

## Admin Panel

Generated by Doxygen 1.9.5



---

<b>1 Admin Panel</b>	<b>1</b>
1.1 Install Packages: . . . . .	1
1.2 Run: . . . . .	1
1.3 Build EXE to export: . . . . .	1
1.4 Template from: . . . . .	1
<b>2 App</b>	<b>3</b>
<b>3 BehaviourTreeHandler</b>	<b>5</b>
<b>4 CodeBox</b>	<b>7</b>
<b>5 CodeMessagingBar</b>	<b>9</b>
<b>6 Stream</b>	<b>11</b>
<b>7 IPList</b>	<b>13</b>
<b>8 NavigationBar</b>	<b>15</b>
<b>9 RaspberryPiConnectBanner</b>	<b>17</b>
<b>10 RaspberryPiSelector</b>	<b>19</b>
<b>11 SaveBehaviourTreeModal</b>	<b>21</b>
<b>12 CodeBox</b>	<b>23</b>
<b>13 CodeButton</b>	<b>25</b>
<b>14 WebsocketServerConfig</b>	<b>27</b>
<b>15 Home</b>	<b>29</b>
<b>16 NotFound</b>	<b>31</b>
<b>17 Validate</b>	<b>33</b>
<b>18 Websocket</b>	<b>35</b>
<b>19 Class Index</b>	<b>37</b>
19.1 Class List . . . . .	37
<b>20 File Index</b>	<b>39</b>
20.1 File List . . . . .	39
<b>21 Class Documentation</b>	<b>41</b>
21.1 Code Class Reference . . . . .	41
21.1.1 Member Function Documentation . . . . .	41
21.1.1.1 constructor() . . . . .	41

---

21.1.1.2 generate()	41
21.1.1.3 get()	41
21.2 WebSocketServer Class Reference	41
21.2.1 Member Function Documentation	42
21.2.1.1 close()	42
21.2.1.2 connect()	42
21.2.1.3 constructor()	42
21.2.1.4 generateCode()	42
21.2.1.5 getCode()	42
21.2.1.6 getPort()	42
21.2.1.7 getRaspberryPiMap()	43
21.2.1.8 getSelectedRaspberryPi()	43
21.2.1.9 isConnected()	43
21.2.1.10 selectRaspberryPi()	43
21.2.1.11 sendBehaviourTree()	43
21.2.1.12 startBehaviourTree()	43
21.2.1.13 startConnections()	43
21.2.1.14 stopBehaviourTree()	44
<b>22 File Documentation</b>	<b>45</b>
22.1 js/behaviour_tree.js File Reference	45
22.1.1 Function Documentation	45
22.1.1.1 getBehaviourTreeList()	45
22.1.1.2 handle()	45
22.1.1.3 isDev()	46
22.1.1.4 removeBehaviourTree()	46
22.1.1.5 saveBehaviourTree()	46
22.1.1.6 sendBehaviourTree()	46
22.1.1.7 startBehaviourTree()	46
22.1.1.8 stopBehaviourTree()	46
22.1.1.9 updateBehaviourTreeList()	46
22.1.1.10 validateBehaviourTree()	46
22.1.2 Variable Documentation	47
22.1.2.1 base_path	47
22.1.2.2 const	47
22.1.2.3 exec	47
22.1.2.4 fs	47
22.1.2.5 Os	47
22.1.2.6 path	47
22.1.2.7 resources_dir	47
22.2 js/device.js File Reference	47
22.2.1 Function Documentation	47

22.2.1.1 getLocalIPList()	48
22.2.1.2 handle()	48
22.2.2 Variable Documentation	48
22.2.2.1 exports	48
22.2.2.2 os	48
22.3 js/main.js File Reference	48
22.3.1 Function Documentation	48
22.3.1.1 createWindow()	48
22.3.1.2 isDev()	49
22.3.1.3 on()	49
22.3.1.4 onClose()	49
22.3.2 Variable Documentation	49
22.3.2.1 const	49
22.3.2.2 loadURL	49
22.3.2.3 path	49
22.3.2.4 serve	49
22.4 src/main.js File Reference	49
22.4.1 Variable Documentation	49
22.4.1.1 app	49
22.4.1.2 pcss	50
22.4.1.3 svelte	50
22.5 js/preload.js File Reference	50
22.6 js/raspberry_pi.js File Reference	50
22.6.1 Function Documentation	50
22.6.1.1 getRaspberryPiList()	50
22.6.1.2 getSelectedRaspberryPi()	50
22.6.1.3 handle()	50
22.6.1.4 selectRaspberryPi()	50
22.6.1.5 unselectRaspberryPi()	51
22.6.2 Variable Documentation	51
22.6.2.1 const	51
22.7 js/store.js File Reference	51
22.8 js/websocket.js File Reference	51
22.8.1 Function Documentation	51
22.8.1.1 getRandomInt()	51
22.8.2 Variable Documentation	51
22.8.2.1 exports	51
22.8.2.2 WebSocket	52
22.9 js/websocket_events.js File Reference	52
22.9.1 Function Documentation	52
22.9.1.1 closeWebSocketServer()	52
22.9.1.2 getWebSocketServer()	52

22.9.1.3 handle()	52
22.9.1.4 startWebSocketServer()	52
22.9.2 Variable Documentation	52
22.9.2.1 const	52
22.9.2.2 tcpPortUsed	53
22.10 README.md File Reference	53
22.11 rollup.config.js File Reference	53
22.11.1 Function Documentation	53
22.11.1.1 serve()	53
22.11.2 Variable Documentation	53
22.11.2.1 commonjs	53
22.11.2.2 default	53
22.11.2.3 import	53
22.11.2.4 livereload	53
22.11.2.5 only	54
22.11.2.6 polyfills	54
22.11.2.7 production	54
22.11.2.8 resolve	54
22.11.2.9 svelte	54
22.11.2.10 tailwindConfig	54
22.11.2.11 tailwindcss	54
22.12 src/App.svelte File Reference	54
22.13 src/lib/BehaviourTreeHandler.svelte File Reference	54
22.14 src/lib/CodeBox_Constants.js File Reference	54
22.14.1 Variable Documentation	54
22.14.1.1 condition_operator_types	54
22.14.1.2 import	55
22.14.1.3 node_hover	55
22.14.1.4 node_information_list	55
22.14.1.5 range_0_100	55
22.14.1.6 range_0_180	55
22.14.1.7 range_0_360	56
22.15 src/lib/home/CodeBox.svelte File Reference	56
22.16 src/lib/validate/CodeBox.svelte File Reference	56
22.17 src/lib/home/CodeMessagingBar.svelte File Reference	56
22.18 src/lib/home/Stream.svelte File Reference	56
22.19 src/lib/IPList.svelte File Reference	56
22.20 src/lib/NavigationBar.svelte File Reference	56
22.21 src/lib/RaspberryPiConnectBanner.svelte File Reference	56
22.22 src/lib/RaspberryPiSelector.svelte File Reference	56
22.23 src/lib/SaveBehaviourTreeModal.svelte File Reference	56
22.24 src/lib/validate/CodeButton.svelte File Reference	56

22.25 src/lib/WebsocketServerConfig.svelte File Reference	56
22.26 src/routes.js File Reference	56
22.26.1 Variable Documentation	56
22.26.1.1 default	56
22.26.1.2 svelte	56
22.27 src/routes/Home.svelte File Reference	57
22.28 src/routes/NotFound.svelte File Reference	57
22.29 src/routes/Validate.svelte File Reference	57
22.30 src/routes/Websocket.svelte File Reference	57
22.31 src/store/behaviour_tree_store.js File Reference	57
22.31.1 Function Documentation	57
22.31.1.1 main()	57
22.31.2 Variable Documentation	57
22.31.2.1 behaviour_tree_list	57
22.31.2.2 behaviour_tree_save_modal_code	57
22.31.2.3 behaviour_tree_xml_code	57
22.31.2.4 import	57
22.31.2.5 show_behaviour_tree_save_modal	57
22.32 src/store/home_store.js File Reference	58
22.32.1 Variable Documentation	58
22.32.1.1 import	58
22.32.1.2 lidar_toggle	58
22.32.1.3 main_pane_size_0_store	58
22.32.1.4 main_pane_size_1_store	58
22.32.1.5 side_pane_size_0_store	58
22.32.1.6 side_pane_size_1_store	58
22.33 src/store/raspberry_pi_store.js File Reference	58
22.33.1 Function Documentation	59
22.33.1.1 main()	59
22.33.1.2 onConnection()	59
22.33.1.3 onDisconnection()	59
22.33.2 Variable Documentation	59
22.33.2.1 import	59
22.33.2.2 raspberry_pi_uuid_list	59
22.33.2.3 selected_raspberry_pi_uuid	59
22.34 src/store/websocket_store.js File Reference	59
22.34.1 Function Documentation	60
22.34.1.1 main()	60
22.34.1.2 onMessage()	60
22.34.2 Variable Documentation	60
22.34.2.1 camera_stream	60
22.34.2.2 import	60

22.34.2.3 lidar_stream . . . . .	60
22.34.2.4 websocket_server_code . . . . .	60
22.34.2.5 websocket_server_connection_state . . . . .	60
22.34.2.6 websocket_server_port . . . . .	60

<b>Index</b>	<b>61</b>
--------------	-----------



# Chapter 1

## Admin Panel

The app to directly control the Sunfounder PiCar.

