

Ventilation System Project Plan

Class: T-CB-CS2-CB01

Group: 6

Date: 15.09.20

Mentor: Gerald Hilderink

Team Members:

Fabian Ifionu - 3886913

Jorrit Triest - 4081722

Toma Krastanov – 3774678

Rami Alramadan - 3936767

Sjuul de Lang - 4023935

# Table of Contents

[1. Project Description 1](#_Toc57886017)

[1.1 Overview 2](#_Toc57886018)

[Constraints: 2](#_Toc57886019)

[1.2 Scope 2](#_Toc57886020)

[1.3 Actors 2](#_Toc57886021)

[1.4 Scenarios 2](#_Toc57886022)

[1.5 Functional Requirements 3](#_Toc57886023)

[1.6 Use Cases 3](#_Toc57886024)

[Actor: User 3](#_Toc57886025)

[Actor: Technician 3](#_Toc57886026)

[Actor: Cleaners 3](#_Toc57886027)

[Actor: CO2 sensor 3](#_Toc57886028)

[Actor: Temperature sensor 3](#_Toc57886029)

[Actor: Humidity sensor 3](#_Toc57886030)

[Actor: Particulate matter sensor 4](#_Toc57886031)

[Actor: ZigBee 4](#_Toc57886032)

[2. Project Organization 5](#_Toc57886033)

[2.1 Roles and Responsibilities 5](#_Toc57886034)

[3. Project way of working 5](#_Toc57886035)

[3.1 Standup Meeting setup 5](#_Toc57886036)

[3.2 Sprint Demos 5](#_Toc57886037)

[3.3 Communication plans 5](#_Toc57886039)

[3.4 Testing Strategies 5](#_Toc57886040)

[4. Timeline & Milestones 6](#_Toc57886041)

[5. Deliverables 6](#_Toc57886042)

[6. Risks analysis 8](#_Toc57886043)

[7. Configuration Management 9](#_Toc57886044)

[8. Project Plan Version 10](#_Toc57886045)

# Project Description

Via Fontys University we have received a big assignment from our client Airios.

This assignment will count towards our final for ICT&Technology and also will contribute to the personal portfolio of each active team member. The most important aspects are:

* **The final product**. Functionality and quality are important two highly important points. The acceptability of the final product is considered if at least these two criteria are met.
* **Team working**. The structure of the team, way of working and engaging are highly important points.

For the final product we are going to build a fully functional ventilation system for a room. This system works wirelessly and consists of diverse sensors that can measure CO2, humidity, temperature, and particulate matter. Furthermore, this sensor will communicate wirelessly with a ventilation box over ZigBee. Based on the measurements provided by the sensor and preferences of the user, the ventilation box regulates the ventilation to improve the indoor climate and air quality.

This project will have a rough duration of six months and will be presented to both our final client (Airios) and our Fontys tutor (Pu Xuemei).

**The stakeholders/actors in this project are:**

**Our team**. Whom interest in the matter is about designing, building, testing, and improving the prototype.

**Pu Xuemei**. Whom interest is to guide us in our journey to complete the assignment. Also, she is responsible to grade our work and progress.

**Airios**. Whom interest is to get the final product for further production, maintaining, innovation and commercialization.

**Final buyer**. Whose interest is to solve his problem regarding the temperature, humidity, CO2 and/or particulate matter in his room.

### Overview

The system will provide a solution to manage temperature, humidity, CO2 level, VOCs in the most energy efficient way. Using sensor means the system having a trouble-free communication is very critical and the use of wireless protocol is the solution, providing for 2-way communication between any sensor and the ventilation box. Measurement of these sensors provides a crucial information to control indoor heating and ventilation of the house.

### 

### Constraints:

* There is not enough hardware, this means we must concentrate on providing these for a limited number of rooms, to be precise 1 room.

### 1.2 Scope

* The system should be able to detect CO2 level, temperature, humidity and send information to the ventilation box in C# which will activate a fan to remove the CO2 or lower the temperature or the humidity and consequently allow fresh air in.
* Build a system that will control Air quality
* It will be possible to also control temperature and humidity in the house
* Provide a healthy environment for inhabitants
* Provide a cozy and pleasant feeling to the customer.
* Remove condensation which causes damage to the home.

### 1.3 Actors

* The User
* The technical maintenance team
* The cleaners
* Multiple sensors
* ZigBee

### Scenarios

* + - John lives in a house with a very small window, this makes it hard to get a enough fresh air into the house.
    - Sometime John may decide to open the windows to get fresh are but it’s very cold outside which means he cannot leave the windows open for a very long time and polluted air will still recirculate in the house.
    - John was introduced to Fontys students who build a ventilation system for him which continuously remove polluted air and allow fresh air into the house.

