



METROPOLIA UNIVERSITY OF APPLIED SCIENCES

INTERNET OF THINGS

GROUP PROJECT

**Web Interface for ABB Ventilation Controller
Technical Documentation**

Authors:

Janine PASCHEK

Maya HORNSCHUH

Theresa BRANKL

Simon SCHÄDLER

Submitted on: October 21, 2021

Contents

| | | |
|----------|--|----------|
| 1 | Introduction | 3 |
| 1.1 | Installation | 3 |
| 2 | Back end implementation | 4 |
| 2.1 | Global database | 4 |
| 2.2 | Routing | 4 |
| 2.3 | Authentication and authorization | 4 |
| 2.4 | Settings functionality | 4 |
| 2.4.1 | Changing the password | 4 |
| 2.4.2 | Adding a new user | 4 |
| 2.4.3 | Login activity | 4 |
| 2.5 | Data transfer | 4 |
| 2.5.1 | MQTT | 4 |
| 2.5.2 | WebSocket | 4 |
| 3 | Front end implementation | 5 |
| 3.1 | Responsive header and navigation menu | 5 |
| 3.2 | Help page | 5 |
| 3.3 | Control panel | 5 |
| 3.3.1 | Activating and inactivating the panels | 5 |
| 3.3.2 | Switching the mode | 5 |
| 3.3.3 | Plotting the data | 5 |
| 3.3.4 | Setting the target pressure and target fan speed | 5 |
| 3.4 | Settings page | 5 |
| 3.4.1 | Changing the current users' password | 5 |
| 3.4.2 | Adding a new user to the system | 5 |
| 3.4.3 | Displaying the login activity | 5 |
| 3.5 | Displaying warnings | 5 |
| 3.6 | Logging a user out | 5 |

1 Introduction

VentPro is a web interface for controlling an ABB ventilation controller. The interface displays all available information about the connected IoT device and enables the user to control the ventilation system using a website.

This technical documentation provides specific information about the implementation of both the front end and back end of the system. It does not include any descriptions of how to use the web interface itself. This information can be found in the [user manual](#).

1.1 Installation

This project is based on Node which needs to be installed to run the server. Please visit nodejs.org and follow the instructions to install node (LTS or latest version). The following node packages are required to run the server. The packages can be installed by running `"npm install <package>"`.

- express
- ejs
- body-parser
- sqlite3
- ws

The server also requires a running MQTT broker. To connect the server to a running broker, please open the `/src/server.js` file and check the MQTT configuration section. Please enter the correct IP, port, and topics by adjusting the following parameters:

```
const mqtt_ip = "mqtt://localhost";  
const mqtt_port = "1883";  
const mqtt_topic_pub = "controller/settings";  
const mqtt_topic_sub = "controller/status";
```

After installing the required packages and setting up the MQTT parameters, the server can be started by running `"node src/server.js"`. The server display status information in the terminal if everything started up correctly.

2 Back end implementation

2.1 Global database

2.2 Routing

2.3 Authentication and authorization

2.4 Settings functionality

2.4.1 Changing the password

2.4.2 Adding a new user

2.4.3 Login activity

2.5 Data transfer

2.5.1 MQTT

2.5.2 WebSocket

Data minimization

Live data

Warnings

Sending commands

Storing identifiers

3 Front end implementation

3.1 Responsive header and navigation menu

3.2 Help page

3.3 Control panel

3.3.1 Activating and inactivating the panels

3.3.2 Switching the mode

3.3.3 Plotting the data

Fetching data of the selected interval

Receiving and displaying live data

Selecting different time intervals

3.3.4 Setting the target pressure and target fan speed

Fetching the current state of the system

Sending a new target pressure and fan speed

3.4 Settings page

3.4.1 Changing the current users' password

3.4.2 Adding a new user to the system

3.4.3 Displaying the login activity

3.5 Displaying warnings

3.6 Logging a user out