**Test Case Template-- ECE 458 Spring 2020**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test Name | TV Remote | | Test Number | 4 | | | |
| Requirement(s) Tested | | 3 & 4 | Verification Method | I | A | D | T |
| Test Setup  -Include HW or SW Versions  -Attach Diagrams as appropriate | | 1. Raspberry Pi 3B+    1. OS: Raspbian v10 2. Connections to Pi:    1. USB Sound Card into USB port    2. HDMI cable into HDMI port    3. Micro USB into Micro USB port    4. Ethernet cable into Ethernet port    5. Keyboard & Mouse into USB ports    6. 3.5mm cable into Microphone (Pink) port of USB soundcard    7. IR Remote shield connected to pins 1 to 26 3. Connections to TV:    1. Micro USB cable into USB port    2. 3.5mm cable into 3.5mm headphone port    3. Power cable into power port    4. Antenna into Coaxial port 4. Other Connections:    1. Connect HDMI cable to secondary Monitor    2. Connect TV Power cable to power outlet 5. Other Materials:    1. Westinghouse TV remote 6. TV Settings:    1. Powered On    2. TV Input set to Raspberry Pi 3B (HDMI) | | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Step | Action (Attach test data, diagrams, etc. as appropriate) | Expected Result | Observed Result | Pass/  Fail |
| 1. | Point the system remote to the Raspberry Pi and press **BUTTON 1** | Commercial detection program will begin to search for commercials with mute selected | Commercial detection program is not completed at this time. | Fail |
| 2. | When commercial is detected send a mute command from IR transmitter to the Westinghouse TV | Westinghouse TV displays mute, Audio symbol with X over it, and mutes volume output | Westinghouse TV audio has been stopped and can no longer hear TV audio. Audio symbol with X over it is shown on the bottom left corner of the TV display. | Fail. See comments for clarification |
| 3. | When the commercial end is detected send a mute command from IR transmitter to the Westinghouse TV | Westinghouse TV display for mute disappears and audio level is restored | Westinghouse TV audio has been restored and can hear the TV program. The mute icon on the TV has disappeared. | Fail. See comments for clarification |
| 4. | Point the system remote to the Raspberry Pi and press **BUTTON 2** | Commercial detection program will begin to search for commercials with volume decrease selected | Commercial detection program is not completed at this time. | Fail |
| 5. | When commercial is detected irsend a KEY VOLUMEDOWN command from IR transmitter to the Westinghouse TV | Westinghouse TV displays volume level bar going down ten button presses | Westinghouse TV audio from the speakers has been lowered. This is also indicated with a volume level bar shown on the bottom of the TV display. A vertical bar indicating the current audio level is moved by 10 spaces to the left. | Fail. See comments for clarification |
| 6. | When the commercial end is detected irsend a KEY VOLUMEUP command from IR transmitter to the Westinghouse TV | Westinghouse TV displays volume level bar going up returning to its previous volume | The volume level bar reappears on the TV display and the vertical bar goes to the right 10 spaces. The TV audio has been increased to its original volume level. | Fail. See comments for clarification |
|  |  |  |  |  |
|  | On the Desktop, select **StartingPycharm.sh** | Execute File dialog box appears | The File or Project window is shown on the computer monitor | Pass |
|  | Select **Execute** in the Execute File dialog box | Pycharm opens to last opened file | The final project file has not been created. It opens up the most recent project file | Fail |

|  |
| --- |
| Comments  Since the commercial detection program has not been completed at this time, the team could not test this case with the final python program. The final python program will integrate the Westinghouse TV remote, the MuteBot system remote and the Commercial Detection python files. In order to at least test that the Westinghouse TV will either mute or decrease/increase its speaker volume, a python file was created to wait for a button press to occur on the MuteBot system. After receiving a button press, the code would send an IR signal to the Westinghouse TV. The team was able to mute, lower, and increase the TV volume. All tests for this test case passed but none with the final python file because we did not have the commercial detection program completed. |

Date March 6, 2020 Test Engineer Thomas Morrissey Witness Steve Ferreira