**PROMAN**



**Developed by**

**Usama Majid 3206-FBAS/BSSE/F16**

**Kamran Nazir 3276-FBAS/BSSE/F16**

Supervisedb𝘺

**Dr. Qamar Abbas**

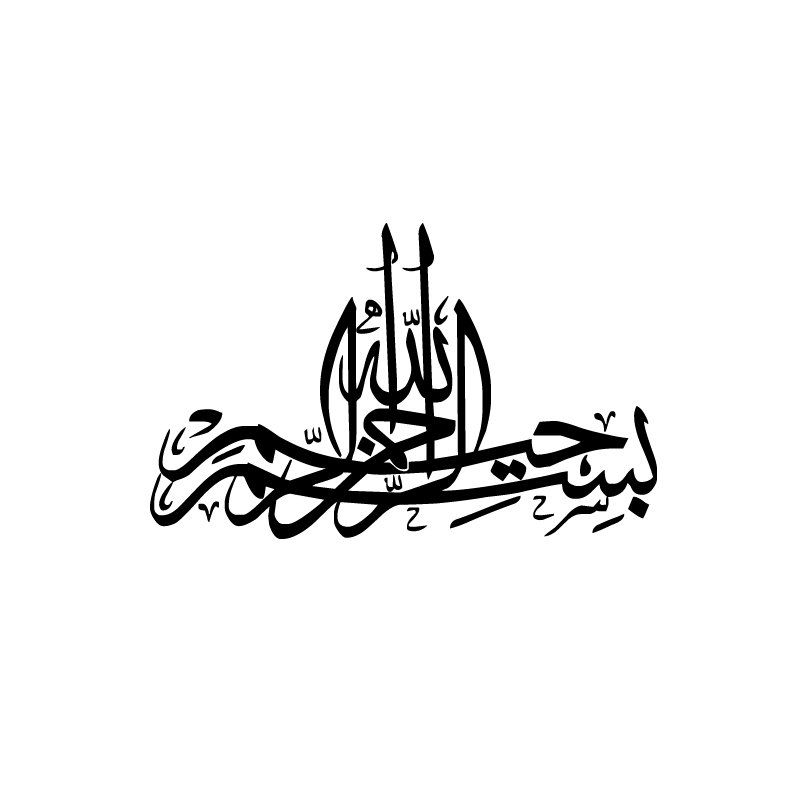
DEPARTMENT OF COMPUTER SCIENCE AND SOFTWARE

ENGINEERING

FACULTY OF BASIC AND APPLIED SCIENCES

INTERNATIONAL ISLAMIC UNIVERSITY ISLAMABAD

(2020)



**Final Approval**

**Date:** \_\_\_\_\_\_\_\_\_\_\_\_

It is certified that we have read the Project report title "ProMan” Submitted by Usama Majid 3206-FBAS/BSSE/F16 and Kamran Nazir 3276-FBAS/BSSE/F16 under the Supervision of our respected Teacher Dr. Qamar Abbas Assistant Professor DCS&SE, FBAS, IIUI. It is our supposition that this project is of enough standard to justify its acceptance by Department of computer science and software engineering, International Islamic University Islamabad for the bachelor’s degree of Software Engineering.

**Dedication**

We dedicate this humble effort to our beloved parents and respectable teachers for their endless support, affection, trust, prayers and encouragement. With completion of our final year project after dedicating this accomplishment and effort to our lovely parents who have constantly supported and provided all the possible facilities that we could complete our studies and the project.

**Declaration**

We hereby declare that the development of this project and its report is thoroughly based on our personal efforts and learning accomplished under the absolute support and guidance of the assigned supervisor **Dr. Qamar Abbas**. Not a single part of this work appeared in this report has been submitted for any other degree or any other university or university or any other educational institute. We further declare that this project, all code, and associated documents and reports are submitted as partial requirements for the degree of Bachelor of Software Engineering.

**Kamran Nazir**

3276-FBAS/BSSE/F16

**Usama Majid**

3206-FBAS/BSSE/F16

**Acknowledgement**

All praise to **Allah Almighty** for his blessings in completing this project and blessings upon our last Prophet Muhammad (PBUH).

We would like to pay our sincere gratitude to our supervisor Dr. Qamar Abbas for his meaningful and continuous support and motivation in our project. We are thankful of his brilliant supervision throughout the completion of the project.

**Kamran Nazir**

3276-FBAS/BSSE/F16

**Usama Majid**

3206-FBAS/BSSE/F16

Project in Brief

|  |  |
| --- | --- |
| **Project Title:** | PROMAN |
| **Undertaken By**: | Usama Majid (3206-FBAS/BSSE/F16)  Kamran Nazir (3276-FBAS/BSSE/F16) |
| **Supervised By:** | Dr. Qamar Abbas  Assistant Professor  Department of Computer Science and Software Engineering International Islamic University Islamabad |
| **Date Started:**  **Date Completed:** | December 2019  July 2020 |
| **Tools, Technologies and language Used:** | React.js, Node.js, Express.js, Mongo dB  Visual Studio Code |
| **System Used:** | Core i5, RAM 6 GB, Core m3 RAM 8GB |
| **Operating System:** | Microsoft Windows 10, Ubuntu 20.04, Chrome |

**Table of Contents**

[1. Introduction: 12](#__RefHeading___Toc4199_2263466818)

[1.1 Overview: 12](#__RefHeading___Toc4201_2263466818)

[1.2 PROMAN: 12](#__RefHeading___Toc4203_2263466818)

[1.3 Problem statement: 12](#__RefHeading___Toc4205_2263466818)

[1.4 Solution of These Problems: 13](#__RefHeading___Toc4207_2263466818)

[We are proposing an environment where people can manage their projects remotely. Projects can be managed individually as well as in team oriented environment. People can manage software projects and can estimate cost as well. The beauty of our proposed environment is that even an ordinary person can use it as a task manager for him and can set due date and time so he can work on time and can check the overall status of the project and the remaining time as well. 13](#__RefHeading___Toc4209_2263466818)

[1.5 Control Flow Diagram (Leader): 14](#__RefHeading___Toc4211_2263466818)

[1.6 Control Flow Diagram (User): 15](#__RefHeading___Toc4213_2263466818)

[2.1 USE CASE DIAGRAM 16](#__RefHeading___Toc4215_2263466818)

[2.2 Use Case Description in Detail Expended Format: 19](#__RefHeading___Toc4217_2263466818)

[2.2.1 Use Case Register: 19](#__RefHeading___Toc4219_2263466818)

[2.2.2 Use Case Login 20](#__RefHeading___Toc4221_2263466818)

[2.2.3 Use Case Add Project: 21](#__RefHeading___Toc4223_2263466818)

[2.2.4 Use Case Add Team: 22](#__RefHeading___Toc4225_2263466818)

[2.2.5 Use Case Check Team Details: 23](#__RefHeading___Toc4227_2263466818)

[2.2.6 Use Case Add Member to Team: 24](#__RefHeading___Toc4229_2263466818)

[2.2.7 Use Case Open Project 25](#__RefHeading___Toc4231_2263466818)

[2.2.8 Use Case Add Member To Project: 26](#__RefHeading___Toc4233_2263466818)

[2.2.9 Use Case Open Leader Dashboard: 27](#__RefHeading___Toc4235_2263466818)

[2.2.10 Use Case Add Task List: 28](#__RefHeading___Toc4237_2263466818)

[2.2.11 Use Case Add Task: 29](#__RefHeading___Toc4239_2263466818)

[2.2.12 Use Case Add Sub Task: 30](#__RefHeading___Toc4241_2263466818)

[2.2.13 Use Case Add Comment: 31](#__RefHeading___Toc4243_2263466818)

[2.2.14 Use Case Add Attachment: 32](#__RefHeading___Toc4245_2263466818)

[2.2.15 Use Case Review Task: 33](#__RefHeading___Toc4247_2263466818)

[2.2.16 Use Case Update Task Status: 34](#__RefHeading___Toc4249_2263466818)

[2.2.17 Use Case Subscribe: 35](#__RefHeading___Toc4251_2263466818)

[2.2.18 Use Case Unsubscribe: 36](#__RefHeading___Toc4253_2263466818)

[2.2.19 Use Case Profile: 37](#__RefHeading___Toc4255_2263466818)

[2.2.20 Use Case Check Project Details: 38](#__RefHeading___Toc4257_2263466818)

[2.2.21 Use Case Check Overview of Project: 39](#__RefHeading___Toc4259_2263466818)

[2.2.22 Use Case Check all Member details of project: 40](#__RefHeading___Toc4261_2263466818)

[2.2.23 Use Case Check Task by task details: 41](#__RefHeading___Toc4263_2263466818)

[2.2.24 Use Case Task by member details: 42](#__RefHeading___Toc4265_2263466818)

[2.3 SEQUENCE DIAGRAMS: 43](#__RefHeading___Toc4267_2263466818)

[2.3.1 Register: 43](#__RefHeading___Toc4269_2263466818)

[2.3.2 Login: 43](#__RefHeading___Toc4271_2263466818)

[2.3.3 Add Project: 44](#__RefHeading___Toc4273_2263466818)

[2.3.4 Add Team: 44](#__RefHeading___Toc4275_2263466818)

[2.3.5 Check Team Details: 45](#__RefHeading___Toc4277_2263466818)

[2.3.6 Add Member to Team: 45](#__RefHeading___Toc4279_2263466818)

[2.3.7 Open Project: 46](#__RefHeading___Toc4281_2263466818)

[2.3.8 Add Member to Project: 46](#__RefHeading___Toc4283_2263466818)

[2.3.9 Open Leader Dashboard: 47](#__RefHeading___Toc4285_2263466818)

[2.3.10 Add Task List 47](#__RefHeading___Toc4287_2263466818)

[2.3.11 Add Task: 48](#__RefHeading___Toc4289_2263466818)

[2.3.12 Add Sub Task: 48](#__RefHeading___Toc4291_2263466818)

[2.3.13 Add Comment: 49](#__RefHeading___Toc4293_2263466818)

[2.3.14 Add Attachment: 49](#__RefHeading___Toc4295_2263466818)

[2.3.15 Review Task: 50](#__RefHeading___Toc4297_2263466818)

[2.3.16 Update Task Status: 50](#__RefHeading___Toc4299_2263466818)

[2.3.17 Subscribe: 51](#__RefHeading___Toc4301_2263466818)

[2.3.18 Unsubscribe: 51](#__RefHeading___Toc4303_2263466818)

[2.3.19 Profile: 52](#__RefHeading___Toc4305_2263466818)

[2.3.20 Check Project Details: 52](#__RefHeading___Toc4307_2263466818)

[2.3.21 Check Overview of Project: 53](#__RefHeading___Toc4309_2263466818)

[2.3.22 Check All Member Details of Project: 53](#__RefHeading___Toc4311_2263466818)

[2.3.23 Check Task By Task Details: 54](#__RefHeading___Toc4313_2263466818)

[2.3.24 Check Task by Member Details: 54](#__RefHeading___Toc4315_2263466818)

[2.4 Domain Model 55](#__RefHeading___Toc4317_2263466818)

[3.1 System Sequence Diagram: (Leader) 56](#__RefHeading___Toc4319_2263466818)

[3.2 System Sequence Diagram: (User) 57](#__RefHeading___Toc4321_2263466818)

[3.4 Alternate to ERD: 59](#__RefHeading___Toc4323_2263466818)

[3.5 Class Diagram 59](#__RefHeading___Toc4325_2263466818)

[4. Implementation: 60](#__RefHeading___Toc4327_2263466818)

[4.1 Functional Features of the Model: 60](#__RefHeading___Toc4329_2263466818)

[5.1 Black Box Testing: 61](#__RefHeading___Toc4331_2263466818)

[5.2 White Box Testing: 61](#__RefHeading___Toc4333_2263466818)

[5.3 Test Cases: 61](#__RefHeading___Toc5507_2553297276)

[5.3.1 Register: 61](#__RefHeading___Toc4335_2263466818)

[5.3.2 Login: 62](#__RefHeading___Toc4337_2263466818)

[5.3.3 Add Project: 62](#__RefHeading___Toc4339_2263466818)

[5.3.4 Add Team: 63](#__RefHeading___Toc4341_2263466818)

[5.3.5 Add Members to Team: 63](#__RefHeading___Toc4343_2263466818)

[5.3.6 Add Task List: 64](#__RefHeading___Toc4345_2263466818)

[5.3.7 Add Task: 65](#__RefHeading___Toc4347_2263466818)

[5.3.8 Add Sub Task: 65](#__RefHeading___Toc4349_2263466818)

[5.3.9 Add Members to Project: 66](#__RefHeading___Toc4351_2263466818)

[5.3.10 Add Comment: 66](#__RefHeading___Toc4353_2263466818)

[5.3.11 Add Attachment: 67](#__RefHeading___Toc4355_2263466818)

[5.3.12 Subscribe Task: 67](#__RefHeading___Toc4357_2263466818)

[5.3.13 Unsubscribe task: 68](#__RefHeading___Toc4359_2263466818)

[5.3.14 See Profile: 68](#__RefHeading___Toc4361_2263466818)

[5.3.15 Review Task: 69](#__RefHeading___Toc4363_2263466818)

[5.4 Result: 70](#__RefHeading___Toc4365_2263466818)

[Conclusion 71](#__RefHeading___Toc4367_2263466818)

[6. Conclusion: 71](#__RefHeading___Toc4369_2263466818)

[Appendix-A 72](#__RefHeading___Toc4371_2263466818)

[User Manual 72](#__RefHeading___Toc4373_2263466818)

[A. User manual: 72](#__RefHeading___Toc4375_2263466818)

[A.1 Public View: 72](#__RefHeading___Toc4377_2263466818)

[A.2 Register: 72](#__RefHeading___Toc4379_2263466818)

[A.3 Login: 73](#__RefHeading___Toc4381_2263466818)

[A.4 Add new Project: 74](#__RefHeading___Toc4383_2263466818)

[A.5 Add new Project: 75](#__RefHeading___Toc4385_2263466818)

[A.6 Project Overview: 76](#__RefHeading___Toc4387_2263466818)

[A.7 Task Overview: 76](#__RefHeading___Toc4389_2263466818)

[A.8 Project overview in table format: 77](#__RefHeading___Toc4391_2263466818)

[78](#__RefHeading___Toc4393_2263466818)

[A.10 Leader Dashboard: 78](#__RefHeading___Toc4395_2263466818)

# 1. Introduction:

“PROMAN" is basically a project manager which is designed for people to manage their projects online with many and many of options. People can use our platform to manage their project easily and more efficiently than any other such existing system. We are giving people multiple options like they can use our online system for individual as a normal project manager and as well as for a team to manage bigger software projects and can calculate software cost as well.

