

Project Management System

Software Requirement Specification

Project Code: PMS

Document Code: PMS\_SR– v1.0

**Fpt University HCMC**

**May 27th 2012**

# 1 Record of change

\*A - Added M - Modified D - Deleted

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Effective Date | Changed Items | A\* M, D | Change Description | New Version |
| 27/05/2012 | Initial | A | Add project over view | 1.0 |
| 27/05/2012 | Introduction | A | Add introduction | 1.0 |
| 27/05/2012 | Overall Description | A | Add Overall Description | 1.0 |
| 27/05/2012 | Functional Requirement | A | Add Functional Requirement | 1.0 |
| 27/05/2012 | Non-functional Require | A | Add Non-functional requirement | 1.0 |
| 27/05/2012 | Supporting Information | A | Add supporting information | 1.0 |
| 27/05/2012 | Overall Description | A | Add overall description | 1.0 |
| 27/05/2012 | Use case model | A | Add usecase model | 1.0 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# 2 SIGNATURE PAGE

ORIGINATOR: **Manh Hoang Truong** May 27th, 2012

Developer

REVIEWERS: **Ngo Duc Duy** May 27th, 2012

Developer

APPROVAL: **Le Ngoc Thach** May 27th, 2012

Project Manager

# 3 TABLE OF CONTENTS

[1 Record of change 2](#_Toc286137244)

[2 SIGNATURE PAGE 2](#_Toc286137245)

[3 TABLE OF CONTENTS 2](#_Toc286137246)

[4 Introduction 2](#_Toc286137247)

[4.1 Purpose 2](#_Toc286137248)

[4.2 Scope 2](#_Toc286137249)

[4.3 Definitions, Acronyms, and Abbreviations 2](#_Toc286137250)

[4.4 References 2](#_Toc286137251)

[4.5 Overview 2](#_Toc286137252)

[5 Overall Description 2](#_Toc286137253)

[5.1 Product perspective: 2](#_Toc286137254)

[5.2 Product functions: 2](#_Toc286137255)

[5.3 User characteristics: 2](#_Toc286137256)

[5.4 Constraints: 2](#_Toc286137257)

[5.5 Assumptions and dependencies: 2](#_Toc286137258)

[5.6 Requirements subsets: 2](#_Toc286137259)

[6 FUNCTIONAL Requirements 2](#_Toc286137260)

[6.1 Home Screen 2](#_Toc286137261)

[6.2 Log-in 2](#_Toc286137262)

[6.3 Log-out 2](#_Toc286137263)

[6.4 Monitor current level 2](#_Toc286137264)

[6.5 Filter Task 2](#_Toc286137265)

[6.6 Add New Task 2](#_Toc286137266)

[6.7 Filter Dashboard 2](#_Toc286137267)

[7 Use case model 2](#_Toc286137268)

[7.1 Actors 2](#_Toc286137269)

[7.2 Use cases 2](#_Toc286137270)

[7.3 Main Use Case Diagrams of the PMS 2](#_Toc286137271)

[7.4 Log-in 2](#_Toc286137272)

[7.5 Log-out 2](#_Toc286137273)

[7.6 Monitor current level 2](#_Toc286137274)

[7.7 Filter Task 2](#_Toc286137275)

[7.8 Add New Task 2](#_Toc286137276)

[7.9 Filter Dashboard 2](#_Toc286137277)

[8. NON-FUNCTIONAL Requirements 2](#_Toc286137278)

[8.1 Usability 2](#_Toc286137279)

[8.2 Reliability 2](#_Toc286137280)

[8.3 Performance 2](#_Toc286137281)

[8.4 Supportability 2](#_Toc286137282)

[8.5 Design Constraints 2](#_Toc286137283)

[8.6 On-line User Documentation and Help System Requirements 2](#_Toc286137284)

[8.7 Purchased Components 2](#_Toc286137285)

[8.8 Interfaces 2](#_Toc286137286)

[8.9 Licensing Requirements 2](#_Toc286137287)

[8.10 Legal, Copyright, and Other Notices 2](#_Toc286137288)

[8.11 Applicable Standards 2](#_Toc286137289)

[9. Supporting Information 2](#_Toc286137290)

# 4 Introduction

In this accelerating of technologies and economics, Project Management is growing extremely importance to organizations because it deals effectively with the management of change. Modern economics and business environment are complicated than ever, organizations are beginning to realize that the traditional forms of management cannot adapt efficiently to the dynamics and its resulting chaos.

In the future, software industries could very well depend on how quickly the procedures and systems of project management are adopted. Therefore, Project Manager will play a very important role within organizations. Our mission is to provide an online system which actively assists those managers in managing their project. The approaches and techniques that are used in the project management process are of interest to all those who wish to be more certain about achieving predetermined targets.

## 4.1 Purpose

This Software Requirements Specification provides a complete description of all the functional requirements, non-functional requirements, constraints and other requirement specification of the “Project Management System”.

## 4.2 Scope

This project aims to create an online software project management system. With friendly interface, powerful tools, OOPMS provides powerful and efficient customized service for numerous kinds of managers from small to medium projects. Besides, OOPMS can be deployed on J2EE portal servers (JSR168, JSR268).

## 4.3Definitions, Acronyms, and Abbreviations

Table 1 Abbreviations

|  |  |  |
| --- | --- | --- |
| No. | Key word | Meaning |
| 1 | PM | Project Manager |
| 2 | PMS | Project Management System |
| 3 | ISO | International Standard Organization |

Table 01: Abbreviations

## 4.4 References

FSOFT PMS System

## 4.5 Overview

The rest of this document includes four chapters:

The 2nd“Overall Description” chapter describes an overview description of this document. It lists all the functions, constraints of the application.