### 1.1 Install Packages:

```
pnpm i
```

### 1.2 Run:

```
pnpm run electron
```

### 1.3 Build EXE to export:

```
pnpm run preelectron-pack  
AND  
pnpm run electron-pack
```

### 1.4 Template from:

<https://github.com/soulehshaikh99/create-svelte-electron-app>



## Chapter 2

# App

```
<script> import NavigationBar from "../lib/NavigationBar.svelte"; import Router from "svelte-spa-router"; import routes from "../routes"; </script>
```

```
<NavigationBar > <main> <Router {routes} /> </main>
```

```
<style> main { text-align: center; max-width: 240px; margin: 0 auto; height: 92vh; }
```

```
@media (min-width: 640px) { main { max-width: none; } } </style>
```



## Chapter 3

# BehaviourTreeHandler

```
<script> import { Button, Card } from "flowbite-svelte";

import { behaviour_tree_xml_code, behaviour_tree_save_modal_code, show_behaviour_tree_save_modal,
behaviour_tree_list, } from "../store/behaviour_tree_store";

</script>

<Button on:click={() => { show_behaviour_tree_save_modal.set(true); behaviour_tree_save_modal_code.<←
set($behaviour_tree_xml_code); }} class="w-full my-3" color="green">Save Behaviour Tree</Button> <←
Card class="w-full max-w-full h-full max-h-full overflow-scroll my-4 gap-y-2" > {#each $behaviour_tree_list as
behaviour_tree} <Card class="w-full max-w-full grid grid-rows-2">

{behaviour_tree.name === "" ? "[Empty Name]" : behaviour_tree.name}

<Button color="red" on:click={()=>>{ api.removeBehaviourTree({uuid: behaviour_tree.uuid}); behaviour_tree.<←
list.set($behaviour_tree_list.filter((item) => item.uuid !== behaviour_tree.uuid)); }}>Remove</Button> <Button
color="blue" on:click={()=>>{ $behaviour_tree_xml_code = behaviour_tree.code; }}>Load</Button> </Card>
{/each} </Card>
```



## Chapter 4

### CodeBox

```
<script> // If this does not work then try preview import CodeMirror from "svelte-codemirror-editor"; import { xml }
from "@codemirror/lang-xml"; import { oneDark } from "@codemirror/theme-one-dark";
```

```
import { node_hover, xml_schema } from "../CodeBox_Constants";
```

```
import { behaviour_tree_xml_code } from "../../store/behaviour_tree_store"; import CodeMessagingBar from "../↵
CodeMessagingBar.svelte"; </script>
```

```
<CodeMessagingBar> <CodeMirror class="text-left h-full flex-auto font-mono text-lg font-bold" bind↵
:value={$behaviour_tree_xml_code} lang={xml(xml_schema)} tabSize={4} theme={oneDark} extensions=[[node_↵
hover]] > <style> .cm-editor { height: 90% !important; } .cm-editor * { font-family: ui-monospace, SFMono-Regular,
Menlo, Monaco, Consolas, "Liberation Mono", "Courier New" !important; } .cm-scroller { height: 100% !important; }
</style>
```





## Chapter 5

# CodeMessagingBar

```
<script> import { selected_raspberry_pi_uuid } from "../store/raspberry_pi_store"; import xmlFormat from "xml-formatter"; import { behaviour_tree_xml_code } from "../store/behaviour_tree_store";
```

```
/** @type {string} */ let send_behaviour_tree_text = "Send Behaviour Tree";
```

```
/** @type {string} */ let send_behaviour_tree_color = "#50AA34";
```

```
/** @type {boolean} */ let send_behaviour_tree_debounce = false;
```

```
/** @type {string} */ let start_behaviour_tree_text = "Start Behaviour Tree";
```

```
/** @type {string} */ let start_behaviour_tree_color = "#89AAFF";
```

```
/** @type {boolean} */ let start_behaviour_tree_debounce = false;
```

```
async function sendBehaviourTree() { if (send_behaviour_tree_debounce) { return; } send_behaviour_tree_debounce = true; try { await api.sendBehaviourTree({ data: xmlFormat.minify($behaviour_tree_xml_code), }); send_behaviour_tree_text = "Sent Behaviour Tree!"; send_behaviour_tree_color = "#3457AA"; } catch (error) { send_behaviour_tree_text = "Unable to send Behaviour Tree!"; send_behaviour_tree_color = "#AA3434"; } setTimeout(() => { send_behaviour_tree_text = "Send Behaviour Tree"; send_behaviour_tree_color = "#50AA34"; send_behaviour_tree_debounce = false; }, 1000); }
```

```
async function startBehaviourTree() { if (start_behaviour_tree_debounce) { return; } start_behaviour_tree_debounce = true; try { await api.startBehaviourTree(); start_behaviour_tree_text = "Started Behaviour Tree!"; start_behaviour_tree_color = "#4277FF"; } catch (error) { start_behaviour_tree_text = "Unable to start Behaviour Tree!"; start_behaviour_tree_color = "#AA3434"; } setTimeout(() => { start_behaviour_tree_text = "Start Behaviour Tree"; start_behaviour_tree_color = "#89AAFF"; start_behaviour_tree_debounce = false; }, 1000); }
```

```
async function stopBehaviourTree() { await api.stopBehaviourTree(); } </script>
```

```
<button on:mousedown={sendBehaviourTree} class="p-2 rounded-lg shadow-lg relative inset-0" style="background-color: {send_behaviour_tree_color}; color: white; width: 100%; border: none; {$selected_raspberry_pi_uuid === " ? 'cursor: not-allowed;' : ''}" disabled={$selected_raspberry_pi_uuid === ""} >{send_behaviour_tree_text}</button> <button on:mousedown={startBehaviourTree} class="p-2 rounded-lg shadow-lg relative inset-0" style="background-color: {start_behaviour_tree_color}; color: white; width: 100%; border: none; {$selected_raspberry_pi_uuid === " ? 'cursor: not-allowed;' : ''}" disabled={$selected_raspberry_pi_uuid === ""} >{start_behaviour_tree_text}</button> <button on:mousedown={stopBehaviourTree} class="p-2 rounded-lg shadow-lg relative inset-0" style="background-color: #FF352F; color: white; width: 100%; border: none; {$selected_raspberry_pi_uuid === " ? 'cursor: not-allowed;' : ''}" disabled={$selected_raspberry_pi_uuid === ""} >Stop Behaviour Tree</button>
```



## Chapter 6

# Stream

```
<script> import { onMount } from "svelte"; import { main_pane_size_0_store, lidar_toggle, } from "../store/home_↵
_store"; import { camera_stream, lidar_stream } from "../store/websocket_store";

const lidar_wall_colour = "#0061FF"; const offset_angle = 45; const fps = 30; const fps_interval = 1000 / fps; let
lidar_bar_height_size = 20000;

/**

•
```



## Chapter 7

# IPList

```
<script> import { Button, Card, Spinner, Tooltip } from "flowbite-svelte"; import { copy } from "svelte-copy";

import { websocket_server_port } from "../store/websocket_store";

if (api.getLocalIPList === undefined) { api.getLocalIPList = async function () { return new Promise((resolve) => {
  resolve([]); }); }; }

export let getIPList = updateIPList();

function refreshIPList() { getIPList = updateIPList(); }

function updateIPList() { return api.getLocalIPList().then((local_ip_list) => { let ip_list = []; console.log(local_ip_list);
  for (const [ip_group_name, ip_group] of Object.entries( local_ip_list, )) { for (const ip of ip_group) { if (ip.family === "IPv4") { ip_list.push({ group_name: ip_group_name, address: ip.address, netmask: ip.netmask, mac: ip.mac, internal: ip.internal, cidr: ip.cidr, }); } } } console.log(ip_list); return ip_list; }); } </script>

<Card class="p-6">
```

