

# CHANGE MANAGEMENT TOOL

## Software Requirements Specification

**TEAM 6**

## Contents

- [1. Introduction](#)
- [2. Purpose](#)
- [3. Scope](#)
- [4. Definitions, Acronyms and Abbreviations](#)
- [5. References](#)
- [6. Overall Description](#)
- [7. External Interface Requirements](#)
- [8. Functional Requirements](#)
- [9. Non Functional Requirements](#)
  - [9.1 Usability](#)
  - [9.2 Security](#)
  - [9.3 Performance](#)
  - [9.4 Capacity](#)
  - [9.5 Recovery](#)
  - [9.6 Availability](#)
  - [9.7 Reliability](#)
  - [9.8 Maintainability](#)
  - [9.9 Portability](#)
  - [9.10 Privacy](#)

## Revision History

Name	Date	Reason For Changes	Version

## 1. Introduction

When an organization introduces a change with a project or initiative, that change needs to be effectively managed on both the technical side and the people side. A technical side focus ensures that the change is developed, designed and delivered effectively. The discipline of project management provides the structure, processes and tools to make this happen. A people side focus ensures that the change is embraced, adopted and utilized by the employees who have to do their jobs differently as a result of the project. The discipline of change management provides the structure, processes and tools to make this happen.

The project aims to provide a platform wherein the registered users can request for necessary changes, which can then be conveyed to the team or individual responsible to carry out that change in a timely manner. Thus, the objective is to provide a system which can enhance the inter departmental communication and make the process of change management as user friendly as possible while keeping in mind the time limitations and priority of the requested changes

## 2. Purpose

The purpose of this Software Requirements Specification (SRS) document is to describe the overall behavior of Change Management TOOL(CMT). This SRS defines and describes the operations, performance, and quality assurance requirements of the CMT that is to be developed. This document also describes the nonfunctional requirements. It also describes the design constraints and technologies that are to be considered when the system is to be designed, and other factors necessary to provide a complete description of the requirements for the system. This Software Requirements Specification (SRS) captures the complete software requirements for the system. Requirements described in this document will be used as guidelines to develop the CHANGE MANAGEMENT TOOL.

## 3. Scope

The Change Management Tool aims at automation of the following processes

- Creation of maintenance tasks and assignment of engineers.
- Submission of change request
- Prioritization of change
- Escalation of unattended request
- Reopening of closed tickets
- Report generation
- Assignment algorithm of ticket management

#### 4. Definitions, Acronyms and Abbreviations

Abbreviation	Designation
CMT	Change Management Tool.
SRS	Software Requirements Specification

#### 5. References

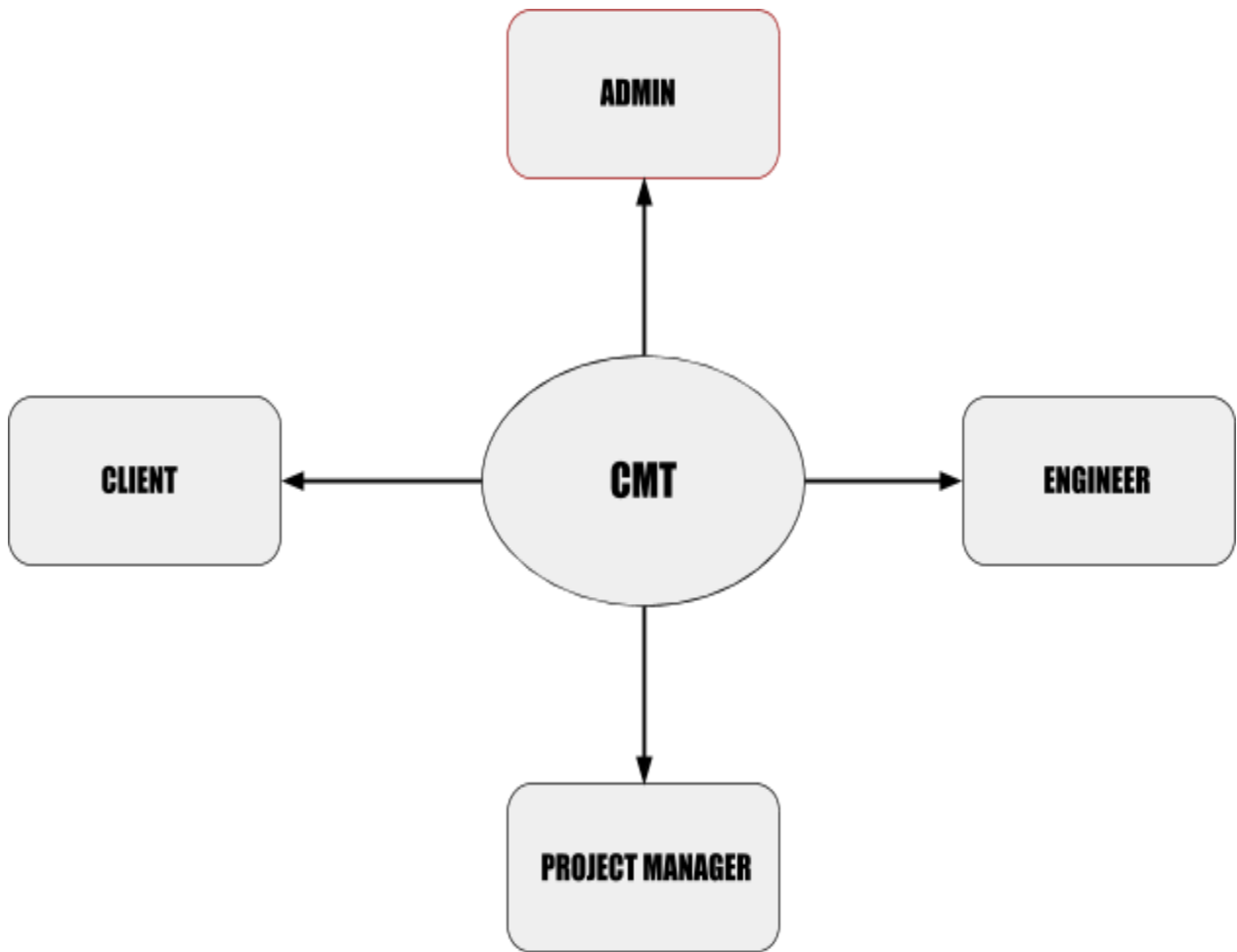
Existing Change Management Tools like PROSCII

#### 6. Overall Description

(Thirumurugan S S 13BCE0293, Ambikan S 13BCE0767)

##### 6.1 Product Perspective

Change Management Tool will provide a medium to facilitate inter departmental communication thus making the process of change management as user friendly as possible while keeping in mind the constraints like time limitations and priority of the requested changes.



*Overview of the proposed system*

## 6.2 Product Functions

The following are the user requirements for the system

6.2.1 The clients register a Problem in a project by entering the following details

6.2.1.1 Project Name

6.2.1.2 Project ID

6.2.1.3 Module

6.2.1.4 what is the Problem

6.2.1.5 priority

6.2.2 Checking the status of the problem

6.2.3 reopening the problem

6.2.4 cancelling the registered problem

6.2.5 The admin performs the following operations

6.2.5.1 Adding a Project

6.2.5.2 Deleting a project

6.2.5.3 Checking the status of the problem

6.2.5.4 Tracking the session

6.2.5.4 Adding clients and engineers

6.2.6 The engineer performs the following operations

6.2.6.1 Checking the active problems

6.2.6.2 Checking the completed problem if it is reopened

6.2.7 the project manager checks the status of the problem and do review of the engineer.

