

|  |
| --- |
|  |
| System Design Document  **Indoor Climate Control System** |
| |  |  |  | | --- | --- | --- | | Group #3 | 14-April-2022 | T2-CB01 Project | |

Logo

Description automatically generated

* Authors :
  + Victor Covalciuc
  + Farros Ramzy
  + Andy Verkooijen
  + Sonam Lama

Table of Contents

[Document history 2](#_Toc83976011)

[Terms, Abbreviations 2](#_Toc83976012)

[1. Introduction 3](#_Toc83976013)

[1.1. Project description 3](#_Toc83976014)

[2. System description 3](#_Toc83976015)

[3. System Design 4](#_Toc83976016)

[3.1. Main features 4](#_Toc83976017)

[3.2. System Context 4](#_Toc83976018)

[3.3. System Hardware Modules 4](#_Toc83976019)

[3.4. Subsystems 5](#_Toc83976020)

[3.4.1. Controller 5](#_Toc83976021)

[3.4.2. Communication Protocol 5](#_Toc83976022)

[3.4.3. more … 5](#_Toc83976023)

[3.4.4. Ventilation 5](#_Toc83976024)

[3.5. State Machine 5](#_Toc83976025)

[References 5](#_Toc83976026)

[Appendix 5](#_Toc83976027)

List of Figures

[Figure 1: an example of a “system context” diagram, replace your diagram here 4](#_Toc83976028)

[Figure 1: an example of a “system hardware modules” diagram, replace your diagram here 4](#_Toc83976029)

[Figure 2: an example of a “state machine” diagram, replace your own diagram here 5](#_Toc83976030)

List of Tables

**No table of figures entries found.**

# Document history

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Version** | **Date** | **Status** | **Author** | **Description** | **Remarks** |
| 1.0.0 | 14-04-2022 | Scarp | All Auth. | * Creation on template * Cover Page * Insertion of Diagrams * Insertion of read-written paragraphs * Sketch writing of missing paragraphs | W.I.P. |
| 1.1.0 | 15-05-2022 | Draft | Victor Covalciuc |  | W.I.P. |
| 2.0.0 | 15-05-2022 | Final | Victor Covalciuc |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

* ­­ Highlighted in green is the current version on which the document is on.

# Terms, Abbreviations

|  |  |
| --- | --- |
| Term | Enlarged Version of said Term |
| SDD | System Design Document |
|  |  |

# Introduction

In this document of our team, the system in described from different point of view shown in various diagrams and written in text. The general view from which the system is presented is a technical one broken down in multiple section which together create an image of the product’s inner workings.

## Project description

As Group number 3 of semester two, for our project of the semester we are building an Indoor Climate Control System which will regulate the interior environment of a home according to the outside conditions, inside measurements and online information.

# System description

The system is comprised of multiple parts which work in tandem to generate and analyse data to regulate the ventilation throughout the entire house in which it is placed. By doing so it controls the condition of the home’s environment, which further on pleases the inhabitants of the house in accordance with their liking or the default settings of the system.

* **Primary Functionality**
  + Regulate indoor climate to the default settings
  + Regulate indoor climate to the custom settings
  + Variable speed fan
  + Variable temperature changeable settings
  + Variable humidity changeable settings
  + Weather display
  + Collection of statistics in cloud
* **Secondary Functionality**
  + Easy navigation throughout the rooms
  + Show energy usage of every room
  + Show small indicator on room when a hardware error occurs
  + Constant display of a shortcut to the support and troubleshooting page

# System Design

# Main features

In this section the system main features/functionalities are described.

Measure

# System Context

Diagram

Description automatically generated

Figure 1: “System Context” diagram

The system context diagram (Figure 1) highlights all communication through out the system between different components. It consists of a central system labelled as “Room Climate System” that displays information gathered from sensors that detect total volatile organic compound (TVOC), carbon dioxide (C02), temperature, humidity, and carbon-monoxide on a touch screen. Similarly, all sensor data is also sent to viewable in an app and subsequently stored in a database. The data shared here is bidirectional as it is stored and can be retrieved if needed. Additionally, the system is also connected to an alarm system and a fan. The system here messages the devices to turn on or off and increase or decrease the speed, respectively.