### Local IP List

```
<Button class="mb-3" id="refresh_ip_address" on:click={refreshIPList} >Refresh IP List</Button> <Tooltip
triggeredBy="#refresh_ip_address" >Remember to determine the appropriate IP Address based on the Raspberry
Pi's network connection.</Tooltip>
```

```
{#await getIPList}
```

Loading IP List

```
<Spinner > { :then ip_list } <Card class="overflow-y-scroll h-screen gap-2 justify-center" style="padding:0px">
{#each ip_list as ip} <Card class="py-6 border-neutral-100 border-b border-t" style="padding:0.5rem">
```

```
{ip.group_name}
```

```
{ip.address}:{$websocket_server_port}
```

```
<button use:copy={$ {ip.address} : {$websocket_server_port}} class="mt-4 p-2" > Copy IP
</button> </Card> {/each} </Card> {/await} </Card>
```



## Chapter 8

# NavigationBar

```
<script> import { DarkMode, Navbar, NavLi, NavUI, NavHamburger, } from "flowbite-svelte";
```

```
let btnClass = "text-gray-500 dark:text-gray-400 hover:bg-gray-100 dark:hover:bg-gray-700 rounded-lg text-xl p-2"
</script>
```

```
<Navbar style="height:7vh;"> <NavUI> <NavLi>  Home </NavLi> <NavLi>  Manage Websockets </NavLi> <NavLi>  Validate Behaviour Tree </NavLi> </NavUI> <NavHamburger class="lg:hidden"
> <DarkMode {btnClass} /> </Navbar>
```





## Chapter 9

# RaspberryPiConnectBanner

```
<script> import { Banner, Button } from "flowbite-svelte"; import { ArrowRightOutline } from "flowbite-svelte-icons";  
</script>
```

```
<Banner class="border-2 z-20 flex-auto p-1 fixed bottom-3 rounded-lg bg-white dark:bg-gray-700 border-gray-500  
dark:border-gray-600" bannerType="info" style="left:50%; transform: translateX(-50%);"
```

### Not Connected!

Connect the Raspberry Pi to the Admin Panel's websocket server by following instructions in the Websocket Tab.

```
<Button href="#/websocket" class="w-fit ml-3"> View Instructions <ArrowRightOutline > </Button>
```

```
</Banner>
```



## Chapter 10

# RaspberryPiSelector

```
<script> import { Button, Card } from "flowbite-svelte";

import { raspberry_pi_uuid_list, selected_raspberry_pi_uuid, } from "../store/raspberry_pi_store";

/** @type {string} */ let selected_uuid = "";

async function connectToRaspberryPi() { if (selected_uuid === "") { return; } console.log(selected_uuid);
const selection_result = await api.selectRaspberryPi({ uuid: selected_uuid, }); console.log(selection_result); if
(selection_result.success) { selected_raspberry_pi_uuid.set(selected_uuid); } else { console.log(selection_result.↵
message); } } </script>
```

## Raspberry Pi UUID List

```
{#each $raspberry_pi_uuid_list as uuid} <Card> <Button on:click={() => { selected_uuid = uuid; }} > {uuid} </↵
Button> </Card> {/each} <Card class="col-start-3 w-full ml-2">
```

## Connect to Raspberry Pi UUID

```
<Button on:click={connectToRaspberryPi}>> {#if selected_uuid === ""} Select a Raspberry Pi UUID {else} Con-
nect to
[{{selected_uuid}}] {/if} </Button> </Card>
```



## Chapter 11

# SaveBehaviourTreeModal

```
<script> import { Button, Input, Modal } from "flowbite-svelte";

import CodeMirror from "svelte-codemirror-editor"; import { xml } from "@codemirror/lang-xml"; import { oneDark }
from "@codemirror/theme-one-dark";

import { xml_schema } from "../CodeBox_Constants"; import { node_hover } from "../CodeBox_Constants";

import { show_behaviour_tree_save_modal, behaviour_tree_save_modal_code, behaviour_tree_list, } from
"../store/behaviour_tree_store";

let behaviour_tree_name = ""; </script>

<Modal title="Save Behaviour Tree" bind:open={$show_behaviour_tree_save_modal} autoclose

    Would you like to save this behaviour tree?

    <Input let:props> <input type="text" {...props} placeholder="Behaviour Tree Name:" bind:value={behaviour_↵
tree_name} required /> </Input> <CodeMirror class="text-left h-full flex-auto font-mono text-lg font-
bold" bind:value={$behaviour_tree_save_modal_code} lang={xml(xml_schema)} tabSize={4} theme={oneDark}
extensions=[[node_hover]] > <svelte:fragment slot="footer"> <Button on:click={() => { const behaviour_tree_↵
_save_state = { name: behaviour_tree_name, code: $behaviour_tree_save_modal_code, }; api.saveBehaviour_↵
Tree(behaviour_tree_save_state); behaviour_tree_list.set([...$behaviour_tree_list, behaviour_tree_save_state]);
}}> Yes</Button> <Button color="alternative">No</Button> </svelte:fragment> </Modal>
```



## Chapter 12

### CodeBox

```
<script> // If this does not work then try preview import CodeMirror from "svelte-codemirror-editor"; import { xml }
from "@codemirror/lang-xml"; import { oneDark } from "@codemirror/theme-one-dark";
```

```
import { node_hover, xml_schema } from "../CodeBox_Constants";
```

```
import { behaviour_tree_xml_code } from "../../store/behaviour_tree_store"; import CodeButton from "../Code↵
Button.svelte"; </script>
```

```
<CodeButton> <CodeMirror class="text-left h-full flex-auto font-mono text-lg font-bold" bind:value={$behaviour↵
_tree_xml_code} lang={xml(xml_schema)} tabSize={4} theme={oneDark} extensions=[[node_hover]] > <style>
.cm-editor { height: 90% !important; } .cm-editor * { font-family: ui-monospace, SFMono-Regular, Menlo, Monaco,
Consolas, "Liberation Mono", "Courier New" !important; } .cm-scroller { height: 100% !important; } </style>
```





## Chapter 13

### CodeButton

```
<script> import xmlFormat from "xml-formatter"; import { behaviour_tree_xml_code } from "../store/behaviour_↵
tree_store";

/** @type {string} */ let validate_behaviour_tree_text = "Validate Behaviour Tree";

/** @type {string} */ let validate_behaviour_tree_color = "#4C9CFF";

/** @type {boolean} */ let validate_behaviour_tree_debounce = false;

async function validateBehaviourTree() { if (validate_behaviour_tree_debounce) { return; } validate_behaviour_↵
tree_debounce = true; try { validate_behaviour_tree_text = "Validating Behaviour Tree"; validate_behaviour_↵
tree_color = "#4C23FF"; const result = JSON.parse(await api.validateBehaviourTree({ data: xmlFormat.↵
minify($behaviour_tree_xml_code), })); if (result.success) { validate_behaviour_tree_text = "The Behaviour Tree
is valid!"; validate_behaviour_tree_color = "#34A83B"; } else { validate_behaviour_tree_text = result.message;
validate_behaviour_tree_color = "#AA3434"; } } catch (error) { validate_behaviour_tree_text = error; validate_↵
behaviour_tree_color = "#AA3434"; } setTimeout(() => { validate_behaviour_tree_text = "Validate Behaviour Tree";
validate_behaviour_tree_color = "#4C9CFF"; validate_behaviour_tree_debounce = false; }, 1000); } </script>

<button on:mousedown={validateBehaviourTree} class="p-2 rounded-lg shadow-lg relative inset-0" style="background-
color: {validate_behaviour_tree_color}; color: white; width: 100%; border: none;">{validate_behaviour_tree_↵
text}</button
```



