Senior Design
ENG EC 463

Memo

To: Professor Pisano, Professor Osama AlShaykh, Jiawei Chen

From: Allison Durkan, EJ Fitzpatrick, Johan Ospina Buitrago, Santiago Beltran, Tess Gauthier

Team: 13

Date: 11/12/16

Subject: Ventana First Deliverable Test Report

1.0 Project Objective

Using Microsoft HoloLens, we will create a platform to find, connect, track, and control existing internet-connected devices in the home. This will be split up into three facets: design the platform that allows these Internet of Things devices to speak to HoloLens, build a piece of hardware that can act as a marker to be tracked by HoloLens, and create an application for the HoloLens that allows it to control the interface of our platform.

2.0 Test Objective and Significance

The test demonstrated that a HoloLens user has the ability to interact with another internet-connected device, a Sonos speaker. This proved that the HoloLens can, in fact, control other internet-connected devices in the home, thereby facilitating a simplified experience between humans and smart devices. The first step in testing involved powering on the HoloLens and Sonos speaker, then verifying the connectivity of both devices to the local server. Then, the test ensured the compilation and deployment of Ventana to the HoloLens, when the user opened the program. Finally, the HoloLens user gazed at the Sonos speaker, saw the holographic display of album artwork, and controlled the device by pressing buttons on the hologram. Upon successfully completing these tasks, expansion of the Ventana platform to other Internet of Things devices, such as lights supported by a Wink hub, can begin.

2.1 Initiate HoloLens and Sonos Speaker

In order to use our application and interact with the IoT devices in the room, we ensured that the HoloLens and the Sonos speaker turned on correctly. This step also verified that both the HoloLens and the Sonos speaker were connected to the internet, which was required in order for the Ventana application to talk to the server. The HoloLens and the Sonos speaker need to be connected to the same network, so this tested that network.

2.2 Launch HoloHub Server

The HoloHub established the local connection between IoT devices and the HoloLens, via a python Flask app. This test observed the functionality of the API server and ensured that it can communicate with the Sonos Speaker. In order for the server to run, the raspberry pi that hosted the code needed to be powered on and connected to the same network that the HoloLens and Sonos speaker were on, “tenda_48e238”. On boot, the raspberry pi runs a cronjob that launches a script containing the necessary instructions to start the server, then it logs the results from the server in a cronlog that is accessible on the raspberry pi to the user. This test established that the HoloHub, the server, was connected to the correct network, and that the server code had launched and was running.

2.3 Deploy Ventana

Ventana needed to be opened to run, like any application. In this step, we navigated to the Hololens’ main menu, and opened Ventana. After opening Ventana, we were able to proceed to the next test and explored the full functionality of the project thus far.

2.4 Recognizing Internet of Things Device

This test examined the ability of the HoloLens’ built-in camera to identify an image target when using Vuforia, an image recognition software-development kit. Vuforia detected when the user gazed upon an image target on the Sonos speaker, and appropriately positioned the holographic display in relation to it. This allowed control over the amount of information provided to the user at any given moment, which represents an essential aspect of the project. The test also quantified the maximum allowable distance between a user and the image target. Since the project focuses on devices in the home, the maximum range represents an important metric for further development of the platform.

2.5 Display Information about Internet of Things Device

The primary goal of displaying information about an IoT device contributes to Ventana’s overall objective of making devices in the home easier to use. In this case, displaying album artwork for a current song represented an enhanced user experience for a Sonos speaker, and moving forward, with other devices this will include more critical information. The test examined the lag time, if any, for the album artwork to update when the song changed, and the accuracy of the album artwork when different songs were played.

2.6 Control Internet of Things Device Using Holographic Display

To test the functionality of the four control buttons, the user gazed at the button she wanted to select. Once the gaze was on the correct button, the user "tapped" with the HoloLens' tapping motion. This was repeated for all buttons on the display; the play, the pause, the next song, and the previous song. When the current song was paused, that button changed to a play button, and the song resumed playing upon selection of that play button. This was repeated multiple times for every button by multiple HoloLens users to ensure selection functionality worked correctly.