The 3rd“Functional Requirements” chapter lists all the required functions for PMS.

The 4th“Non-functional Requirements” chapter provides non-functional requirements, constraints of the system that shall be satisfied.

The final “Supporting Information” chapter gives supporting and additional information in order to make the PMS easier to use.

# 5 Overall Description

This chapter describes an overview description of this document by listingall the functions, characteristics, constraints of the application.

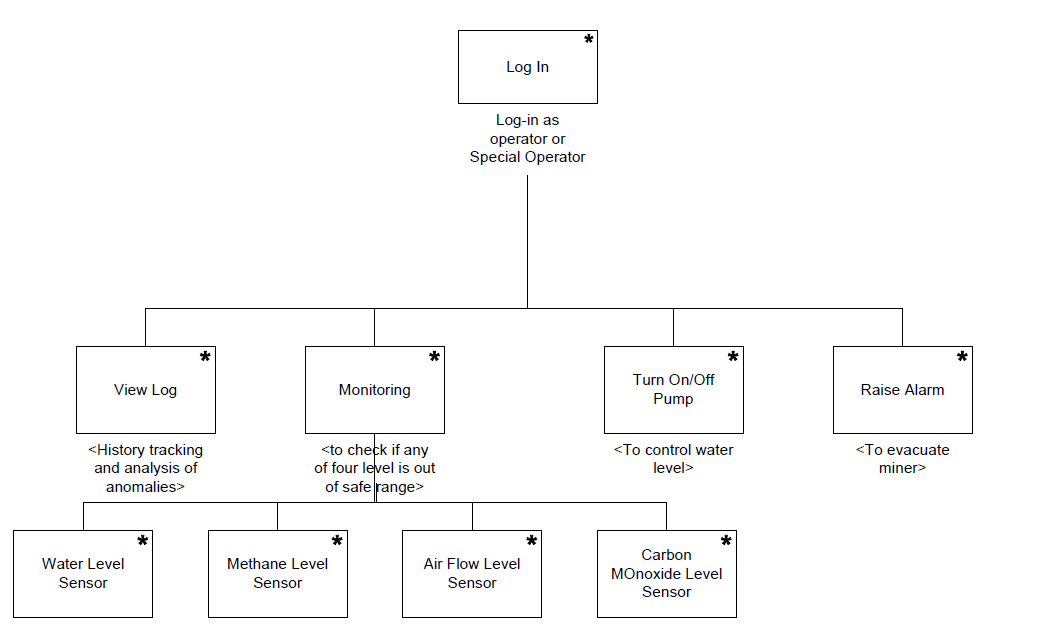


Figure 1PMSOverview

## 5.1 Product perspective:

The PMS shall be developed to manage requirements, tasks, defects, time tracking and report in software projects.

## 5.2 Product functions:

The PMS provides functions as below :

## 1. Dashboard: show present status of projects which have been created in OOPMS. Project managers and others senior manager can easily track and monitor status of their project.

## 2. Planner:

## a. Allow users import Microsoft Project file (\*.mpp) into OOPMS and in the future, users also can edit and export \*.mpp files at OOPMS;

## b. Create task and assign task to team members.

## 3. Report: support creating report of project weekly and allow Team leader of project add more comments to the report.

## 4. Project Eye: manage detail information about a specific project.

## 5. Time Tracking: manage working time of team members.

## 6. DMS: manage defects of projects.

## 7. Requirement: include list of requirements (function, non-function). Allow users to keep track status of those requirements

## 8. Admin: Allow admin to manage users and manage system data

## .5.3 User characteristics:

The PMSis designed for human operatorswho are assumed to have the following characteristics:

+ Fully understand the process of Safety Control System and how it is important inside a mine.

+ No special knowledge or skills.

+ No training in computer system use.

## 5.4 Constraints:

The PMS is a real time system that is responsible for managing project information. It must operate correctly, accurately and in time.

## 5.5 Assumptions and dependencies:

None

## 5.6 Requirements subsets:

None

# 6 FUNCTIONAL Requirements

Dashboard

Planner

Report

Admin

Project Eye

(Product + stages + Deliverables)

Timesheet

DMS

Requirement

Include: Description, Use Case, Use Case scenario

This chapter of the document describes all the functions that the PMSapplication shall provide, and how the system operating these functions using natural language and model, chart.

## 6.1 Home Screen

### 6.1.1 Screen Layout

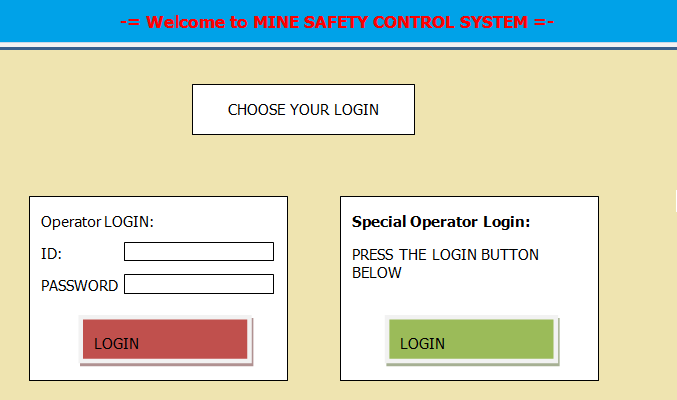


Figure 2 Home Screen

## 6.2 Log-in

### 6.2.1 Screen Layout

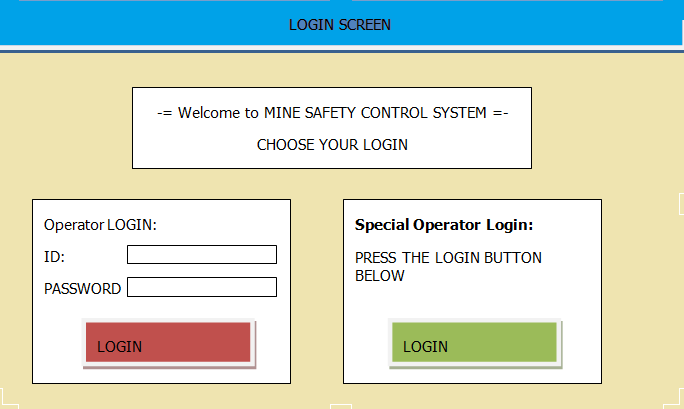


Figure 3 Login Screen

### 6.2.2 Description

This function shall beused to check username and password to log into the system. Only logged in user can use the system.

