**Comfort Home**

**Project Plan**

**Victor Florea**

**Ivo Belitts**

**Valentin Stoyanov**

**Yvan Koopmans**

# Project description

The goal of the project is to create a system that can ventilate a room based on measurements done by wireless sensors that can measure CO2, humidity, temperature or particulate matter. The way it works is using sensor readings sent to a ventilation box over ZigBee we determine if and how much the room should be ventilated.

## Use cases:

### Users:

* **The client;**
* **The maintenance (us);**

**The client:**

Depending on where the system is mounted the client could vary from a family to an entire department. Therefore the system should have a user friendly device in order to be easy to use with no importance to the age of the user.

**The maintenance:**

The team (us) should be prepared for any kind of request from the user or problem. Either one of the sensors failed or the user wants to add a new sensor to the system, we need to already have a solution prepared for the possible problems that can appear when using this system. Keeping a clean and organized system, starting from the hardware and going to the actual code, is a good solution to avoid unforeseen problems.

Scenarios**:**

**SYS-1: Mounting the system**

Pre-conditions: The client purchases the system

Actors: The maintainers, the client

**Step 1:** The maintainers set up the system wiring at the desired location.

**Step 2:** The maintainers calibrate the sensors and set the client’s preferences.

**SYS-2: System is operating normally**

Pre-condition: The system is installed properly

Actors: The client

**Step-1:** The system is ventilating the room as expected.

**Step-2:** The client checks from time to time the system to make sure it is working properly.

**SYS-3: System is malfunctioning**

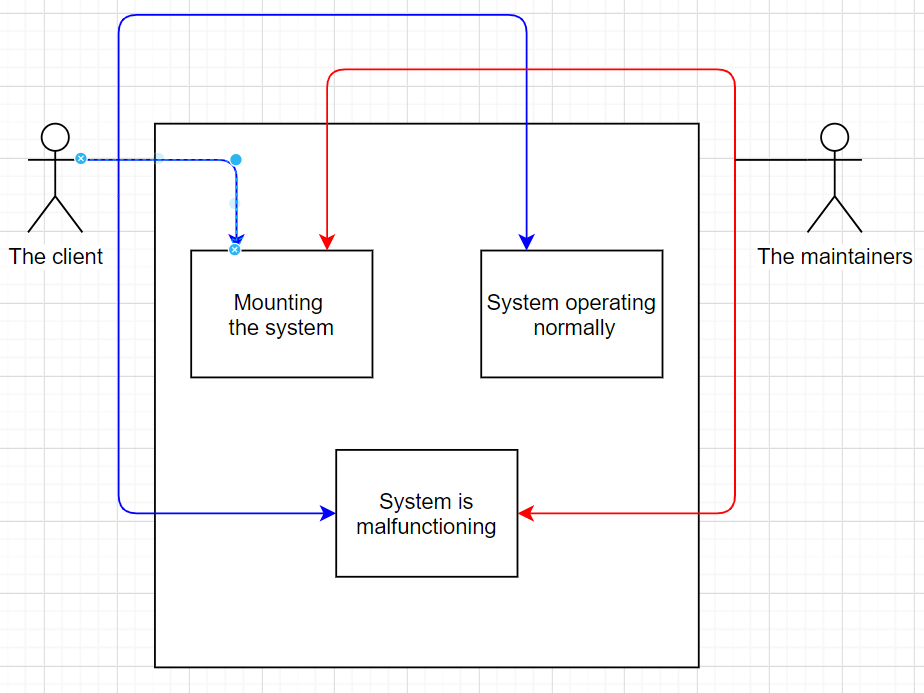
Pre-conditions: System is functioning as expected

Trigger: One or more sensors are faulty

Actors: The client, the maintenance

**Step 1:** The client perceives a visual response from the system which indicates that one or more sensors gone defective or the data stream is not sent as expected (red LED blinking). The client contacts the maintenance to come and repair or replace the damaged sensor/ overhaul the problem.

**Step 2:** One of the team members will connect to the system and analyze the systems feedback to identify the problem. The custom error messages designed will help us identify if the problem is related to a malfunction of the sensors, Nucleo board, data reading/receiving or a connecting one. That will enable us to fix the problem quickly without affecting the client too much.



## Functional requirements:

* COM-1: Bi-directional communication
* COM-2: Signals from sensors every 15 minutes or in case of a sharp increase
* COM-3: LED signals the state of the communication between the sensors and the ventilation box(green – communicating, red – not communicating)
* VB-1: The ventilation box stores logs all measurements
* VB-2: Ventilation control based on a configurable algorithm

## Non-functional requirements:

* Performance
* Maintainability
* Scalability

# Project organization

## Roles and responsibilities :

We decided that everyone will take part in coding, reviewing as well as testing code. That way we all get to experience all parts of the project and develop the most as programmers. We determined that Victor is going to be the scrummaster of our group. He is also going to take part in coding, reviewing and testing, but he will also have the responsibility of arranging meetings, taking notes of how meetings went and checking documentation.

# Project way of working

## Standup meeting setups

We plan on meeting every Wednesday and Thursday because we don’t have any lectures on these days, but we also agreed on meeting on other dates in case something urgent is to be done or we have to discuss something that won’t take more than 20 minutes.

## Sprint demos

We will have a sprint demo every 3 weeks. We also plan on documenting goals for our next sprint demo on what needs improving and what needs implementing.

## Communication plan

We plan on meeting at least 2x per week. We also set up a WhatsApp group, a GitLab repository and a Discord chat. We plan on using WhatsApp for setting up meetings and notifying others when somebody won’t be able to attend. GitLab will be used as our version control software and we will use Discord when we want to discuss something on weekends when we are all home.

## Testing strategies

We will try to develop similar systems simultaneously so that once we reach a milestone, we can try to combine them and test them together. That way we can do more tests and react quicker to potential problems.

# Deliverables

1. Working sensors used to determine state of air in rooms
2. Working way of communication between sensors and ventilation box
3. Working ventilation box used to determine if and how much rooms should be ventilated
4. Frequent communication with stakeholders

# Risks analysis

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk** | **Probability** | **Impact** | **Countermeasures** |
| Members not attending meeting | Medium | Member misses out on information and feedback | Maintain communication and arrange a second meeting |
| Problems understanding and implementing requirements | Medium | Delivering a final product that does not fit the requirements | Frequent meetings with stakeholders discussing progress |
| Lack of communication | High | Misunderstandings that lead to incompatible systems | Maintain a good tone and try to be friendly and understanding |
| Trying to overachieve and running out of time# | Low | Delivering a product that Is not yet finished | Try to manage our time and plan ahead |
| Problems with equipment | Medium | The system will malfunction without us knowing the problem | Frequent testing of the hardware |
| Conflicts between team members code | High | System not working as expected , merging problems | Maintain communication and testing |

# Configuration management

The master branch will be the release branch which will keep all the final versions. From the master branch we will have a “Development” branch which will be split in separate branches containing each feature of our system.

Once a feature is done we will push it to the “Development” branch and start testing it along with other features.

After testing two or more features together we will push the files to the master branch and release it as a version.