Store

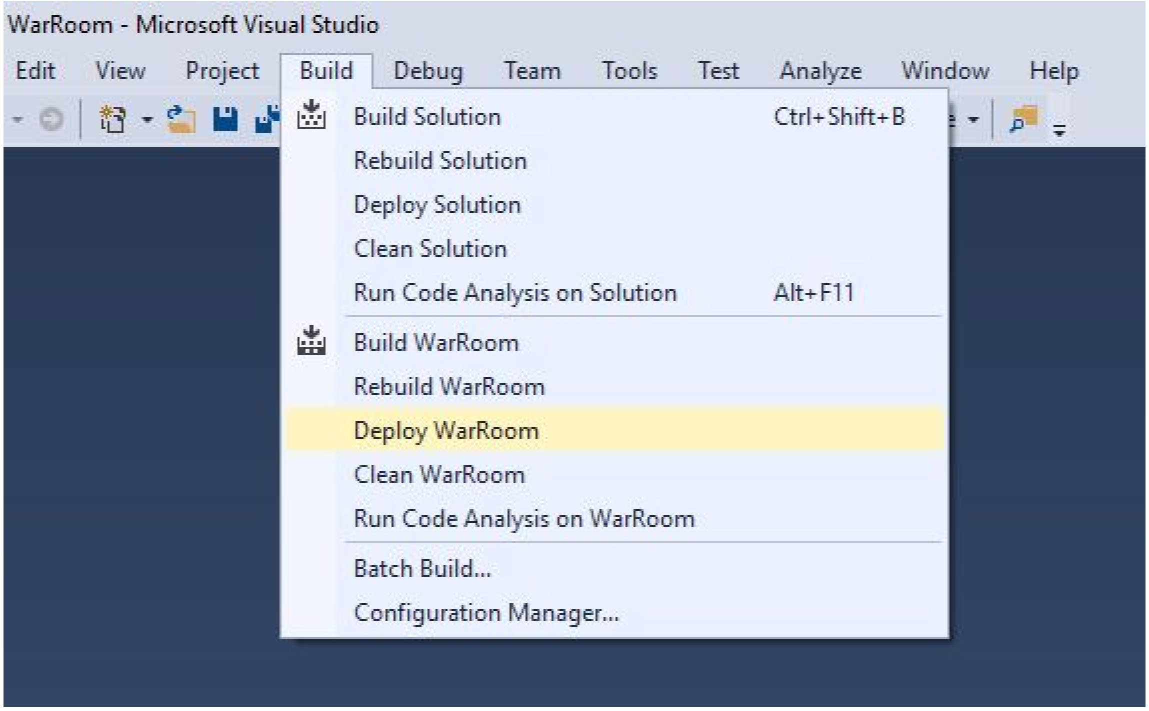
SDK	Universal 10
Target device	HoloLens
UWP Build Type	D3D
UWP SDK	Latest installed
Build and Run on	Local Machine
Copy References	□
Debugging	
Unity C# Projects	☑
Development Build	□
Autoconnect Profiler	□

- Connect the hololens and open the Visual Studio Solution
  https://developer.microsoft.com/en-us/windows/mixed-reality/using_visual_studio

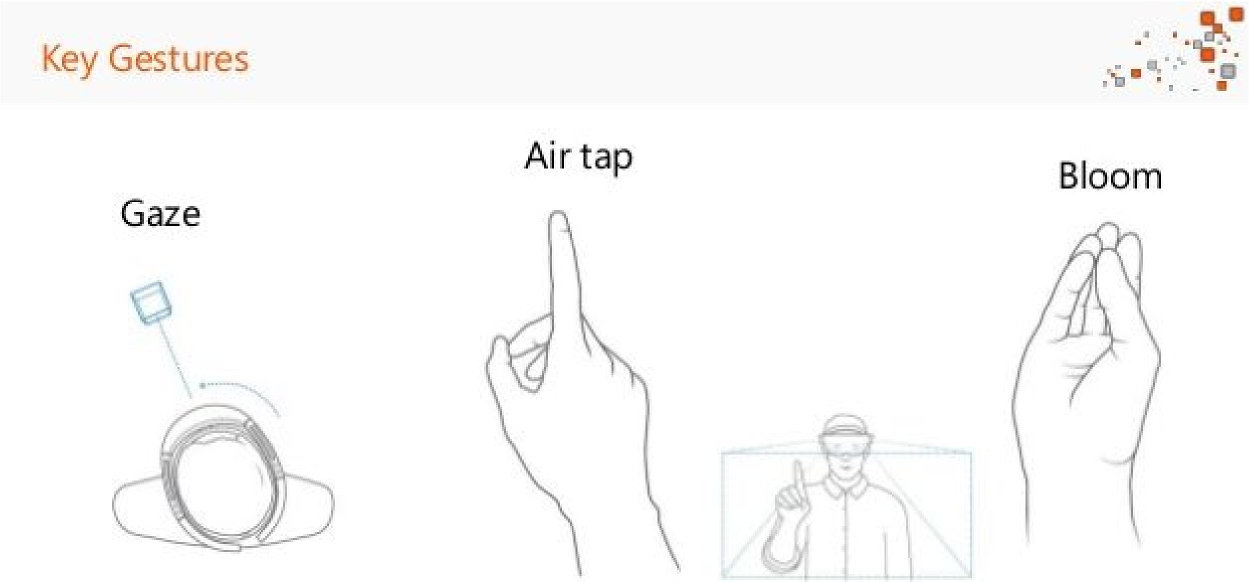
  In the hololens:
    - Setting->Update->For Developers->Developer Mode should be set to ON
    - Pair device - The Hololens will display a pin for Visual Studio
    - The Device portal for your hololens can be accessed in Settings->Update and Security->For Developers->Device Portal
    - https://docs.microsoft.com/en-us/windows/uwp/debug-test-perf/device-portal-hololens