The PMSallows Special Operator adding new user, deleting current user, and editing current user’s information.

PMSshall clearly define Operator or Special Operator with different authorization from Log-in phase.

Users’ attributes shall include:

Table 2 Users' attributes

|  |  |  |
| --- | --- | --- |
| No. | Field name | Remarks |
| 1 | User ID | Each user has a unique identification number |
| 2 | Name | User’s name |
| 3 | Address | User’s publications receiving address |
| 4 | Phone | User’s contact phone number |
| 5 | Position | Position in Organization |
| 6 | Department | Department that user belong to |
| 7 | Working status | Working status : on work or on vacation |

Table 02: Users’ attributes

## 6.3 Log-out

### 6.3.1 Screen Layout

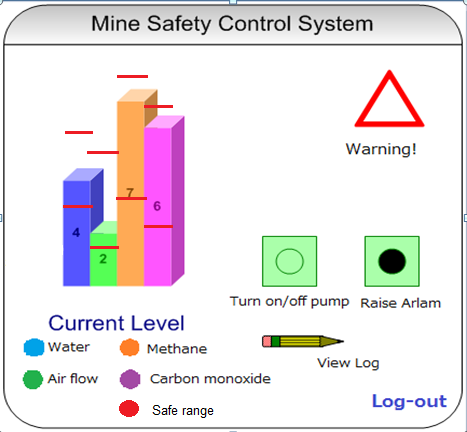


Figure 4 Logout Screen

### 6.3.2 Description

This function provide user to log out the system. Only logged in user can log out the system.

## 6.4 Monitor current level

### 6.4.1 Screen Layout

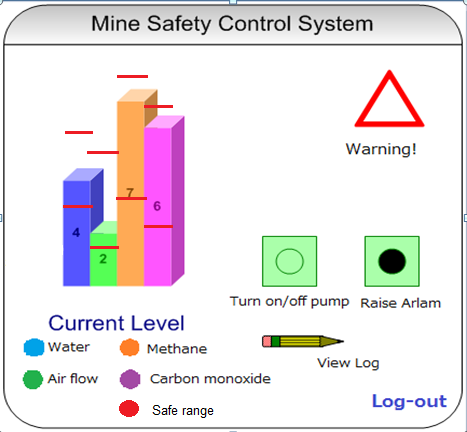


Figure 5 Monitor Screen

### 6.4.2 Description

This function provide user to monitor Water, Methane, Air Flow, Carbon monoxide Level inside a mine, give warning when the level reach nearby the critical range and allow user to Filter Task as well as Add New Task when needed, Filter Dashboard.

Item (Water, Airflow, Methane, Carbon monoxide) Level attributes shall include:

Table 3 Items' attributes

|  |  |  |
| --- | --- | --- |
| No. | Field name | Remarks |
| 1 | Item ID | Each Item has a unique identification number |
| 2 | Name | Item’s name |
| 3 | Description | Item’s description |
| 4 | High Level | Min acceptable level |
| 5 | Low Level | Max acceptable level |
| 6 | Critical range | Critical range |

Table 03: Items’ attributes

## 6.5 Filter Task

### 6.5.1 Screen Layout

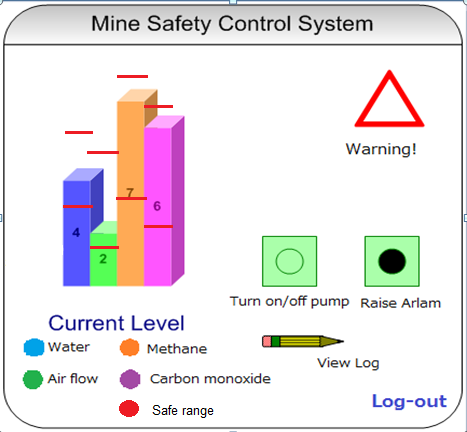


Figure 6 Filter Task

### 6.5.2 Description

This function provide user to Filter Task to control water level.

Pump actuator can be control automatically by PMS controller or manually by operators.

Manually By Operators - Use Case 03a:

Name : Filter Task

Goal : Control water level

Actors : All users (include operator and special operator), pump actuator, water sensors

Pre-conditions: Logged user

Post-conditions: None

Main Flow:

Currently in monitor screen

Raise warning to user whenever water level enter critical range

Receive Filter Task demand from user

Filter Task as order

Display monitor screen for user keep checking

Automatically by PMS- Use Case 03b:

Name : Filter Task

Goal : Control water level

Actors : pump actuator, water sensors

Pre-conditions: None

Post-conditions: None

Main Flow:

Currently in monitor screen

Raise warning to user whenever water level enter critical range

Automatically Filter Task by pump actuator whenever water level reach high or low point.

