

**PUSS214201**

v. 0.2

---

SRS - Software Requirements Specification

---

Group 2

Responsible: System Group

Authors: System Group, UG

2021-02-02

## Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Reference documents</b>	<b>3</b>
<b>3</b>	<b>Background and goals</b>	<b>3</b>
3.1	Main goals . . . . .	3
3.2	Actors and their objectives . . . . .	3
<b>4</b>	<b>Terminology</b>	<b>3</b>
<b>5</b>	<b>Context diagram</b>	<b>4</b>
<b>6</b>	<b>Functional requirements</b>	<b>4</b>
6.1	Login and logout . . . . .	4
6.2	Data . . . . .	4
6.3	User . . . . .	5
6.4	Project Leader . . . . .	8
6.5	Administrator . . . . .	9
<b>7</b>	<b>Design requirements</b>	<b>11</b>
<b>8</b>	<b>Project requirements</b>	<b>12</b>
8.1	Development environment . . . . .	12

## 1 Introduction

This document presents requirements for the *TimeMate*. *TimeMate* is a system which purpose and main functionality is to administer time reporting with web-usage capabilities.

## 2 Reference documents

1. Software Requirements Specification: BaseBlockSystem, v. 1.0, Doc. number: PUSS12002
2. The requirements on the usability of the system applies to this document and are therefore implemented in *TimeMate*, following requirements are *not* implemented; 6.2.3, 6.3.4

## 3 Background and goals

### 3.1 Main goals

The goal is to develop and distribute a web-based system where the user can report time and administrate the system according to their roles.

### 3.2 Actors and their objectives

The system can be used and administrated by following actors;

- **User:** The user has the authority to log in to the system to report and change past reported time as well as review their reported times. The user also inherits roles as either "PG", "SG", "UG", and "TG".
- **Project Leader:** The project leaders main objective is to administer groups within the project, this implies that the project leader has the authority to add and remove as well as assign users from/to designated roles.
- **Administrator:** The administrator has the authority to add and remove users from the system along with creating and removing project groups. The admin is the only role that can assign the role "Project leader" to a user. The main goal of this role is to be able to administrate creation and removal of users.

## 4 Terminology

**UG, SG TG** These roles are only name tags for users and grants no special permissions or rights.

**Administrator:** Has access to all functions of the system.

**Time report:** A document containing fields detailing time spent on various parts of development.

**SHA-2:** A set of cryptographic hash functions used to protect users passwords.

**Salt string:** A salt string is a random string that is used in conjunction with SHA-2 to create safer passwords and protect against attacks.

## 5 Context diagram

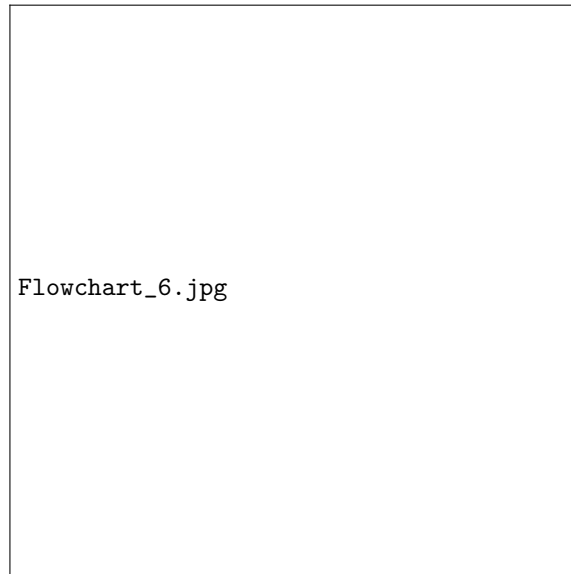


Figure 1: Context diagram

## 6 Functional requirements

### 6.1 Login and logout

#### 6.1.1 Requirement

If the user has forgotten their password, they should be able to get a newly generated password from the server by email.

#### 6.1.2 Requirement

New users should be assigned passwords generated by the server according to the password ruleset and receive it by email.

### 6.2 Data

#### 6.2.1 Requirement

A password should consist of at least one character of each type. `[a-z][A-Z][1-9]` with a minimum of 8 characters.

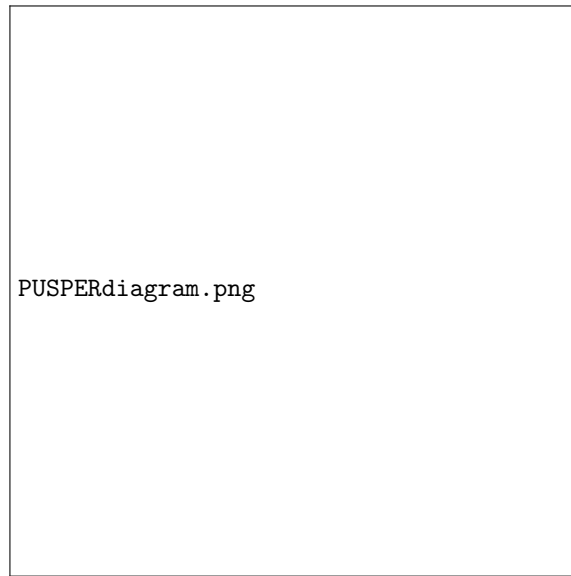


Figure 2: ER-diagram

### 6.2.2 Requirement

Passwords should be encrypted using SHA-2 and include a randomly generated salt string in the encryption process.

