So nutzen wir das Internet of Things –

Eine Analyse der vielfältigen Anwendungsgebiete des IoT

## 1 Introduction

#### 1.1 Purpose of the document

This document can be seen as the basis for evaluation and feedback of this project. It further summarizes the most important technical and non-technical aspects of this project.

#### 1.2 Validity of the document

This document is valid for all parts of the project and is not draw up on an already existing specification. Changes to this specification can be made by project team members, as well as the supervisor of this project.

#### 1.3 Definitions and abbreviations

* HTTP – Hypertext Transfer Protocol
* HTTPS – Hypertext Transfer Protocol Secure
* SSL – Secure Socket Layer
* JSON – JavaScript Object Notation
* XML – Extensible Mark-up Language
* API – Application Programming Interface

#### 1.4 Overview over the document

The general structure of this document can be seen in the table of content. In the first chapter, the general validity and purpose of the document are laid out. In chapter two the focus lies on the product, in this case the Hardware(Sensors), the Backend, and the Application Prototype. In the third chapter the features of the applications prototype are defined, specified and described. The fourth chapters’ main constituent are the standards set to the application by the project team and client. The fifth and last chapter depicts the obligations the client set within the scope of the project.

## 2 General description over the document

#### 2.1 Aim of the product

The main purpose of the developed product is to be used as showcase application. It should show its user, what’s possible, using various Sensors and Actors and build up an Internet of Things System on their own. For this purpose, the application features several distinct features, which try to cover a broad array of available Sensors.

#### 2.2 General restrictions

The purpose of the system is to create an Internet of Things System, which should possibly easily be rebuilt and used by anyone, that owns the Hardware, or at least parts of it. The application will not be released, the only group of people, which will have access to the application are the product clients, the team from Butleroy. The application needs to run on a mobile device or Web Browser, the mobile device can either run Android, iOS or Windows Phone or any Web Browser. The application needs an active internet connection when used, to be able to communicate with the Integrator(Node-Red) and the Database.

#### 2.3 Pre-settings for hardware and software

The application can be executed on every mobile and Web Browser. The device needs a connection to the internet for the application to work.

#### 2.4 User of the product

The user of the product is the client. The team of Butleroy. The application will not be released to the public.

#### 2.5 Overview over the demanded functionality

The client wanted the following features and functionality:

* Should contain several Hardware-Sensors
* An Application that represents the Data of the Sensors
* A database, where the collected data should be saved
* An Integrator which works as a wiring tool between sensors and the database

#### 3.1 Delivery contents

The delivery content consists of the applications source code, the SQL database description, the php files and the source code of the various sensors and actors used. A Raspberry Pi, Temperature Sensor DS18b20Six Wemos D1 Mini, Joystick PS2, Passive Buzzer, l2C LCD 1602 Module, WS2812 LED-Strip(NeoPixel), Reed-Switch, Raindrop Sensor, Humiture Sensor, Ks0087 Keystudio Electronic including i2c and l2C LCD 1602 Module, Power, Lan, Anti-reverse and jumper wire cables to connect the Sensor with the Wemos and the Wemos to the power grid are also needed for the project to work fully.

#### 3.3 Aims of the user

The main goal, when using the application is, to see how an Internet of Things System can be implemented and easily be represented in an Application. The system can be used as a point of reference for further implementations and should be easily adaptable if anyone wants to add further sensors or actors.

#### 3.4 Required functions of the product

##### 3.4.1 Function „ Show Humidity “

|  |  |
| --- | --- |
| Name of the use case | Humiture Sensor |
| Abstract | The user of the system has the possibility to see the current Humidity in the room and can decide if he should open a window to let in fresh air. |
| Normal sequence of events | The user starts the application and goes to the Environment tab, where he sees the current Humidity of the last few minutes, to decide whether the window should be opened to let in fresh air. |
| Exceptions | No exceptions are handled. |
| Trigger | There are no triggers |
| Assumption | The user has an up and running internet connection. The Raspberry Pi and Node-Red Integrator are up and running. The Database is functioning. The Humiture Sensor is running and is connected to the Raspberry Pi’s Network |
| Pre-conditions | The users phone must have a network connection. |
| Post-conditions | The use case does not fulfil any post-conditions. |
| Author | Patrick Felbauer |
| Date | 10.06.2018 |

##### 3.4.2 Function „ Show Temperature “

|  |  |
| --- | --- |
| Name of the use case | Temperature Sensor |
| Abstract | The user of the system has the possibility to see the current Temperature in the room and can decide if he should open a window to let in fresh air. |
| Normal sequence of events | The user starts the application and goes to the Environment tab, where he sees the current Temperature of the last few minutes, to decide where the window should be opened to let in fresh air. |
| Exceptions | No exceptions are handled. |
| Trigger | There are no triggers |
| Assumption | The user has an up and running internet connection. The Raspberry Pi and Node-Red Integrator are up and running. The Database is functioning. The Temperature Sensor is running and is connected to the Raspberry Pi’s Network |
| Pre-conditions | The users phone must have a network connection. |
| Post-conditions | The use case does not fulfil any post-conditions. |
| Author | Patrick Felbauer |
| Date | 10.06.2018 |