Display monitor screen for user keep checking

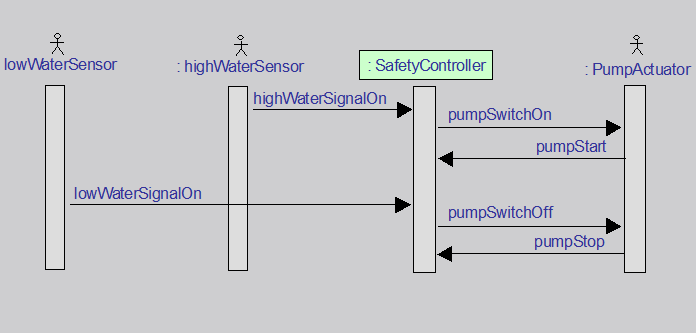


Figure 7 Water sensor operation - Sequence Diagram

## 6.6 Add New Task

### 6.6.1 Screen Layout

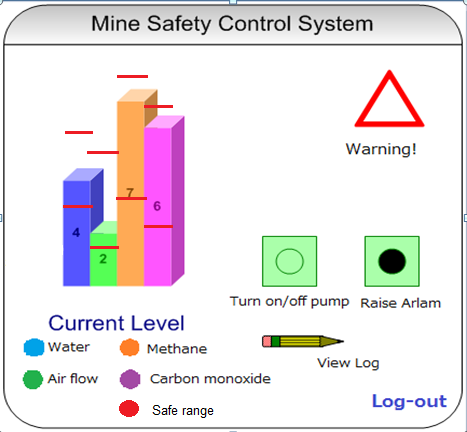


Figure 8 Add New Task

### 6.6.2 Description

This function provide user to Add New Taskwhen one of the item’s level out of safe rangeto evacuate miners.

Alarm actuator can be control automatically by PMScontroller or manually by operators.

Manually by Operators - Use Case 04a:

Name : Add New Task

Goal : Evacuate miners

Actors : All users (include operator and special operator), alarm actuator, methane sensors, or airflow sensors, or carbon monoxide sensors.

Pre-conditions: Logged user

Post-conditions: None

Main Flow:

Currently in monitor screen

Raise warning to user when item level enter critical range

Receive Add New Task demand from user

Activate alarm system as order

Save log data into hard drive and display monitor screen for user keep checking

Automatically by PMS- Use Case 04b:

Name : Add New Task

Goal : Evacuate miners

Actors : Alarm actuator,methane sensors, or airflow sensors, or carbon monoxide sensors.

Pre-conditions: None

Post-conditions: None

Main Flow:

Currently in monitor screen.

Raise warning to user when item level enter critical range.

Add New Task by alarm actuator whenever one of those level reach high or low point.

Save log data into hard drive and display monitor screen for user keep checking.

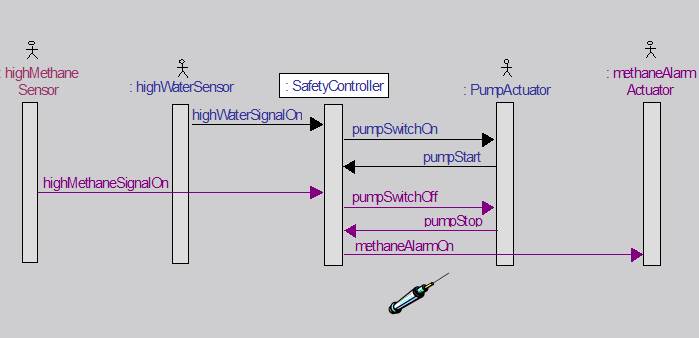


Figure 9 Methane Sensor operation - Sequence Diagram

## 6.7 Filter Dashboard

### 6.7.1 Screen Layout

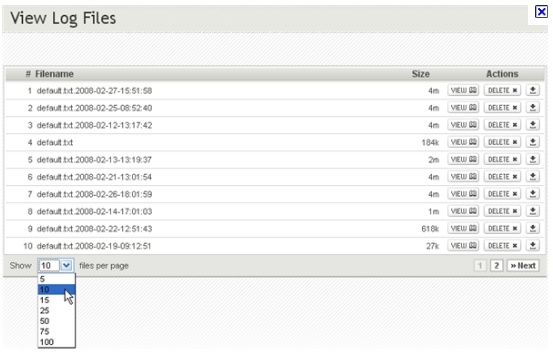


Figure 10 Filter Dashboard Screen

### 6.7.2 Description

This function provide user to Filter Dashboard of the PMS.

Use Case 05:

Name : Filter Dashboard

Goal : Provide log data

Actors : All users (include operator and special operator)

Pre-conditions: Logged user

Post-conditions: None

Main Flow:

Currently in monitor screen

Receive Filter Dashboard command

Go to Filter Dashboard screen and display Log data

# 7 Usecase model

This chapter describes PMS’s features and functions using abstract Use Cases and detail scenarios.

A use case is a description of a sequence of actions (including itsvariations) that the system carries out to create an observable result foran actor.

A scenario is atemporal sequence of interaction events among agent instances.

## 7.1 Actors

Table of all main functions’ actors:

This table indicates which actors associating with which functions.

Table 4 Main functions' actors

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Main functions | Operator | Special Operator | Water Sensors | Methane, Airflow, Carbon monoxide sensors | Pump actuator | Alarm actuator |
| 1 | Log-in | x | x |  |  |  |  |
| 2 | Log-out | x | x |  |  |  |  |
| 3 | Monitor current level |  |  | x | x |  |  |
| 4 | Filter Task | x | x | x |  | x |  |
| 5 | Add New Task | x | x |  | x |  | x |
| 6 | Filter Dashboard | x | x |  |  |  |  |

Table 04: Main functions’ actors

## 7.2 Use cases

This table list all use cases for each corresponding system functions.