### **6.3 User characteristics**

The Graphical User Interface and online help must be sufficient to educate the users on how to use the system without any problems. End users without any basic knowledge of computer systems will be able to adapt to the form based user interfaces offered by the system with sufficient documentation and end user training.

### **6.4 Constraints**

- 6.4.1 Assigning the problem to the engineer according to the priority.
- 6.4.2 Reopening the completed problem when the engineer is assigned to a new problem.

### **6.5 Assumptions and dependencies**

- 6.5.1 Clients and the engineers are communicated about the problem only through mail.
- 6.5.2 The client who selects the priority is well of the problem's nature and its severity.
- 6.5.3 The projects manager decides the engineer for a problem.
- 6.5.4 The project manager submits report of a project to the admin.

## **7. External Interface Requirements**

### **7.1 User Interfaces**



7.1.1 The system will provide GUI for the Clients/engineers/admin/project managers.

## **7.2 Hardware Interfaces**

N/A

## **7.3 Software Interfaces**

7.3.1 The system will have an interface with the existing project management software for the synchronization of the project details required by the system.

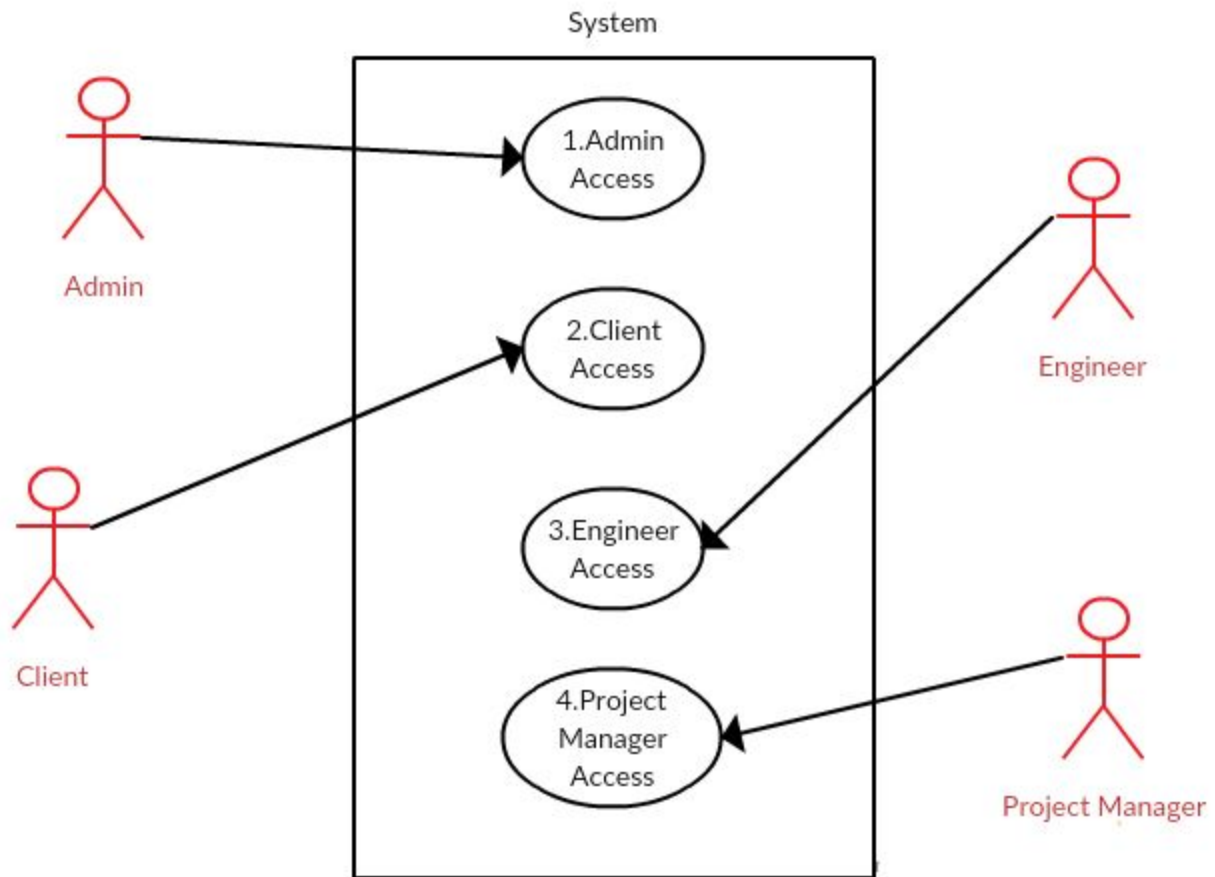
## **7.4 Communications Interfaces**

7.4.1 The system will use web technologies and with the help of http headers will be able to carry out the tasks that it has to do.