## Chapter 14

# WebsocketServerConfig

```
<script> import { Label, Input, Button, Helper } from "flowbite-svelte";

import { websocket_server_connection_state, websocket_server_port, websocket_server_code, } from
"../store/websocket_store";

import { raspberry_pi_uuid_list, } from "../store/raspberry_pi_store";

/** @type {number} */ let websocketServerPort = $websocket_server_port;

/** @type {string} */ let toggleWebsocketServerMessage = "Start Websocket Server";

/** @type {string} */ let helperText = "";

/** @type {string} */ let websocketServerPortInputColor = "";

function displayWebsocketServerStatus() { if ($websocket_server_connection_state) { websocketServerPortInputColor = "green"; helperText = "Websocket Server is running"; toggleWebsocketServerMessage = "Stop Websocket Server"; } else { websocketServerPortInputColor = ""; helperText = ""; toggleWebsocketServerMessage = "Start Websocket Server"; } }

websocket_server_connection_state.subscribe(displayWebsocketServerStatus);

async function toggleWebsocketServer() { const websocketServerPortNumber = websocketServerPort * 1; if (typeof websocketServerPortNumber !== "number") { helperText = "Port must be a number"; websocketServerPortInputColor = "red"; return; } if (!$websocket_server_connection_state) { websocket_server_port.set(websocketServerPortNumber); const response = await api.startWebSocketServer({ port: $websocket_server_port, }); if (response.success) { websocket_server_connection_state.set(true); websocket_server_code.set(response.code); } else { // Error: Websocket Server is already running websocketServerPortInputColor = "red"; helperText = response.message; toggleWebsocketServerMessage = "Start Websocket Server"; } } else { await api.closeWebSocketServer(); websocket_server_connection_state.set(false); raspberry_pi_uuid_list.set([]); } } </script>

<Label for="websocket-server-port" class="block mb-2" >Input Port for Websocket Server</Label> <Input id="websocket-server-port" size="lg" placeholder="Websocket Server Port" bind:value={$websocketServerPort} disabled={$websocket_server_connection_state} color={$websocketServerPortInputColor} > <Button class="mt-4" on:click={toggleWebsocketServer} >{toggleWebsocketServerMessage}</Button> <Helper class="mt-2" color={$websocketServerPortInputColor} >{helperText}</Helper>
```



## Chapter 15

# Home

```
<script> import BehaviourTreeHandler from "../lib/BehaviourTreeHandler.svelte"; import Stream from "../lib/home/↵
Stream.svelte"; import CodeBox from "../lib/home/CodeBox.svelte"; import RaspberryPiConnectBanner from
"../lib/RaspberryPiConnectBanner.svelte";

import { websocket_server_connection_state } from "../store/websocket_store";

import { main_pane_size_0_store, main_pane_size_1_store, side_pane_size_0_store, side_pane_size_1_store, }
from "../store/home_store";

import { Pane, Splitpanes } from "svelte-splitpanes"; import SaveBehaviourTreeModal from "../lib/SaveBehaviour↵
TreeModal.svelte";

/** @type {number} */ let stream_split_height = 0;

/** @type {number} */ let stream_split_width = 0;

let main_pane_size_0 = $main_pane_size_0_store; let main_pane_size_1 = $main_pane_size_1_store; let side↵
_pane_size_0 = $side_pane_size_0_store; let side_pane_size_1 = $side_pane_size_1_store;

function onMainResize(event) { main_pane_size_0_store.set(event.detail[0].size); main_pane_size_1_store.↵
set(event.detail[1].size); }

function onSideResize(event) { side_pane_size_0_store.set(event.detail[0].size); side_pane_size_1_store.↵
set(event.detail[1].size); } </script>

<main style="height: 92vh"> {#if $websocket_server_connection_state === false} <RaspberryPiConnect↵
Banner > {/if} <SaveBehaviourTreeModal > <Splitpanes style="height:100%;padding:1em;background-color↵
:rgb(20,20,40)" on:resize={onSideResize} > <Pane size={side_pane_size_0}>> <Splitpanes horizontal={true}
on:resize={onMainResize}>> <Pane size={main_pane_size_0} minSize={5}>> <Stream > </Pane> <↵
Pane size={main_pane_size_1} minSize={7.5}>> <CodeBox > </Pane> </Splitpanes> </Pane> <Pane
size={side_pane_size_1} minSize={17} maxSize={80}>> <BehaviourTreeHandler > </Pane> </Splitpanes>
</main>
```



## Chapter 16

# NotFound

## Not Found

This route doesn't exist.

```
<style> h1 { color: #ff0040; text-transform: uppercase; font-size: 4em; font-weight: 100; } </style>
```





## Chapter 17

# Validate

```
<script> import BehaviourTreeHandler from "../lib/BehaviourTreeHandler.svelte";

import { side_pane_size_0_store, side_pane_size_1_store, } from "../store/home_store";

import { Pane, Splitpanes } from "svelte-splitpanes"; import SaveBehaviourTreeModal from "../lib/SaveBehaviourTreeModal.svelte"; import CodeBox from "../lib/validate/CodeBox.svelte";

let side_pane_size_0 = $side_pane_size_0_store; let side_pane_size_1 = $side_pane_size_1_store;

function onSideResize(event) { side_pane_size_0_store.set(event.detail[0].size); side_pane_size_1_store.set(event.detail[1].size); }

</script>

<main style="height: 92vh"> <SaveBehaviourTreeModal > <Splitpanes style="height:100%;padding:1em;background-color:rgb(20,20,40)" on:resize={onSideResize} > <Pane size={side_pane_size_0}>> <CodeBox > </Pane> <Pane size={side_pane_size_1} minSize={17} maxSize={80}>> <BehaviourTreeHandler > </Pane> </Splitpanes> </main>
```



## Chapter 18

# WebSocket

```
<script> import IPList from "../lib/IPList.svelte";
```

```
import WebSocketServerConfig from "../lib/WebSocketServerConfig.svelte"; import RaspberryPiSelector from  
"../lib/RaspberryPiSelector.svelte"; </script>
```

```
<main> <WebSocketServerConfig >
```

---

```
<RaspberryPiSelector > <IPList > </main>
```



## Chapter 19

# Class Index

### 19.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">Code</a> .....	41
<a href="#">WebSocketServer</a> .....	41



# Chapter 20

## File Index

### 20.1 File List

Here is a list of all files with brief descriptions:

<a href="#">rollup.config.js</a>	53
<a href="#">js/behaviour_tree.js</a>	45
<a href="#">js/device.js</a>	47
<a href="#">js/main.js</a>	48
<a href="#">js/preload.js</a>	50
<a href="#">js/raspberry_pi.js</a>	50
<a href="#">js/store.js</a>	51
<a href="#">js/websocket.js</a>	51
<a href="#">js/websocket_events.js</a>	52
<a href="#">src/main.js</a>	49
<a href="#">src/routes.js</a>	56
<a href="#">src/lib/CodeBox_Constants.js</a>	54
<a href="#">src/store/behaviour_tree_store.js</a>	57
<a href="#">src/store/home_store.js</a>	58
<a href="#">src/store/raspberry_pi_store.js</a>	58
<a href="#">src/store/websocket_store.js</a>	59





## Chapter 21

# Class Documentation

### 21.1 Code Class Reference

#### Public Member Functions

- [constructor](#) ()
- [get](#) ()
- [generate](#) ()

#### 21.1.1 Member Function Documentation

##### 21.1.1.1 constructor()

```
Code::constructor ( ) [inline]
```

##### 21.1.1.2 generate()

```
Code::generate ( ) [inline]
```

##### 21.1.1.3 get()

```
Code::get ( ) [inline]
```

The documentation for this class was generated from the following file:

- [js/websocket.js](#)

### 21.2 WebSocketServer Class Reference

#### Public Member Functions

- [constructor](#) ()
- [connect](#) (port)
- [isConnected](#) ()
- [close](#) ()
- [generateCode](#) ()
- [getCode](#) ()
- [getPort](#) ()
- [getRaspberryPiMap](#) ()
- [getSelectedRaspberryPi](#) ()
- [selectRaspberryPi](#) (uuid)

- [async startConnections](#) ()
- [sendBehaviourTree](#) (behaviour\_tree)
- [startBehaviourTree](#) ()
- [stopBehaviourTree](#) ()

## 21.2.1 Member Function Documentation

### 21.2.1.1 close()

```
WebSocketServer::close ( ) [inline]
```

Call this function to close the WebSocket server.

### 21.2.1.2 connect()

```
WebSocketServer::connect (
    port ) [inline]
```

Call this function to start the WebSocket server.