### 1.5 Functional Requirements

* If a **sensor** detects a value higher than that of the given threshold it will send a signal to the ventilation box which will activate the fan so the air can be filtered.
* The **user** finds the controlled temperature to be unsatisfactory. The user changes the temperature parameters to a different, more comfortable level. The system now maintains a more comfortable temperature.
* The **user** notices that the system is not working properly and reports it to the **tech maintenance** team. A technician can check the logs from the sensors and figure out which sensor is not working. Once fixed the system is restarted and continues to work accordingly.
* One of the **sensors** stops logging their data. The **technical maintenance** team notices that in the logs and sends a technician to fix the problem.
* If the fan is unable to filter the air the **cleaners** are to clean the ventilation box.

### 1.6 Use Cases

### Actor: User

**UC-01**: The user does not like the temperature in the room and changes it.

**UC-02**: The user notifies technicians for a bug in the ventilation box.

**UC-03**: User notifies cleaners that the ventilation box should be cleaned.

### Actor: Technician

**UC-04**: Technician fixes a technical problem in the system.

**UC-05**: Technicians make a routine maintenance check.

### Actor: Cleaners

**UC-06**: The cleaners clean the ventilation system

### Actor: CO2 sensor

**UC-07**: CO2 level is too high and triggers the fan

**UC-08**: CO2 sensor has a malfunction

### Actor: Temperature sensor

**UC-09**: Temperature is too high and triggers the fan

**UC-10**: Temperature sensor has a malfunction

### Actor: Humidity sensor

**UC-11**: Humidity levels are too high and trigger the fan

**UC-12**: Humidity sensor has a malfunction

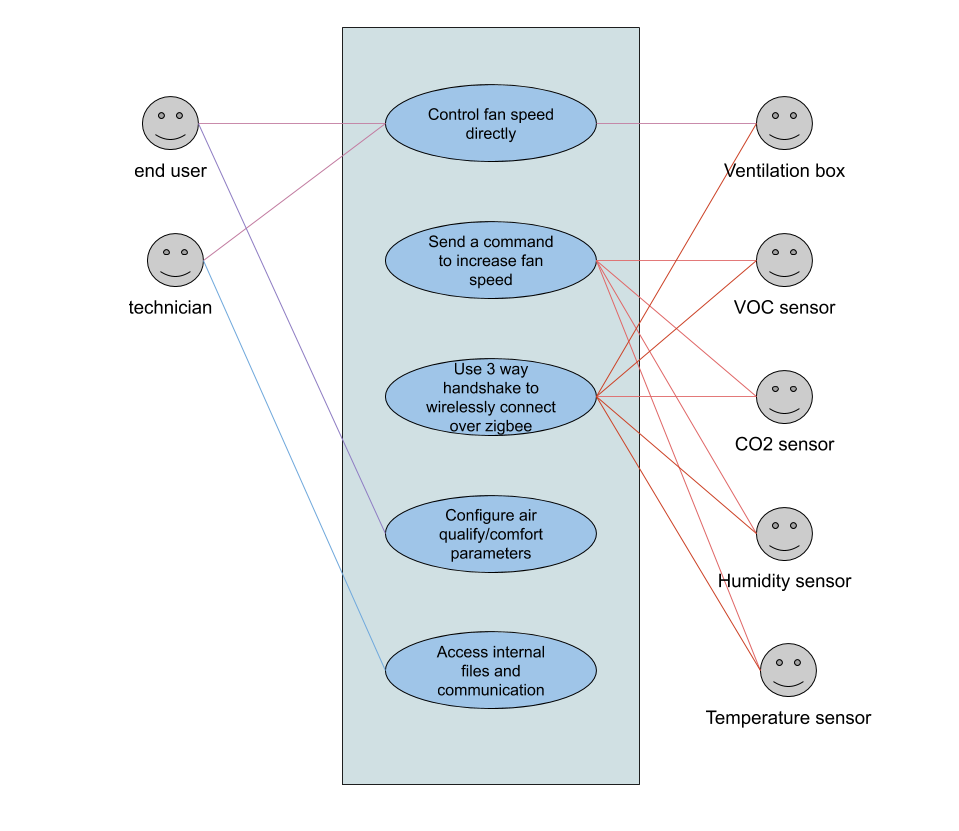
### Actor: Particulate matter sensor

**UC-13**: Particulate matter levels are too high and trigger the fan

**UC-14**: Particulate matter sensor has a malfunction

### Actor: ZigBee

**UC-15**: ZigBee communication fails



# Project Organization

### 2.1 Roles and Responsibilities

|  |  |  |
| --- | --- | --- |
| **Team Member** | **Role** | **Responsibilities** |
| Fabian Ifionu | Developer | The Project, |
| Jorrit Triest | Developer | The Project, Scrum Board Upkeep |
| Toma Krastanov | Scrum Master, Developer | The Project |
| Rami Alramadan | Developer | The Project |
| Sjuul de Lang | Developer | The Project |

# Project way of working

### 3.1 Standup Meeting setup

We plan to meet every Monday for a standup meeting to discuss updates concerning new things we added or removed or to talk about our findings regarding research we are doing for the project.