## **1.1 Overview:**

It is an online platform that provides people an opportunity to manage their business and software projects very easily and efficiently with was a headache for them before.

## **1.2 PROMAN:**

“PROMAN” is an online platform for people to manage their projects easily and efficiently. People can manage their individual projects as well as team oriented projects. People can manage multiple project at the same time and can be leader in one project and can be ordinary member in other project at the same time. People can calculate cost of software project as well and can chat and discuss issues related to project with other project members. Leader can assign task to members and members can complete and can push it to the leader for his approval before due date. Every member have an efficiency through which one can easily check his performance and leader have many more options to check projects and its details.

## **1.3 Problem statement:**

* Managing Your Project is a difficult task.
* It is difficult to keep track what the team members are doing.
* Assigning tasks to different members is sometimes headache especially if members are not in one place (working remotely).
* It is difficult to see the overall progress of the project.
* It is difficult for members to discuss issues with each other while working remotely.

## **1.4 Solution of These Problems:**

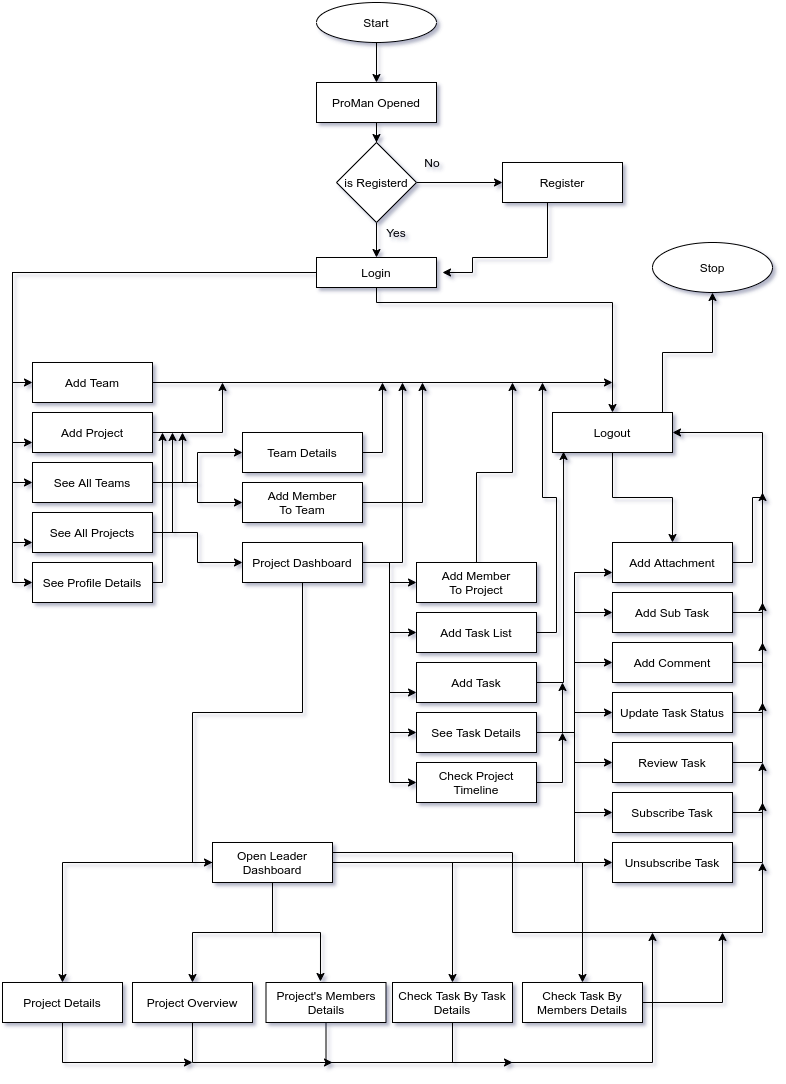
## We are proposing an environment where people can manage their projects remotely. Projects can be managed individually as well as in team oriented environment. People can manage software projects and can estimate cost as well. The beauty of our proposed environment is that even an ordinary person can use it as a task manager for him and can set due date and time so he can work on time and can check the overall status of the project and the remaining time as well.

In PROMAN, user can create project and tasks and can assign it to different members while they are working remotely. User can add comments and attachments to their tasks and can talk to their project fellow members through chat and can easily discuss issues without leaving the environment. One can track and see every minor detail about the project that what is going on and how much work is completed. Leader have other multiple options to approve the work of member, can add members to project and have many options to see every minor detail about the project, task and members.

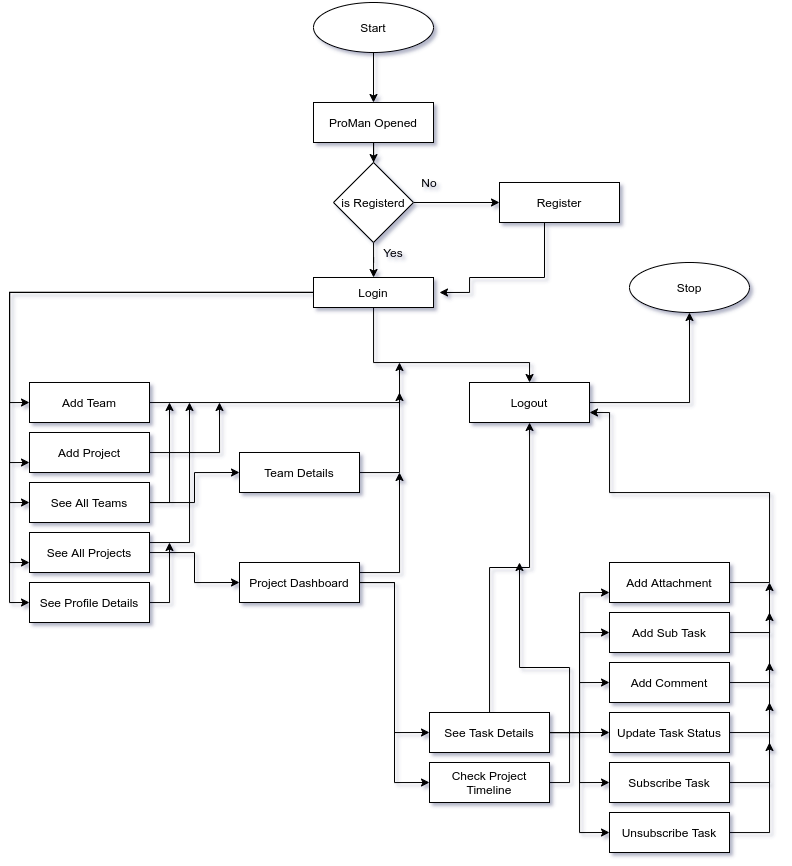
Following are the benefits of our proposed system:

* People can easily manage their project.
* Members can work on different projects at the same time without any conflicts.
* Members can chat with other project members to discuss issues and avoid conflicts.
* Provide large functionality for leader to see the overall project, task and member details.
* Will provide the overall statistics and performance of members which will help leader to identify who is doing well and who is not.

## **1.5 Control Flow Diagram (Leader):**



## **1.6 Control Flow Diagram (User):**



# **2. System Analysis:**

System analysis is the way of a methodology or business to decide its aims and purposes and make agendas and strategies that will achieve them in a creative way. Analyze the problem, understand the problem and its background. One is studying the existing system and other is to understand the requirements and domain of the new system. Ensuing to investigating the fundamentals of the undertaking to be played out, the following stage is to examine the issue and understand its exceptional circumstance. The main movement in the stage is examining the current framework. What's more, other is to understand the necessities and space of the new framework. Understanding the properties and necessities of another framework is increasingly upsetting and requires innovative reasoning and command of existing running framework is likewise troublesome, ill-advised comprehension of present framework can lead redirection from arrangement.

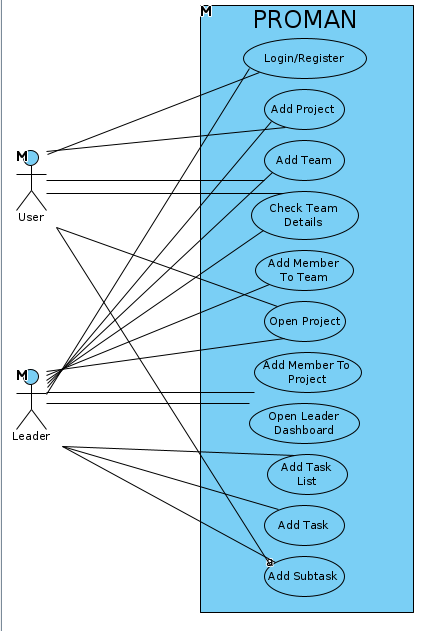
Analysis model includes: -

1. Use case diagram

2. Use case diagram with description

3. System sequence diagram

## **2.1 USE CASE DIAGRAM**



# 

## **2.2 Use Case Description in Detail Expended Format:**

### **2.2.1 Use Case Register:**

1. Details of use case registration which tells how a user is registered to the application.

|  |  |
| --- | --- |
| 1. **Use Case ID no**. | 1. **UC-1** |
| 1. **Scopes** | 1. Online Platform for managing Projects |
| 1. **Name** | 1. Use Case Register |
| 1. **Primary Actors** | 1. User, Leader |
| 1. **Goals** | 1. To enable users to run the system. |
| 1. **Pre-Conditions** | 1. System must be running. |
| 1. **Post Conditions** | 1. Account is Created, User is registered successfully. |
| 1. **Success Scenarios** | 1. User requests for registration by providing his details. Then he clicks on register button and will be registered. |
| 1. **Alternate** | 1. The user email already exists. (User already registered). |

### **2.2.2 Use Case Login**

1. Details of use case login which tells how a user is login to their respective account.

|  |  |
| --- | --- |
| 1. **Use Case ID no**. | 1. **UC-2** |
| 1. **Scopes** | 1. Online Platform for managing Projects |
| 1. **Name** | 1. Use Case Login |
| 1. **Primary Actors** | 1. User, Leader |
| 1. **Goals** | 1. To enable users to login to their respective account. |
| 1. **Pre-Conditions** | 1. System is running and user, leader already registered their account. |
| 1. **Post Conditions** | 1. User enter the credentials and login to their account successfully. |
| 1. **Success Scenarios** | 1. User entered its email and password and hit login and user successfully login to their account. |
| 1. **Alternate** | 1. The user email or password is wrong so login failed. |

### **2.2.3 Use Case Add Project:**

1. Details of use case login which tells how a user can create a new project to start work on it.

|  |  |
| --- | --- |
| 1. **Use Case ID no**. | 1. **UC-3** |
| 1. **Scopes** | 1. Online Platform for managing Projects |
| 1. **Name** | 1. Use Case Add Project |
| 1. **Primary Actors** | 1. User, Leader |
| 1. **Goals** | 1. To enable user to create a new project |
| 1. **Pre-Conditions** | 1. User, Leader is already logged in. |
| 1. **Post Conditions** | 1. Project creation successful. |
| 1. **Success Scenarios** | 1. User entered project name, description, start date, end date etc. 2. And click on create and project created successfully. |
| 1. **Alternate** | 1. User does not fill form properly. |

### **2.2.4 Use Case Add Team:**

1. Details of use case add team which tells how a user can create a new team for its future needs.

|  |  |
| --- | --- |
| 1. **Use Case ID no**. | 1. **UC-4** |
| 1. **Scopes** | 1. Online Platform for managing Projects |
| 1. **Name** | 1. Use Case Add Team |
| 1. **Primary Actors** | 1. User, Leader |
| 1. **Goals** | 1. To enable user to create a new team |
| 1. **Pre-Conditions** | 1. User is already logged in. |
| 1. **Post Conditions** | 1. Team creation successful. |
| 1. **Success Scenarios** | 1. User entered Team name, description. 2. And click on create and team created successfully. |
| 1. **Alternate** | 1. User does not fill form properly. |

### **2.2.5 Use Case Check Team Details:**

1. Details of use case check team details which tells how a user can check the team details.

|  |  |
| --- | --- |
| 1. **Use Case ID no**. | 1. **UC-5** |
| 1. **Scopes** | 1. Online Platform for managing Projects |
| 1. **Name** | 1. Use Case Check Team Details |
| 1. **Primary Actors** | 1. User, Leader |
| 1. **Goals** | 1. To enable user to check its team details e. g. its members. |
| 1. **Pre-Conditions** | 1. User is already logged in. And team is already created. |
| 1. **Post Conditions** | 1. Check team details successfully. |
| 1. **Success Scenarios** | 1. User select Team name, and saw its details like its name, description its members. |
| 1. **Alternate** | 1. No team is present to check details. |