#### Parameters

<code>{number}</code>	port
-----------------------	------

### 21.2.1.3 constructor()

```
WebSocketServer::constructor ( ) [inline]
```

```
@type {WebSocket.Server | undefined}
```

```
@type {number | undefined}
```

```
@type {Code}
```

```
@type {Map<string, WebSocket>}
```

```
@type {RaspberryPi | undefined}
```

### 21.2.1.4 generateCode()

```
WebSocketServer::generateCode ( ) [inline]
```

#### Returns

```
{number}
```

### 21.2.1.5 getCode()

```
WebSocketServer::getCode ( ) [inline]
```

#### Returns

```
{number}
```

### 21.2.1.6 getPort()

```
WebSocketServer::getPort ( ) [inline]
```

#### Returns

```
{number}
```

### 21.2.1.7 getRaspberryPiMap()

```
WebSocketServer::getRaspberryPiMap ( ) [inline]
```

#### Returns

```
{Map<string, WebSocket>}
```

### 21.2.1.8 getSelectedRaspberryPi()

```
WebSocketServer::getSelectedRaspberryPi ( ) [inline]
```

#### Returns

```
{RaspberryPi | undefined}
```

### 21.2.1.9 isConnected()

```
WebSocketServer::isConnected ( ) [inline]
```

### 21.2.1.10 selectRaspberryPi()

```
WebSocketServer::selectRaspberryPi (
    uuid ) [inline]
```

Select a Raspberry Pi to send / receive messages from.

#### Parameters

{string}	uuid
----------	------

#### Returns

### 21.2.1.11 sendBehaviourTree()

```
WebSocketServer::sendBehaviourTree (
    behaviour_tree ) [inline]
```

#### Parameters

{string}	behaviour_tree
----------	----------------

### 21.2.1.12 startBehaviourTree()

```
WebSocketServer::startBehaviourTree ( ) [inline]
```

### 21.2.1.13 startConnections()

```
async WebSocketServer::startConnections ( ) [inline]
```

#### 21.2.1.14 stopBehaviourTree()

WebSocketServer::stopBehaviourTree ( ) [inline]

The documentation for this class was generated from the following file:

- [js/websocket.js](#)

# Chapter 22

## File Documentation

### 22.1 js/behaviour\_tree.js File Reference

#### Functions

- function `isDev` ()
- function `sendBehaviourTree` (\_event, args)
- function `startBehaviourTree` (\_event, \_args)
- function `stopBehaviourTree` (\_event, \_args)
- function `getBehaviourTreeList` ()
- function `updateBehaviourTreeList` (behaviour\_tree\_list)
- function async `saveBehaviourTree` (\_event, args)
- function async `removeBehaviourTree` (\_event, args)
- function async `validateBehaviourTree` (\_event, args)
- ipcMain `handle` ('sendBehaviourTree', sendBehaviourTree)

#### Variables

- `const` { `app`, `ipcMain` } = require('electron')
- `const` `exec` = util.promisify(require('child\_process').exec)
- `const` `Os` = require('os')
- `const` `path` = require("path")
- `const` `fs` = require("fs")
- `const` `base_path` = app.getAppPath()
- `const` `resources_dir` = `isDev`() ? path.join(`base_path`, 'resources') : path.join(`base_path`, '..', '..', 'resources')

#### 22.1.1 Function Documentation

##### 22.1.1.1 `getBehaviourTreeList()`

```
function getBehaviourTreeList ( )
```

##### 22.1.1.2 `handle()`

```
ipcMain handle (
    'sendBehaviourTree' ,
    sendBehaviourTree )
```

#### 22.1.1.3 isDev()

```
function isDev ( )
```

#### 22.1.1.4 removeBehaviourTree()

```
function async removeBehaviourTree (
    _event,
    args )
```

#### 22.1.1.5 saveBehaviourTree()

```
function async saveBehaviourTree (
    _event,
    args )
```

#### 22.1.1.6 sendBehaviourTree()

```
function sendBehaviourTree (
    _event,
    args )
```

#### 22.1.1.7 startBehaviourTree()

```
function startBehaviourTree (
    _event,
    _args )
```

#### 22.1.1.8 stopBehaviourTree()

```
function stopBehaviourTree (
    _event,
    _args )
```

#### 22.1.1.9 updateBehaviourTreeList()

```
function updateBehaviourTreeList (
    behaviour_tree_list )
```

##### Parameters

<code>{any[]}</code>	<code>behaviour_tree_list</code>
----------------------	----------------------------------

#### 22.1.1.10 validateBehaviourTree()

```
function async validateBehaviourTree (
    _event,
    args )
```

@type {string}

## 22.1.2 Variable Documentation

### 22.1.2.1 base\_path

```
const base_path = app.getAppPath()
```

### 22.1.2.2 const

```
const { app, ipcMain } = require('electron')
```

### 22.1.2.3 exec

```
const exec = util.promisify(require('child_process').exec)
```

### 22.1.2.4 fs

```
const fs = require("fs")
```

### 22.1.2.5 Os

```
const Os = require('os')
```

### 22.1.2.6 path

```
const path = require("path")
```

### 22.1.2.7 resources\_dir

```
const resources_dir = isDev() ? path.join(base_path, 'resources') : path.join(base_path, '..', '..', 'resources')
```

## 22.2 js/device.js File Reference

### Functions

- function [getLocalIPList](#) (\_event, \_args)
- ipcMain [handle](#) ('getLocalIPList', [getLocalIPList](#))

### Variables

- [const](#) [os](#)
- module [exports](#)

### 22.2.1 Function Documentation

### 22.2.1.1 getLocalIPList()

```
function getLocalIPList (
    _event,
    _args )
```

### 22.2.1.2 handle()

```
ipcMain handle (
    'getLocalIPList' ,
    getLocalIPList )
```

## 22.2.2 Variable Documentation

### 22.2.2.1 exports

```
module exports
Initial value:
= {
    getLocalIPList: getLocalIPList
}
```

### 22.2.2.2 os

```
const os
Initial value:
= require('os')
const { ipcMain } = require('electron')
```

## 22.3 js/main.js File Reference

### Functions

- function `isDev` ()
- function `createWindow` ()
- `app` on ('ready', `createWindow`)
- function `onClose` ()

### Variables

- `const` { `app`, `BrowserWindow` } = `require`('electron')
- `const` `path` = `require`('path')
- `const` `serve` = `require`('electron-serve')
- `const` `loadURL` = `serve`({ `directory`: 'public' })

### 22.3.1 Function Documentation

#### 22.3.1.1 createWindow()

```
function createWindow ( )
```



### 22.3.1.2 isDev()

```
function isDev ( )
```

### 22.3.1.3 on()

```
app on (
    'ready' ,
    createWindow )
```

### 22.3.1.4 onClose()

```
function onClose ( )
```

## 22.3.2 Variable Documentation

### 22.3.2.1 const

```
const { app, BrowserWindow } = require('electron')
```

### 22.3.2.2 loadURL

```
const loadURL = serve({ directory: 'public' })
```

### 22.3.2.3 path

```
const path = require('path')
```

### 22.3.2.4 serve

```
const serve = require('electron-serve')
```

## 22.4 src/main.js File Reference

### Variables

- `import app` `pcss`
- `import App` from `App` `svelte`
- `const app`

### 22.4.1 Variable Documentation

#### 22.4.1.1 app

```
export default app
```

##### Initial value:

```
= new App({
  target: document.body,
  props: {
    name: "world",
  },
})
```

#### 22.4.1.2 pcss

```
import app pcss
```

#### 22.4.1.3 svelte

```
import App from App svelte
```

## 22.5 js/preload.js File Reference

## 22.6 js/raspberry\_pi.js File Reference

### Functions

- function `unselectRaspberryPi` (`_event`, `_args`)
- function `selectRaspberryPi` (`_event`, `args`)
- function `getRaspberryPiList` (`_event`, `_args`)
- function `getSelectedRaspberryPi` (`_event`, `_args`)
- ipcMain `handle` (`'selectRaspberryPi'`, `selectRaspberryPi`)

### Variables

- `const` { `ipcMain` } = `require('electron')`

### 22.6.1 Function Documentation

#### 22.6.1.1 getRaspberryPiList()

```
function getRaspberryPiList (
    _event,
    _args )
```

#### 22.6.1.2 getSelectedRaspberryPi()

```
function getSelectedRaspberryPi (
    _event,
    _args )
```

#### 22.6.1.3 handle()

```
ipcMain handle (
    'selectRaspberryPi' ,
    selectRaspberryPi )
```

#### 22.6.1.4 selectRaspberryPi()

```
function selectRaspberryPi (
    _event,
    args )
```

### 22.6.1.5 unselectRaspberryPi()

```
function unselectRaspberryPi (
    _event,
    _args )
```

## 22.6.2 Variable Documentation

### 22.6.2.1 const

```
const { ipcMain } = require('electron')
```

## 22.7 js/store.js File Reference

## 22.8 js/websocket.js File Reference

### Classes

- class [Code](#)
- class [WebSocketServer](#)