Table 5Use cases List

|  |  |  |
| --- | --- | --- |
| System Functions | Main Use Cases | Use Case ID |
| Log-in | | |
|  | Log-in Use Case | UC\_1.1 |
| Log-out | | |
|  | Log-out Use Case | UC\_1.2 |
| Monitor current level | | |
|  | Monitor current level Use Case | UC\_1.3 |
| Filter Task | | |
|  | Manually Filter Task Use Case | UC\_1.4a |
|  | Automatically Filter Task Use Case | UC\_1.4b |
| Add New Task | | |
|  | Manually Add New Task Use Case | UC\_1.5a |
|  | Automatically Add New Task Use Case | UC\_1.5b |
| Filter Dashboard | | |
|  | Filter Dashboard Use Case | UC\_1.6 |

Table 05: Use cases List

## 7.3 Main Use Case Diagrams of the PMS

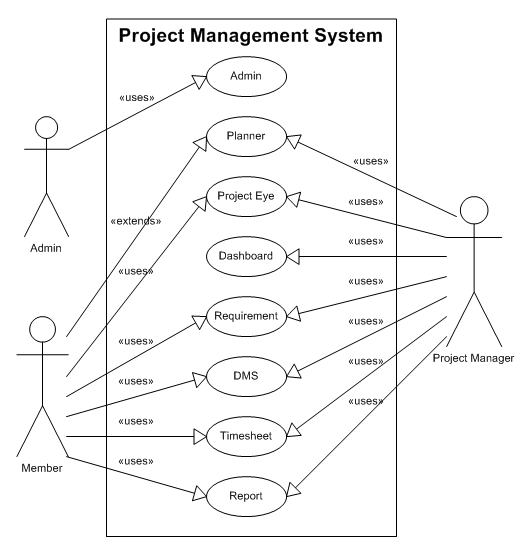


Figure 11 Main Use Case Diagrams of the PMS

This main Use Case Diagrams of the PMS show all main functions placinginside the system boundary and all actors that associate with those functions.

## 

## 7.1 Log-in

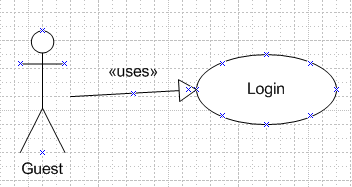


Figure 12 Log-in Use Case model

Use Case scenario:

|  |  |  |
| --- | --- | --- |
| User Case ID | General\_UC01 | |
| Name | Log-in Use Case | |
| Goal | Authenticate guests then authorize them. | |
| Actors | Guest | |
| Pre-conditions | None | |
| Post-conditions | None | |
| Main Flow | 1. User goes into PMS Homepage.  3. Users enter user’s information: Username, Password and hit “Log-in” button. | 2. Display Home Screen and request user to log in.  4. Validate user’s information.  5.Display monitor screen to corresponding user. |
| Exception | If username or password is not correct, the PMSwill show error message and ask user to log in again. | |
| Open Issues | N/A | |

Table 06: Log-in scenario

## 7.2 Log-out

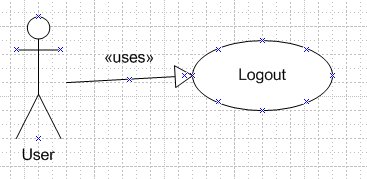


Figure 13:Log-out Use Case model

Use Case scenario:

|  |  |  |
| --- | --- | --- |
| User Case ID | General\_UC02 | |
| Name | Log-out Use Case | |
| Goal | Provide Log-out function for User to check out | |
| Actors | User(include member, admin, project manager) | |
| Pre-conditions | Logged user | |
| Post-conditions | None | |
| Main Flow | 1. Users are using PMS and click “Log-Out” button. | 2. Log user out and display Home Screen. |
| Exception | None | |
| Open Issues | N/A | |

Table 07: Log-out scenario

## 7.3 Search Timesheet

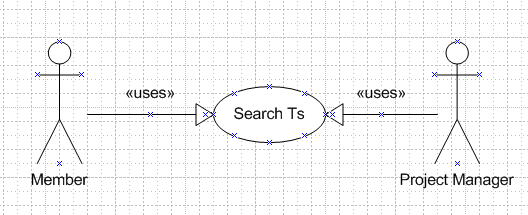


Figure Search timesheet Use Case model

Use Case scenario:

|  |  |  |
| --- | --- | --- |
| User Case ID | Timesheet\_UC01 | |
| Name | **Search Timesheet Use Case** | |
| Goal | This function allows users can search timesheet. | |
| Actors | **Add Timesheet** | |
| Pre-conditions | User logins must be member of project | |
| Post-conditions | None | |
| Main Flow | 1. Users logins to Timesheet system.  3. Click button Search | 2. Select search condition. |
|  |  | |
| Open Issues | N/A | |

Table 06: Search timesheet scenario

## 7.4 Add timesheet

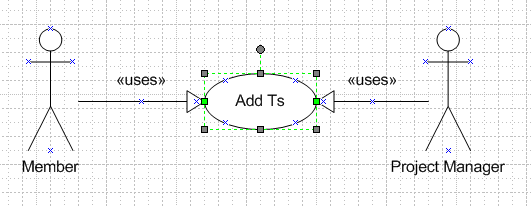


Figure : Add time sheet Use Case model

Use Case scenario:

Table Log-out scenario

|  |  |  |
| --- | --- | --- |
| User Case ID | Timesheet\_UC02 | |
| Name | **Add Timesheet t Use Case** | |
| Goal | This function allows users can add a new timesheet record | |
| Actors | **Add Timesheet** | |
| Pre-conditions | User logins must be member of project | |
| Post-conditions | None | |
| Main Flow | 1. Users logins to Timesheet system. | 2. Click button add new. |
| Exception | None | |
| Open Issues | N/A | |