## 6.3 User

### 6.3.1 Requirement

Users should be able to create and submit time reports

### 6.3.2 Requirement

The following scenario should be supported by the system.

**Scenario:** The user wants to submit a time report

**Prerequisites:** The user is logged in to the system

1. The user navigates to the page “Time Report”.
2. The user is presented with an overview of their previously reported time.
3. The user presses the button “Create New Time Report”
4. The user fills in the blank textboxes with the time in minutes they have spent on the different activities.
5. The user presses the button for “Submit”.

6. The user is presented with a popup window confirming that the changes have been made.
7. An updated view of the page is displayed.

### 6.3.3 Requirement

A user should be able to see a summarized view of previous reported time.

### 6.3.4 Requirement

The following scenario should be supported by the system.

**Scenario:** The user wants to see a summary of their reported time.

**Prerequisites:** The user is logged in to the system

1. The user navigates to the page “Time report”.
2. A page is displayed which includes the option to report time, edit old time reports but also review a summarized view of their previously logged time.

### 6.3.5 Requirement

A user should be able to change their time report after it has been submitted to the system.

### 6.3.6 Requirement

The following scenario should be supported by the system.

**Scenario:** A user wants to change previously reported time.

**Prerequisites:** The user is logged in to the system

1. The user navigates to the page “Time report”.
2. The user is presented with an overview of their previously reported time.
3. The user chooses a time report.
4. The user presses the button “Edit time report”
5. The user is presented with a new page which includes the previously reported time table.
6. The user fills in the new time
7. The user clicks on the button “Submit Change”.
8. The system confirms that the changes have been made.
9. An updated view of the page is displayed.

### 6.3.7 Requirement

Users should be able to change their password on the “My profile” page.

### 6.3.8 Requirement

Users changing their passwords should not be able to set their new password to be the same as their current password.

### 6.3.9 Requirement

The following scenario should be supported by the system.

**Scenario:** A user wants to change their password.

**Prerequisites:** The user is logged in to the system

1. The user clicks on the “Change Password” button in the menu.
2. The user is presented with a page which includes three relevant textboxes.
3. The user fills in the first textbox with their current password.
4. The user fills in the second textbox with their new password.
5. The user repeats their new password in the third textbox.
6. The user clicks the button “Change Password”.
7. The user’s password is changed in the system.
8. The user is presented with a popup window confirming that the changes has been made.

### 6.3.10 Requirement

The following scenario should be supported by the system.

**Scenario:** A user wants to change their password with invalid entry.

**Prerequisites:** The user is logged in to the system

1. The user clicks on the “Change Password” button in the menu.
2. The user is presented with a page which includes three relevant textboxes.
3. The user fills in the first textbox with their current password.
4. The user fills in the second textbox with their new password which does not meet requirement 6.2.1.
5. The user repeats their new password in the third textbox.
6. The user clicks the button “Change Password”.
7. The user’s password is not changed and a popup window with an error message is displayed.
8. The user is sent back to stage 3.

### 6.3.11 Requirement

Users should not be able to edit signed time-reports.

### 6.4 Project Leader

#### 6.4.1 Requirement

The following scenario should be supported by the system.

**Scenario:** The project leader tries to acquire a summary of reported time.

**Prerequisites:** Time to be summarized exists, user has the role "Project leader" and the user is on the "Time Reporting" page.

1. The project leader clicks on the "SHOW SIGNED REPORTS" button.
2. A list with user's name and the total reported time will be displayed,
3. The project leader clicks on a user name and it displays the user's time reports.

#### 6.4.2 Requirement

The project leader should be able to create and submit a time report.

#### 6.4.3 Requirement

Project leader should be able to view all project members and their individual roles.

#### 6.4.4 Requirement

The project leader should be able to change a project members role except themselves and other project leaders.

#### 6.4.5 Requirement

Project leaders should be able to sign time reports, making them uneditable. This should be reversible.

#### 6.4.6 Requirement

The following scenario should be supported by the system.