### Functions

- function [getRandomInt](#) (min, max)

### Variables

- `const WebSocket = require('ws')`
- module `exports = { websocket_server: new WebSocketServer() }`

## 22.8.1 Function Documentation

### 22.8.1.1 getRandomInt()

```
function getRandomInt (
    min,
    max )
```

#### Parameters

<code>{number}</code>	min
<code>{number}</code>	max

#### Returns

```
{number}
```

## 22.8.2 Variable Documentation

### 22.8.2.1 exports

```
module exports = { websocket_server: new WebSocketServer() }
```

### 22.8.2.2 WebSocket

```
const WebSocket = require('ws')
```

## 22.9 js/websocket\_events.js File Reference

### Functions

- function async [startWebSocketServer](#) ([\\_event](#), [args](#))
- function [closeWebSocketServer](#) ([\\_event](#), [\\_args](#))
- function [getWebSocketServer](#) ([\\_event](#), [\\_args](#))
- ipcMain [handle](#) (['startWebSocketServer'](#), [startWebSocketServer](#))

### Variables

- [const](#) { [ipcMain](#) } = [require](#)('electron')
- [const](#) [tcpPortUsed](#) = [require](#)('tcp-port-used')

### 22.9.1 Function Documentation

#### 22.9.1.1 closeWebSocketServer()

```
function closeWebSocketServer (  
    _event,  
    _args )
```

#### 22.9.1.2 getWebSocketServer()

```
function getWebSocketServer (  
    _event,  
    _args )
```

#### 22.9.1.3 handle()

```
ipcMain handle (  
    'startWebSocketServer' ,  
    startWebSocketServer )
```

#### 22.9.1.4 startWebSocketServer()

```
function async startWebSocketServer (  
    _event,  
    args )
```

### 22.9.2 Variable Documentation

#### 22.9.2.1 const

```
const { ipcMain } = require('electron')
```

### 22.9.2.2 tcpPortUsed

```
const tcpPortUsed = require('tcp-port-used')
```

## 22.10 README.md File Reference

## 22.11 rollup.config.js File Reference

### Functions

- function `serve` ()

### Variables

- `import` `svelte` from rollup plugin `svelte`
- `import` `commonjs` from rollup plugin `commonjs`
- `import` `resolve` from rollup plugin node `resolve`
- `import` `livereload` from rollup plugin `livereload`
- `import` { `terser` } from 'rollup-plugin-terser'
- `import` `css` from rollup plugin `css` `only`
- `import` `postcss` from rollup plugin `postcss` `import` `tailwindcss` from `tailwindcss`
- `import` `nodePolyfills` from rollup plugin node `polyfills`
- `const` `tailwindConfig` = `require`('./tailwind.config.cjs')
- `const` `production` = `!process.env.ROLLUP_WATCH`
- `export` `default`

### 22.11.1 Function Documentation

#### 22.11.1.1 `serve()`

```
function serve ( )
```

### 22.11.2 Variable Documentation

#### 22.11.2.1 `commonjs`

```
import commonjs from rollup plugin commonjs
```

#### 22.11.2.2 `default`

```
export default
```

#### 22.11.2.3 `import`

```
import { terser } from 'rollup-plugin-terser'
```

#### 22.11.2.4 `livereload`

```
import livereload from rollup plugin livereload
```

#### 22.11.2.5 only

```
import css from rollup plugin css only
```

#### 22.11.2.6 polyfills

```
import nodePolyfills from rollup plugin node polyfills
```

#### 22.11.2.7 production

```
const production = !process.env.ROLLUP_WATCH
```

#### 22.11.2.8 resolve

```
import resolve from rollup plugin node resolve
```

#### 22.11.2.9 svelte

```
import NotFound from routes NotFound svelte
```

#### 22.11.2.10 tailwindConfig

```
const tailwindConfig = require('./tailwind.config.cjs')
```

#### 22.11.2.11 tailwindcss

```
import postcss from rollup plugin postcss import tailwindcss from tailwindcss
```

## 22.12 src/App.svelte File Reference

## 22.13 src/lib/BehaviourTreeHandler.svelte File Reference

## 22.14 src/lib/CodeBox\_Constants.js File Reference

### Variables

- `import { hoverTooltip } from "@codemirror/view"`
- `export const range_0_180`
- `export const range_0_360`
- `export const range_0_100`
- `export const condition_operator_types`
- `const node_information_list`
- `export const node_hover`

### 22.14.1 Variable Documentation

#### 22.14.1.1 condition\_operator\_types

```
export const condition_operator_types
```

### 22.14.1.2 import

```
import { hoverTooltip } from "@codemirror/view"
```

### 22.14.1.3 node\_hover

```
export const node_hover
```

#### Initial value:

```
= hoverTooltip((view, pos, side) => {
  const { from, to, text } = view.state.doc.lineAt(pos);
  let start = pos,
      end = pos;
  while (start > from && /\w:/ .test(text[start - from - 1])) start--;
  while (end < to && /\w:/ .test(text[end - from])) end++;
  if ((start == pos && side < 0) || (end == pos && side > 0)) return null;
  const highlighted_node = text.slice(start - from, end - from);
  const node_information = node_information_list.get(highlighted_node);
  if (!node_information) return null;
  return {
    pos: start,
    end,
    above: true,
    create(_) {
      const dom = document.createElement("div");
      dom.textContent = node_information;
      return { dom };
    },
  };
});
```

### 22.14.1.4 node\_information\_list

```
const node_information_list
```

#### Initial value:

```
= new Map([
  ["Root", "Start of every node tree"],
  ["Selector", "Runs children until one succeeds"],
  ["Sequence", "Runs children until one fails"],
  ["Random", "Runs a random child"],
  ["UseRoot", "Goes to the given root node id"],
  ["Invert", "Inverts the result of the child"],
  ["Repeat", "Repeats the child given the amount of times"],
  ["Fail", "Always fails"],
  ["Succeed", "Always succeeds"],
  ["Blackboard:ChangeInteger", "Changes the integer value of the blackboard"],
  ["Blackboard:IntegerCondition", "Checks the integer value of the blackboard"],
  ["Action:PauseExecution", "Pauses the execution for the given amount of time"],
  ["Action:Print", "Prints the given message"],
  ["Action:Drive", "Set the speed and direction of the car"],
  ["Action:Turn", "Turns the car by the given angle"],
  ["Action:SetAngle", "Turns the car by the given angle"],
  ["Action:SetSpeed", "Moves the car by the given speed"],
  ["Action:SetWheelDirection", "Sets the direction of the car"],
  ["Condition:SucceedOnAverageNearbyScan", "Checks the average distance between two angles and succeeds if the average is below the specified cm"],
  ["Condition:SucceedOnAnyNearbyScan", "Checks the distance of all nearby points and succeeds if any is below the specified cm"],
  ["Condition:SucceedOnAverageColour", "Checks the average colour of the camera and succeeds if the colour is within the tolerance"],
]);
@type {Map<string, string>}
```

### 22.14.1.5 range\_0\_100

```
export const range_0_100
```

### 22.14.1.6 range\_0\_180

```
export const range_0_180
```

### 22.14.1.7 range\_0\_360

```
export const range_0_360
```

## 22.15 src/lib/home/CodeBox.svelte File Reference

## 22.16 src/lib/validate/CodeBox.svelte File Reference

## 22.17 src/lib/home/CodeMessagingBar.svelte File Reference

## 22.18 src/lib/home/Stream.svelte File Reference

## 22.19 src/lib/IPList.svelte File Reference

## 22.20 src/lib/NavigationBar.svelte File Reference

## 22.21 src/lib/RaspberryPiConnectBanner.svelte File Reference

## 22.22 src/lib/RaspberryPiSelector.svelte File Reference

## 22.23 src/lib/SaveBehaviourTreeModal.svelte File Reference

## 22.24 src/lib/validate/CodeButton.svelte File Reference

## 22.25 src/lib/WebsocketServerConfig.svelte File Reference

## 22.26 src/routes.js File Reference

### Variables

- `import` Home from routes Home `svelte`
- `export default`

### 22.26.1 Variable Documentation

#### 22.26.1.1 default

```
export default
```

##### Initial value:

```
{  
  ' / ': Home,  
  '/websocket': Websocket,  
  '/validate': Validate,  
  
  ' * ': NotFound  
}
```