- Select the “Release x86 Device” then press deploy



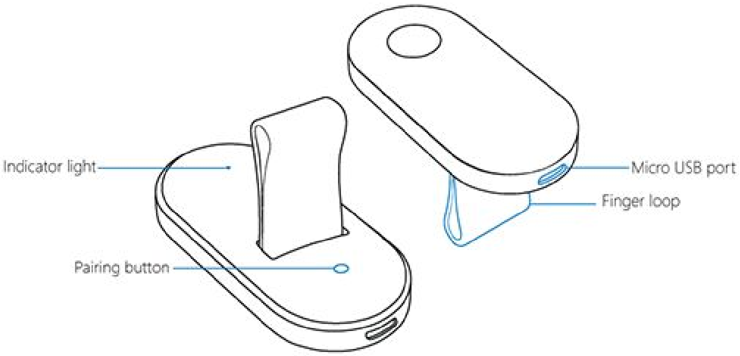


Application User Guide



- Tap on the Menu Button to open the Menu
- Menu options
    - Tap on X to close the Menu
    - Tap on Reset Area to place the Terrain and the launchers again
    - Tap on Reset Target and Launcher to place the target and the launcher again

- First ‘Tap’ is to place the Terrain for our scene
- Second ‘Tap’ is to place the Launcher
- Third tap is to place the Target
- The Next Taps launch missiles
- Tap and Hold to pick up and move the launcher or the Target
- Full Scale mod
    - Rotation tap+hold
- Double Tap on the launcher to open Detailed view
- Hololens Clicker



Application Server Multiplayer Guide

- Run the SharingService.exe from the desired computer
- Launch the WarRoom on the hololens that will give the presentation
- Enter the IP address of the computer running SharingService.exe into the keypad and press enter
    - Or press the offline button to view the scene alone offline without a server
- Wait for the hololens to display the text spawn to tap
    - If there is a connection problem there may be a compatability issue with the firewall on the wifi/internet, try again with a diffrent IP address or on a different network
- Spawn the terrain in the master Hololens
- Launch the WarRoom on all other hololenses
- Enter the correct IP addresses into other hololenses and wait for each hololens to download the necessary anchors to sync the locations of the objects in the room
- The Master can now proceed with the intended display

Features and Implementations

- Menu
    - Script LerpToPosition.cs
        - Float StoppingDistance
            - The distance from the center of the gaze of view in which the menu will stop centering itself
        - Float MovingDistance
            - The distance from the center of the of the gaze in which the menu will start to center itself
        - The Menu will lerp to the location of the object “position” which is a child of the hololens camera
            - Changing the Transform.Position.Z of the object “position” changes the distance from the camera in which the menu will appear
    - Script LookAtScript
        - GameObject TargetToLookAt
            - The Target that the canvas will Look at - if null the Default is main menu
    - Script RemoveMenuIfViewer
        - Removes the menu in all hololenses except from the one giving the presentation

- Sharing
    - Script SharingStage - (Additions)
        - Bool AutoUpdateIP
            - If true the IP address will be set to the last stored IP address used or to the IP address inputted into the keypad
    - Script MasterManager
        - Bool IAmMaster
            - This boolean lets the hololens know if he is the master of the simulation. There is only one master and it is the first hololens to enter the simulation/create the room.

- Placement + Spawning on other hololenses
    - Script SpawnStartArea
        - GameObject StartAreaPrefab
            - The prefab spawned for the terrain
        - Bool IsSpawnStartArea
            - Allows the spawning of the terrain.
        - Public void Reset()
            - Resets the terrain (and all other spawned objects)
    - Script SpawnObjectsWithAirTap
        - List<GameObject> objectsToSpawn
            - A list of the objects that will be spawned, in this order, The last object will be spawned continuously after all the previous objects have been spawned.
        - Void SpawnNext()
            - Spawns the next object from the list objectsToSpawn
        - Public void Reset()
            - Resets all spawned objects other than the terrain

- Specs viewer
    - Script AutoRotation2
        - Float Speed
            - Sets the rotation speed
        - Bool rotating
            - If true will rotate the object in the Specs mode
    - Prefab rotation_mode_rock_tank
        - In The child Textbox's child Text, The text in the field text is what is displayed during specs mode.