### **2.2.6 Use Case Add Member to Team:**

1. Details of use case add member to team which tells how a user can add a new member to the team.

|  |  |
| --- | --- |
| 1. **Use Case ID no**. | 1. **UC-6** |
| 1. **Scopes** | 1. Online Platform for managing Projects |
| 1. **Name** | 1. Use Case Add Member To Team |
| 1. **Primary Actors** | 1. Leader |
| 1. **Goals** | 1. To enable leader to add a new member to existing team. |
| 1. **Pre-Conditions** | 1. Leader is already logged in. And team is already created. |
| 1. **Post Conditions** | 1. Add new member to team successfully. |
| 1. **Success Scenarios** | 1. Leader clicked on add member and entered the user email and then add user to team. |
| 1. **Alternate** | 1. No team is present to add new member. 2. User not found due to wrong email. |

### **2.2.7 Use Case Open Project**

1. Details of use case open project which tells how a user can open the project in which user is enrolled or user created it yourself.

|  |  |
| --- | --- |
| 1. **Use Case ID no**. | 1. **UC-7** |
| 1. **Scopes** | 1. Online Platform for managing Projects |
| 1. **Name** | 1. Use Case Open Project |
| 1. **Primary Actors** | 1. User, Leader |
| 1. **Goals** | 1. To enable user to open the project and see its details. |
| 1. **Pre-Conditions** | 1. User is already logged in. And project is already created in which user in enrolled. |
| 1. **Post Conditions** | 1. Open the project successfully. |
| 1. **Success Scenarios** | 1. User clicked on proceed button on desired project. And project is open successfully. |
| 1. **Alternate** | 1. No project was present in which user is enrolled. |

### **2.2.8 Use Case Add Member To Project:**

1. Details of use case add member to project which tells how a leader can add a new member to the existing project.

|  |  |
| --- | --- |
| 1. **Use Case ID no**. | 1. **UC-8** |
| 1. **Scopes** | 1. Online Platform for managing Projects |
| 1. **Name** | 1. Use Case Add Member To Project |
| 1. **Primary Actors** | 1. Leader |
| 1. **Goals** | 1. To enable leader to add a new member to existing project. |
| 1. **Pre-Conditions** | 1. Leader already opened the project. |
| 1. **Post Conditions** | 1. Add new member to project successfully. |
| 1. **Success Scenarios** | 1. Leader clicked on add member and entered the user email and then add user to project. |
| 1. **Alternate** | 1. User not found due to wrong email. |

### **2.2.9 Use Case Open Leader Dashboard:**

1. Details of use case open leader dashboard which tells how a user can open the project’s leader dashboard in which all details related to the project is showing (only for leader).

|  |  |
| --- | --- |
| 1. **Use Case ID no**. | 1. **UC-9** |
| 1. **Scopes** | 1. Online Platform for managing Projects |
| 1. **Name** | 1. Use Case Open Leader Dashboard |
| 1. **Primary Actors** | 1. Leader |
| 1. **Goals** | 1. To enable leader to open the leader dashboard. |
| 1. **Pre-Conditions** | 1. User is already logged in. And project is already created in which user is a leader. |
| 1. **Post Conditions** | 1. Open the leader dashboard successfully. |
| 1. **Success Scenarios** | 1. User clicked on open leader dashboard button. And then leader dashboard opened successfully. |
| 1. **Alternate** | 1. User was not leader of the project. |

### **2.2.10 Use Case Add Task List:**

1. Details of use case add task list which tells how a leader can add a new task list.

|  |  |
| --- | --- |
| 1. **Use Case ID no**. | 1. **UC-10** |
| 1. **Scopes** | 1. Online Platform for managing Projects |
| 1. **Name** | 1. Use Case Add Task List |
| 1. **Primary Actors** | 1. Leader |
| 1. **Goals** | 1. To enable leader to create new task list. |
| 1. **Pre-Conditions** | 1. User is already logged in. And project is already created in which user is a leader. |
| 1. **Post Conditions** | 1. Add a new Task List successfully. |
| 1. **Success Scenarios** | 1. Leader clicked on add task list, and leader enter name and description for task list and task list created successfully. |
| 1. **Alternate** | 1. User was not leader of the project. |

### **2.2.11 Use Case Add Task:**

1. Details of use case add task which tells how a leader can add a new task.

|  |  |
| --- | --- |
| 1. **Use Case ID no**. | 1. **UC-11** |
| 1. **Scopes** | 1. Online Platform for managing Projects |
| 1. **Name** | 1. Use Case Add Task |
| 1. **Primary Actors** | 1. Leader |
| 1. **Goals** | 1. To enable leader to create new task. |
| 1. **Pre-Conditions** | 1. User is already logged in. And project is already created in which user is a leader. |
| 1. **Post Conditions** | 1. Add a new Task successfully. |
| 1. **Success Scenarios** | 1. Leader clicked on add task, and leader enter name and description, assign members etc for task and task created successfully. |
| 1. **Alternate** | 1. User was not leader of the project. |

### **2.2.12 Use Case Add Sub Task:**

1. Details of use case add sub Task which tells how a user can add a new sub task.

|  |  |
| --- | --- |
| 1. **Use Case ID no**. | 1. **UC-12** |
| 1. **Scopes** | 1. Online Platform for managing Projects |
| 1. **Name** | 1. Use Case Add Sub Task |
| 1. **Primary Actors** | 1. User, Leader |
| 1. **Goals** | 1. To enable user to create new sub task. |
| 1. **Pre-Conditions** | 1. User is already logged in. And project is already created in which user is already enrolled. A task is already assigned. |
| 1. **Post Conditions** | 1. Add a new Sub Task successfully. |
| 1. **Success Scenarios** | 1. User clicked on create new sub task button and enter its details and sub task created successfully. |
| 1. **Alternate** | 1. User was not among the assigned member of the task. |

### **2.2.13 Use Case Add Comment:**

Details of use case add comment which tells the leader or user how to add comment.

|  |  |
| --- | --- |
| **Use Case ID no**. | **UC-13** |
| **Scopes** | Online Platform for managing Projects |
| **Name** | Add Comment |
| **Primary Actors** | Leader, User |
| **Goals** | To Enable Leader and user to add comments. |
| **Pre-Conditions** | Leader or user must be registered and logged in.  Project must be existed.  Task must be existed. |
| **Post Conditions** | Comment will be added. |
| **Success Scenarios** | User will write comment and it will be posted when user will click add comment button. |
| **Alternate** | Task does not exists. |

### **2.2.14 Use Case Add Attachment:**

Details of use case add attachment which tells the leader or user how to add attachment.

|  |  |
| --- | --- |
| **Use Case ID no**. | **UC-14** |
| **Scopes** | Online Platform for managing Projects |
| **Name** | Add Attachment |
| **Primary Actors** | Leader, User |
| **Goals** | To Enable Leader or User to add attachments. |
| **Pre-Conditions** | Leader or user must be registered and logged in.  Project must be existed.  Task must be existed. |
| **Post Conditions** | Attachment will be added. |
| **Success Scenarios** | User will add attachment by clicking on add attachment or by drag and drop on add attachment button. |
| **Alternate** | Task does not exists. |

### **2.2.15 Use Case Review Task:**

Details of use case Review task which tells the Leader how to review task.

|  |  |
| --- | --- |
| **Use Case ID no**. | **UC-15** |
| **Scopes** | Online Platform for managing Projects |
| **Name** | Review Task |
| **Primary Actors** | Leader |
| **Goals** | To enable Leader to review task. |
| **Pre-Conditions** | Leader must be registered and logged in.  Project must be existed.  Task must be existed. |
| **Post Conditions** | Task will be reviewed. |
| **Success Scenarios** | Leader will be able to change the status of task from pending to done and pending to in progress. |
| **Alternate** | Task is not available for review. |

### **2.2.16 Use Case Update Task Status:**

Details of use case Update Task status which tells the Leader and user how to Update Task Status.

|  |  |
| --- | --- |
| **Use Case ID no**. | **UC-16** |
| **Scopes** | Online Platform for managing Projects |
| **Name** | Update Task Status |
| **Primary Actors** | Leader, User |
| **Goals** | To enable User to forward task from in progress to pending. |
| **Pre-Conditions** | Leader must be registered and logged in.  Project must be existed.  Task must be existed. |
| **Post Conditions** | Status of task will be updated. |
| **Success Scenarios** | Leader or User will click on Push button to change the status from in progress to pending. |
| **Alternate** | No alternate. |

### **2.2.17 Use Case Subscribe:**

Details of use case Subscribe which tells the leader and user how to subscribe a task.

|  |  |
| --- | --- |
| **Use Case ID no**. | **UC-17** |
| **Scopes** | Online Platform for managing Projects |
| **Name** | Subscribe |
| **Primary Actors** | Leader, User |
| **Goals** | To enable Leader or User to get notifications about the task. |
| **Pre-Conditions** | Leader or User must be registered and logged in.  Project must be existed.  Task must be existed. |
| **Post Conditions** | Task will be subscribed. |
| **Success Scenarios** | Leader or User will click on Subscribe button and will be able to get notifications about that task. |
| **Alternate** | User is among the assigned member of the task so will not be able to subscribe. |

### **2.2.18 Use Case Unsubscribe:**

Details of use case Unsubscribe which tells the leader and user how to unsubscribe a task.

|  |  |
| --- | --- |
| **Use Case ID no**. | **UC-18** |
| **Scopes** | Online Platform for managing Projects |
| **Name** | Unsubscribe |
| **Primary Actors** | Leader, User |
| **Goals** | To enable Leader or User to not be able to get notifications about that particular task. |
| **Pre-Conditions** | Leader or User must be registered and logged in.  Project must be existed.  Task must be existed.  Task must be subscribed. |
| **Post Conditions** | Task will be subscribed. |
| **Success Scenarios** | Leader or User will click on Unsubscribe button and will not be able to get notifications about that task in future. |
| **Alternate** | Task is not Subscribed. |

### **2.2.19 Use Case Profile:**

Details of use case Profile which tells the leader and user how to check Profile.

|  |  |
| --- | --- |
| **Use Case ID no**. | **UC-19** |
| **Scopes** | Online Platform for managing Projects |
| **Name** | Check Profile |
| **Primary Actors** | Leader, User |
| **Goals** | To enable Leader or User to see profile details. |
| **Pre-Conditions** | Leader or User must be registered and logged in. |
| **Post Conditions** | Leader or User will be able to see profile. |
| **Success Scenarios** | Leader or User will click on avatar (Navbar) and then will click on profile option to see the profile. |
| **Alternate** | Leader or User is not registered or logged in. |

### **2.2.20 Use Case Check Project Details:**

Details of use case Check Project Details which tells the Leader how to Check Project Details.

|  |  |
| --- | --- |
| **Use Case ID no**. | **UC-20** |
| **Scopes** | Online Platform for managing Projects |
| **Name** | Check Project Details |
| **Primary Actors** | Leader |
| **Goals** | To enable Leader to see project details. |
| **Pre-Conditions** | Leader must be registered and logged in.  Project must be existed and User must be the Leader. |
| **Post Conditions** | Leader will be able to see project details. |
| **Success Scenarios** | Leader will click on Leader Dashboard and then Project details button in side bar to see project details. |
| **Alternate** | User is not a leader. |

### **2.2.21 Use Case Check Overview of Project:**

Details of use case Check Overview of Project which tells the Leader how to Check Overview of Project.

|  |  |
| --- | --- |
| **Use Case ID no**. | **UC-21** |
| **Scopes** | Online Platform for managing Projects |
| **Name** | Check Overview of Project |
| **Primary Actors** | Leader |
| **Goals** | To enable Leader to see overview of project. |
| **Pre-Conditions** | Leader must be registered and logged in.  Project must be existed and User must be the Leader. |
| **Post Conditions** | Leader will be able to see overview of project. |
| **Success Scenarios** | Leader will click on leader dashboard from project and then click on project overview button from sidebar to see project overview. |
| **Alternate** | User is not a leader. |

### **2.2.22 Use Case Check all Member details of project:**

Details of use check all Member details of project which tells the Leader how to see all Member details of project.

|  |  |
| --- | --- |
| **Use Case ID no**. | **UC-22** |
| **Scopes** | Online Platform for managing Projects |
| **Name** | Check all Member details of project |
| **Primary Actors** | Leader |
| **Goals** | To enable Leader to see member details of project. |
| **Pre-Conditions** | Leader must be registered and logged in.  Project must be existed.  Members must be existed.  User must be the Leader. |
| **Post Conditions** | Leader will be able to see member details of project. |
| **Success Scenarios** | Leader will click on leader dashboard from project and then click on member details button from sidebar to see project member’s details. |
| **Alternate** | User is not a leader. |

### **2.2.23 Use Case Check Task by task details:**

Details of use case Check Task by task details which tells the Leader how to see Task by task details.

|  |  |
| --- | --- |
| **Use Case ID no**. | **UC-23** |
| **Scopes** | Online Platform for managing Projects |
| **Name** | Task by task details |
| **Primary Actors** | Leader |
| **Goals** | To enable Leader to see the overall task details. |
| **Pre-Conditions** | Leader must be registered and logged in.  Project must be existed.  Task must be existed.  User must be the Leader. |
| **Post Conditions** | Leader will be able to see the overall task details. |
| **Success Scenarios** | Leader will click on leader dashboard from project and then click on task by task button from sidebar to see overall task details. |
| **Alternate** | User is not a leader. |

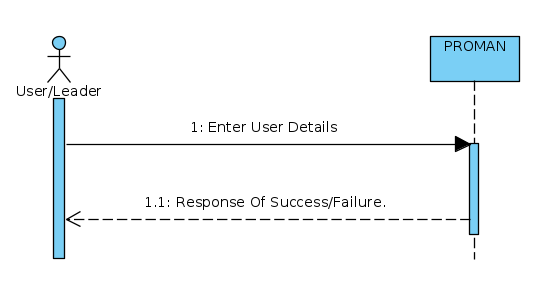
### **2.2.24 Use Case Task by member details:**