### 3.2 Sprint Demos

## Once every three weeks, we are going to do sprint demos in front of the product owner in which we are going to demonstrate our progress from the previous sprint demo.

### 3.3 Communication plans

There is a group WhatsApp where we communicate regularly to give update on our project or to notify each other on meetings. There is also a team meeting, this is where we meet every Monday to discuss the progress, update scrum board and assign new tasks to team members

### 3.4 Testing Strategies

We will have each member do some reviewing of code and testing how reliable the code is and if it can be broken. This will be done so that we ensure a good code design with no bugs or malfunctions.

# 4. Timeline & Milestones

|  |  |  |  |
| --- | --- | --- | --- |
| **Sprint** | **Task/Deliverable** | **Start Date** | **End Date** |
| 0 | Project plan | 30-08-2020 | 16-09-2020 |
| 1 | System Design document | 20-09-2020 | 08-10-2020 |
| 1 | Test code for temperature and humidity sensor that will display the values in the serial monitor | 20-09-2020 | 08-10-2020 |
| 2 | Test code for the CO2 and VOC sensors that will display the values in the serial monitor | 10-10-2020 | 29-10-2020 |
| 3 | Demo C# application | 30-10-2020 | 19-11-2020 |
| 3 | Combined code for all the sensors | 20-11-2020 | 10-12-2020 |
| 4 | ZigBee protocol | 11-12-2020 | 07-01-2021 |
| 5 | Final version of the project | 08-01-2021 | 28-01-2021 |
| 5 | Presentation | 08-01-2021 | 28-01-2021 |
| 5 | Project report | 08-01-2021 | 28-01-2021 |

# Deliverables

By the end of the project we will deliver a fully operational ventilation system consisting of:

* Diverse sensors capable to read data from the room environment. The data read by these sensors are CO2, humidity, temperature, and particulate matter.
* C# ventilation box which uses a ZigBee wireless communication to receive data from the sensor module.
* Project documentation (Project plan, system design, and use cases).
* Test report.
* Source code.

|  |  |  |
| --- | --- | --- |
| **Deliverable** | **Description** | **Person Responsible** |
| Project Plan | The project document containing the information on how our team is going to approach the project and what will be delivered, in addition to the use cases of the system. | All group members |
| Use Cases | A list of actions done by the actors that are going to interact with our system | -------- |
| C# Ventilation Box | A ventilation box simulated on C# that takes in the measurements from the sensors and ventilates the air in the room | Jorrit Triest  Rami Alramadan  Sjuul De Lang |
| Arduino Program | An Arduino application which takes all the values from the sensors on the shield and transmits them to the C# ventilation box via ZigBee | Toma Krastanov |
| Test Codes for All the Sensors on The Shield | The test codes will be presented on a sprint demo to demonstrate the team's ability to work with the sensors | Fabian Ifionu |
| System Design Document | A document showing the system design and how the team is going to approach the project | All group members |

# 6. Risks analysis

|  |  |  |  |
| --- | --- | --- | --- |
| Risk | Probability | Impact | Countermeasures |
| Hardware only being available to one group member | High | It is hard to split coding work as only one person can test it | We will have to send code to the member that has the hardware for testing. Code will have to have significant changes before testing to not waist the testers time |
| Overwriting on shared files | Low | Not knowing which file has the correct code for example | Updating everyone when there are changes made in the project. |
| The tasks are not divided well enough | Medium | One person gets stressed out and the other person is relaxed | Helping each other through the tasks so no one gets stressed out |
| Damaged hardware parts. | Medium | Incorrect results or unexpected circuit behavior. | Immediately replace the damaged part. |
| Team member cancels at last moment | Medium | Depending on the task of this member the project could be compromised. | Engage all members in everyone’s part, in this way if one member fails the rest will know how to proceed. |
| Using unfamiliar technologies | Low | The additional time needed for researching and understanding new concepts can be longer than expected. | Split the research between more people. |

# 7. Configuration Management

# 

All the documentation as well as the test and source codes will be uploaded to a Git repository which all the teachers have access to. Everyone will be pushing their work to the repository and when a change is made in either the documentation or the code the newest version will be uploaded to Git so that the teachers can see the changes immediately. By working with Git, we also can keep track of every modification to our code. If a mistake is made, we can always turn back the clock and compare earlier versions of the code to help fix the mistake while minimizing the disruption to all the team members.

# 8. Project Plan Version

|  |  |  |
| --- | --- | --- |
| Version | Changes | Date |
| V 0.1 | Initial project plan | 15.09.2020 |
| V 0.2 | Changed scenarios and actors | 16.09.2020 |
| V 0.3 | Updated project plan based on our mentor’s criticism | 03.12.2020 |