#### 22.26.1.2 svelte

```
import NotFound from routes NotFound svelte
```



## 22.27 src/routes/Home.svelte File Reference

## 22.28 src/routes/NotFound.svelte File Reference

## 22.29 src/routes/Validate.svelte File Reference

## 22.30 src/routes/Websocket.svelte File Reference

## 22.31 src/store/behaviour\_tree\_store.js File Reference

### Functions

- function async [main](#) ()

### Variables

- [import](#) { writable } from 'svelte/store'
- export [const](#) [behaviour\\_tree\\_xml\\_code](#) = writable("")
- export [const](#) [show\\_behaviour\\_tree\\_save\\_modal](#) = writable(false)
- export [const](#) [behaviour\\_tree\\_save\\_modal\\_code](#) = writable("")
- export let [behaviour\\_tree\\_list](#) = writable([])

### 22.31.1 Function Documentation

#### 22.31.1.1 main()

```
main ( )
```

### 22.31.2 Variable Documentation

#### 22.31.2.1 behaviour\_tree\_list

```
export let behaviour_tree_list = writable([])
```

#### 22.31.2.2 behaviour\_tree\_save\_modal\_code

```
export const behaviour_tree_save_modal_code = writable("")
```

#### 22.31.2.3 behaviour\_tree\_xml\_code

```
export const behaviour_tree_xml_code = writable("")
```

#### 22.31.2.4 import

```
import { writable } from 'svelte/store'
```

#### 22.31.2.5 show\_behaviour\_tree\_save\_modal

```
export const show_behaviour_tree_save_modal = writable(false)
```

## 22.32 src/store/home\_store.js File Reference

### Variables

- `import { writable } from 'svelte/store'`
- `export const main_pane_size_0_store = writable(80)`
- `export const main_pane_size_1_store = writable(20)`
- `export const side_pane_size_0_store = writable(83)`
- `export const side_pane_size_1_store = writable(17)`
- `export const lidar_toggle = writable(true)`

### 22.32.1 Variable Documentation

#### 22.32.1.1 import

```
import { writable } from 'svelte/store'
```

#### 22.32.1.2 lidar\_toggle

```
export const lidar_toggle = writable(true)
```

#### 22.32.1.3 main\_pane\_size\_0\_store

```
export const main_pane_size_0_store = writable(80)
```

#### 22.32.1.4 main\_pane\_size\_1\_store

```
export const main_pane_size_1_store = writable(20)
```

#### 22.32.1.5 side\_pane\_size\_0\_store

```
export const side_pane_size_0_store = writable(83)
```

#### 22.32.1.6 side\_pane\_size\_1\_store

```
export const side_pane_size_1_store = writable(17)
```

## 22.33 src/store/raspberry\_pi\_store.js File Reference

### Functions

- `function async main ()`
- `window api onConnection ((value)=> { value=JSON.parse(value);console.log(value);raspberrypi_uuid_↔list.update((list)=> { list.push(value.uuid);return list;});})`
- `window api onDisconnection ((value)=> { value=JSON.parse(value);console.log(value);raspberrypi_uuid_↔list.update((list)=> { list=list.filter((uuid)=> uuid !==value.uuid);return list;});})`

### Variables

- `import { writable } from 'svelte/store'`
- `export const raspberrypi_uuid_list = writable([])`
- `export const selected_raspberrypi_uuid = writable("")`

## 22.33.1 Function Documentation

### 22.33.1.1 main()

```
function async main ( )
```

### 22.33.1.2 onConnection()

```
window api onConnection (
    (value) ,
    { value=JSON.parse(value);console.log(value);raspberry_pi_uuid_list.update((list)=>
{ list.push(value.uuid);return list;});} )
```

### 22.33.1.3 onDisconnection()

```
window api onDisconnection (
    (value) ,
    { value=JSON.parse(value);console.log(value);raspberry_pi_uuid_list.update((list)=>
{ list=list.filter((uuid)=> uuid !==value.uuid);return list;});} )
```

## 22.33.2 Variable Documentation

### 22.33.2.1 import

```
import { writable } from 'svelte/store'
```

### 22.33.2.2 raspberry\_pi\_uuid\_list

```
export const raspberry_pi_uuid_list = writable([])
```

### 22.33.2.3 selected\_raspberry\_pi\_uuid

```
export const selected_raspberry_pi_uuid = writable("")
```

## 22.34 src/store/websocket\_store.js File Reference

### Functions

- function async [main](#) ()
- window api [onMessage](#) ((value)=> { value=JSON.parse(value);lidar\_stream.set(value.lidar);camera\_↔stream.set(value.frame\_buffer);})

### Variables

- [import](#) { writable } from 'svelte/store'
- export [const websocket\\_server\\_connection\\_state](#) = writable(false)
- export [const websocket\\_server\\_port](#) = writable(3000)
- export [const websocket\\_server\\_code](#) = writable(-1)
- export [const lidar\\_stream](#) = writable([ ])
- export [const camera\\_stream](#) = writable("")

## 22.34.1 Function Documentation

### 22.34.1.1 main()

```
function async main ( )
```

### 22.34.1.2 onMessage()

```
window api onMessage (
    (value) ,
    { value=JSON.parse(value);lidar_stream.set(value.lidar);camera_stream.set(value.↵
frame_buffer);} )
```

## 22.34.2 Variable Documentation

### 22.34.2.1 camera\_stream

```
export const camera_stream = writable("")
```

### 22.34.2.2 import

```
import { writable } from 'svelte/store'
```

### 22.34.2.3 lidar\_stream

```
export const lidar_stream = writable([])
```

### 22.34.2.4 websocket\_server\_code

```
export const websocket_server_code = writable(-1)
```

### 22.34.2.5 websocket\_server\_connection\_state

```
export const websocket_server_connection_state = writable(false)
```

### 22.34.2.6 websocket\_server\_port

```
export const websocket_server_port = writable(3000)
```

# Index

- app
  - main.js, [49](#)
- base\_path
  - behaviour\_tree.js, [47](#)
- behaviour\_tree.js
  - base\_path, [47](#)
  - const, [47](#)
  - exec, [47](#)
  - fs, [47](#)
  - getBehaviourTreeList, [45](#)
  - handle, [45](#)
  - isDev, [45](#)
  - Os, [47](#)
  - path, [47](#)
  - removeBehaviourTree, [46](#)
  - resources\_dir, [47](#)
  - saveBehaviourTree, [46](#)
  - sendBehaviourTree, [46](#)
  - startBehaviourTree, [46](#)
  - stopBehaviourTree, [46](#)
  - updateBehaviourTreeList, [46](#)
  - validateBehaviourTree, [46](#)
- behaviour\_tree\_list
  - behaviour\_tree\_store.js, [57](#)
- behaviour\_tree\_save\_modal\_code
  - behaviour\_tree\_store.js, [57](#)
- behaviour\_tree\_store.js
  - behaviour\_tree\_list, [57](#)
  - behaviour\_tree\_save\_modal\_code, [57](#)
  - behaviour\_tree\_xml\_code, [57](#)
  - import, [57](#)
  - main, [57](#)
  - show\_behaviour\_tree\_save\_modal, [57](#)
- behaviour\_tree\_xml\_code
  - behaviour\_tree\_store.js, [57](#)
- camera\_stream
  - websocket\_store.js, [60](#)
- close
  - WebSocketServer, [42](#)
- closeWebSocketServer
  - websocket\_events.js, [52](#)
- Code, [41](#)
  - constructor, [41](#)
  - generate, [41](#)
  - get, [41](#)
- CodeBox\_Constants.js
  - condition\_operator\_types, [54](#)
  - import, [54](#)
- node\_hover, [55](#)
- node\_information\_list, [55](#)
- range\_0\_100, [55](#)
- range\_0\_180, [55](#)
- range\_0\_360, [55](#)
- commonjs
  - rollup.config.js, [53](#)
- condition\_operator\_types
  - CodeBox\_Constants.js, [54](#)
- connect
  - WebSocketServer, [42](#)
- const
  - behaviour\_tree.js, [47](#)
  - main.js, [49](#)
  - raspberry\_pi.js, [51](#)
  - websocket\_events.js, [52](#)
- constructor
  - Code, [41](#)
  - WebSocketServer, [42](#)
- createWindow
  - main.js, [48](#)
- default
  - rollup.config.js, [53](#)
  - routes.js, [56](#)
- device.js
  - exports, [48](#)
  - getLocalIPList, [47](#)
  - handle, [48](#)
  - os, [48](#)
- exec
  - behaviour\_tree.js, [47](#)
- exports
  - device.js, [48](#)
  - websocket.js, [51](#)
- fs
  - behaviour\_tree.js, [47](#)
- generate
  - Code, [41](#)
- generateCode
  - WebSocketServer, [42](#)
- get
  - Code, [41](#)
- getBehaviourTreeList
  - behaviour\_tree.js, [45](#)
- getCode
  - WebSocketServer, [42](#)