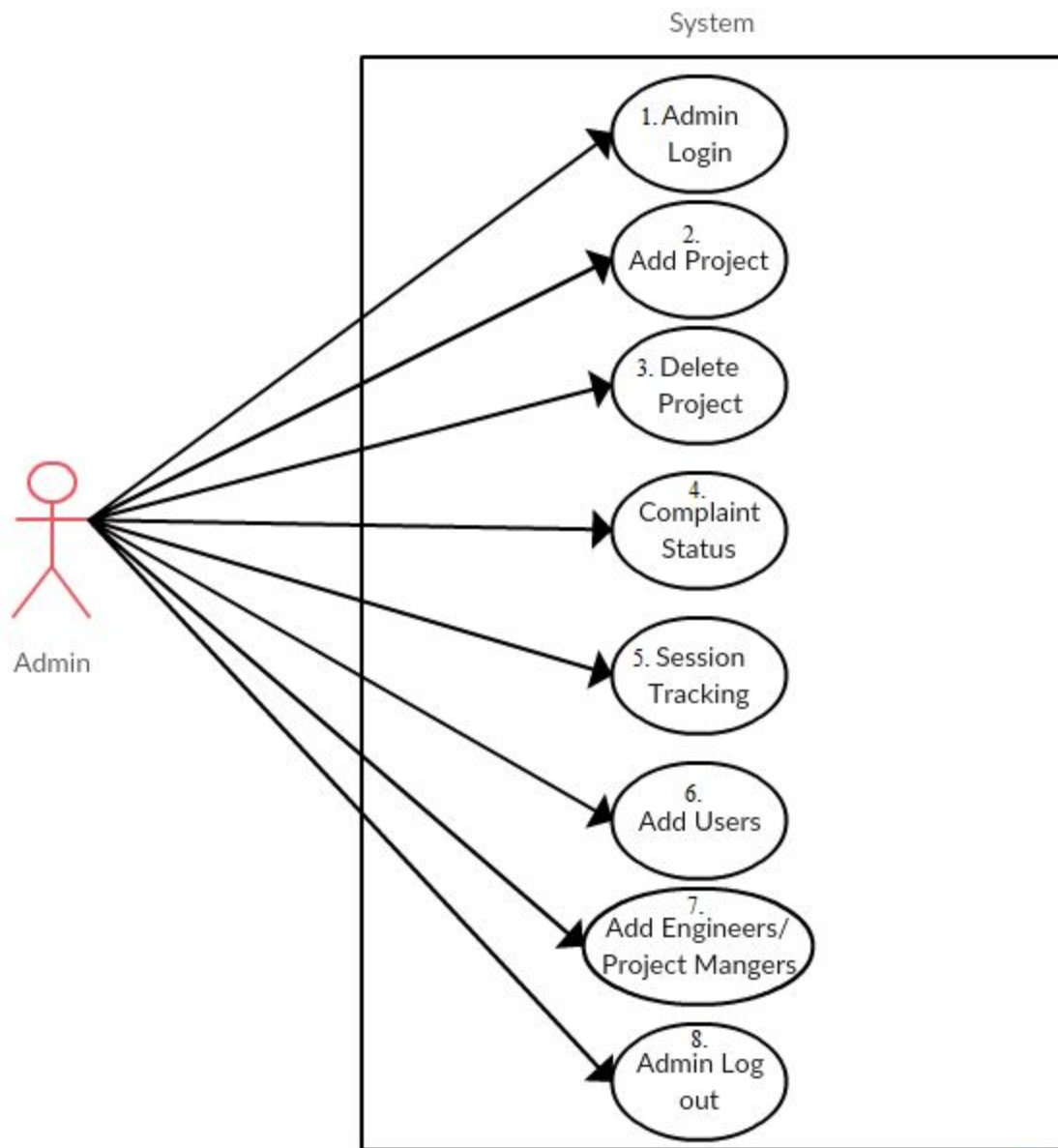
## 8. Functional Requirements

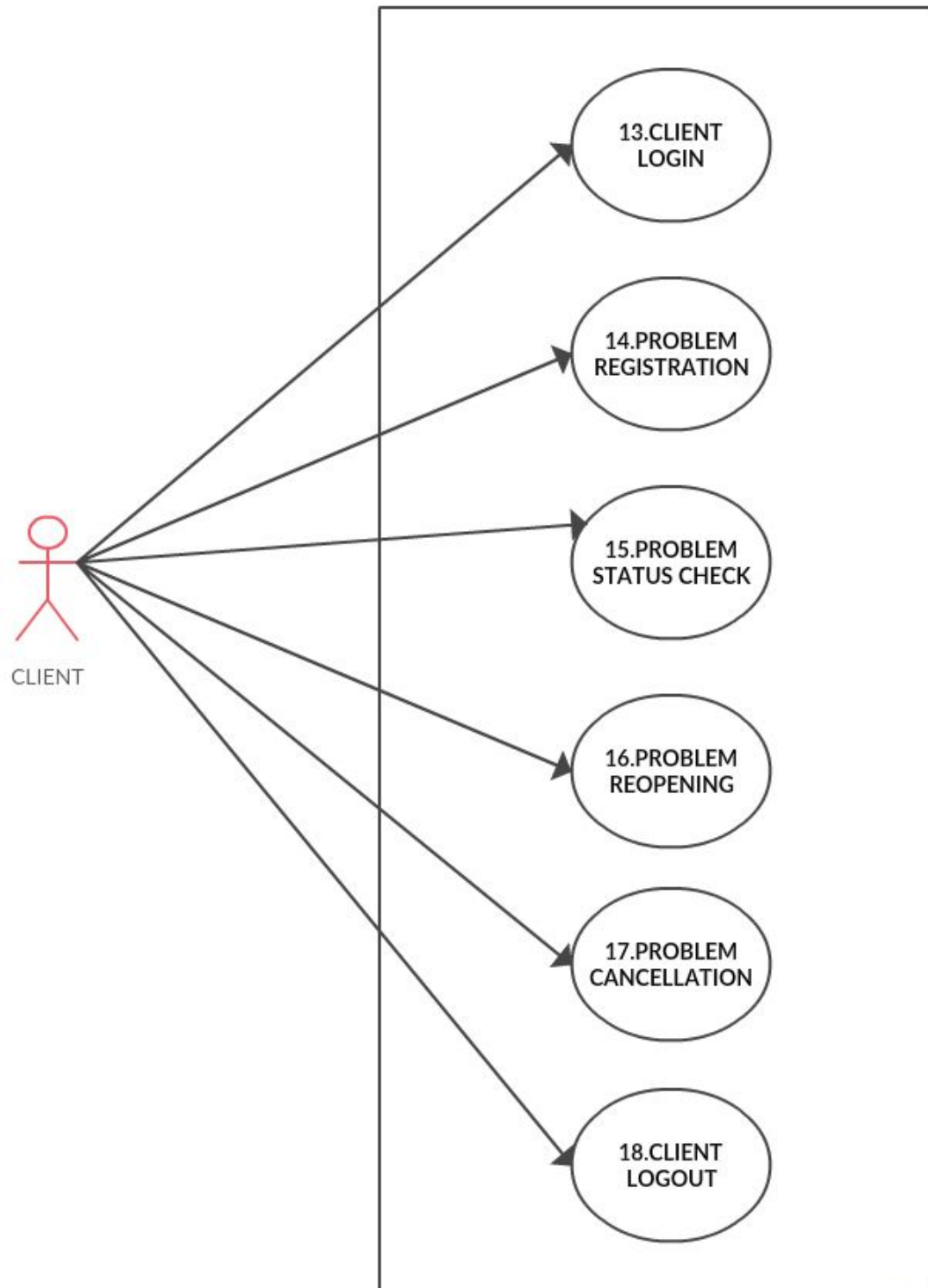
### 8.1 Use Case Diagrams

(Harsh Saxena 13BCE0093, Mohit Awasthi 13BCE0855)

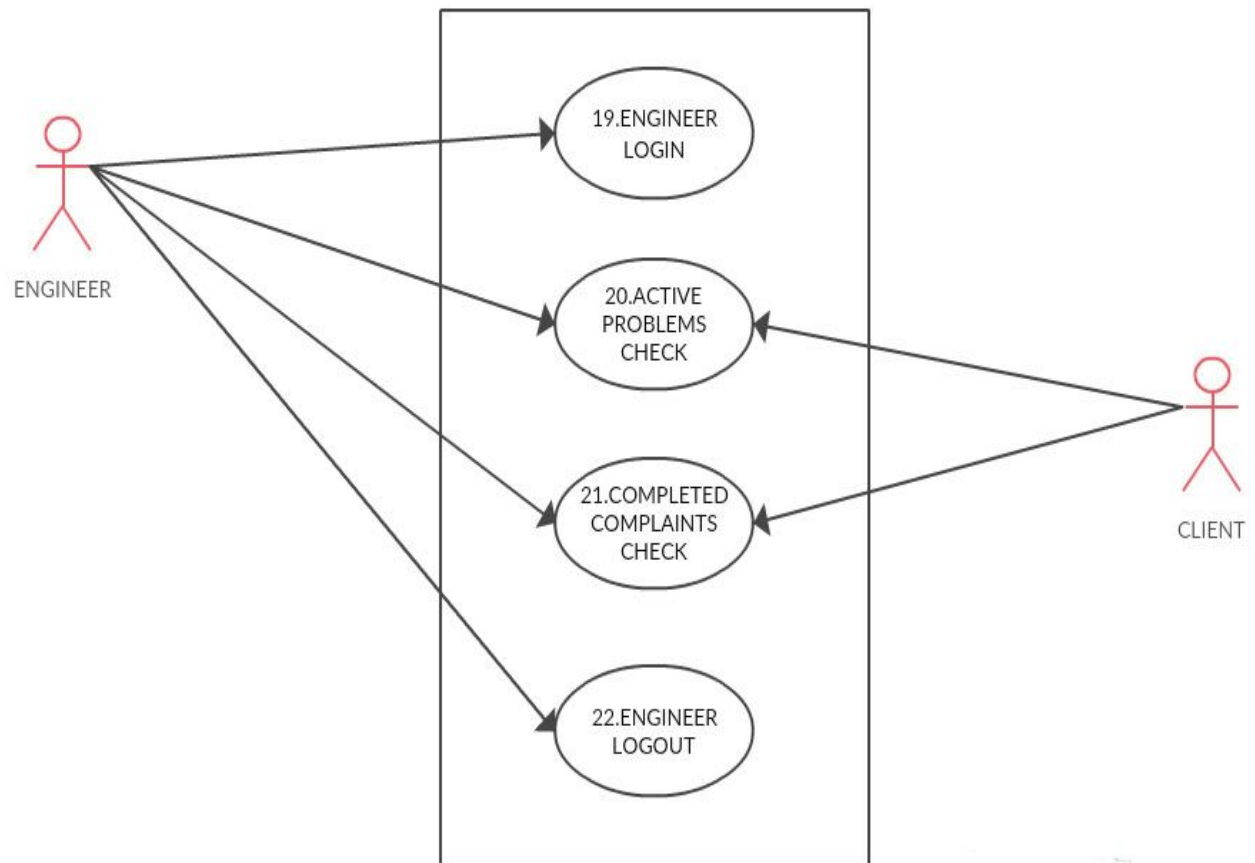


(Karan Pittie 13BCE0360,Varun Manocha 13BCE0141)