Table 07: Add timesheet scenario

## 7.5 Update timesheet

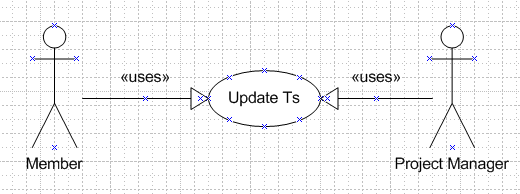


Figure Update timesheet Use Case model

Use Case scenario:

|  |  |  |
| --- | --- | --- |
| User Case ID | Timesheet\_UC03 | |
| Name | Update timesheet Use Case | |
| Goal | This function allows users can update timesheet records. | |
| Actors | Member , Project Manager | |
| Pre-conditions | User logins must be member of project | |
| Post-conditions | None | |
| Main Flow | 1. User logins to Timesheet system.  3. Click button update | 2. Select timesheet record to update |
| Exception | None | |
| Open Issues | N/A | |

Table 08: Update timesheet scenario

## 7.6 Delete timesheet

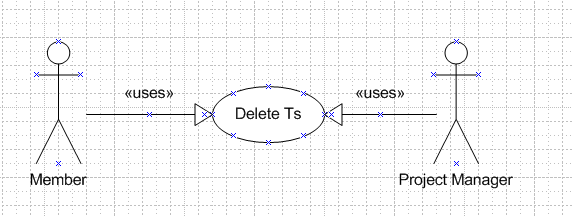


Figure Delete timesheet Use Case model

Use Case scenario:

|  |  |  |
| --- | --- | --- |
| User Case ID | Timesheet\_UC04 | |
| Name | Delete timesheet Use Case | |
| Goal | This function allows users can delete timesheet records. | |
| Actors | Member , Project Manager | |
| Pre-conditions | User logins must be member of project | |
| Post-conditions | None | |
| Main Flow | 1. User logins to Timesheet system.  3. Click button delete | 2. Select timesheet record to delete |
| Exception | None | |
| Open Issues | N/A | |

Table 09: Delete timesheet scenario

## 7.7 Approve/ Reject timesheet

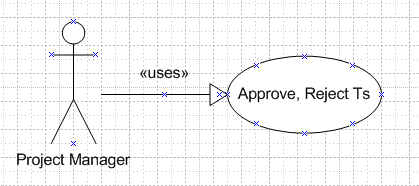


Figure Approve/ Reject timesheet Use Case model

Use Case scenario:

|  |  |  |
| --- | --- | --- |
| User Case ID | Timesheet\_UC05 | |
| Name | Approve/ Reject timesheet Use Case | |
| Goal | Allow project manager to approves or reject members‘ timesheet records | |
| Actors | Project manager | |
| Pre-conditions | Logged users as role project manager | |
| Post-conditions | None | |
| Main Flow | 1. User logins to Timesheet system  as role project manager  3. Click button approve or reject | 2. Select Timesheet to approve or reject |
| Exception | N/A | |
| Open Issues | N/A | |

Table 10: Approve/ Reject timesheet scenario

## 7.8 Search defect

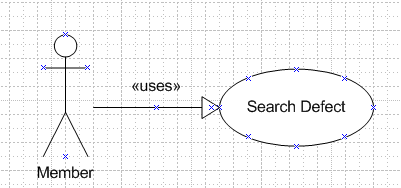


Figure Search defect Use Case model

Use Case scenario:

Table Search defect Scenario

|  |  |  |
| --- | --- | --- |
| User Case ID | DMS\_UC01 | |
| Name | Search defect Use Case | |
| Goal | This function allows users can search defect.  Defect has 3 kinds : open defect, closed defect, leakage | |
| Actors | Member | |
| Pre-conditions | Logged user | |
| Post-conditions | None | |
| Main Flow | 1. Users login into DMS system 2. Click button search | 1. Select search condition |
| Exception | None | |
| Open Issues | N/A | |

Table 11: Search defect scenario

## 7.9 Add defect

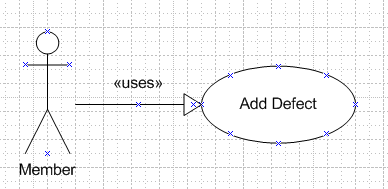


Figure Search defect Use Case model

Use Case scenario:

|  |  |  |
| --- | --- | --- |
| User Case ID | DMS\_UC02 | |
| Name | Add defect Use Case | |
| Goal | This function allows users can add defect.  When defect was added, member become creator, defect change status to opened | |
| Actors | Member | |
| Pre-conditions | Logged user | |
| Post-conditions | None | |
| Main Flow | 1Users login into DMS system | 1. Click button add new |
| Exception | None | |
| Open Issues | N/A | |

Table 11: Add defect scenario

## 7.11 Update defect

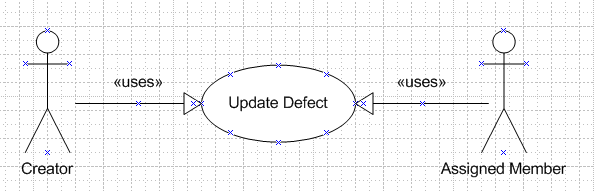


Figure Update defect Use Case model

Use Case scenario:

|  |  |  |
| --- | --- | --- |
| User Case ID | UC\_1.6 | |
| Name | Update defect Use Case | |
| Goal | This function allows users can update defect.  Creator can change status of defect from opened to cancelled or corrected.  Assigned member can change status of defect to opened, canceled or closed. | |
| Actors | Member | |
| Pre-conditions | Logged user | |
| Post-conditions | None | |
| Main Flow | 1Users login into DMS system  3..Click button update | 1. Select defect |
| Exception | None | |
| Open Issues | N/A | |