Details of use case Check Task by member details which tells the Leader how to see Task by member details.

|  |  |
| --- | --- |
| **Use Case ID no**. | **UC-24** |
| **Scopes** | Online Platform for managing Projects |
| **Name** | Task by member details |
| **Primary Actors** | Leader |
| **Goals** | To enable Leader to see task by member details. |
| **Pre-Conditions** | Leader must be registered and logged in.  Project must be existed.  Task must be existed.  Members must be existed.  User must be the Leader. |
| **Post Conditions** | Leader will be able to see task by member details. |
| **Success Scenarios** | Leader will click on leader dashboard from project and then click on member by task details button from sidebar to see task by member details. |
| **Alternate** | User is not a leader. |

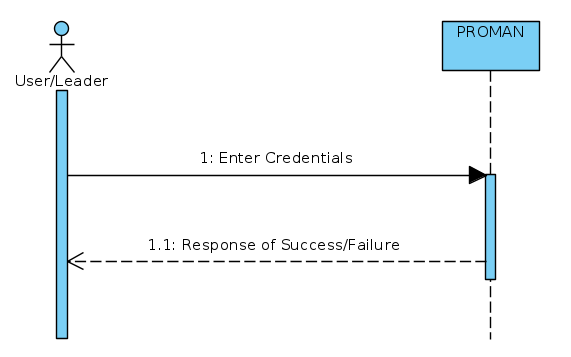
## **2.3 SEQUENCE DIAGRAMS:**

### **2.3.1 Register:**

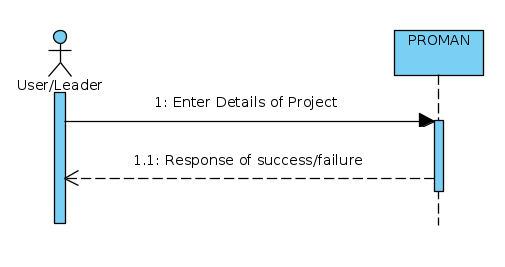


### 

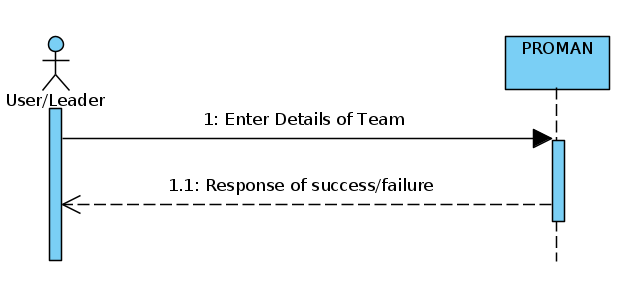
### **2.3.2 Login:**



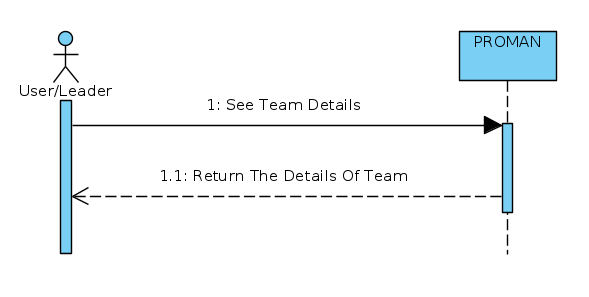
### **2.3.3 Add Project:**



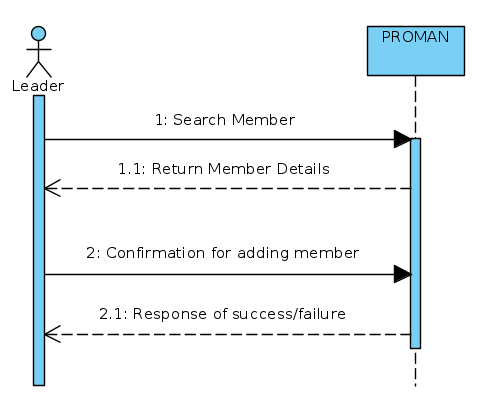
### **2.3.4 Add Team:**



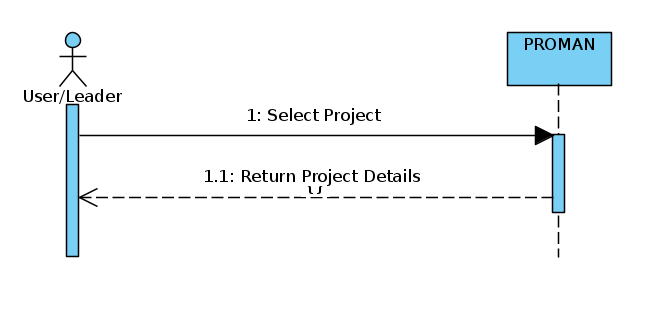
### **2.3.5 Check Team Details:**



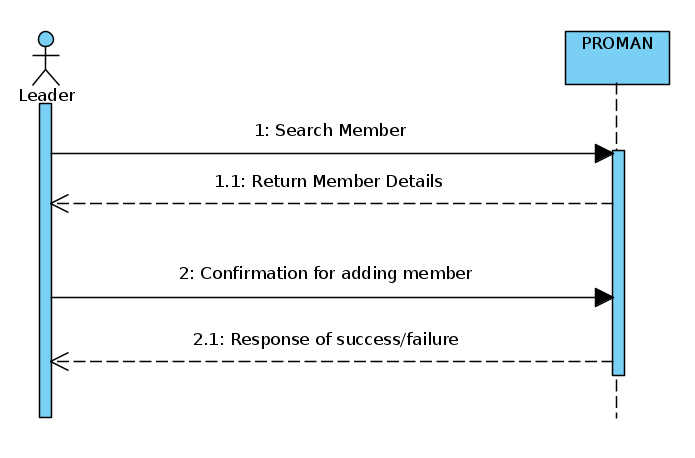
### **2.3.6 Add Member to Team:**



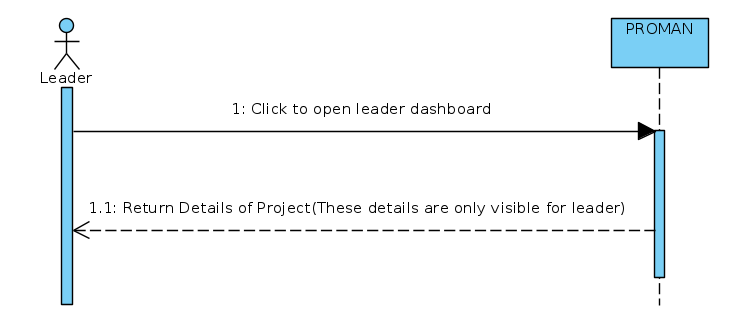
### **2.3.7 Open Project:**



### **2.3.8 Add Member to Project:**

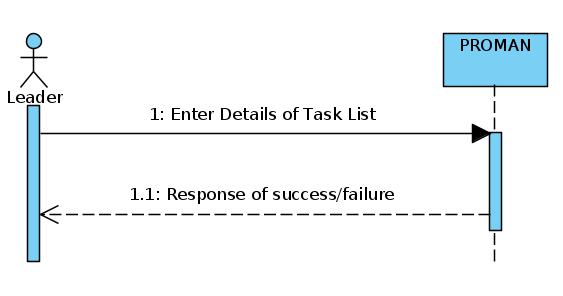


