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

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test Name | System Remote | | Test Number | 2 | | | |
| Requirement(s) Tested | | 3 & 6 | 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** | Python console output  displays “++Algorithm is ON, Mute is selected” | “++Algorithm is ON, Mute is selected” is shown on the computer’s display. | Pass, see comments for clarification |
| 2. | Point the system remote to the Raspberry Pi and press **BUTTON 1** | Python console output displays “++Algorithm is OFF” | “++Algorithm is OFF” is shown on the computer’s display. | Pass, see comments for clarification |
| 3. | Point the system remote to the Raspberry Pi and press **BUTTON 2** | Python console output displays “++Reduced Volume ON” | “++Reduced Volume ON” is displayed onto the computer screen. | Pass, see comments for clarification |
| 4. | Point the system remote to the Raspberry Pi and press **BUTTON 2** | Python console output displays “++Reduced Volume OFF” | “++Reduced Volume OFF” is shown on the computer’s display | Pass, see comments for clarification |
|  |  |  |  |  |
|  | On the Desktop, select **StartingPycharm.sh** | Execute File dialog box appears | The Execute File dialog box is shown on the computer monitor. | Pass |
|  | Select **Execute** in the Execute File dialog box | Pycharm opens to last opened file | Pycharm boots up and displays the python code from the most recent file | Pass |
|  | Select **File** in top left corner, Select **Open** | Open File or Project window opens | The File or Project window is shown on the computer monitor | Pass |
|  | Type in search bar **/home/pi/remotecontrol\_v1/buttonPressed.py**  Then press **OK** | The buttonPressed program will open | The buttonPressed project file opens after clicking OK. The python code from this file is shown on the computer monitor. | Pass |
|  | Select **Run** at the top of the screen, Select **Run “buttonPressed”** | Python console will open and button presses will now be read | PyCharm’s console window appears on the computer monitor and waits for buttons to be pressed from the MuteBot system. | Pass |

|  |
| --- |
| Comments  The team did some prototyping with MuteBot IR remote to see if a button press could start and stop the commercial detection program. When prototyping, the team found out that the IR remote shield on the MuteBot can only receive or transmit signals to the Raspberry Pi. The IR expansion board cannot send and receive IR signals simultaneously with the software LIRC; which controls the IR transmitter and receiver. There must be a change to the lirc\_options file in order to change the IR expansion board from sending to receiving IR signals. After this change in the file, the Raspberry Pi must be rebooted in order to apply the changes. With time being the number one priority, the team decided to use the IR expansion board’s two buttons to run the tests for this test case. All tests passed with the buttons but not with the MuteBot system’s IR remote since it was not incorporated into the system yet. The team has been researching to see if the file change in LIRC could be done differently so that the Raspberry Pi didn’t have to be rebooted every time the file is changed. Since students can now longer be on campus, the team cannot do any testing to see if this can be accomplished with the MuteBot’s IR remote. |

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