3.0 Equipment and Setup

3.0.1 List of Equipment

To test our project, we had a Microsoft HoloLens ,a Sonos speaker, a Vuforia Image Target, a Raspberry Pi 2, and a Tenda router.

3.1 Initiate HoloLens and Sonos Speaker

To set the HoloLens up for the first test case, we put the HoloLens on the user's head and adjusted the band for comfort. The inner band rests at the top of the forehead and gets tightened until the HoloLens is snug. Then the front part of the device is moved in front of the eyes, with the nose bridge centered on the user's nose. This allows the user to press the power button on the back of the HoloLens comfortably to pass the first test case.

To set the Sonos speaker up for the first test case, we plugged the power cord into a wall socket and launched the Sonos application on a laptop. From that application, we connected the Sonos speaker to the correct wireless network and started a playlist.

3.2 Launch HoloHub Server

To set up the HoloHub Server, we had to take a raspberry pi and plug the power cord into a wall socket. We then plugged it into an ethernet port via the on-board ethernet adapter. We cloned the *Ventana-Server* GitHub repository we created into a folder on the Desktop of the raspberry pi, which runs with a script called on boot from a cronjob. Once it was connected to our network and set up with our server code, we were able to launch the server in the second test case.

3.3 Deploy Ventana

In order to deploy Ventana, the application must be exported to the HoloLens from Unity. With Ventana opened in Unity, the exporter navigated to file menu then selected “build settings.” In the build menu, under the scenes box, the main scene box was checked, and under the platform box, the windows store was checked. On the right-hand side, universal 10 was selected as the sdk, D3D was selected as the UWP build type, and local machine was selected as build and run on. Lastly, under debugging, the unity C# box was checked. With these settings selected, the exporter then pressed build. This created a Microsoft Visual Studio Solution, which was then opened in Visual Studio. On the top toolbar in Visual Studio, the exporter selected release mode, x86, and device. The exporter then pressed the green play button, and then received a successful deploy message in Visual Studio. Once Ventana has been successfully deployed from Visual Studio, it is accessible from the HoloLens’ start menu for any user to open.

3.4 Recognize Internet of Things Device

To test the recognition of the Internet of Things device, in this test case, a Sonos speaker, we had to put the Vuforia image marker that we connected to the Sonos speaker in Unity. The image marker is displayed below in Figure 3.4.1. The figure is also the correct orientation, because Vuforia detects what orientation the image marker is at so it can place the Unity objects in the correct place in reference to the image. By attaching this image marker in the correct orientation, we were able to recognize the Sonos speaker and pass this test case.

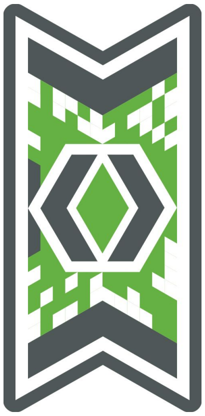


Figure 3.4.1

3.5 Display Information about Internet of Things Device

To display information about our Sonos speaker, we used the results from the previous test cases, i.e. recognizing the device and the deployment of Ventana. When Ventana was deployed and the application recognized the view marker placed on the Sonos speaker, Ventana was able to associate that device with the current playlist of that Sonos speaker.

3.6 Control Internet of Things Device Using Holographic Display

To set up this test case, the user had to have completed the other test cases and had Ventana deployed on the HoloLens. To interact with the Holographic Display, the user had to direct her gaze at the correct action item on the hologram.

4.0 Measurements and Data

4.1 Initiate HoloLens and Sonos Speaker

- Turning on the HoloLens was successful upon pressing the power button in Figure 4.1.1.