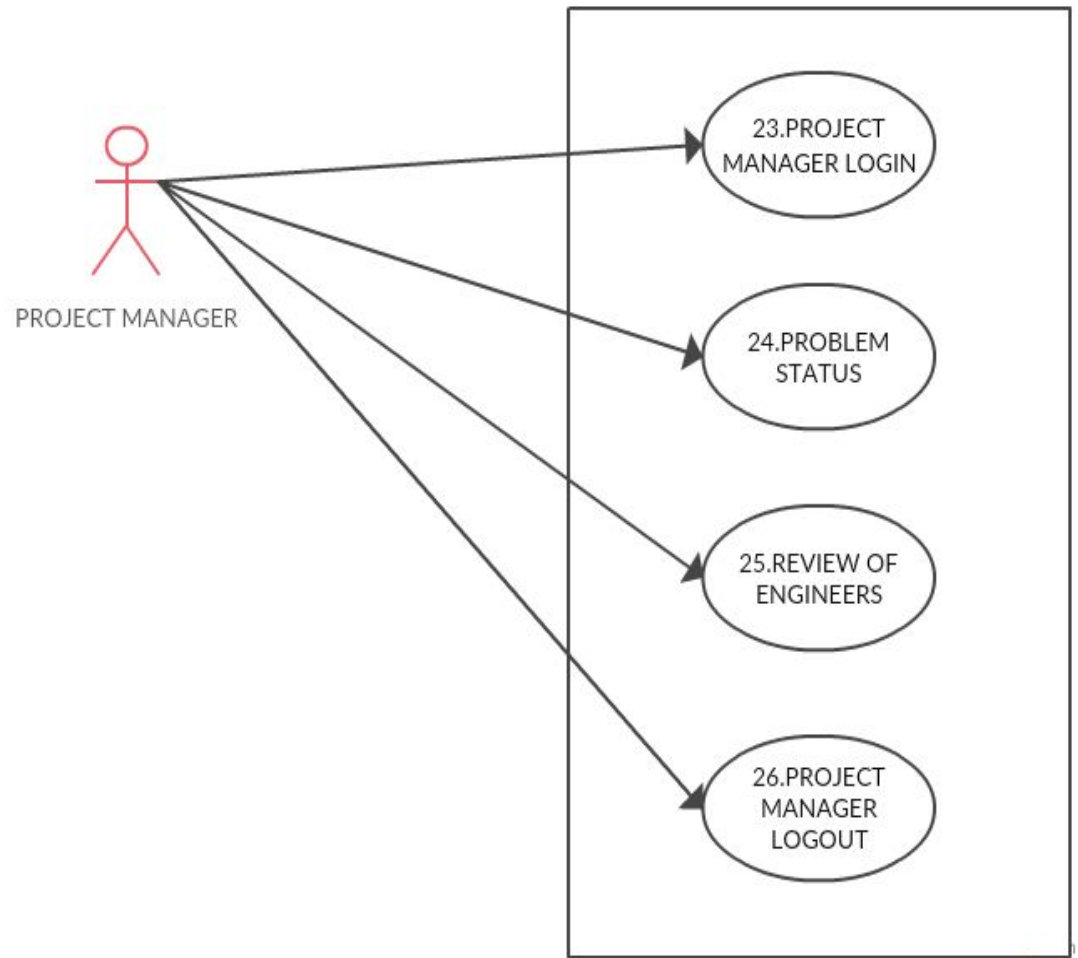




**(INDRANIL OJHA 13BCE0857)**



**(INDRANIL OJHA 13BCE0857)**



**(INDRANIL OJHA 13BCE0857)**

## 8.2 Use Case Scenario Description

(Harsh Saxena 13BCE0093,Mohit Awasthi 13BCE0855)

Use Case ID	1	
Use Case name	Admin Login	
Summary	The action will allow the admin to go to the admin login page.	
Preconditions	The user should visit the tool website.	
Success End Condition	The user successfully reaches the admin login place.	
Failed End Condition	An error message stating no such page is available.	
Primary, Secondary Actors	Admin, Client, Engineer, Project Manager	
Trigger	This use case is triggered when the actor clicks on the "Admin Access" button.	
Description	Step	Action
	1	The admin visits the system website.
	2	The admin clicks the Admin Login button which opens the admin login page.

Use Case ID	2
-------------	---

<b>Use Case name</b>	<b>Client Login</b>	
<b>Summary</b>	<b>The action will allow the client to go to the client login page.</b>	
<b>Preconditions</b>	<b>The user should visit the tool website.</b>	
<b>Success End Condition</b>	<b>The user successfully reaches the client login page.</b>	
<b>Failed End Condition</b>	<b>An error message stating no such page is available.</b>	
<b>Primary, Secondary Actors</b>	<b>Admin, Client, Engineer, Project Manager</b>	
<b>Trigger</b>	<b>This use case is triggered when the actor clicks on the “Client Login” button.</b>	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	<b>1</b>	<b>The client visits the system website.</b>
	<b>2</b>	<b>The user clicks on the “Client Login” button opens the Client Login Page</b>

<b>Use Case ID</b>	<b>3</b>
--------------------	----------



<b>Use Case name</b>	<b>Engineer Login</b>	
<b>Summary</b>	<b>The action will allow the user to go to the Engineer Login page.</b>	
<b>Preconditions</b>	<b>The user should visit the tool website.</b>	
<b>Success End Condition</b>	<b>The user successfully reaches the Engineer Login page.</b>	
<b>Failed End Condition</b>	<b>An error message stating no such page is available.</b>	
<b>Primary, Secondary Actors</b>	<b>Admin, Client, Engineer, Project Manager</b>	
<b>Trigger</b>	<b>This use case is triggered when the actor clicks on the “Engineer Login” button.</b>	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	<b>1</b>	<b>The user visits the system website.</b>
	<b>2</b>	<b>The user clicks on the “Engineer Login” button which opens the Engineer Login Page</b>

<b>Use Case ID</b>	<b>4.</b>
<b>Use Case name</b>	<b>Project Manager Login</b>
<b>Summary</b>	<b>The action will allow the user to go to the project manager login page.</b>

<b>Preconditions</b>	The user should visit the tool website.	
<b>Success End Condition</b>	The user successfully reaches the project manager login place.	
<b>Failed End Condition</b>	An error message stating no such page is available.	
<b>Primary, Secondary Actors</b>	Admin, Client, Engineer, Project Manager	
<b>Trigger</b>	This use case is triggered when the user clicks on the “Project Manager Login” button.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	The user visits the system/website.
	2	The user clicks on the “Project Manager Login” button which opens the Project Manager Login Page