### **2.3.9 Open Leader Dashboard:**

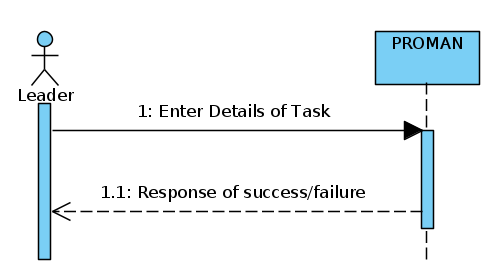


### 

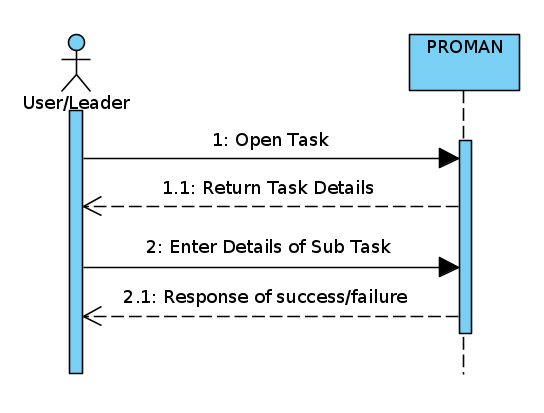
### **2.3.10 Add Task List**



### **2.3.11 Add Task:**

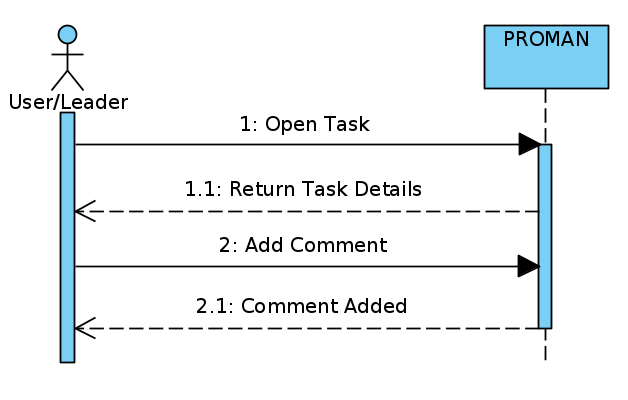


### **2.3.12 Add Sub Task:**

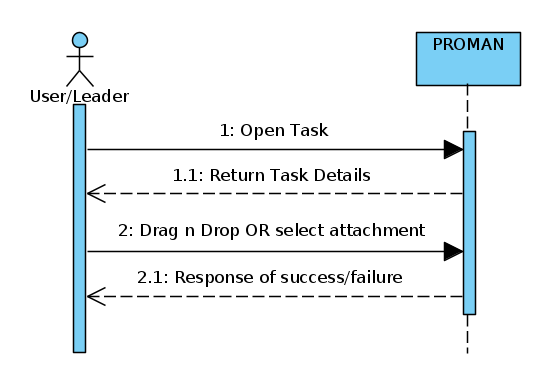


### 

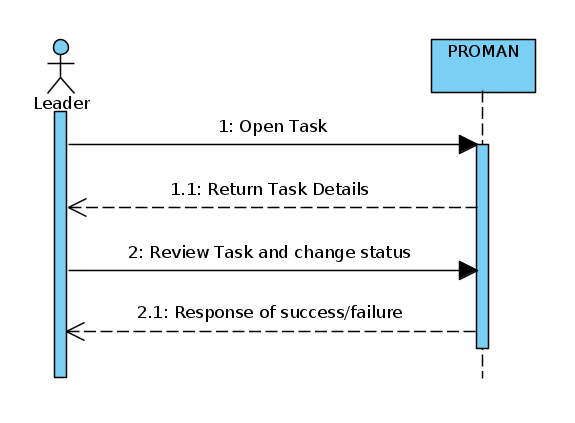
### **2.3.13 Add Comment:**



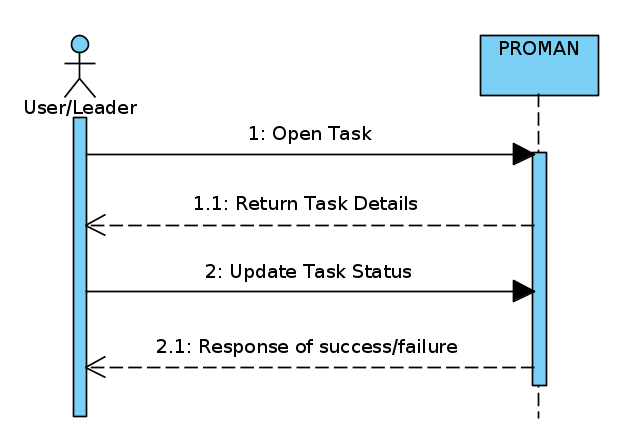
### **2.3.14 Add Attachment:**



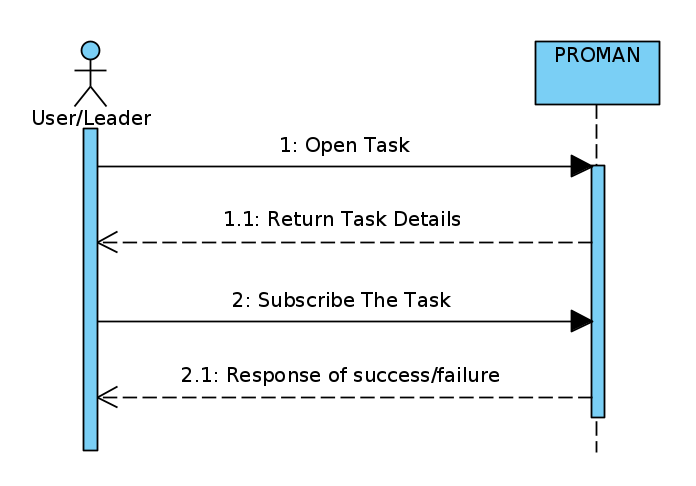
### **2.3.15 Review Task:**



### **2.3.16 Update Task Status:**

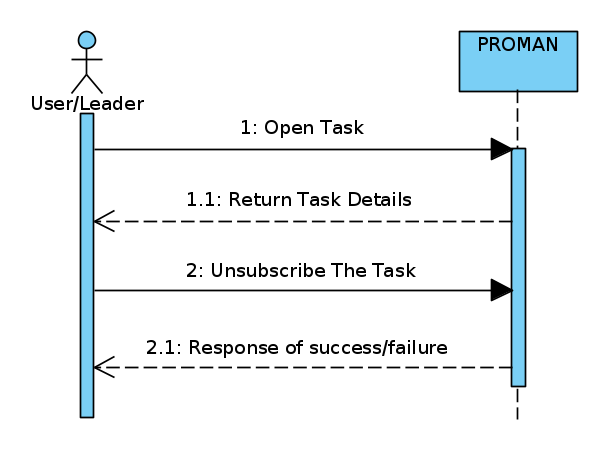


### **2.3.17 Subscribe:**

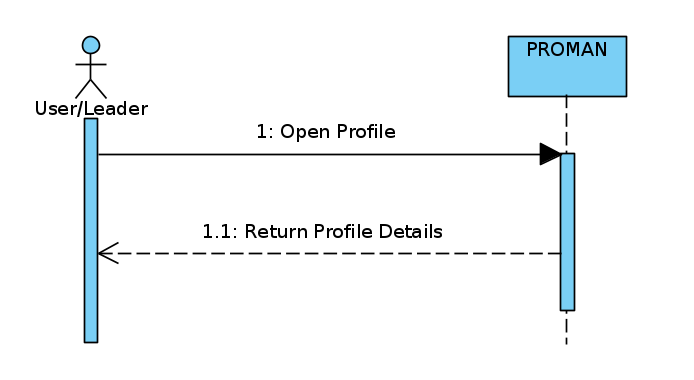


### 

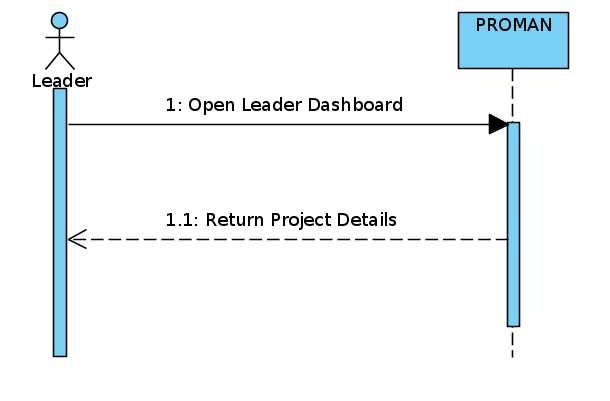
### **2.3.18 Unsubscribe:**



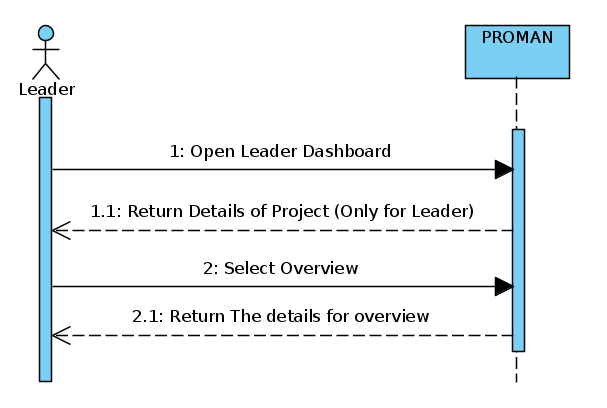
### **2.3.19 Profile:**



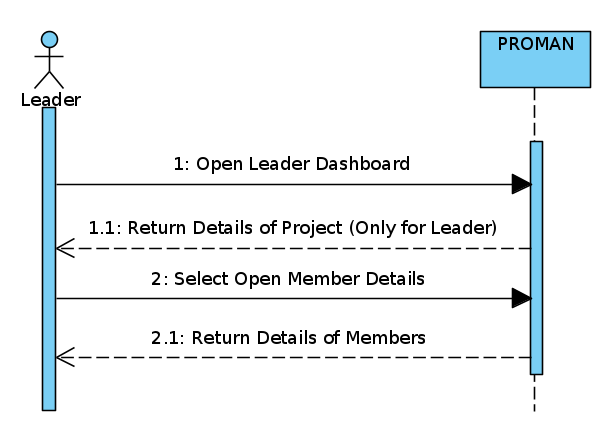
### **2.3.20 Check Project Details:**



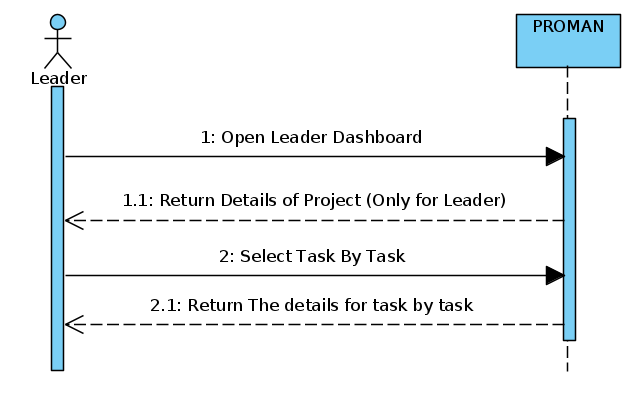
### **2.3.21 Check Overview of Project:**



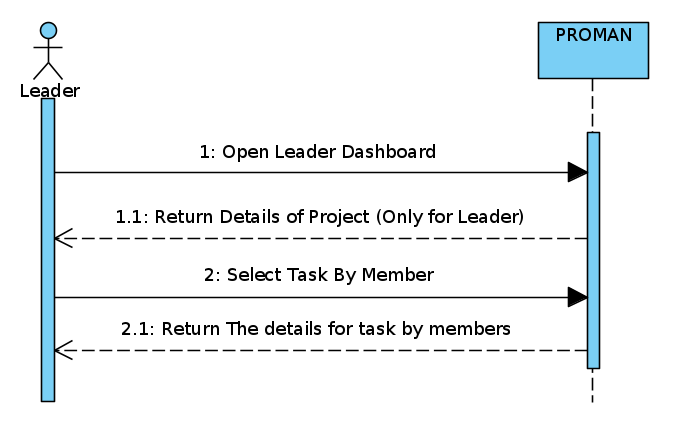
### **2.3.22 Check All Member Details of Project:**



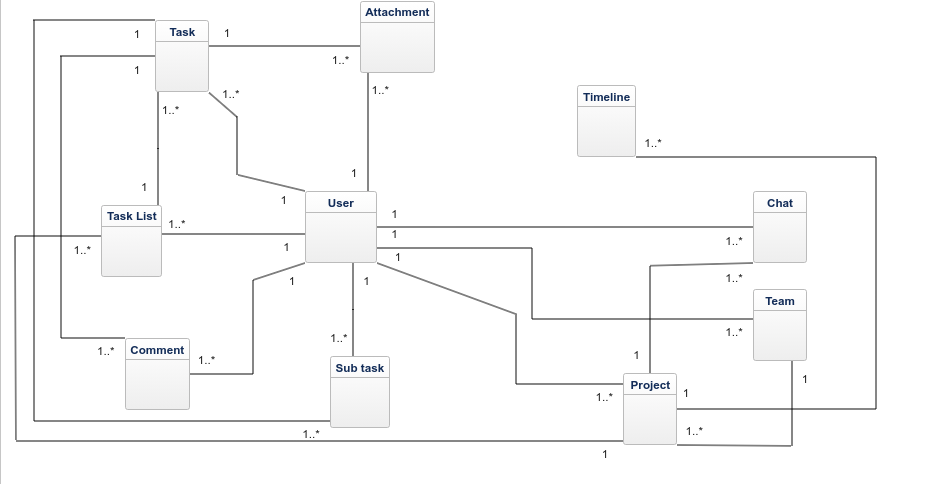
### **2.3.23 Check Task By Task Details:**



### **2.3.24 Check Task by Member Details:**

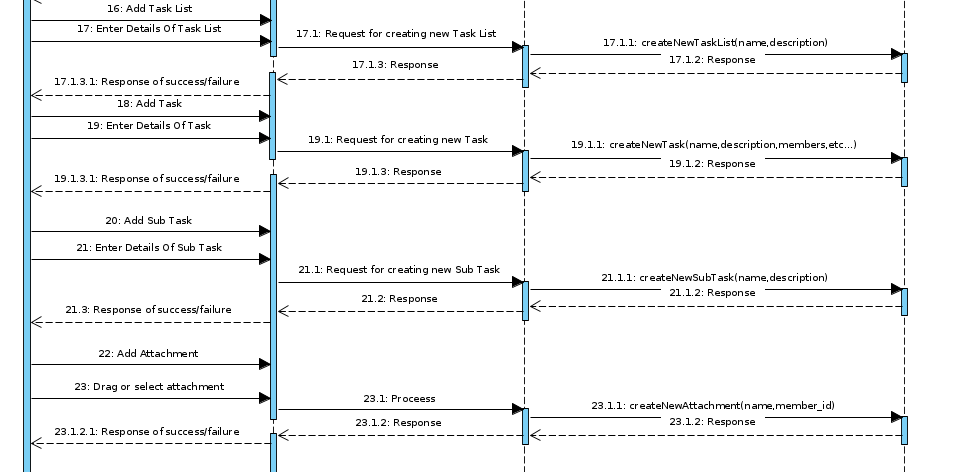
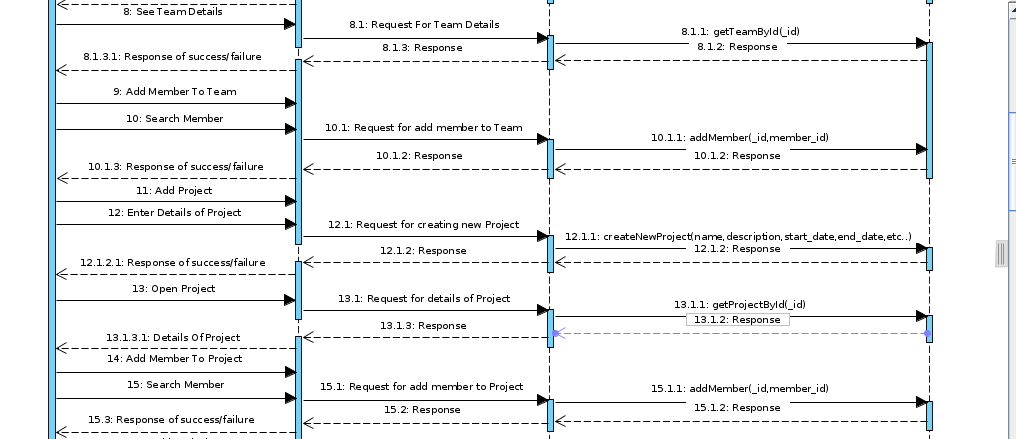
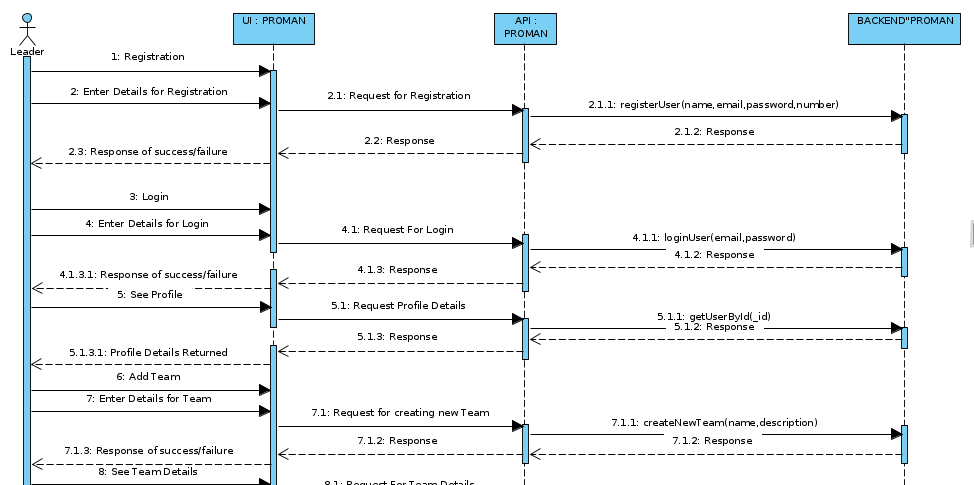


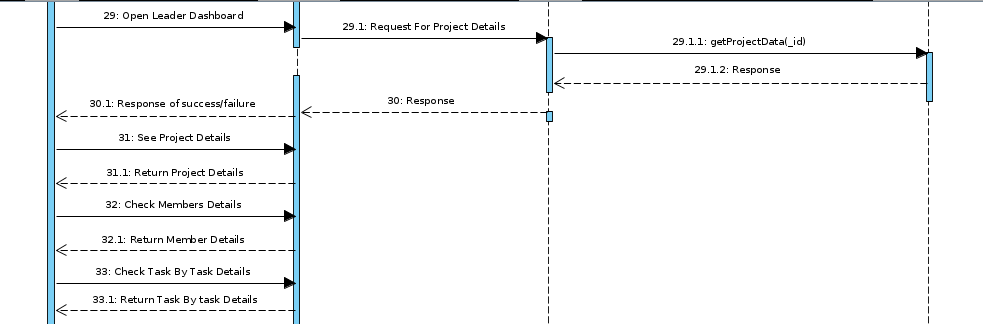
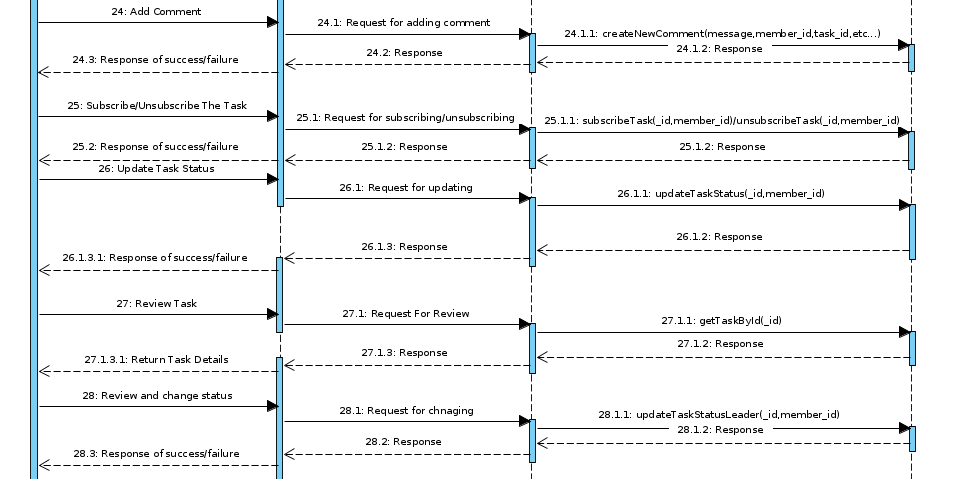
## **2.4 Domain Model**