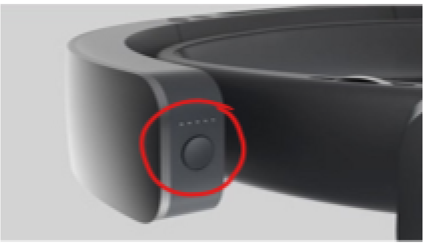


Figure 4.1.1

- Upon pressing the Hololens with the Bloom gesture, the main menu was activated as seen in Figure 4.1.2. This demonstrated the correct behavior and setup with both the wireless connection setup correctly and Ventana application stored in the HoloLens. The wireless connections reads, “Tenda_48e238” as expected from our test plan.

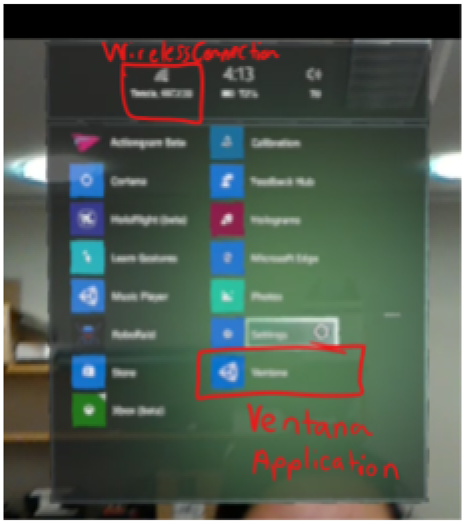
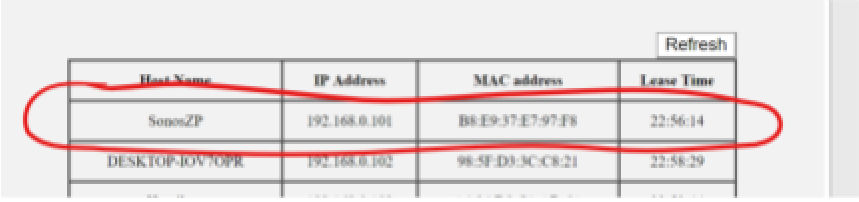


Figure 4.1.2

- Successful connection between the Sonos Speaker and the router was validated via the DHCP Client connection which displayed the dynamic IP address assigned by the router to the speaker indicating successful internet connection. As seen in figure 4.1.3:



Host Name	IP Address	MAC address	Lease Time
SonosZP	192.168.0.101	B8:E9:37:E7:97:F8	22:56:14
DESKTOP-IOV7OPR	192.168.0.102	98:5F:D3:3C:C8:21	22:58:29

Figure 4.1.2

4.2 Launch HoloHub Server

- Powering on the router resulted in five status turing on. The Power and System lights indicate proper activity on the router and are determined a normal behavior. The WLAN light indicates the wireless radio is activated, as expected. The Port 4 light indicates there is an active link between the Raspberry Pi and the router. Lastly the WAN light indicates there is a connection between the BU ethernet port and the router. This is demonstrated in figure 4.2.1 below.

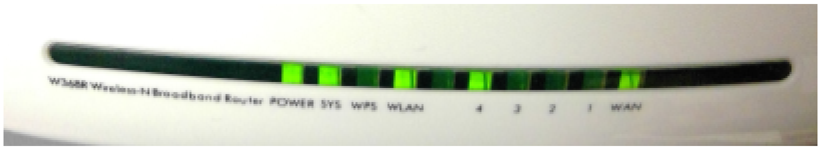


Figure 4.2.1

- On the Raspberry Pi, we observed two status indicators showing correct setup of the hardware. Both the power light and network activity light showed successful indicator lights, Figure 4.2.2.