**(Karan Pittie 13BCE0360,Varun Manocha 13BCE0141)**

<b>Use Case ID</b>	5
<b><u>Use Case name</u></b>	Admin Login
<b>Summary</b>	<p>The System will allow the Admin to</p> <ul style="list-style-type: none"> <li>· Log in to the system .</li> </ul>

	<p>Only verified admin users, whose details (user name, password) are stored in the database will be allowed access.</p> <p>The following entries will be retrieved from the database :</p> <ul style="list-style-type: none"> <li>· User name</li> <li>· Password</li> </ul> <p>and matched against the user's entry.</p>	
Preconditions	<ul style="list-style-type: none"> <li>● The Admin needs to be registered in the database.</li> </ul>	
Success End Condition	The admin successfully logs into the system.	
Failed End Condition	<ul style="list-style-type: none"> <li>· Display of appropriate prompt message.</li> </ul>	
Primary, Secondary Actors	<ul style="list-style-type: none"> <li>● Admin – primary actor</li> <li>● N/A – secondary actor</li> </ul>	
Trigger	This use case is triggered on clicking the admin button.	
Description	Step	Action
	1	The admin enters the system.
	2	He clicks on the admin button which will open a prompt wherein the admin name and password fields are displayed.
	3	The admin then has to enter his valid details into the fields and click on the submit button.
	4	If admin fails to authenticate his details then appropriate message will be displayed else he will be logged in.

Extensions	Step	Branching Action
	1	The system retrieves information of the admin from the database and then verifies the details entered.

Use Case ID	6	
<u>Use Case name</u>	Add project	
Summary	<p>The System will allow the admin to:</p> <ul style="list-style-type: none"> <li>• Add projects along with its associated modules.</li> </ul> <p>The admin will provide the following values :-</p> <ul style="list-style-type: none"> <li>• Project ID</li> <li>• Project Name</li> <li>• Project Modules</li> <li>● Project Manager ID</li> <li>● Project Manager Name</li> </ul>	
Preconditions	<ul style="list-style-type: none"> <li>● The admin should be logged into the system.</li> </ul>	
Success End Condition	The project has been added to the database successfully.	
Failed End Condition	<ul style="list-style-type: none"> <li>● If any of the fields are left blank then the project won't be inserted.</li> <li>● Multiple projects having the same project ID.</li> </ul>	
Primary, Secondary Actors	<ul style="list-style-type: none"> <li>• Admin – primary actor</li> <li>• N/A –secondary actors</li> </ul>	
Trigger	This use case is triggered when the admin clicks on the add project button.	
Description	Step	Action

	1	The admin logs into his account by entering his valid details.
	2	The admin clicks on the add project button.
	3	The admin enters the fields that are required to be entered.
	4	Now the admin clicks on the submit button.
	5	A confirmation message will be displayed. Also the added project id and project name will be shown.
	6	Using the back button the admin can move to the previous page.
	7	The admin can also sign out by clicking on the logout button.
Extensions	Step	Branching Action
	1	The system adds the project details into the database.

Use Case ID	7
<u>Use Case name</u>	Delete Project
Summary	The System will allow the admin to delete a project that is already in the system by searching for the project to be deleted
Preconditions	<ul style="list-style-type: none"> <li>● The admin must be logged into the system.</li> <li>● There should be a project existing corresponding to the mentioned project ID.</li> </ul>

Success End Condition	The project has been successfully deleted from the system.	
Failed End Condition	<ul style="list-style-type: none"> <li>If the project to be removed by the admin doesn't exist in the database.</li> </ul>	
Primary, Secondary Actors	<ul style="list-style-type: none"> <li>Admin – primary actor</li> <li>PROJECT MANAGER,ENGINEER, USERS –secondary actors</li> </ul>	
Trigger	This use case is initiated by the Admin to delete/remove a particular project.	
Description	Step	Action
	1	The admin has already logged into his account by giving his registered login-id, password and entering the captcha.
	2	The admin has selected the Delete Project link.
	3	The admin then enters the ID of the project that he wishes to delete from the system.
	4	Now the admin will hit the DELETE button.
	5	A confirmation message will be displayed. Also the deleted project id and project name will be shown.
	6	Using the BACK button the admin can move to the previous page.
	7	The admin can also sign out by clicking on the LOGOUT link.
Extensions	Step	Branching Action
	1	The System will update the database.The following entries are deleted:

		<ul style="list-style-type: none"> <li>• Project ID</li> <li>• Project name</li> <li>• Project modules</li> </ul>
--	--	---

Use Case ID	8
<u>Use Case name</u>	Complaint Status
Summary	<p>The System will allow the admin to</p> <ul style="list-style-type: none"> <li>• Check the status of the problems according to the ID of the problem.</li> </ul> <p>The system will retrieve from the database the following information based on the complaint id provided by the admin:-</p> <ul style="list-style-type: none"> <li>● Complaint ID</li> <li>● Complaint Details</li> <li>● Project ID</li> <li>● Project Name</li> <li>● Project Module</li> <li>● Priority</li> <li>● Status</li> </ul>
Preconditions	<ul style="list-style-type: none"> <li>● The admin must be logged into the system.</li> </ul>
Success End Condition	The correct status of the complaint can be viewed.
Failed End Condition	<ul style="list-style-type: none"> <li>● If the complaint doesn't exist.</li> </ul>
Primary, Secondary Actors	<ul style="list-style-type: none"> <li>• Admin – primary actor</li> <li>• N/A –secondary actors</li> </ul>

<b>Trigger</b>	This use case is initiated by the Admin to view a particular status of a problem.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	The admin has already logged into his account by giving his registered login-id, password and entering the captcha.
	2	The admin has clicked the view status button.
	3	The admin then enters the ID of the complaint.
	4	Now the admin will hit the search button.
	5	The details associated with the problem whose ID has been entered in the search box.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	1	The details of the problem are retrieved from the database.

**(Gagandeep Singh 13BCE0822)**

<b>Use Case ID</b>	9
<b>Use Case name</b>	Session Tracking
<b>Summary</b>	<p>The System will allow the Admin to track the working log of individual engineers.</p> <p>The following entries will be retrieved from the database :</p>



	<ul style="list-style-type: none"> <li>Engineer_name</li> <li>Problem_name</li> <li>Timestamp_logged_in</li> <li>Timestamp_problem_submit</li> </ul>	
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>The Admin needs to be registered in the database.</li> <li>The problem needs to be rectified by the assigned engineer.</li> </ul>	
<b>Success End Condition</b>	The admin successfully sees the session of the engineer.	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>Display of appropriate prompt message.</li> </ul>	
<b>Primary, Secondary Actors</b>	<ul style="list-style-type: none"> <li>Admin – primary actor</li> <li>Engineer – secondary actor</li> </ul>	
<b>Trigger</b>	This use case is triggered on clicking the session tracking button.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	The admin enters the system.
	2	He clicks on the admin button which will open a prompt wherein the admin name and password fields are displayed.
	3	The admin then has to enter his valid details into the fields and click on the submit button.
	4	On the next screen the admin has to choose the session tracking option.

	5	He then enters the engineer_id of the engineer whose session he wishes to track.
	6	He then sees the session details of the engineer.
	7	The admin may then choose to logout from the system by clicking on the logout button.
Extensions	Step	Branching Action
	1	The system retrieves information of the engineer from the database and the problem and then displays the appropriate details on the page.