3. System Design:

## **3.1 System Sequence Diagram: (Leader)**





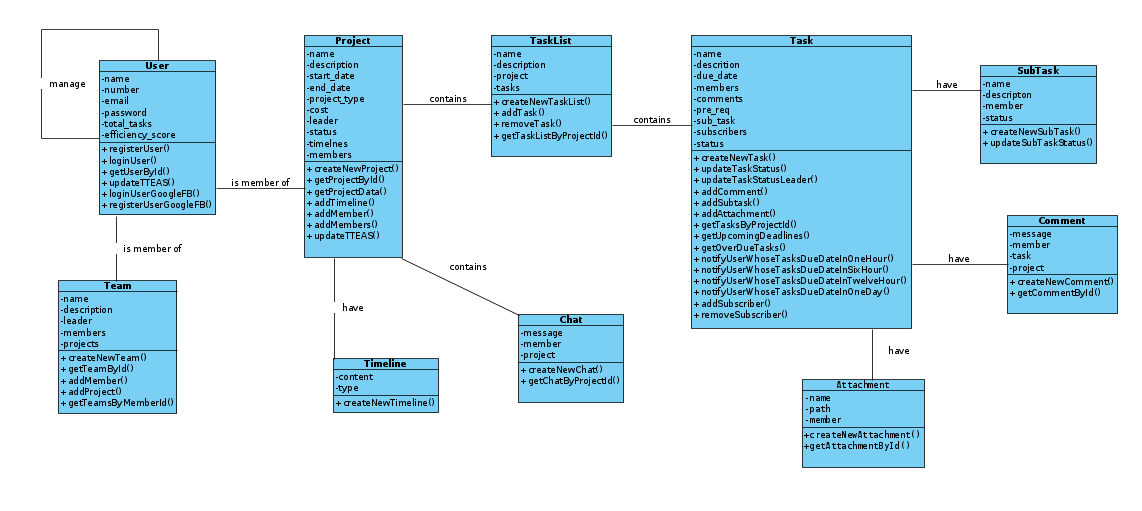
### **3.2 System Sequence Diagram: (User)**

### 

## **3.4 Alternate to ERD:**

Because we are using Mongo DB

## **3.5 Class Diagram**



# 4. Implementation:

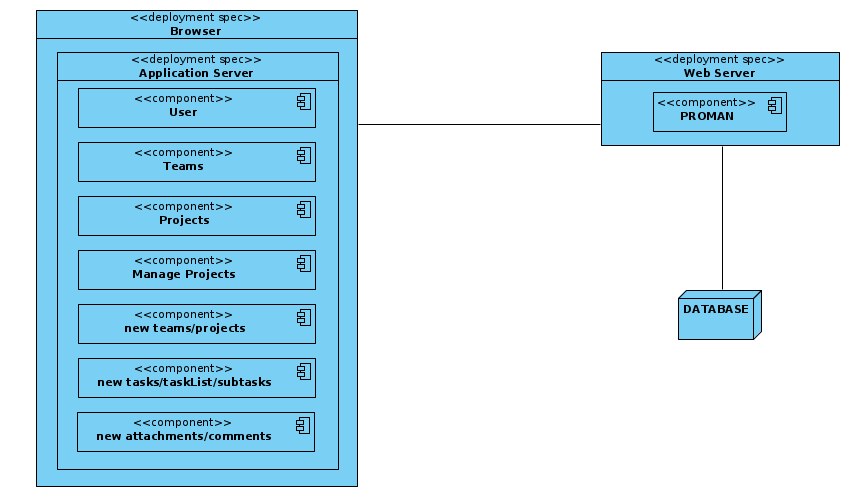
Implementation is basically another word for coding/software programming. This chapter includes the component diagrams and deployment diagram**.**

## **4.1 Functional Features of the Model:**

The main features of the system are:

* User friendly
* Friendly interface
* Responsiveness
* Reliability
* Security

# 4.2 Deployment Diagram.



# 5. System Testing:

Software testing is a post development step. It will be performed to minimize any potential errors or uncertainties. We have tested all the use cases and each of them is passed. The testing of the software gives a satisfaction to the developer as well as the user and it gives a quality assurance to all the participants worried.

## **5.1 Black Box Testing:**

Black box testing, also known as behavioral testing, is a software testing method in which the internal structure/design/implementation of the item being tested is not these tests can be functional or non-functional though usually useful.

## **5.2 White Box Testing:**

White box also known as Clear Box Testing, Open Box Testing, Glass Box Testing, Transparent Box Testing, Code-Based Testing and Structural Testing. This is a software testing method in which the internal structure/design/implementation of the item being tested is known to the tester.

## **5.3 Test Cases:**

### **5.3.1 Register:**

|  |  |
| --- | --- |
| **Test Case ID** | TC-01 |
| **Functional Area/Module** | Register |
| **Action to Perform** | 1. User starts the application. 2. Clicks on register button. 3. Enters name, email, password and other personal information. 4. Clicks on register button. 5. Finish |
| **Purpose** | To register a user. |
| **Prerequisites** | Application must be running. |
| **Test Case Engineer** | Kamran Nazir |
| **Environment** | Windows 10, Chrome browser |
| **Expected Result(s)** | User registered successfully. |
| **Comment:** Test passed successfully. | |

## **5.3.2 Login:**

|  |  |
| --- | --- |
| **Test Case ID** | TC-02 |
| **Functional Area/Module** | Login |
| **Action to Perform** | 1. User starts the application. 2. Clicks on Login button. 3. Enters his credentials. 4. Clicks on login button. 5. Finish |
| **Purpose** | To create a project. |
| **Prerequisites** | Application must be running.  User must be registered and logged in. |
| **Test Case Engineer** | Kamran Nazir |
| **Environment** | Windows 10, Chrome browser |
| **Expected Result(s)** | User logged in successfully. |
| **Comment:** Test passed successfully. | |

## **5.3.3 Add Project:**

|  |  |
| --- | --- |
| **Test Case ID** | TC-03 |
| **Functional Area/Module** | Add Project |
| **Action to Perform** | 1. User starts the application. 2. Clicks on Login button. 3. Enters his credentials. 4. Clicks on login button. 5. Finish |
| **Purpose** | To create a project. |
| **Prerequisites** | Application must be running.  User must be registered and logged in. |
| **Test Case Engineer** | Kamran Nazir |
| **Environment** | Windows 10, Chrome browser |
| **Expected Result(s)** | Project created successfully. |
| **Comment:** Test passed successfully. | |

## **5.3.4 Add Team:**

|  |  |
| --- | --- |
| **Test Case ID** | TC-04 |
| **Functional Area/Module** | Add Team |
| **Action to Perform** | 1. User starts the application. 2. User logs in by entering right  credentials. 3. Clicks on add team. 4. Enter team information. 5. Click on create button. 6. Finish |
| **Purpose** | To create a team. |
| **Prerequisites** | Application must be running.  User must be registered and logged in. |
| **Test Case Engineer** | Kamran Nazir |
| **Environment** | Windows 10, Chrome browser |
| **Expected Result(s)** | Team added successfully. |
| **Comment:** Test passed successfully. | |

## **5.3.5 Add Members to Team:**

|  |  |
| --- | --- |
| **Test Case ID** | TC-05 |
| **Functional Area/Module** | Add Members to Team |
| **Action to Perform** | 1. User starts the application. 2. User logs in by entering right  credentials. 3. Clicks Add members to team  button from sidebar. 4. Click on add member and Enter his email to search. 5. Click on add button. 6. Finish. |
| **Purpose** | To add members to a team. |
| **Prerequisites** | Application must be running.  User must be registered and logged in.  User you want to add must exists and should not be a part of team. |
| **Test Case Engineer** | Kamran Nazir |
| **Environment** | Windows 10, Chrome browser |
| **Expected Result(s)** | Member added to team successfully. |
| **Comment:** Test passed successfully. | |

## **5.3.6 Add Task List:**

|  |  |
| --- | --- |
| **Test Case ID** | TC-06 |
| **Functional Area/Module** | Add task list |
| **Action to Perform** | 1. User starts the application. 2. User logs in by entering right  credentials. 3. Clicks on existing project. 4. Clicks on add task list. 5. Enter its name and description. 6. Click on create button. 7. Finish. |
| **Purpose** | To add task list to projects. |
| **Prerequisites** | Application must be running.  User must be registered and logged in.  Project must be existed and you must be the leader. |
| **Test Case Engineer** | Kamran Nazir |
| **Environment** | Windows 10, Chrome browser |
| **Expected Result(s)** | Task list added successfully. |
| **Comment:** Test passed successfully. | |

## **5.3.7 Add Task:**

|  |  |
| --- | --- |
| **Test Case ID** | TC-07 |
| **Functional Area/Module** | Add task |
| **Action to Perform** | 1. User starts the application. 2. User logs in by entering right  credentials. 3. Clicks on existing project. 4. Clicks on add task. 5. Enter its name and other detail. 6. Click on create button. 7. Finish. |
| **Purpose** | To add task to project. |
| **Prerequisites** | Application must be running.  User must be registered and logged in.  Project must be existed and you must be the leader. |
| **Test Case Engineer** | Kamran Nazir |
| **Environment** | Windows 10, Chrome browser |
| **Expected Result(s)** | Task added successfully. |
| **Comment:** Test passed successfully. | |

## **5.3.8 Add Sub Task:**

|  |  |
| --- | --- |
| **Test Case ID** | TC-08 |
| **Functional Area/Module** | Add Subtask |
| **Action to Perform** | 1. User starts the application. 2. User logs in by entering right  credentials. 3. Clicks on existing project. 4. Clicks on existing task. 5. Click on add subtask button. 6. Enter its name and description. 7. Click on create button. 8. Finish. |
| **Purpose** | To add Sub task to projects. |
| **Prerequisites** | Application must be running.  User must be registered and logged in.  Project and task must be existed. |
| **Test Case Engineer** | Kamran Nazir |
| **Environment** | Windows 10, Chrome browser |
| **Expected Result(s)** | Sub Task added successfully. |
| **Comment:** Test passed successfully. | |

## **5.3.9 Add Members to Project:**

|  |  |
| --- | --- |
| **Test Case ID** | TC-09 |
| **Functional Area/Module** | Add members to project |
| **Action to Perform** | 1. User starts the application. 2. User logs in by entering right  credentials. 3. Clicks on existing project. 4. Clicks on add member button. 5. Enter its email. 6. Click on add. 7. Finish. |
| **Purpose** | To add task list to projects. |
| **Prerequisites** | Application must be running.  User must be registered and logged in.  Project must be existed and you must be the leader. |
| **Test Case Engineer** | Kamran Nazir |
| **Environment** | Windows 10, Chrome browser |
| **Expected Result(s)** | Member added to project successfully. |
| **Comment:** Test passed successfully. | |

## **5.3.10 Add Comment:**

|  |  |
| --- | --- |
| **Test Case ID** | TC-10 |
| **Functional Area/Module** | Add comment |
| **Action to Perform** | 1. User starts the application. 2. User logs in by entering right  credentials. 3. Clicks on existing project. 4. Clicks on existing task. 5. Enter its comment. 6. Click on post button. 7. Finish. |
| **Purpose** | To add comments on tasks. |
| **Prerequisites** | Application must be running.  User must be registered and logged in.  Project and task must be existed. |
| **Test Case Engineer** | Kamran Nazir |
| **Environment** | Windows 10, Chrome browser |
| **Expected Result(s)** | Comment posted successfully. |
| **Comment:** Test passed successfully. | |

## **5.3.11 Add Attachment:**

|  |  |
| --- | --- |
| **Test Case ID** | TC-11 |
| **Functional Area/Module** | Add attachment |
| **Action to Perform** | 1. User starts the application. 2. User logs in by entering right  credentials. 3. Clicks on existing project. 4. Clicks on existing task. 5. Click on attachment button or drag and drop file to add attachment. 6. Finish. |
| **Purpose** | To add attachment in task. |
| **Prerequisites** | Application must be running.  User must be registered and logged in.  Project and task must be existed. |
| **Test Case Engineer** | Kamran Nazir |
| **Environment** | Windows 10, Chrome browser |
| **Expected Result(s)** | Attachment posted successfully. |
| **Comment:** Test passed successfully. | |