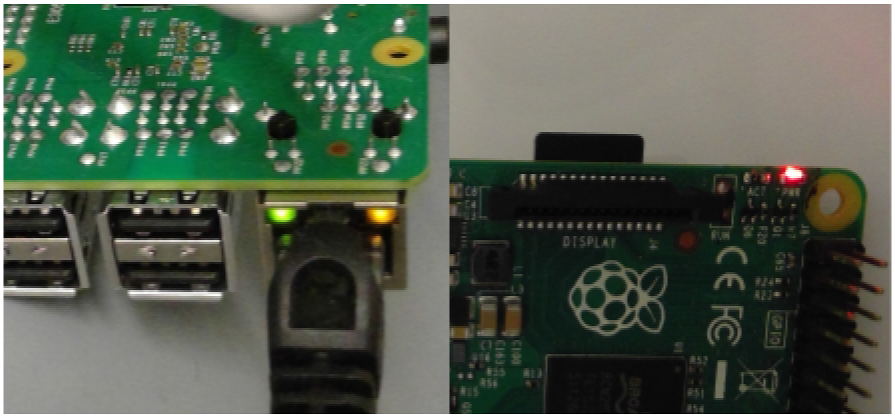
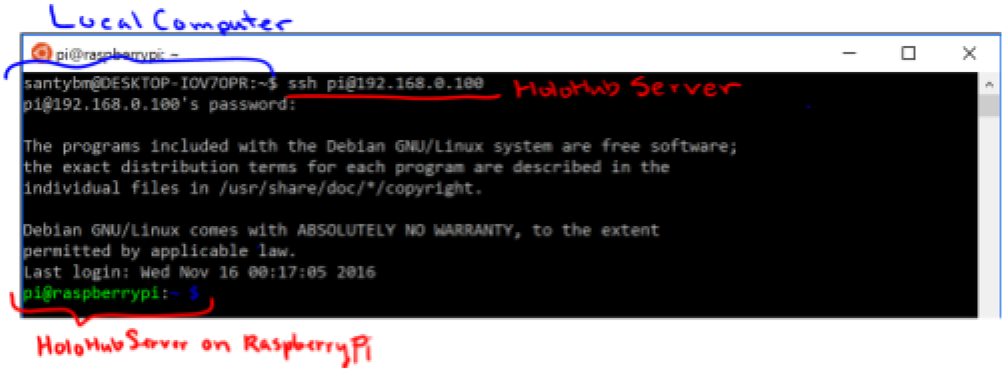


Figure 4.2.2

- From a local computer, we tested the connection and status of the server on the Raspberry Pi. By running the SSH command we observed correct behavior asseen in figure 4.2.3:



```
santybm@DESKTOP-IOV7OPR:~$ ssh pi@192.168.0.100
pi@192.168.0.100's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Nov 16 00:17:05 2016
pi@raspberrypi:~ $
```

Figure 4.2.3

4.3 Deploy Ventana

- Test was verified by seeing the ‘Ventana’ icon on the start screen on the hololens. Once we see the ‘Made in Unity’ logo, the application launch is determined to be successful as seen in Figure 4.3.1.

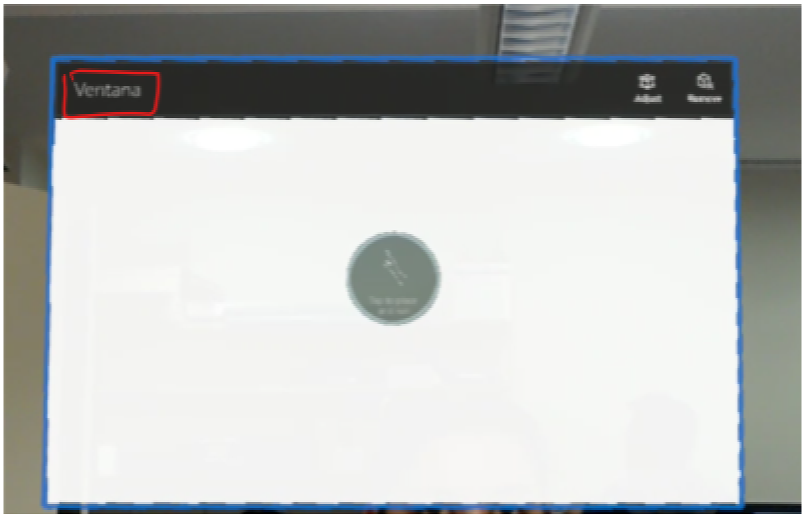


Figure 4.3.1

4.4 Recognize Internet of Things Device

- Wearing the HoloLens and looking at the Sonos speaker, we were able to recognize the image target and the hologram appeared indicating success.

