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: 11/15/16

Subject: Ventana First Deliverable Test Plan

1.0 Initiate HoloLens and Sonos Speaker

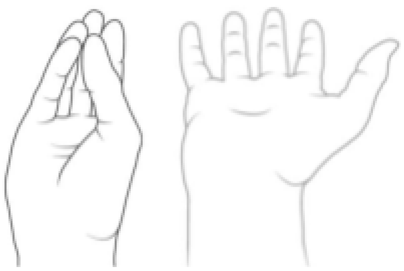
1.1 Description & Goal

In order to use our application and interact with the Internet of Things devices in the room, we need to make sure the HoloLens and the Sonos speaker turn on correctly. This step will also ensure that both the HoloLens and the Sonos speaker are connected to the internet, which is 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, which is the current network that our server is connected to, so this tests and verifies that network.

1.2 Procedure

There are two steps to make sure the HoloLens and the Sonos speaker are launched correctly. First, press the power button on the back of the HoloLens. Place the HoloLens headset on your head and adjust the band for comfort. Next, press the power button on the top of the Sonos speaker.

After both the HoloLens and Sonos speaker have been turned on correctly, we need to ensure they are both connected to "tenda_48e238," the wireless network. To test the HoloLens' network, we will open the Start Menu. Use bloom, a built-in HoloLens gesture, to launch the Start Menu. To bloom, put your arm out with your palm facing upwards, and bring your fingers together, so they are touching. Then, release your fingers outward in one motion. This motion is depicted in figure 1.2.1. When the menu appears, you should see the wireless connection in the upper left of the menu, with the correct name appearing below the wireless connected image.



Bloom Gesture

FIGURE 1.2.1

To verify that the Sonos speaker is connected to the correct network, we need to connect a laptop to “tenda_48e238,” the wireless network. From that computer, we can pull up a web browser and go to the IP address of the router, 192.168.0.1. Then we click “Advanced Settings”, “DHCP Client Server” and “DHCP Client List”. This will list all the devices connected to this router. One of them will be the Sonos speaker, called “SonosZP”. During this step we can double-check the HoloLens is connected because it will also appear in this list.

1.3 Verifiable Result

The HoloLens power button lights up, and once the headset is on, it will probably display a “mapping your surroundings” message. The Sonos speaker power button will light up. These indicate that the devices are powered on correctly. In the Start Menu on the HoloLens, there should be a wireless symbol with the network “tenda_48e238” listed below. The router’s dhcp services list will show the Sonos device at the IP address 192.168.0.104. These two results indicate that the HoloLens and the Sonos speaker are connected to the correct network.

2.0 Launch HoloHub Server

2.1 Description & Goal

The HoloHub establishes the local connection between Internet of Thing devices and the HoloLens via a python Flask app. This test observes the functionality of the API server and ensures that it can communicate with the Sonos Speaker. In order for the server to run, the raspberry pi that hosts the code needs to be powered on and connected to the same network that the HoloLens and Sonos speaker are 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 establishes that the HoloHub, the server, is connected to the correct network, and that the server code has launched and is running.

2.2 Procedure

The first step of launching the server is to power on the raspberry pi that contains the server code. This consists of plugging the power cord from the raspberry pi into a socket. The raspberry pi should also be connected by an ethernet code to the router whose network is “tenda_48e238”. A separate computer on the same network, like the one used in the previous test, should be used to SSH into the raspberry pi. This can be done by checking the IP address of the raspberry pi through the router interface previously discussed, which should be 192.168.0.101. The command to ssh from a terminal is “ssh pi@192.168.0.101” and when prompted for a password, enter “raspberry”. Once SSH’ed into the pi, run the command “curl -XGET “localhost:5000”” which will make sure the server is transmitting on port 5000.

2.3 Verifiable Result

Once powered on and connected to the network of the router, the ethernet jack status light on the Raspberry Pi will be lit. Once the curl command is entered, a json response should be returned.

3.0 Deploy Ventana

3.1 Description & Goal

Ventana needs to be opened to run, like any application. In this step, we will navigate to the Hololens main menu, and open Ventana. After opening Ventana, we will be able to explore the full functionality of the project.