## **5.3.12 Subscribe Task:**

|  |  |
| --- | --- |
| **Test Case ID** | TC-12 |
| **Functional Area/Module** | Subscribe Task |
| **Action to Perform** | 1. User starts the application. 2. User logs in by entering right  credentials. 3. Clicks on existing project. 4. Clicks on existing task. 5. Press Subscribe button. 6. Finish. |
| **Purpose** | To get notifications about a tasks. |
| **Prerequisites** | Application must be running.  User must be registered and logged in.  Project and task must be existed.  User should not be a part of that task. |
| **Test Case Engineer** | Kamran Nazir |
| **Environment** | Windows 10, Chrome browser |
| **Expected Result(s)** | Task Subscribed successfully. |
| **Comment:** Test passed successfully. | |

## **5.3.13 Unsubscribe task:**

|  |  |
| --- | --- |
| **Test Case ID** | TC-13 |
| **Functional Area/Module** | Unsubscribe Task |
| **Action to Perform** | 1. User starts the application. 2. User logs in by entering right  credentials. 3. Clicks on existing project. 4. Clicks on existing task. 5. Press Unsubscribe button. 6. Finish. |
| **Purpose** | To stop getting notifications about a tasks. |
| **Prerequisites** | Application must be running.  User must be registered and logged in.  Project and task must be existed.  User must be subscribed to that task. |
| **Test Case Engineer** | Kamran Nazir |
| **Environment** | Windows 10, Chrome browser |
| **Expected Result(s)** | Task Unsubscribed successfully. |
| **Comment:** Test passed successfully. | |

## **5.3.14 See Profile:**

|  |  |
| --- | --- |
| **Test Case ID** | TC-14 |
| **Functional Area/Module** | See Profile |
| **Action to Perform** | 1. User starts the application. 2. User logs in by entering right  credentials. 3. User clicks on avatar. 4. Then clicks on profile button. 5. Finish. |
| **Purpose** | To add comments on tasks. |
| **Prerequisites** | Application must be running.  User must be registered and logged in. |
| **Test Case Engineer** | Kamran Nazir |
| **Environment** | Windows 10, Chrome browser |
| **Expected Result(s)** | Profile loaded successfully. |
| **Comment:** Test passed successfully. | |

## **5.3.15 Review Task:**

|  |  |
| --- | --- |
| **Test Case ID** | TC-15 |
| **Functional Area/Module** | Review Task |
| **Action to Perform** | 1. User starts the application. 2. User logs in by entering right  credentials. 3. User opens the project. 4. User opens task. 5. User clicks on push button. 6. Finish. |
| **Purpose** | To push task for Leader’s review. |
| **Prerequisites** | Application must be running.  User must be registered and logged in.  Project and task must be existed. |
| **Test Case Engineer** | Kamran Nazir |
| **Environment** | Windows 10, Chrome browser |
| **Expected Result(s)** | Task is pushed for review successfully. |
| **Comment:** Test passed successfully. | |

## **5.4 Result:**

|  |  |
| --- | --- |
| **Test Case Number** | **Results** |
| **TC-01** | **Pass** |
| **TC-02** | **Pass** |
| **TC-03** | **Pass** |
| **TC-04** | **Pass** |
| **TC-05** | **Pass** |
| **TC-06** | **Pass** |
| **TC-07** | **Pass** |
| **TC-08** | **Pass** |
| **TC-09** | **Pass** |
| **TC-10** | **Pass** |
| **TC-11** | **Pass** |
| **TC-12** | **Pass** |
| **TC-13** | **Pass** |
| **TC-14** | **Pass** |
| **TC-15** | **Pass** |

# Conclusion

## **6. Conclusion:**

PROMAN is web based software which provides users to manage their projects while working remotely. This application provides best features to manage the normal projects and software projects as well and can estimate cost as well. The Application “PROMAN” is developed with hard work and passion by both of the Group members. We also have plans for this project to deploy it in the real environment. Moreover, we have some ideas to modify this project in the future for improvement and evolution purposes. The main motive of this application is to provide the ease and user-friendly environment to users and thus they can work using our application.

# Appendix-A

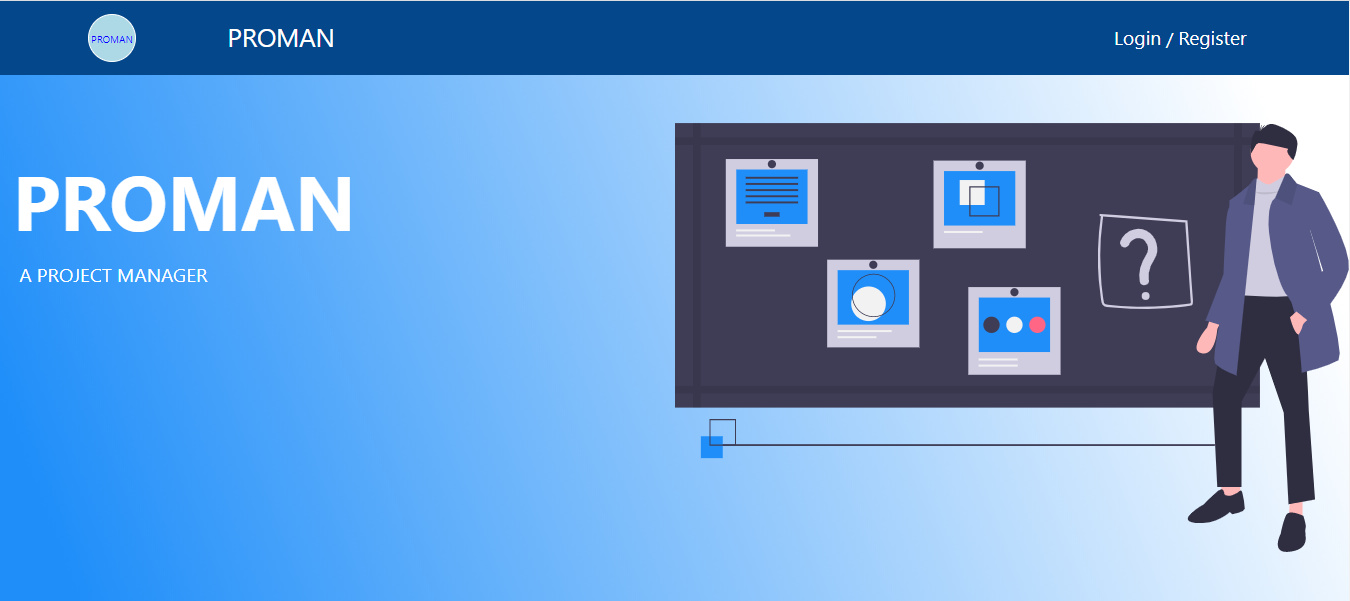
# User Manual

## **A. User manual:**

The user manual contains all necessary information for the user to make full use of the application. This manual includes description of the application features, and step-by step procedures for the use of application.

## **A.1 Public View:**

First page is the homepage where user can see the public view of the website without signing in.

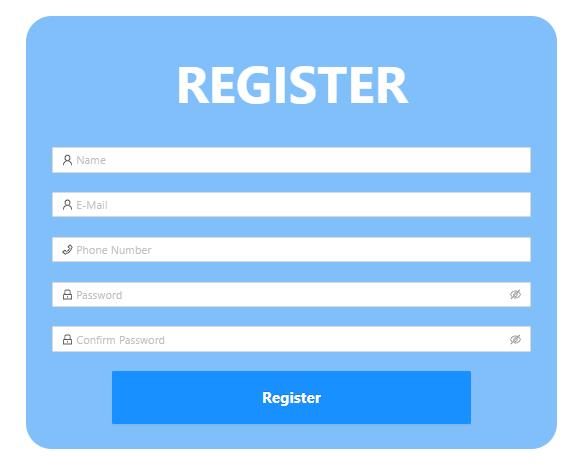


* Figure shows the page for user to select:

1. Click on register button to register.
2. Click on Login button to Login to our system.

## **A.2 Register:**

If you are a new user and you want to use this application and first you have to do is to register to this application by entering your personal details.

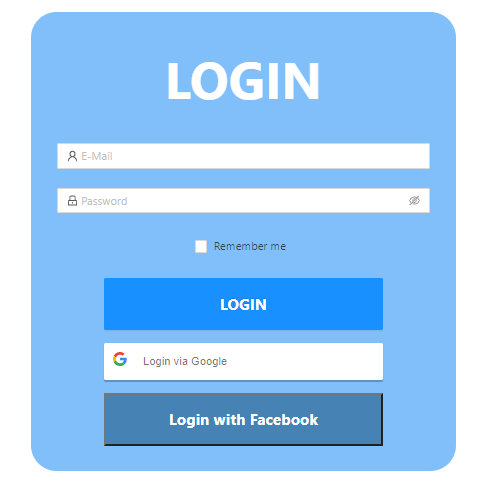


* Do the following procedure to get registered in our application.

1. Fill out the whole information in the above figure and click on register button to get yourself register on our application.
2. Or you can register directly via Google or Facebook by entering your credentials.

## **A.3 Login:**

Login page is used to login after the successful registration.



* Do the following procedure for logging in in our application.

1. Enter your credentials and click on login button and you will be redirected to the next page.
2. Or you can login directly via Google or Facebook by entering your credentials.

## **A.4 Add new Project:**

User can create new project using this Add new Project page.

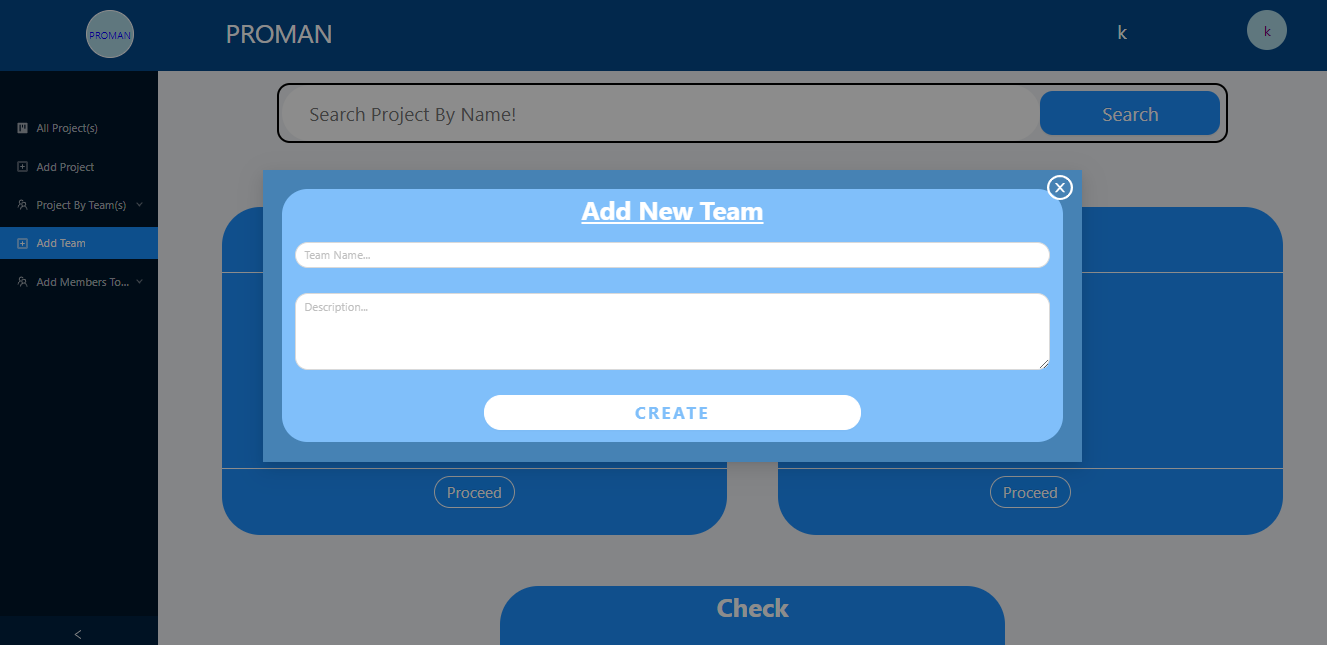


Do the following procedure for Adding project in our application.

1. After logging in you will see this main dashboard page, from there click on add   
   project.
2. Enter details and click on create button and project will be created.

## **A.5 Add new Project:**

User can create new team using this Add new team page.

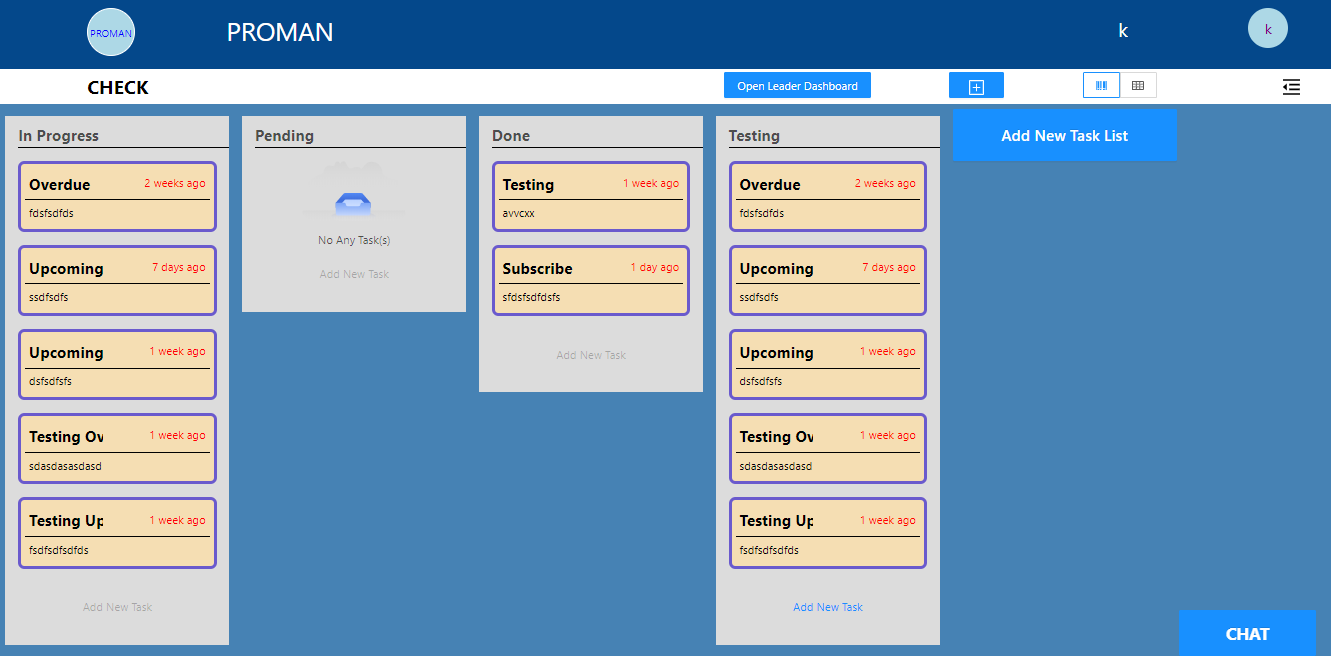


* Do the following procedure for Adding Team in our application.