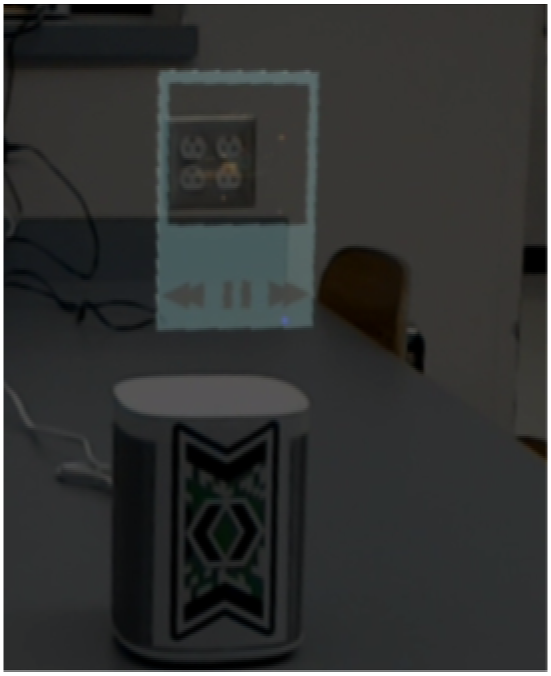


Figure 4.4.1

4.5 Display Information about Internet of Things Device

- The controller floating above the speaker should display the correct album art and the correct state for the play button. If music is playing it should show a pause button and it should show a play button if the music is paused. We observed both in our tests. Figure 4.5.1 shows how the album artwork changes dynamically and that while playing music, the play/pause button displays the correct state