- getLocalIPList
  - device.js, 47
- getPort
  - WebSocketServer, 42
- getRandomInt
  - websocket.js, 51
- getRaspberryPiList
  - raspberrypi.js, 50
- getRaspberryPiMap
  - WebSocketServer, 42
- getSelectedRaspberryPi
  - raspberrypi.js, 50
  - WebSocketServer, 43
- getWebSocketServer
  - websocket\_events.js, 52
- handle
  - behaviour\_tree.js, 45
  - device.js, 48
  - raspberrypi.js, 50
  - websocket\_events.js, 52
- home\_store.js
  - import, 58
  - lidar\_toggle, 58
  - main\_pane\_size\_0\_store, 58
  - main\_pane\_size\_1\_store, 58
  - side\_pane\_size\_0\_store, 58
  - side\_pane\_size\_1\_store, 58
- import
  - behaviour\_tree\_store.js, 57
  - CodeBox\_Constants.js, 54
  - home\_store.js, 58
  - raspberrypi\_store.js, 59
  - rollup.config.js, 53
  - websocket\_store.js, 60
- isConnected
  - WebSocketServer, 43
- isDev
  - behaviour\_tree.js, 45
  - main.js, 48
- js/behaviour\_tree.js, 45
- js/device.js, 47
- js/main.js, 48
- js/preload.js, 50
- js/raspberrypi.js, 50
- js/store.js, 51
- js/websocket.js, 51
- js/websocket\_events.js, 52
- lidar\_stream
  - websocket\_store.js, 60
- lidar\_toggle
  - home\_store.js, 58
- livereload
  - rollup.config.js, 53
- loadURL
  - main.js, 49
- main
  - behaviour\_tree\_store.js, 57
  - raspberrypi\_store.js, 59
  - websocket\_store.js, 60
- main.js
  - app, 49
  - const, 49
  - createWindow, 48
  - isDev, 48
  - loadURL, 49
  - on, 49
  - onClose, 49
  - path, 49
  - pcss, 49
  - serve, 49
  - svelte, 50
- main\_pane\_size\_0\_store
  - home\_store.js, 58
- main\_pane\_size\_1\_store
  - home\_store.js, 58
- node\_hover
  - CodeBox\_Constants.js, 55
- node\_information\_list
  - CodeBox\_Constants.js, 55
- on
  - main.js, 49
- onClose
  - main.js, 49
- onConnection
  - raspberrypi\_store.js, 59
- onDisconnection
  - raspberrypi\_store.js, 59
- only
  - rollup.config.js, 53
- onMessage
  - websocket\_store.js, 60
- Os
  - behaviour\_tree.js, 47
- os
  - device.js, 48
- path
  - behaviour\_tree.js, 47
  - main.js, 49
- pcss
  - main.js, 49
- polyfills
  - rollup.config.js, 54
- production
  - rollup.config.js, 54
- range\_0\_100
  - CodeBox\_Constants.js, 55
- range\_0\_180
  - CodeBox\_Constants.js, 55
- range\_0\_360
  - CodeBox\_Constants.js, 55

- raspberrypi.js
  - const, 51
  - getRaspberrypiList, 50
  - getSelectedRaspberrypi, 50
  - handle, 50
  - selectRaspberrypi, 50
  - unselectRaspberrypi, 50
- raspberrypi\_store.js
  - import, 59
  - main, 59
  - onConnection, 59
  - onDisconnection, 59
  - raspberrypi\_uuid\_list, 59
  - selected\_raspberrypi\_uuid, 59
- raspberrypi\_uuid\_list
  - raspberrypi\_store.js, 59
- README.md, 53
- removeBehaviourTree
  - behaviour\_tree.js, 46
- resolve
  - rollup.config.js, 54
- resources\_dir
  - behaviour\_tree.js, 47
- rollup.config.js, 53
  - commonjs, 53
  - default, 53
  - import, 53
  - livereload, 53
  - only, 53
  - polyfills, 54
  - production, 54
  - resolve, 54
  - serve, 53
  - svelte, 54
  - tailwindConfig, 54
  - tailwindcss, 54
- routes.js
  - default, 56
  - svelte, 56
- saveBehaviourTree
  - behaviour\_tree.js, 46
- selected\_raspberrypi\_uuid
  - raspberrypi\_store.js, 59
- selectRaspberrypi
  - raspberrypi.js, 50
  - WebSocketServer, 43
- sendBehaviourTree
  - behaviour\_tree.js, 46
  - WebSocketServer, 43
- serve
  - main.js, 49
  - rollup.config.js, 53
- show\_behaviour\_tree\_save\_modal
  - behaviour\_tree\_store.js, 57
- side\_pane\_size\_0\_store
  - home\_store.js, 58
- side\_pane\_size\_1\_store
  - home\_store.js, 58
- src/App.svelte, 54
- src/lib/BehaviourTreeHandler.svelte, 54
- src/lib/CodeBox\_Constants.js, 54
- src/lib/home/CodeBox.svelte, 56
- src/lib/home/CodeMessagingBar.svelte, 56
- src/lib/home/Stream.svelte, 56
- src/lib/IPList.svelte, 56
- src/lib/NavigationBar.svelte, 56
- src/lib/RaspberrypiConnectBanner.svelte, 56
- src/lib/RaspberrypiSelector.svelte, 56
- src/lib/SaveBehaviourTreeModal.svelte, 56
- src/lib/validate/CodeBox.svelte, 56
- src/lib/validate/CodeButton.svelte, 56
- src/lib/WebsocketServerConfig.svelte, 56
- src/main.js, 49
- src/routes.js, 56
- src/routes/Home.svelte, 57
- src/routes/NotFound.svelte, 57
- src/routes/Validate.svelte, 57
- src/routes/Websocket.svelte, 57
- src/store/behaviour\_tree\_store.js, 57
- src/store/home\_store.js, 58
- src/store/raspberrypi\_store.js, 58
- src/store/websocket\_store.js, 59
- startBehaviourTree
  - behaviour\_tree.js, 46
  - WebSocketServer, 43
- startConnections
  - WebSocketServer, 43
- startWebSocketServer
  - websocket\_events.js, 52
- stopBehaviourTree
  - behaviour\_tree.js, 46
  - WebSocketServer, 43
- svelte
  - main.js, 50
  - rollup.config.js, 54
  - routes.js, 56
- tailwindConfig
  - rollup.config.js, 54
- tailwindcss
  - rollup.config.js, 54
- tcpPortUsed
  - websocket\_events.js, 52
- unselectRaspberrypi
  - raspberrypi.js, 50
- updateBehaviourTreeList
  - behaviour\_tree.js, 46
- validateBehaviourTree
  - behaviour\_tree.js, 46
- WebSocket
  - websocket.js, 52
- websocket.js
  - exports, 51
  - getRandomInt, 51

- WebSocket, [52](#)
- websocket\_events.js
  - closeWebSocketServer, [52](#)
  - const, [52](#)
  - getWebSocketServer, [52](#)
  - handle, [52](#)
  - startWebSocketServer, [52](#)
  - tcpPortUsed, [52](#)
- websocket\_server\_code
  - websocket\_store.js, [60](#)
- websocket\_server\_connection\_state
  - websocket\_store.js, [60](#)
- websocket\_server\_port
  - websocket\_store.js, [60](#)
- websocket\_store.js
  - camera\_stream, [60](#)
  - import, [60](#)
  - lidar\_stream, [60](#)
  - main, [60](#)
  - onMessage, [60](#)
  - websocket\_server\_code, [60](#)
  - websocket\_server\_connection\_state, [60](#)
  - websocket\_server\_port, [60](#)
- WebSocketServer, [41](#)
  - close, [42](#)
  - connect, [42](#)
  - constructor, [42](#)
  - generateCode, [42](#)
  - getCode, [42](#)
  - getPort, [42](#)
  - getRaspberryPiMap, [42](#)
  - getSelectedRaspberryPi, [43](#)
  - isConnected, [43](#)
  - selectRaspberryPi, [43](#)
  - sendBehaviourTree, [43](#)
  - startBehaviourTree, [43](#)
  - startConnections, [43](#)
  - stopBehaviourTree, [43](#)