##### 3.4.3 Function „Show amount of Rain”

|  |  |
| --- | --- |
| Name of the use case | Rain Sensor |
| Abstract | The user of the system has the possibility to see if it is currently raining. |
| Normal sequence of events | The user starts the application and goes to the Environment tab, where he sees the current Rain status of the last few minutes and also the amount of rain in the last few minutes. |
| Exceptions | No exceptions are handled. |
| Trigger | There are no triggers |
| Assumption | The user has an up and running internet connection. The Raspberry Pi and Node-Red Integrator are up and running. The Database is functioning. The Rain Sensor is running and is connected to the Raspberry Pi’s Network |
| Pre-conditions | The users phone must have a network connection. |
| Post-conditions | The use case does not fulfil any post-conditions. |
| Author | Patrick Felbauer |
| Date | 10.06.2018 |

##### 3.4.4 Function „Show Shower Time”

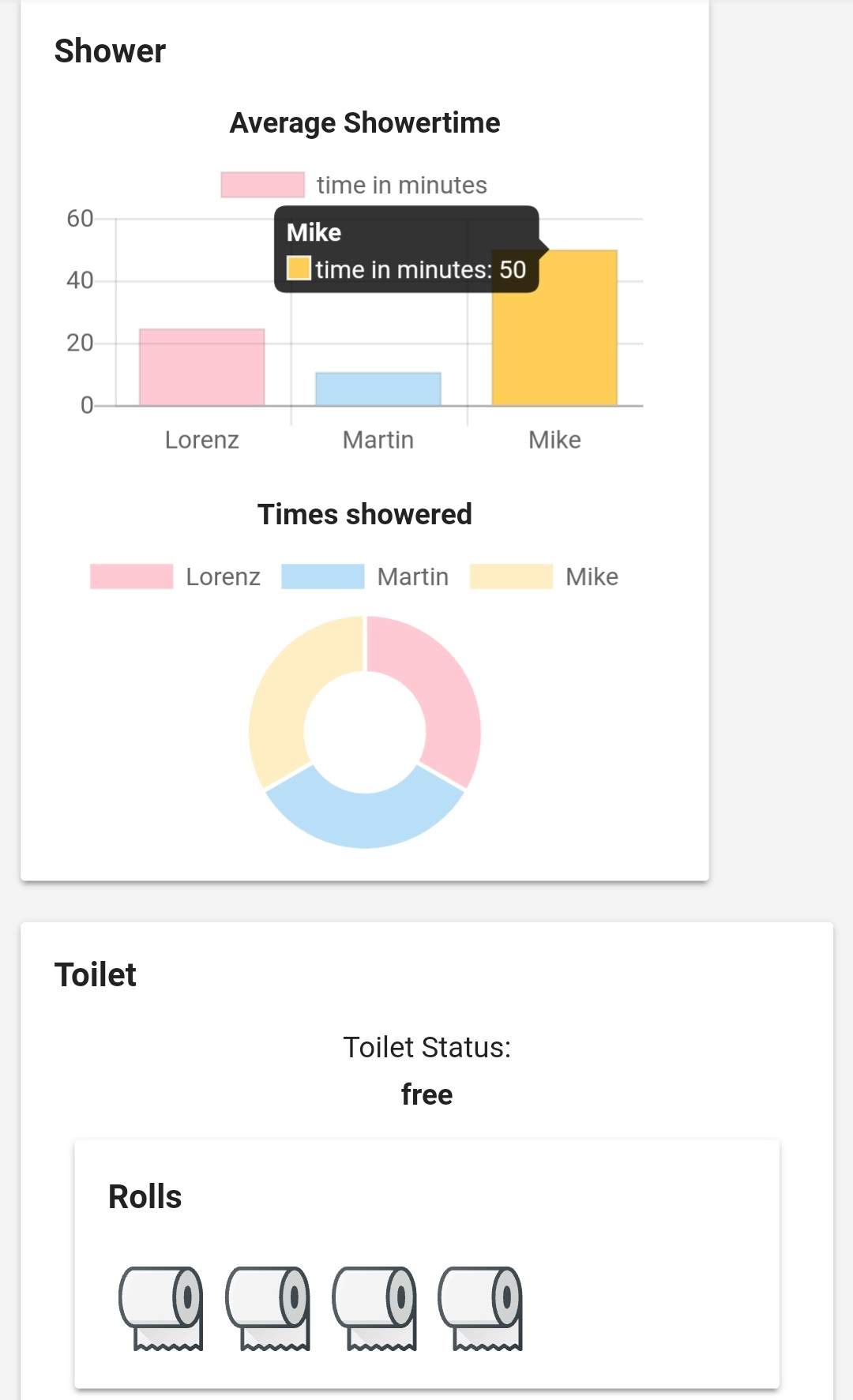
|  |  |
| --- | --- |
| Name of the use case | Shower Time stats |
| Abstract | The user of the system has the possibility to see the average shower time and amount of the users of the Shower Control Unit in the last 30 days. |
| Normal sequence of events | The user starts the application and goes to the Statistics tab, where he sees the average shower time of the users of the Shower Control Unit in the last 30 days. |
| Exceptions | No exceptions are handled. |
| Trigger | There are no triggers |
| Assumption | The user has an up and running internet connection. The Raspberry Pi and Node-Red Integrator are up and running. The Database is functioning. The Shower Control Unit, including the Joystick, LCD-Screen, LED-Stripe and Buzzer are running and the Wemos is connected to the Raspberry Pi’s Network |
| Pre-conditions | The users phone must have a network connection. |
| Post-conditions | The use case does not fulfil any post-conditions. |
| Author | Patrick Felbauer |
| Date | 10.06.2018 |