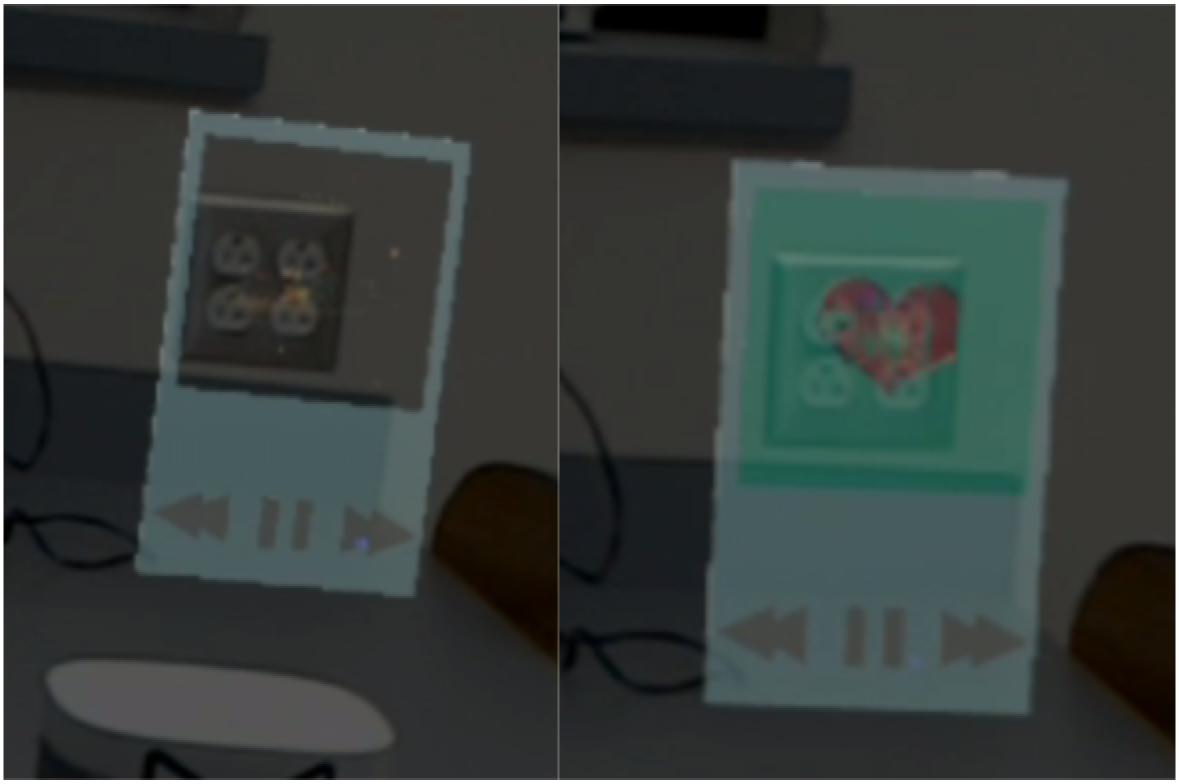


Figure 4.5.1

4.6 Control Internet of Things Device Using Holographic Display

- In the holographic controller we should be able to do 4 actions. play, pause, skip track, and go to previous track. The figure below shows each action and the desired behavior before and after the action is completed.

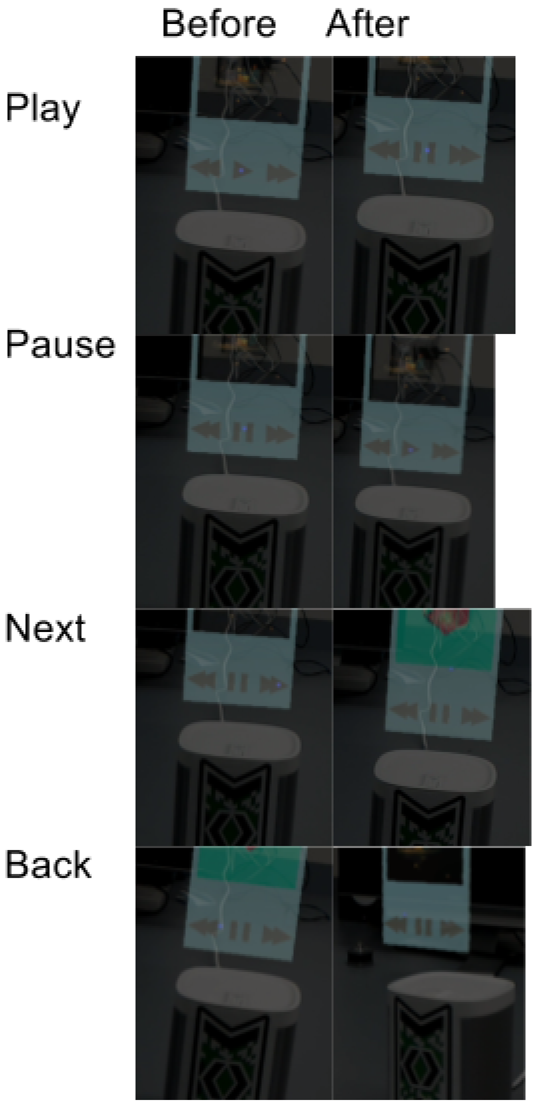


Figure 4.6.1

5.0 Conclusions

Through successfully completing this phase of testing, we have proven that the camera on the HoloLens can successfully use Vuforia image targets to locate and interact with holographic objects. This image recognition technology paired with the augmented reality technology of HoloLens allowed us to deploy an initial implementation of a location-aware holographic interface. In fact, through our testing we found that at close ranges the HoloLens can effectively use the image targets to serve up a context aware user interface. Furthermore, the architecture proposed herein has proven to have both little user lag time and scalability as it stands now.

Looking forward, the team will need to further branch out from the comfort zone of a Sonos speaker which will mean reworking the Ventana application to accommodate more than one 3D model at a time. In addition to this architectural change, some thought will need to be placed on how to approach long distance positional tracking of IoT devices in the home. This approach could be a hardware component for this project that will allow for more a precise and unobtrusive way to detect a device, as well as monitor its locational data. Finally, improvements need to be made with regards to how the application queries the server for updates so that Ventana is not unnecessarily wasting network resources by polling the server every second. Though the system works as it is now, this change will be pivotal in expanding the number of devices that Ventana can support in the near future.