Table 11: Update defect scenario

## 7.9 Filter Dashboard



Figure Filter Dashboard Use Case model

Use Case scenario:

|  |  |  |
| --- | --- | --- |
| User Case ID | Dashboard \_UC01 | |
| Name | Filter Dashboard | |
| Goal | This function allows user to filter projects, which appear on Dashboard. Project on system can be filtered by status or category. | |
| Actors | Project Manager | |
| Pre-conditions | Users must log in with role “Project Manager”; Users must go to Dashboard page. | |
| Post-conditions |  | |
| Main Flow | 1. Select conditions to filter and Click “Search” button. | 2. Display filtered project list. |
| Exception | N/A | |
| Open Issues | N/A | |

Table 11: Filter Dashboard scenario

## 7.10 Export Dashboard



Figure 22 Export Dashboard Use Case model

Use Case scenario:

|  |  |  |
| --- | --- | --- |
| User Case ID | Dashboard \_UC02 | |
| Name | Export Dashboard | |
| Goal | This function allows Project Manager to export Dashboard to Microsoft Excel file. | |
| Actors | Project Manager | |
| Pre-conditions | Users must log in with role “Project Manager”; Users must go to Dashboard page. | |
| Post-conditions |  | |
| Main Flow | 1. Filter project list (optional).  2. User Click on “Export” button. | 3. Export file. |
| Exception | N/A | |
| Open Issues | N/A | |

Table 08: Export Dashboard scenario

## 7.12 Filter Task



Figure 23 Filter Task Use Case model

Use Case scenario:

|  |  |  |
| --- | --- | --- |
| User Case ID | Planner\_UC01 | |
| Name | Filter Task | |
| Goal | This function allows Project Manager and Team Member to filter tasks that are displayed on screen. | |
| Actors | Project Manager; Team Member | |
| Pre-conditions | Users must log in with role “Project Manager” or “Team Member”; Users must go to Planner page | |
| Post-conditions |  | |
| Main Flow | 1. Select conditions to filter and click “Search” button. | 2. Show task list after filtering. |
| Exception | N/A | |
| Open Issues | N/A | |

Table 09: Filter Task scenario

## 7.13 Add New Task



Figure 24Add New Task Use Case model

Use Case scenario:

|  |  |  |
| --- | --- | --- |
| User Case ID | Planner\_UC02 | |
| Name | Add New Task | |
| Goal | This function allows Project Manager to plan and add specific tasks to his/her project. | |
| Actors | Project Manager | |
| Pre-conditions | Users must log in with role “Project Manager”; Users must go to Planner page | |
| Post-conditions |  | |
| Main Flow | 1. Click “Add New Task” button  3. Fill up task’s information.  6. Click “Save” button. | 2. Redirect to Task Adding Page.  3. Validate task’s information.  5. Display “Complete!” message. |
| Exception | 3A: if task information is not correct, show error message and ask to input again. | |
| Open Issues | N/A | |

Table 10: Add New Task scenario

## 7.14 Update Task



Figure Update Task Use Case model

Use Case scenario:

|  |  |  |
| --- | --- | --- |
| User Case ID | Planner\_UC02 | |
| Name | Update Task | |
| Goal | This function allows Project Manager to plan and add specific tasks to his/her project. | |
| Actors | Project Manager | |
| Pre-conditions | Users must log in with role “Project Manager”; Users must go to Planner page | |
| Post-conditions |  | |
| Main Flow | 1. Click “Update Task” button  3. Fill up task’s information.  6. Click “Save” button. | 2. Redirect to Task Adding Page.  3. Validate task’s information.  5. Display “Complete!” message. |
| Exception | 3A: if task information is not correct, show error message and ask to input again. | |
| Open Issues | N/A | |

Table 10: Update Task scenario

## 7.15 Delete Task



Figure Delete Task Use Case model

Use Case scenario:

Table Update Task scenario

|  |  |  |
| --- | --- | --- |
| User Case ID | Planner\_UC04 | |
| Name | Delete Task | |
| Goal | This function allows Project Manager delete task on Planner page. | |
| Actors | Project Manager | |
| Pre-conditions | Users must log in with role “Project Manager”; Users must go to Planner page | |
| Post-conditions |  | |
| Main Flow | 1. Select a task, which he/she want to delete.  2. Click “Delete” button. | 3. Display “Delete Successfully” message. |
| Exception | N/A | |
| Open Issues | N/A | |

Table 10: Delete Task scenario

## 7.15 Import Task



Figure Delete Task Use Case model

Use Case scenario:

Table Update Task scenario

|  |  |  |
| --- | --- | --- |
| User Case ID | Planner\_UC05 | |
| Name | Import Task | |
| Goal | This function allows Project Manager to import task from Microsoft Project file. | |
| Actors | Project Manager | |
| Pre-conditions | Users must log in with role “Project Manager”; Users must go to Planner page | |
| Post-conditions |  | |
| Main Flow | 1. User clicks “Import” button.  3. User selects a \*.mpp file from PC and clicks “OK” button. | 2. Display window to user select \*.mpp file.  4. Import tasks in the file to Planner. |
| Exception | 3A: if the file is not correct, show error message and ask to input again. | |
| Open Issues | N/A | |

Table 10: Import Task scenario

## 7.16 Report Task



Figure Report Task Use Case model

Use Case scenario:

|  |  |  |
| --- | --- | --- |
| User Case ID | Planner\_UC06 | |
| Name | Print Report | |
| Goal | This function allows Project Manager and Team Member to view and print report about planner. | |
| Actors | Project Manager; Team Member | |
| Pre-conditions | Users must log in with role “Project Manager” or “Team Member” ; Users must go to Planner page | |
| Post-conditions |  | |
| Main Flow | 1. User clicks “Report” button.  3. User clicks “Report” button. | 2. Display Report of Planner  4. Export Report. |
| Exception | N/A | |
| Open Issues | N/A | |

Table 10: Report Task scenario

# 8. NON-FUNCTIONAL Requirements

This chapter of the document describes all the non-functions requirement of the PMS.