##### 3.4.4 Function „Show Toilet Stats”

|  |  |
| --- | --- |
| Name of the use case | Shower Toilet stats |
| Abstract | The user of the system has the possibility to see the current occupation of the Toilet and the amount of Toilet Paper stored in there. |
| Normal sequence of events | The user starts the application and goes to the Statistics tab, where he the current occupation of the Toilet and the amount of Toilet Paper stored in there.. |
| Exceptions | No exceptions are handled. |
| Trigger | There are no triggers |
| Assumption | The user has an up and running internet connection. The Raspberry Pi and Node-Red Integrator are up and running. The Database is functioning. The Scale including the LCD-Screen and the I2C are running, the Reed Switch is placed next on the door and the door is magnetic, or there is a magnetic plate for detection on it and the Wemos is connected to the Raspberry Pi’s Network |
| Pre-conditions | The users phone must have a network connection. |
| Post-conditions | The use case does not fulfil any post-conditions. |
| Author | Patrick Felbauer |
| Date | 10.06.2018 |

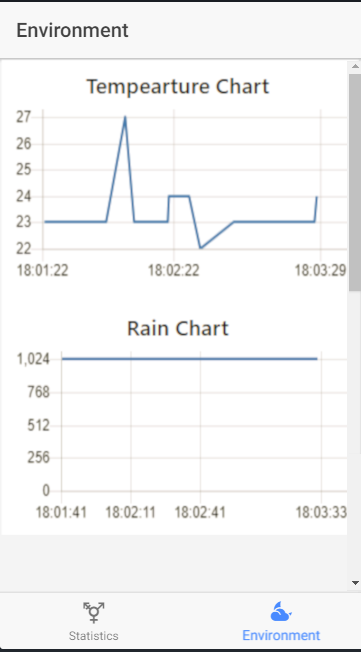
#### 3.5 External interfaces of the product

##### 3.5.1 User Interfaces



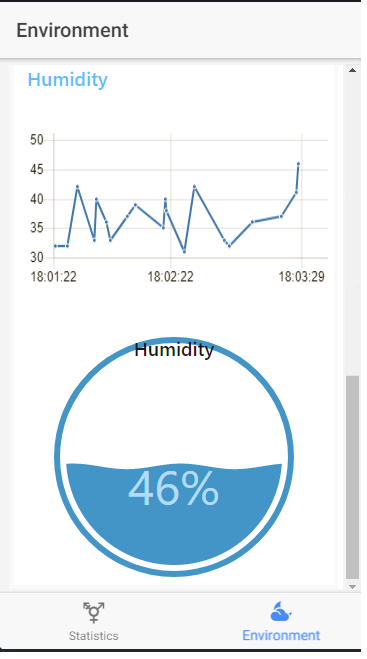
Displays the current occupation of the Toilet and the amount of toilet paper rolls stored on the scale in the toilet

Displays the average shower time and the times showered in the last 30 days of the users and you can press on them to get further details



Displays the current amount of rain for the last few minutes

Displays the current Temperature for the last few minutes



Displays the current Humidity

Displays the current Humidity for the last few minutes

## 4 Standards to the project execution

#### 4.1 Requirements to the realization

For the execution of the project in a controlled environment the following components are necessary:

Hardware:

* Computer running Windows, MacOS or Linux
* A Raspberry Pi (minimum Version 3)
* Temperature Sensor DS18b20
* Six Wemos D1 Mini
* Joystick PS2
* Passive Buzzer
* l2C LCD 1602 Module
* WS2812 LED-Strip(NeoPixel)
* Reed-Switch
* Raindrop Sensor
* Humiture Sensor
* Ks0087 Keystudio Electronic including i2c and l2C LCD 1602 Module
* Power, Lan, Anti-reverse and jumper wire cables to connect the Sensor with the Wemos and the Wemos to the power grid.

Software:

* UlnoIoT
* Ionic including Cordova
* Atom
* Node-Red
* MySQL Database

#### 4.4 Acceptance requirements

* Circumstances
  + The project gets approved against the specification, which was made in this file.  
    The functions, which are specified for the implementation of the project, must be fulfilled.
  + The project is accepted by evaluating the presentation of the project and the project itself executed by the supervisor of this project and its team.

## 5 Client obligations

The client is obligated in the project scope to test and evaluate the project and the presentation of it. Furthermore, continually support of the project during the duration of it is to be provided (like regular meetings with the supervisor of the project).