Use Cases for SBC Remote

## Table of Contents

[1.1 Rename a GPIO pi](#h.8c5axl4nl0f4)n ……………………………………………………………………………. 2

[1.2 Change a pin’s high and low labels](#h.tfcjlftzesh) …………………………………………………………….. 3

[1.3 Change a pin’s state](#h.l52hdkgszje3) ……………………………………………………………………………. 4

[1.4 Enable push notifications](#h.u3ofwjt4np2w) ……………………………………………………………………….. 5

[1.5 Set a pin to Monitor, Control, or Ignore](#h.sln2v7uxd4c8) ……………………………………………………..….. 6

[1.6 Set a pin’s natural state](#h.qh4coru3wmfk) ………………………………………………………………………… 7

# 1.1 Rename a GPIO pin

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 1.1 | | |
| Use Case Name: | Rename a GPIO pin | | |
| Created By: | Peter Welsh | Last Updated By: |  |
| Date Created: | April 15, 2016 | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: | User |
| Description: | The user gives a name to a specific GPIO pin. |
| Preconditions: | 1. User has a WebIOPi-enabled Raspberry Pi. 2. User has registered an account with Weaved. |
| Postconditions: | Specific GPIO pin has a new name |
| Priority: | Medium |
| Frequency of Use: | Infrequent |
| Normal Course of Events: | 1. User opens app from his Android or iOS smartphone. 2. User logs in to Weaved 3. User selects a Pi from the list 4. User goes to Configuration screen 5. User taps the name of the pin he wants to rename 6. User enters a new name for the pin |
| Alternative Courses: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: | 1. The Pi is on and working 2. The user has internet access |
| Notes and Issues: |  |

# **1.2** Change a pin’s high and low labels

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 1.2 | | |
| Use Case Name: | Change a pin’s high and low labels | | |
| Created By: | Peter Welsh | Last Updated By: |  |
| Date Created: | April 15, 2016 | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: | User |
| Description: | Change the label displayed when a pin is in its high or low state. |
| Preconditions: | 1. User has a WebIOPi-enabled Raspberry Pi. 2. User has registered an account with Weaved. |
| Postconditions: | Specified pin’s high state now has a label other than High (default) |
| Priority: | Medium |
| Frequency of Use: | Infrequent |
| Normal Course of Events: | 1. User opens app from his Android or iOS smartphone. 2. User logs in to Weaved 3. User selects a Pi from the list 4. User goes to the Configuration page 5. User clicks the label of the pin he wants to relabel 6. User types the new name for the pin’s high or low state |
| Alternative Courses: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: | 1. The Pi is on and working 2. The user has internet access |
| Notes and Issues: |  |

# 1.3 Change a pin’s state

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 1.3 | | |
| Use Case Name: | Change a pin’s state | | |
| Created By: | Peter Welsh | Last Updated By: |  |
| Date Created: | April 14, 2016 | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: | User |
| Description: | User changes the state of a GPIO pin on his Pi from high to low or from low to high. |
| Preconditions: | 1. User has a WebIOPi-enabled Raspberry Pi. 2. User has registered an account with Weaved. |
| Postconditions: | 1. Previously high GPIO pin on Pi is now low.   or   1. Previously low GPIO pin is now high. |
| Priority: | High |
| Frequency of Use: | Frequent |
| Normal Course of Events: | 1. User opens app from his Android or iOS smartphone. 2. User provides login information for Weaved. 3. User selects his Pi from the list of devices presented onscreen. 4. User selects the pin he wants to change from the pins displayed onscreen. 5. User goes to Monitor/Control screen 6. User flips switch from the High position to Low. |
| Alternative Courses: | As above, except   1. User flips switch from the Low position to High. |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: | 1. The Pi is on and working 2. The user has internet access |
| Notes and Issues: |  |

# 1.4 Enable push notifications

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 1.4 | | |
| Use Case Name: | Enable push notifications | | |
| Created By: | Peter Welsh | Last Updated By: |  |
| Date Created: | April 15, 2016 | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: | User |
| Description: | The user goes to the Configuration page and flips the switch enabling push notifications. |
| Preconditions: | 1. User has a WebIOPi-enabled Raspberry Pi. 2. User has registered an account with Weaved. |
| Postconditions: | Push notifications are enabled. All monitor pins will send a push notification on state change. |
| Priority: | High |
| Frequency of Use: | Infrequent |
| Normal Course of Events: | 1. User opens app from his Android or iOS smartphone. 2. User logs in to Weaved 3. User selects a Pi from the list 4. User goes to Configuration page 5. User taps switch enabling push notifications |
| Alternative Courses: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: | 1. The Pi is on and working 2. The user has internet access |
| Notes and Issues: |  |

# 1.5 Set a pin to Monitor, Control, or Ignore

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 1.5 | | |
| Use Case Name: | Set a pin to Monitor, Control, or Ignore | | |
| Created By: | Peter Welsh | Last Updated By: |  |
| Date Created: | April 15, 2016 | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: | User |
| Description: | The user changes a pin from Monitor to Control, Control to Monitor, Monitor to Ignore, Ignore to Monitor, Control to Ignore, or Ignore to Control. |
| Preconditions: | 1. User has a WebIOPi-enabled Raspberry Pi. 2. User has registered an account with Weaved. |
| Postconditions: | Specific pin is now a Monitor, Control, or Ignore pin. Monitor pins have the ability to send push notifications on a state change. Control pins are able to be switched from High to Low. Ignore pins are not shown on the Monitor/Control screen. |
| Priority: | High |
| Frequency of Use: | Infrequent |
| Normal Course of Events: | 1. User opens app from his Android or iOS smartphone. 2. User logs in to Weaved 3. User selects a Pi from the list 4. User goes to Configuration page 5. User selects a specific pin’s M/C/I state |
| Alternative Courses: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: | 1. The Pi is on and working 2. The user has internet access |
| Notes and Issues: |  |

# 1.6 Set a pin’s natural state

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 1.6 | | |
| Use Case Name: | Set a pin’s natural state | | |
| Created By: | Peter Welsh | Last Updated By: |  |
| Date Created: | April 15, 2016 | Date Last Updated: |  |

|  |  |
| --- | --- |
| Actor: | User |
| Description: | The user sets a specific pin’s natural state, i.e. Low or High |
| Preconditions: | 1. User has a WebIOPi-enabled Raspberry Pi. 2. User has registered an account with Weaved. |
| Postconditions: | Specific pin’s natural state is either High or Low, as set by the user |
| Priority: | Medium |
| Frequency of Use: | Infrequent |
| Normal Course of Events: | 1. User opens app from his Android or iOS smartphone. 2. User logs in to Weaved 3. User selects a Pi from the list 4. User goes to Configuration page 5. User selects a specific pin 6. User sets the pin’s natural state (High or Low) |
| Alternative Courses: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: | 1. The Pi is on and working 2. The user has internet access |
| Notes and Issues: |  |