(Karan Pittie 13BCE0360,Varun Manocha 13BCE0141)

Use Case ID	10
<u>Use Case name</u>	Add clients/users
Summary	The System will allow the admin to · Add new users/clients that can use the system.
Preconditions	<ul style="list-style-type: none"> <li>The admin must be logged into the system.</li> </ul>
Success End Condition	User/client details have successfully been inserted into the database.
Failed End Condition	<ul style="list-style-type: none"> <li>If any of the fields are left blank.</li> </ul>

<b>Primary, Secondary Actors</b>	<ul style="list-style-type: none"> <li>• Admin – Primary Actor</li> <li>• N/A – Secondary Actors</li> </ul>	
<b>Trigger</b>	This use case is initiated by the Admin to add a user/client into the system.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	The admin has already logged into his account by giving his registered login-id, password and entering the captcha.
	2	The admin has clicked the add user/client button.
	3	The admin then enters the ID along with password for the user.
	4	Now the admin will hit the submit button.
	5	The client/user details have successfully been entered into the database prompt is viewed.
	6	Using the back button the admin can move to the previous page.
	7	The admin can also sign out by clicking on the logout link.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	1	<p>The System will insert the following contents into the database :-</p> <ul style="list-style-type: none"> <li>• User ID</li> <li>• Password</li> </ul>

Use Case ID	11	
<u>Use Case name</u>	Add Engineers / Project Managers.	
Summary	<p>The System will allow the admin to</p> <ul style="list-style-type: none"> <li>Add the details of the engineers/project managers into the database.</li> </ul>	
Preconditions	<ul style="list-style-type: none"> <li>The admin must be logged into the system.</li> </ul>	
Success End Condition	The details of the engineer/project manager have successfully been inserted.	
Failed End Condition	<ul style="list-style-type: none"> <li>If any of the fields are left blank.</li> </ul>	
Primary, Secondary Actors	<ul style="list-style-type: none"> <li>Admin – Primary Actor</li> <li>N/A – Secondary Actors</li> </ul>	
Trigger	This use case is initiated by the Admin add an engineer/project manager to the team.	
Description	Step	Action
	1	The admin has already logged into his account by giving his registered login-id, password and entering the captcha.
	2	The admin has clicked the add engineer button.
	3	The admin then enters the details of the engineer.
	4	Now the admin will hit the submit button.
	5	A prompt saying that the engineer has been added is then displayed.

	6	Using the back button the admin can move to the previous page.
	7	The admin can also sign out by clicking on the logout link.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	1	<p>The System will insert the following contents into the database as in :-</p> <ul style="list-style-type: none"> <li>• Engineer ID</li> <li>• Engineer Name</li> <li>• Field of Work</li> <li>▪ Status</li> </ul>

<b>Use Case ID</b>	12
<b><u>Use Case name</u></b>	Admin logout.
<b>Summary</b>	This option will make the Admin sign-out of his account.
<b>Preconditions</b>	The Admin should be a registered user of the system i.e. should have an account.
<b>Success End Condition</b>	The Admin has successfully logged out.
<b>Failed End Condition</b>	N/A
<b>Primary, Secondary Actors</b>	Admin-primary actor  N/A-secondary actor

<b>Trigger</b>	<b>This use case is initiated based on the request of the Admin to log out from his account.</b>	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	<b>1</b>	<b>The Admin has already logged into his account by giving his registered login-id, password and entering the captcha.</b>
	<b>2</b>	<b>The Admin clicks on the logout button to logout.</b>
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	<b>1</b>	<b>The system logs the Admin out.</b>

**(Thirumurugan S S 13BCE0293, Ambikan S 13BCE0767)**

<b>Use Case ID</b>	<b>13</b>
<b>Use Case name</b>	<b>Client Login</b>
<b>Summary</b>	<b>The System will allow the client to</b> <ul style="list-style-type: none"> <li><b>· Login to the system.</b></li> <li><b>· Only verified users are allowed to access it.</b></li> </ul>
<b>Preconditions</b>	<b>The client needs to be registered</b>

<b>Success End Condition</b>	<b>The client successfully logs into the system and directed to his work page.</b>	
<b>Failed End Condition</b>	<b>Error message indicating unsuccessful login due to invalid credentials (username &amp; password).</b>	
<b>Primary, Secondary Actors</b>	<ul style="list-style-type: none"> <li>· <b>Client – Primary actor</b></li> <li>· <b>N/A- Secondary actors</b></li> </ul>	
<b>Trigger</b>	<b>This use case is triggered by clicking the client button</b>	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	<b>1</b>	<b>Client enters the system</b>
	<b>2</b>	<b>By clicking on the client button he is directed to the page where he has to enter the username and password.</b>
	<b>3</b>	<b>If the entered credentials are correct he is logged in and can work.</b>
<b>Extensions</b>	<b>Step</b>	<b>Branching action</b>
	<b>1</b>	<b>The following entries will be retrieved from the database as the user enters.</b> <ul style="list-style-type: none"> <li>· <b>User ID</b></li> <li>· <b>Password</b></li> </ul>

<b>Use Case ID</b>	<b>14</b>
<b>Use Case name</b>	<b>Problem/Change registration</b>

<b>Summary</b>	<p>The System will allow the client to</p> <ul style="list-style-type: none"> <li>· Register a change.</li> <li>· Specify its priority.</li> </ul>	
<b>Preconditions</b>	The client needs to be logged in.	
<b>Success End Condition</b>	The change request has been registered successfully.	
<b>Failed End Condition</b>	If any of the fields are left blank (unattended) then the change request won't be registered.	
<b>Primary, Secondary Actors</b>	<ul style="list-style-type: none"> <li>· Client – primary actor</li> <li>· N/A-secondary actors</li> </ul>	
<b>Trigger</b>	This use case is triggered by clicking the change request button	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	Client logs into the system
	2	By clicking on the login button he/she enters the page where he/she has to enter the details for the problem registration.
	3	If the entered credentials are valid then a success message is displayed on clicking the submit button.
<b>Extensions</b>	<b>Step</b>	<b>Branching action</b>
	1	The system stores the data entered by the client in the database.



<b>Use Case ID</b>	<b>15</b>	
<b>Use Case name</b>	<b>Problem/change status check</b>	
<b>Summary</b>	<b>The System will allow the client to</b> <ul style="list-style-type: none"> <li>· <b>Verify the status of the change he/she registered.</b></li> </ul>	
<b>Preconditions</b>	<b>The client should have registered a change request.</b>	
<b>Success End Condition</b>	<b>If the client has successfully registered a change and the details of the registered change are displayed.</b>	
<b>Failed End Condition</b>	<b>If the client has not registered a change.</b>	
<b>Primary, Secondary Actors</b>	<ul style="list-style-type: none"> <li>· <b>Client – primary actor</b></li> <li>· <b>N/A-secondary actors</b></li> </ul>	
<b>Trigger</b>	<b>This use case is triggered by clicking on one of the registered change request.</b>	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	<b>1</b>	<b>Client logs into the system.</b>
	<b>2</b>	<b>After the client logs in he/she directed into the page which displays the list of requests registered by the client.</b>
	<b>3</b>	<b>By clicking on the requests the client can view the status on the changes request.</b>
<b>Extensions</b>	<b>Step</b>	<b>Branching action</b>

	1	The details are retrieved from the database and displayed.
--	---	--

Use Case ID	16	
Use Case name	Problem/Change reopening	
Summary	<p>The System will allow the client to</p> <ul style="list-style-type: none"> <li>· Reopen a change request registered by him/her.</li> </ul>	
Preconditions	The client should have registered a change request.	
Success End Condition	The client successfully reopens the request he/she has registered.	
Failed End Condition	If the client has not registered a request or invalid id is given.	
Primary, Secondary Actors	<ul style="list-style-type: none"> <li>· Client – primary actor</li> <li>· N/A-secondary actors</li> </ul>	
Trigger	This use case is triggered by clicking the request to be reopened.	
Description	Step	Action
	1	Client logs into the system.
	2	After the client logs in he/she directed into the page which displays the list of requests registered by the client.
	3	By clicking on the request the client can reopen it and view the details.