3.2 Procedure

Use bloom, the built-in HoloLens gesture mentioned previously, to launch the main menu. Interact with the main menu as you would with a typical touch screen. Place your hand in the shape of an “L” until a cursor appears on the application labeled Ventana, listed on the main menu, and then tap to launch it. Tap is another built-in HoloLens gesture, displayed below in Figure 3.2.1. The program window will open, and need to be placed in the user’s world. Tap anywhere to place the window, as its location will not affect how Ventana runs.

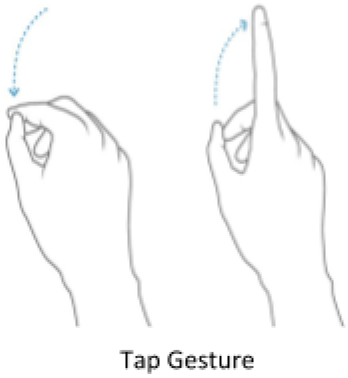


FIGURE 3.2.1

3.3 Verifiable Result

A “Made with Unity” logo will show up, which means Ventana has started loading. Once that screen disappears, Ventana has fully launched, and the user can begin to utilize it.

4.0 Recognize Internet of Things Device

4.1 Description & Goal

Using gaze recognition, when the HoloLens’ camera detects a device supported by Ventana in its field of view, a holographic display, tailored to the device, will automatically appear in the user’s world. In order to implement this, we utilized Vuforia, an image recognition software-development kit. Vuforia connects actions with specific pre-defined images, known as Image Targets or VuMarks, which allows us to connect the holographic display to the Sonos speaker's position. The test examines the capabilities of Vuforia, and the maximum distance allowed between a HoloLens' and the Internet of Things Device.

4.2 Procedure

With the HoloLens on and the Ventana application running, look at the Sonos speaker with the gaze circle appearing on the hologram. The user should back away from the Sonos speaker slowly, approximately a foot at a time, until the user can no longer see the holographic display. We will record how far away the user can be from the device, while still being able to see the holographic display.

4.3 Verifiable Result

An interactive holographic display will show up near the image target.

5.0 Display Information about Internet of Things Device

5.1 Description & Goal

Displaying information about an Internet of Things 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 represents an enhanced user experience for a Sonos speaker, and moving forward, with other devices this will include more critical information. The test examines the lag time, if any, for the album artwork to update when the song changes, and the accuracy of the album artwork when five different songs are played.

5.2 Procedure

With the Ventana app opened, the first step will be to look at the Sonos speaker while a song is playing. The Ventana hologram will appear and display forward, back, and pause buttons. Change the song at least four times by navigating the Sonos application on the computer. Check the accuracy of the information displayed, specifically the album artwork, by comparing it to the Sonos application on a separate computer.

5.3 Verifiable Result

A holographic display of the album artwork for the current song will appear in the user's world above the IoT device, in this test a Sonos speaker.

6.0 Control Internet of Things Device Using Holographic Display

6.1 Description & Goal

The holographic display will have next song, previous song, and pause buttons that the user can select. The buttons are Unity assets created using Blender, a 3D modeling software. Since the Sonos speaker and Hololens are connected to the same local Internet server, being run on a Raspberry Pi 3, the control inputs to the Holographic display can be sent to the Sonos speaker, controlled by SoCo, a Python class. The test observes the functionality of the display, and the responsiveness of each of the buttons.

6.2 Procedure

To test the functionality of the next, previous, and pause/play buttons, the user should direct her gaze at the button she wants to select. Once the gaze is on the correct button, the user should "tap" with the HoloLens tapping motion. This should be repeated on all buttons on the display. When the current song is paused, that button will change to a play button, and the song will resume playing upon selection. Repeat this at least three times with every button to ensure selection functionality works correctly.

6.3 Verifiable Result

Once the pause button is pressed, music will stop playing from the Sonos speaker and that button will change to a play button. Once the play button is pressed, music will begin playing again from the Sonos speaker and the button will change back to a pause button. Once the next song button is pressed, the next song in the playlist will begin playing. Once the previous song button is pressed, the previous song in the playlist will begin playing. These expected changes can be heard by everyone in the room, and verified by comparing to the current song status on the Sonos application on a computer.

7.0 Appendix

Reference for all images:

https://support.microsoft.com/en-us/help/12644/hololens-use-gestures