1. After logging in you will see this main dashboard page, from there click on add   
   team.
2. Enter details and click on create button and new team will be created.

## **A.6 Project Overview:**

User can see the details about project in the following page.

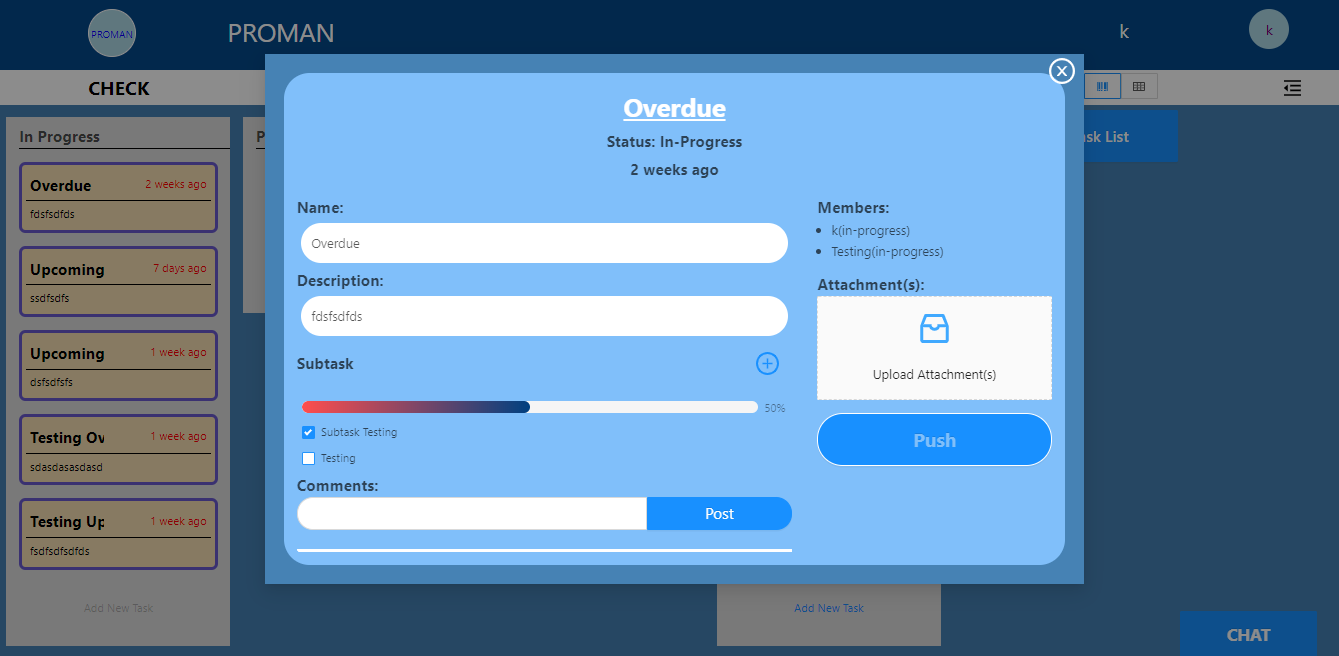


* Do the following procedure to see project information.

1. From dashboard page click on proceed button on the project and you will be   
   redirected to this page where you have all information about project.

## **A.7 Task Overview:**

User can see task detail using following page.

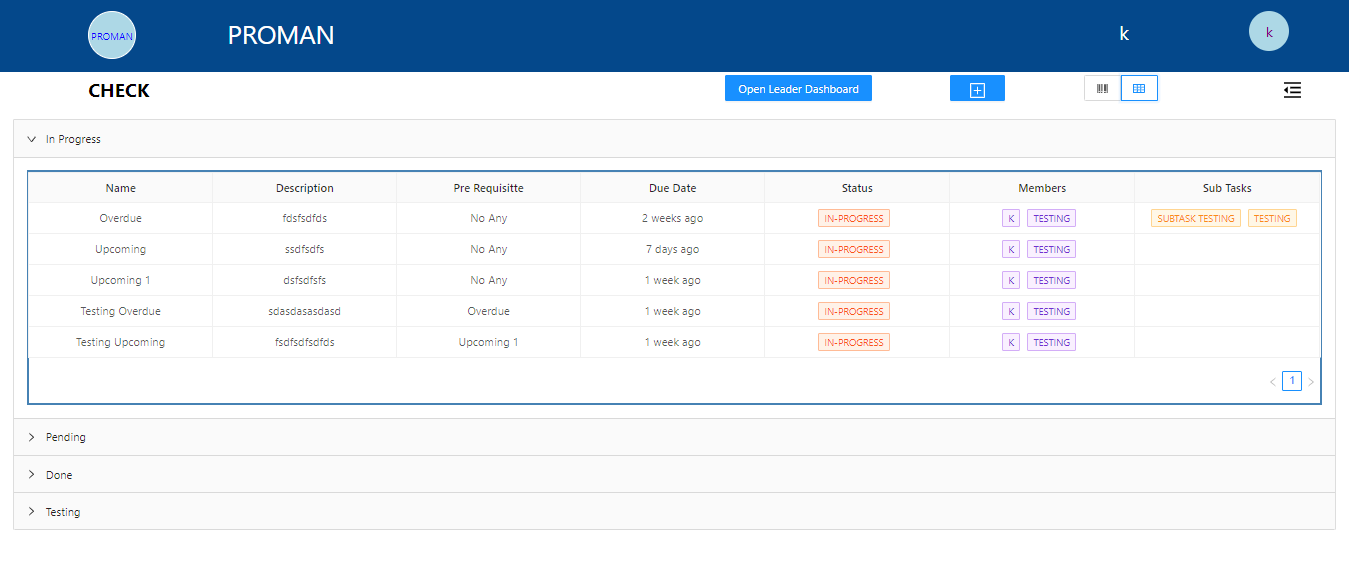


Do the following procedure to see task information.

1. From dashboard page click on proceed button on the project and you will be   
   redirected to this page where you have all information about project, you can click on any task to see its information.

## **A.8 Project overview in table format:**

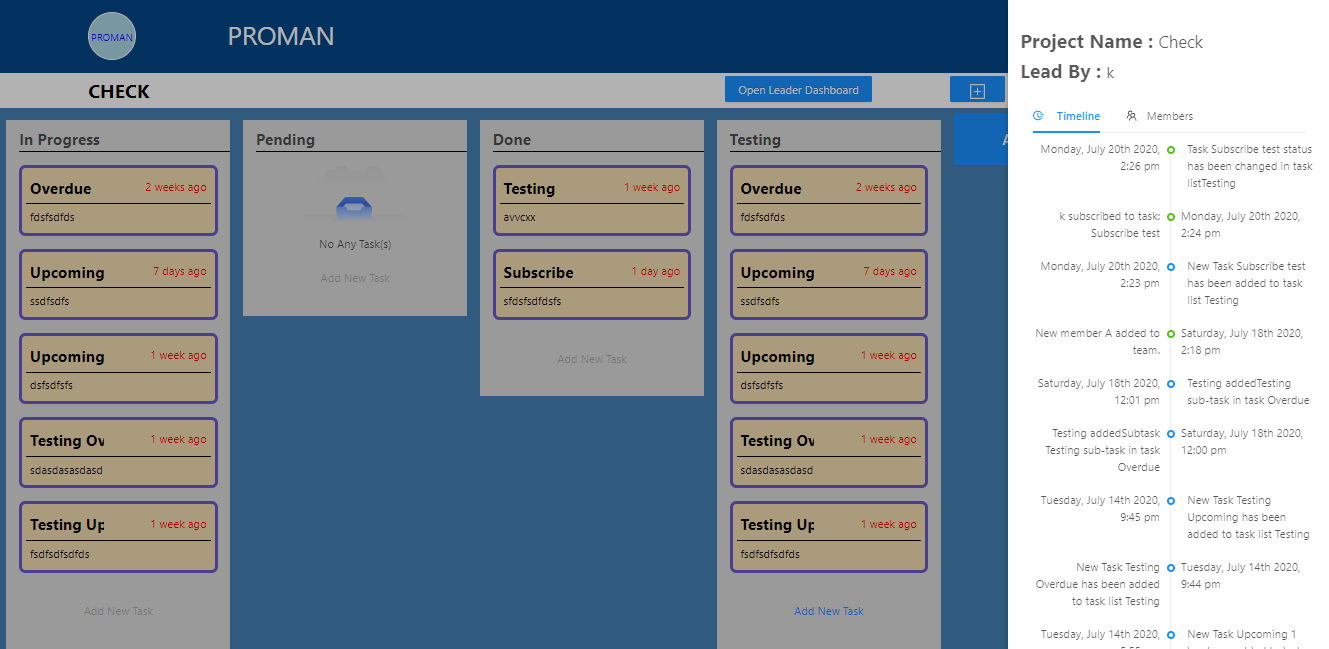
User can view the information about project in the table format as well which is easier to understand.



* Do the following procedure to see the detailed information in table view.

1. From dashboard page click on proceed button on the project and you will be   
   redirected to project information page.
2. From there click on table view button which is right bottom of navbar to see the information in table view.

**A.9 Timeline:**   
 It is just like a notification bar in which User can view the information about what happened to tasks and who did what.



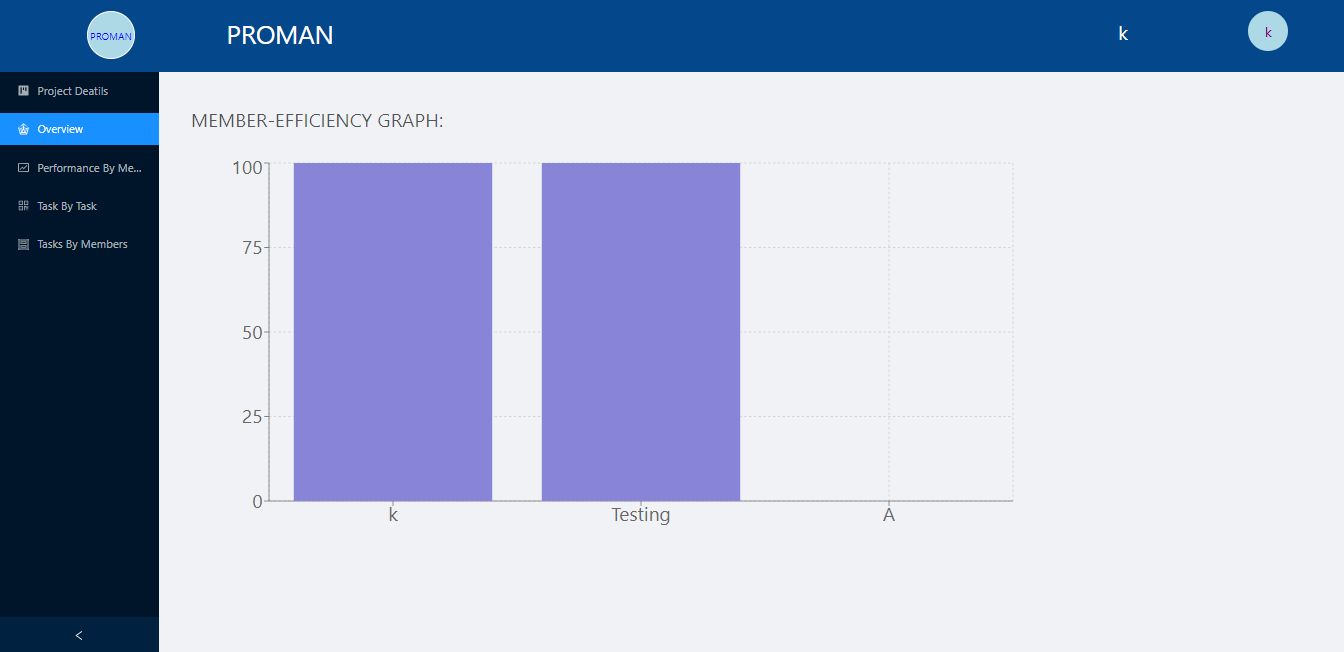
* Do the following procedure to see project information.

1. From dashboard page click on proceed button on the project and you will be   
   redirected to this page where you have all information about project.
2. From there you can click on timeline button which is at right corner below the taskbar to see the recent notifications about project.

## 

## **A.10 Leader Dashboard:**

This is basically an additional tab for Leader where he can get a detailed overview of project.



* Do the following procedure to see project information.

1. From dashboard page click on proceed button on the project and you will be   
   redirected to the project information page.
2. From there you can click leader dashboard button to see this information but for that purpose, you must be the leader of that project.