# System Hardware Modules

The system is composed of different types of modules connected to Arduino Uno R3. They are intuitive touchscreen panels to display information to the user, various sensors to measure corresponding values for temperature, carbon monoxide, carbon dioxide, total volatile organic compounds, and fans to ventilate rooms. *Figure 2* shows a diagram of all the hardware modules.

Diagram

Description automatically generated

Figure 2: Different hardware modules connected to the Arduino

# Subsystems

Main blocks (software) of the system. The subsystems exchange information with each other, which is indicated as a received message or sent message. Messages can contain parameters that carry the data that are exchanged between different subsystems.

# Controller

In this section, the algorithms will be described.

# Communication Protocol

# Symbols & Characters

|  |  |  |
| --- | --- | --- |
| Symbols & Characters | Name | Meaning |
| # | Number Sign | Start character, initiate the start of the message. |
| & | Ampersand | Split message character, split up the initiated words per message. |
| { | Open Curly Braces | Payload start character, initiate the start of the payload state/value. |
| } | Closed Curly Braces | Payload end character, initiate the end of a payload state/value. |
| | | Vertical Bar | Split payload character, split up the initiated state/values per payload. |
| ; | Semicolon | End character, initiate the end of a message line. |

# Type of Messages

## System ID

Since the system will use the Wi-fi as its communication medium, the system will have its own MAC address as its default device ID. However, this ID can be renamed, by the user later in the output implementation. The system ID can be the name of the room or something else in that implementation.

In example:

Default ID (MAC address): 00:00:2a:00:72:ff

Preferred ID (Name): Bedroom

## Device ID

Device ID is the identifier for each device that is implemented in this system. For example:

|  |  |
| --- | --- |
| Device Name / Device Type | Device ID |
| Ventilation Box | VB |
| User Interface | UI |
| Sensor | SENSOR |

## Payload Type

Payload type is a word that explains the kind of the payload inside of the message.

Also, payload means the part in the transmitted that is the actual message intended for use in the application.

For example:

1. TVOC sensor = VOC

2. CO2 sensor = CO2

3. Temperature sensor = TMP

4. Humidity sensor = HUM

5. Fan Speed sensor = FAN

## Payload State

Payload state is a status of the message type. For example, if the payload value came in from the sensor automatically, the state will be **READ**. If the payload value came from the UI, the state will be **WRITE**. If the payload is feedback of the previous message, the state will be **REPLY**, and if the payload is a feedback error from the previous message, the state will be **ERROR**.

## Payload Value

It is the value of each payload that is carried by each message. **The values here are basically written as a string in a message line**. However, it could be changed into a float or integer type to be implemented by the system after it is received. For a reply, the value can be written as **ACK** or **NACK**.

# Line of A Message

**#<System\_ID>&<Device\_ID>&<Payload\_Type>{<Payload\_state>|<Value>};**

Example:

#**BEDROOM**&**SENSOR**&**TEMP**{**READ**|**20**};

Meaning:

The temperature sensor in the bedroom is reading 20 degree Celsius.

# Error Types

|  |  |  |
| --- | --- | --- |
| No. | Error Type | Error Message |
| 1. | Unknown System ID/Device ID/Payload Type/Payload State | 404 |
| 4. | Payload Unaccepted | 406 |
| 5. | Multiple Message Lines | 429 |
| 6. | Messages Process Conflicted | 409 |

Example:

#**BEDROOM**&**VB**&**VOC**{**ERROR**|**406**};

Meaning:

The ventilation box detected an unaccepted payload of TVOC sensor in the bedroom.

# more …

# Ventilation

Simulation, how the ventilation box will be simulated?

# State Machine

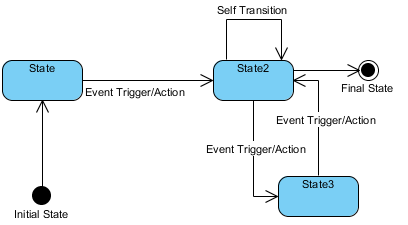


Figure 3: an example of a “state machine” diagram, replace your own diagram here

# References

# 5. Appendix