Extensions	Step	Branching action
	1	The system retrieves the details of the registered request from the database and displays it.

(INDRANIL OJHA 13BCE0857)

Use Case ID	17
<u>Use Case name</u>	Problem Cancellation
Summary	<p>The System will allow the CLIENT to</p> <ul style="list-style-type: none"> <li>Cancel the request irrespective of its priority (high/medium/low)</li> </ul>
Preconditions	<ul style="list-style-type: none"> <li>The CLIENT needs to be registered as a valid user.</li> <li>The CLIENT should have filed at least one complaint.</li> </ul>
Success End Condition	CLIENT has successfully deleted a request/problem and that problem is no longer in his list of available complaints.

<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>● If the problem to be removed by the CLIENT doesn't exist in his list.</li> <li>● If the problem to be removed by the CLIENT has already been dealt with by the required ENGINEER.</li> <li>● If the CLIENT enters the wrong id that is not present in his list.</li> <li>● If the CLIENT gives NULL values for problem id</li> <li>● If the CLIENT gives text/string characters instead of problem-id.</li> </ul>	
<b>Primary, Secondary Actors</b>	<ul style="list-style-type: none"> <li>● CLIENT – primary actor</li> <li>● N/A –secondary actors</li> </ul>	
<b>Trigger</b>	This use case is initiated based on the request from the CLIENT to delete/remove a particular problem/request.	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	1	The CLIENT has already logged into his account by giving his registered login-id, password and entering the captcha.
	2	The CLIENT has selected the DELETE PROBLEM link.
	3	Now out of the available problem names and problem ids, the CLIENT will select the desired problem id by entering it into the text box.
	4	Now the CLIENT will hit the DELETE button.
	5	A confirmation message will be displayed. Also the deleted problem id and problem name will be shown.

	6	Similarly the client can delete the other problems he wishes to.
	7	Using the BACK button the CLIENT can move to the previous page.
	8	The Client can also sign out by clicking on the LOGOUT link.
Extensions	Step	Branching Action
	1	<p>The System will update the following fields in the database :-</p> <ul style="list-style-type: none"> <li>• PROBLEM ID</li> <li>• PROBLEM NAME</li> <li>• PROBLEM CONTENT</li> <li>• The assigned ENGINEER to that problem</li> <li>• The assigned PROJECT MANAGER under whom the ENGINEER was handling that problem.</li> </ul>

(INDRANIL OJHA 13BCE0857)

Use Case ID	18
<u>Use Case name</u>	CLIENT LOGOUT
Summary	This option will make the CLIENT log-out of his account

<b>Preconditions</b>	<b>The CLIENT should be a registered user of the system i.e. should have an account.</b>	
<b>Success End Condition</b>	<b>The CLIENT has successfully logged out and move to CLIENT LOGIN page.</b>	
<b>Failed End Condition</b>	<b>N/A</b>	
<b>Primary, Secondary Actors</b>	<b>CLIENT-primary actor</b> <b>N/A-secondary actor</b>	
<b>Trigger</b>	<b>This use case is initiated based on the request of the CLIENT to sign-out from his account.</b>	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	<b>1</b>	<b>The CLIENT has already logged into his account by giving his registered login-id, password and entering the captcha.</b>
	<b>2</b>	<b>The CLIENT clicks on the LOGOUT link to sign-out.</b>
	<b>3</b>	<b>The CLIENT clicks on the LOGOUT link after registering the Problem.</b>
	<b>4</b>	<b>The CLIENT clicks on the LOGOUT link after checking the status of the problem.</b>
	<b>5</b>	<b>The CLIENT clicks on the LOGOUT link after problem cancellation.</b>

	6	The CLIENT clicks on the LOGOUT link after reopening the problem.
Extensions	Step	Branching Action
	1	The system logs the CLIENT out.

### Apoorv Sharma (13BCE0589)

Use Case ID	19	
<u>Use Case name</u>	Engineer Login	
Summary	This option will allow an engineer to log into his account.	
Preconditions	The engineer should be a registered user of the system i.e. should have an account added by the Admin.	
Success End Condition	The engineer has successfully logged in .	
Failed End Condition	Error message to indicate unsuccessful login. The user ID and/or password may be invalid.	
Primary, Secondary Actors	Engineer-primary actor	
Trigger	This use case is initiated by the engineer on the login page.	
Description	Step	Action
	1	The engineer enters the system.

	2	He clicks on the engineer login button on the login page which will open a webpage which allows him to enter his engineer ID and password.
	3	If the details entered are correct, he is logged in and can now work on his assigned tasks.
Extensions	Step	Branching Action
	1	The system retrieves information of the engineer from the database and then verifies the details entered.

**Apoorv Sharma (13BCE0589)**

Use Case ID	20
Use Case name	View Active Complaints (Engineer)
Summary	<p>This button will allow the engineer to</p> <ul style="list-style-type: none"> <li>· View the tasks assigned to him that have not yet been completed.</li> </ul> <p>The System will retrieve from the database the following details :-</p> <ul style="list-style-type: none"> <li>● Complaint ID</li> <li>● Complaint Details</li> <li>● Project ID</li> <li>● Project Name</li> <li>● Project Module</li> <li>● Priority</li> <li>● Status</li> </ul>
Preconditions	<ul style="list-style-type: none"> <li>● The engineer must be logged into the system.</li> </ul>



	<ul style="list-style-type: none"> <li>● The task status must be 'unfinished'.</li> </ul>	
Success End Condition	<ul style="list-style-type: none"> <li>● The details of the task can be viewed</li> </ul>	
Failed End Condition	<ul style="list-style-type: none"> <li>● The task has been finished.</li> </ul>	
Primary, Secondary Actors	<ul style="list-style-type: none"> <li>● engineer – primary actor</li> <li>● Client –secondary actors</li> </ul>	
Trigger	This use case is initiated by the engineer to view the details of a problem.	
Description	Step	Action
	1	The engineer has already logged into his account by giving his registered login-id, password.
	2	The admin has clicked the view active complaints button.
	3	A list of all the unfinished tasks assigned to engineer appears.
Extensions	Step	Branching Action
	1	The details of all the unlisted problems assigned to the engineer are retrieved from the database.

(Jahnvi Jaiswal 13BCE0604 and Aarushi Arya 13BCE0678)

Use Case Number	21
-----------------	----

<b>Use case Name</b>	<b>Completed or rectified complaints check</b>	
<b>Summary</b>	<b>This option allows the engineer to check if the complaints have been completed or rectified.</b>	
<b>Preconditions</b>	<b>The engineer must be logged in to check the complaints.</b>	
<b>Success end Condition</b>	<b>The status of the complaints is displayed.</b>	
<b>Failed end Condition</b>	<b>The problem id specified does not correspond to any of the registered complaints.</b>	
<b>Primary, secondary, actors</b>	<b>This use case is based on the request of the project manager or admin to check the status of the complaints.</b>	
<b>Triggers</b>	<b>This event is triggered when the user clicks on the check status of complaints button.</b>	
<b>Description</b>	<b>Steps</b>	<b>Actions</b>
	<b>1</b>	<b>The engineer logs in using the user id and password.</b>
	<b>2</b>	<b>The engineer clicks to check the status of the complaints.</b>
	<b>3</b>	<b>The list of all the complaints is shown and the engineer can view the status of the registered problems.</b>

<b>Extensions</b>	<b>Steps</b>	<b>Branching Action</b>
	<b>1</b>	<b>The complaint status is retrieved from the database.</b>

<b>Use Case ID</b>	<b>22</b>
<b>Use Case Name</b>	<b>Engineer logout</b>
<b>Summary</b>	<b>This option will allow an engineer to log out of his account.</b>
<b>Preconditions</b>	<b>The Engineer should be a registered user of the system i.e. should have an account and should be logged into the system</b>
<b>Success end Condition</b>	<b>The Engineer has successfully logged out and is directed to the home page.</b>
<b>Failed end condition</b>	<b>Error message to indicate unsuccessful logout.</b>
<b>Primary, secondary, actors</b>	<b>Engineer-primary actor.</b>
<b>Trigger</b>	<b>This event is triggered when the engineer clicks on the logout button.</b>

Description	Steps	Actions
	1	The Engineer has already logged into his account by giving his registered login-id, password and entering the captcha.
	2	The engineer clicks on the LOGOUT button to sign-out.
Extensions	Steps	Branching Action
	1.	N/A.