The PMS’s usability, availability, performance are very important factors to ensure that the system operate effectively and keep hundred miners safe.

## 8.1 Usability

The PMS usability is the key factor to ensure that the system run exactly as well as the operators can control the system efficiently.

The PMS application shall provide clear, friendly and easy interface to operate so that system users have to spend no more than one hour learning to use the system.

### 8.1.1 Background knowledge

PMS users are assumed to have very basic knowledge at using computer systems.

### 8.1.2 Training

Operators’ training time requirement: 60 minutes.

## 8.2 Reliability

### 8.2.1 Availability

Whenever the system is during operation time no matter how many people working inside, the PMS has to be ON.

### 8.2.2 Mean Time between Failures (MTBF)

Mean Time between Failures (MTBF): more than 6 months.

### 8.2.3 Mean Time to Repair (MTTR)

Mean Time To Repair (MTTR): less than 48 hours.

### 8.2.4 Accuracy

Accuracy: 100%

### 8.2.5 Maximum Bugs and Defect Rate

Maximum Bugs and Defect Rate: 0.3 bugs per thousand lines of code (0.3 bugs/KLOC).

### 8.2.6 Critical Bugs

Critical bugs:

+ Loss ofLog: No

+ Unable to operate any function: No

## 8.3 Performance

### 8.3.1 Response Time

Response time for a respond:

Average: 50 milliseconds

Maximum: 80 milliseconds

### 8.3.2 Capacity

Only one operator at the same time.

### 8.3.3 Resource utilization

Memory:

+ 512MB of RAM

Operating System:

+ Microsoft Windows XP or newer version

+ Mac OS X 10.0 or newer version

+ Linux 3.5 or newer version

## 8.4 Supportability

### 8.4.1 Coding standards

According to “Standard Java Coding Convention”

– 09be-HD/PM/HDCV/FSOFT - Version 1/1.

### 8.4.2 Maintenance Utilities

Support working hours phone call technical support: 8:00 to 16:00 from Monday to Friday.

## 8.5 Design Constraints

Coding standard:

+ This application will be developed in Java programming language, version J2EE 6.

Software process requirements:

+ The software process shall confront to the CMMI 5 standard.

Developmental tools:

+ This application will be developed using Eclipse from Sun Microsystems.

### 8.5.1 Software Languages

GUI , Help documents, all other support documents are in English.

### 8.5.2 Software Process Requirement

The software process has to meet the CMMI 5 standard.

### 8.5.3 Development Tools

The PMS shall be developed using Eclipse 3.6 , SQA activity using FSOFT template and j-unit.

## 8.6 On-line User Documentation and Help System Requirements

The deployment of the application shall be provided by a technical agent include full help document for user.

## 8.7 Purchased Components

None

## 8.8 Interfaces

### 8.8.1 User Interfaces

As shown in Functional requirements – chapter 6

8.8.2Hardware Interfaces

API to interact with sensors are provided in advance.

## 8.9 Licensing Requirements

The PMS is only applicable for Project Management Department, which legally buy this product. No other use is legal.

## 8.10 Legal, Copyright, and Other Notices

This PMS is protected under the national law of copyright.

## 8.11 Applicable Standards

The system shall confront to the ISO 90003:2001 standard.

# 9. Supporting Information

**List of figures**:

[Figure 1 PMSOverview 2](#_Toc286136508)

[Figure 2 Derived use case diagram **Error! Bookmark not defined.**](#_Toc286136509)

[Figure 3 Home Screen 2](#_Toc286136510)

[Figure 4 Login Screen 2](#_Toc286136511)

[Figure 5 Logout Screen 2](#_Toc286136512)

[Figure 6 Monitor Screen 2](#_Toc286136513)

[Figure 7 Filter Task 2](#_Toc286136514)

[Figure 8 Water sensor operation - Sequence Diagram 2](#_Toc286136515)

[Figure 9 Add New Task 2](#_Toc286136516)

[Figure 10 Methane Sensor operation - Sequence Diagram 2](#_Toc286136517)

[Figure 11 Filter Dashboard Screen 2](#_Toc286136518)

[Figure 12 Main Use Case Diagrams of the PMS 2](#_Toc286136519)

[Figure 13 Log-in Use Case model 2](#_Toc286136520)

[Figure 14: Log-out Use Case model 2](#_Toc286136521)

[Figure 15 Monitor current level Use Case model 2](#_Toc286136522)

[Figure 16 Filter Task Use Case model 2](#_Toc286136523)

[Figure 17 Add New Task Use Case model 2](#_Toc286136524)

[Figure 18 Filter Dashboard Use Case model 2](#_Toc286136525)

**List of tables:**

[Table 1 Abbreviations 2](#_Toc286136530)

[Table 2 Users' attributes 2](#_Toc286136531)

[Table 3 Items' attributes 2](#_Toc286136532)

[Table 4 Main functions' actors 2](#_Toc286136533)

[Table 5 Use cases List 2](#_Toc286136534)

[Table 6 Log-in scenario 2](#_Toc286136535)

[Table 7 Log-out scenario 2](#_Toc286136536)

[Table 8 Monitor current level scenario 2](#_Toc286136537)

[Table 9 Filter Task scenario 2](#_Toc286136538)

[Table 10 Add New Task scenario 2](#_Toc286136539)

[Table 11 Filter Dashboard Scenario 2](#_Toc286136540)