**Scenario:** Project leader tries to sign a weekly report

**Prerequisites:** User has the role "Project leader", weekly reports exists

1. The project leader clicks on the "Sign Time Reports" button.
2. A list of all the unsigned weekly reports is shown to the project leader.
3. The project leader can sign the reports by clicking on the corresponding checkbox.
4. The project leader confirms their selection by pressing on the "Confirm" button
5. The system sends the changes to the server.
6. The system returns an updated list of unsigned reports, or an empty list if none exists



### 6.4.7 Requirement

The following scenario should be supported by the system.

**Scenario:** Project leader tries to assign roles to members.

**Prerequisites:** User has the role “Project leader”, members exists and the user is on the “User Management” page.

1. The project leader can change members’ roles by clicking on the corresponding radio button.
2. Project leader confirms changes made by pressing “Confirm”.
3. The system sends the changes to the server.

## 6.5 Administrator

### 6.5.1 Requirement

Only the administrator should have access to the administration page.

### 6.5.2 Requirement

All requirements and scenarios listed for “Users” and “Project Leaders” should also apply to the administrator.

### 6.5.3 Requirement

The following scenario should be supported by the system.

**Scenario:** The administrator tries to add a user.

**Prerequisites:** The administrator is on the “Administration” page

1. Administrator clicks on the button to add a user.
2. Administrator inputs the name of the user.
3. The inputs are sent to the server which adds it to the database and generates a password.
4. The system creates a new entry in the user list and inserts the inputted name.
5. The server returns the password to the corresponding field in the user list.
6. The user clicks the button “Change Password”.
7. The updated user list is shown to the administrator.

### 6.5.4 Requirement

The following scenario should be supported by the system.

**Scenario:** The Administrator tries to add a user with the same name.

**Prerequisites:** The administrator is on the “Administration” page

1. Administrator clicks on the button to add a user.

2. Administrator inputs the name of the user.
3. The inputs are sent to the server, which notices that the name already exists in the database.
4. The server does not accept the inputted name.
5. Administrator is told that the user already exists with a message.

### 6.5.5 Requirement

The following scenario should be supported by the system.

**Scenario:** The administrator tries to remove a user.

**Prerequisites:** The administrator is on the “Administration” page

1. Administrator clicks on the button to remove the user.
2. Administrator must confirm the removal by typing in the user’s name and click on the “confirm” button.
3. The information is sent to the server.
4. The user is removed from the database.
5. The page is updated with the new user list, with a message confirming the change.

### 6.5.6 Requirement

The following scenario should be supported by the system.

**Scenario:** The administrator tries to acquire a summary of reported time. **Prerequisites:** Time to be summarized exists.

1. Administrator clicks on the “SHOW SIGNED REPORTS” button.
2. A list with the names of all the users and their total reported time will be displayed,
3. Administrator clicks on a user name and it displays the user’s time reports.

### 6.5.7 Requirement

The following scenario should be supported by the system.

**Scenario:** The administrator tries to assign roles to members.

**Prerequisites:** The administrator is on the “User Management” page and members exist.

1. Administrator can change members’ roles by clicking on the corresponding radio button.
2. Administrator confirms changes made by pressing “Confirm”.
3. The system sends the changes to the server.

## 7 Design requirements

### 7.0.1 Requirement

The main font should be web safe and belong to one of the following font families: Serif or Sans-serif.

### 7.0.2 Requirement

The main text should have a font size of 16 pixels (+-2 pixels). All main text should be of the same size. Textfields and textfield captions should have a font size of at least 16 pixels.

### 7.0.3 Requirement

The color scheme should consist of three primary colors. These should, if possible, follow the 60/30/10 design rule. Secondary colors are permitted only when dealing with less important information. The background should consist of a lighter color and the main text should consist of a darker shade.

### 7.0.4 Requirement

The content of the web page should be centered in the browser window. The content should keep its centered position when the window is extended.

### 7.0.5 Requirement

A fixed navigation/menu bar shall be placed horizontally at the very top of the page.

### 7.0.6 Requirement

Selections Main-page, Administration, User Management, Time Reporting and My profile should be included in the menu bar. Other menu selections should be categorized as subheadings belonging to one of the primary menu choices and should not be visible or clickable by default. The administration page should only be visible to the administrator

### 7.0.7 Requirement

When a menu selection containing at least one subheading is selected a vertical menu bar containing its subheadings should be revealed. A downward pointing arrowhead, should be placed next all menu selections containing at least one subheading. When a menu selection containing one or more subheadings is selected, the arrowhead should point upwards.

### 7.0.8 Requirement

The system logo and/or the system name should be placed on the far left of the menu bar.

### 7.0.9 Requirement

The logout button should always be visible.

## **8 Project requirements**

### **8.1 Development environment**

#### **8.1.1 Requirement**

The back end for the system should use Java JDK 11 as development environment.

#### **8.1.2 Requirement**

The front end for the system should use Bootstrap v.4.5.3.

#### **8.1.3 Requirement**

The database should be developed in MySQL.

#### **8.1.4 Requirement**

Error messages should be in the following format: "Error code – Reason".