Use Case ID	23
Use Case Name	Project Manager Login.
Summary	This option allows the project manager to log in to his account.
Preconditions	The project manager should be registered user of the system i.e., should have an account added by the Admin.
Success End Condition	The project manager is successfully logged in.
Failed end conditions	Error message to indicating unsuccessful login. The user ID and/or password may be invalid.

<b>Primary Secondary Actors</b>	<b>Project manager-Primary actor</b>	
<b>Triggers</b>	<b>The event is triggered when the project manager clicks on the login button after entering his or her login details.</b>	
<b>Description</b>	<b>Steps</b>	<b>Actions</b>
	<b>1</b>	<b>The project manager enters the system.</b>
	<b>2</b>	<b>He clicks on the engineer login button on the login page which allows him to enter is engineer ID and password.</b>
	<b>3</b>	<b>If the details entered are correct, he is logged in and can now work on his assigned tasks</b>
<b>Extensions</b>	<b>Steps</b>	<b>Branching Actions</b>
	<b>1</b>	<b>The system receives the information of the project manager as an engineer from the database and verifies the details entered.</b>

<b>Use case ID</b>	<b>24</b>
<b>Use Case Name</b>	<b>Problem Status</b>
<b>Summary</b>	<b>This allows the Project manager to view the status of the problems registered by the clients.</b>
<b>Preconditions</b>	<b>The problems should be registered by the respective clients.</b>
<b>Success end Condition</b>	<b>The status of all the problems is listed.</b>

<b>Failed end Condition</b>	<b>The problem id specified does not correspond to any of the registered complaints.</b>	
<b>Primary, secondary, actors</b>	<b>Project manager- primary actor</b>	
<b>Trigger</b>	<b>The event is triggered on clicking the check problem status button.</b>	
<b>Description</b>	<b>Steps</b>	<b>Actions</b>
	<b>1</b>	<b>The project manager logs into the system.</b>
	<b>2</b>	<b>He clicks on the check problem status button.</b>
	<b>3</b>	<b>The status of each problem reported by the client can now be seen.</b>
<b>Extensions</b>	<b>Steps</b>	<b>Branching Action</b>
	<b>1.</b>	<b>The information regarding the status of each problem is retrieved from the database.</b>

**(Karan Pittie 13BCE0360,Varun Manocha 13BCE0141,Gagandeep Singh 13BCE0822)**

<b>Use case ID</b>	<b>25</b>
<b>Use Case Name</b>	<b>Review of engineers</b>
<b>Summary</b>	<b>This allows the Project manager to the view the work allocated to every engineer.</b>

<b>Preconditions</b>	The project manager must be signed into the system.	
<b>Success end Condition</b>	The status of all the engineers is displayed.	
<b>Failed end Condition</b>	N/A	
<b>Primary, secondary, actors</b>	Project manager- primary actor Engineers -secondary actor	
<b>Trigger</b>	The event is triggered on clicking the review of engineers button.	
<b>Description</b>	<b>Steps</b>	<b>Actions</b>
	1	The project manager logs into the system.
	2	He clicks on the review engineers button.
	3	The status of the engineers along with the complaints assigned ,complaints worked on and the project is displayed.
<b>Extensions</b>	<b>Steps</b>	<b>Branching Action</b>
	1.	The information of the engineers is retrieved from the database along with their current status.

(Harsh Saxena 13BCE0093,Mohit Awasthi 13BCE0855)

<b>Use Case ID</b>	26
--------------------	----

<b>Use Case name</b>	<b>Project Manager logout.</b>	
<b>Summary</b>	<b>This option will allow the Project Manager to log out of the account.</b>	
<b>Preconditions</b>	<b>The Project Manager should be a registered user of the system i.e. should have an account and should be logged into the system.</b>	
<b>Success End Condition</b>	<b>The Project Manager has successfully logged out and is directed to the home page.</b>	
<b>Failed End Condition</b>	<b>N/A.</b>	
<b>Primary, Secondary Actors</b>	<b>Project Manager-primary actor.</b>	
<b>Trigger</b>	<b>This use case is initiated based on the request of the Project Manager to log-out from his account.</b>	
<b>Description</b>	<b>Step</b>	<b>Action</b>
	<b>1</b>	<b>The Project Manager has already logged into his account by giving his registered login-id, password and entering the captcha.</b>
	<b>2</b>	<b>The Project Manager clicks on the LOGOUT button to sign-out.</b>
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
		<b>N/A</b>



## 9. Non Functional Requirements - Team 6

This section describes in detail all the non-functional requirements

### 9.1 Usability

9.1.1 The system shall allow the users to access the system from the Internet.

9.1.2 The end users will be able to adapt to the system with a minimum training of 10 hours.

### 9.2 Security

#### 9.2.1 Login requirements -

9.2.1.1 All the users belonging to the various categories will be having a password verification procedure if they would like to access the system.

9.2.1.2 The passwords entered by the users, in order to provide security, will be encrypted first and then stored into the database.

9.2.1.3 In order to prevent the system from attacks via robots we will be implementing captcha functionality.

#### 9.2.2 Password requirements

9.2.2.1 Passwords must have a minimum length of 8 characters

#### 9.2.3 Inactivity timeouts

9.2.3.1 N/A

### 9.3 Performance

#### 9.3.1 Response time

9.3.1.1 The response time will be less than 10 seconds for 95% requests made to the system

### 9.4 Capacity

#### 9.4.1 Storage

9.4.1.1 Hard disk space –

100 Mb – Content

10 Mb – Transaction Logs

## **9.5 Recovery**

### **9.5.1 Recovery time scales**

9.5.1.1 The system will be recovered within four hours from the down time

### **9.5.2 Backup Frequencies**

9.5.2.1 A full backup of CMT data will be done on the last day of every month

## **9.6 Availability**

### **9.6.1 Hours of operation**

9.6.1.1 The system will be available on all days 24\*7

## **9.7 Reliability**

### **9.7.1 Mean Time between Failures**

9.7.1.1 The mean time between failures for the system will be 1000 hours

## **9.8 Maintainability**

### **9.8.1 Mean Time to Recovery**

9.8.1.1 The Mean Time To Recovery (MTTR) shall not exceed one person day.

## **9.9 Portability**

9.9.1 The system will run on Windows 7 and higher versions.

9.9.2 The system will run on UBUNTU 12.04 LTS and higher versions.

9.9.3 The system will run on MAC 10.0 and higher versions.

## **9.10 Privacy**

9.10.1 Reports cannot be accessed by end users other than the project manager

9.10.2 Each engineer can access only his/her own report

9.10.3 Users can only be added by the admin, no user can independently register.

