# GLOBAL SOFTWARE PROJECT MANAGEMENT TOOL

**Emaan Saqib (BCS191039)**



**Spring 2022 Supervised By Hafiz Anas Bilal**

**Department of Computer Science**

### Capital University of Science & Technology, Islamabad

Submission Form for Final-Year

PROJECT REPORT

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | V1.0 | **NUMBER OF MEMBERS** | 1 |
|  |  |  |  |
| **TITLE** | **GLOBAL SOFTWARE PROJECT MANAGEMENT TOOL** | |  |
|  |  | |  |
| **SUPERVISOR NAME** | | **Hafiz Anas Bilal** |  |

|  |  |  |
| --- | --- | --- |
| **MEMBER NAME** | **REG. NO.** | **EMAIL ADDRESS** |
|  |  |  |
| Emaan Saqib | BCS191039 | [Saqibemaan2027@gmail.com](mailto:Saqibemaan2027@gmail.com) |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **MEMBERS’ SIGNATURES** |  | |
|  | |  |
|  | |  |
|  | | **Supervisor’s Signature** |

**APPROVAL CERTIFICATE**

This project, entitled as “GLOBAL SOFTWARE PROJECT MANAGEMENT TOOL” has been approved for the award of

## Bachelors of Science in Computer Science

###### Committee Signatures:

Supervisor:

(Mr. Hafiz Anas Bilal)

Project Coordinator:

(Mr. Abdul Basit)

Head of Department:

(Dr. Nayyer Masood)

## DECLARATION

I/We, hereby, declare that “No portion of the work referred to, in this project has been submitted in support of an application for another degree or qualification of this or any other university/institute or other institution of learning”. It is further declared that this undergraduate project, neither as a whole nor as a part thereof has been copied out from any sources, wherever references have been provided.

**MEMBERS’ SIGNATURES**

## ACKNOWLEGMENT

We are thankful to the Almighty Allah to lend us this chance in life and making us capable of doing it. We also thank our parents, guardians and most specifically Sir Hafiz Anas Bilal for guiding us to the right project and making us enlighten with the knowledge without which we would have been unable to go our Final Year Project.

### Table of Contents

**CHAPTER 1** Error! Bookmark not defined.

[INTRODUCTION 1](#_bookmark0)

* 1. [Project Introduction 1](#_bookmark1)
  2. [Existing Examples / Solutions 3](#_bookmark2)
     1. [BitBucket: 3](#_bookmark3)
     2. [DSpace BU Repository: 3](#_bookmark4)
     3. [Final Year Project Management System: 3](#_bookmark5)
  3. [Comparison of Existing Solutions 5](#_bookmark6)
  4. [Business Scope 5](#_bookmark7)
  5. [Useful Tools and Technology 6](#_bookmark8)
     1. [Languages used: 6](#_bookmark9)
     2. [Visual studio and visual studio code: 6](#_bookmark10)
     3. Asp.net: Error! Bookmark not defined.
     4. [MySQL: 7](#_bookmark11)
     5. [Platform 7](#_bookmark12)
  6. [Project Work Breakdown 8](#_bookmark13)
  7. [Project Timeline 9](#_bookmark14)

[Chapter 2 10](#_bookmark15)

[Requirement specification and analysis 10](#_bookmark16)

* 1. [Functional Requirement: 10](#_bookmark17)
  2. [Non-Functional Requirements 13](#_bookmark18)
  3. [Selected Functional Requirement 13](#_bookmark19)
  4. [System Use Case Modeling 15](#_bookmark20)
  5. [System Sequence diagram 34](#_bookmark21)
  6. [Domain Model 62](#_bookmark22)

[Chapter 3 63](#_bookmark23)

[System Design 63](#_bookmark24)

* 1. [Layer Definition 63](#_bookmark25)
  2. [Software Architecture 64](#_bookmark26)
  3. [Class Diagram 65](#_bookmark27)
  4. [Sequence Diagram 66](#_bookmark28)
  5. Data Flow Diagram Error! Bookmark not defined.
  6. [User Interface Design 69](#_bookmark29)

Tables of figures

List of tables

## CHAPTER 1 INTRODUCTION

### Project Introduction

When working on a new project, both the project manager and the project team lead come across several problems. The first thing they need is digital repository where they can see which similar projects have been done before. Then arises the problem of determining that to what extent that project idea has been implemented. Moreover, there is alsoa problem of project monitoring in which the team leader and the project team are communicating verbally, this can lead to clear misunderstanding and delay in essential tasks. The repositories for maintaining a record of the previous projects already do exist but they don’t offer an all in one solution where you can save, monitor, and check similarity of the project artifacts.

To overcome these problems, we are proposing an all-in-one solution, “Global Software Project Management Tool”. This system will have a projects repository that will be accessible to both team leads and the project team members according to their level of authority and domain. The repositories will hold the following elements of the projects:

* + 1. Project Scope
    2. Functional Requirements
    3. Non-Functional Requirements
    4. UML Artifacts
       1. Use Cases
       2. System Sequence Diagrams (SSDs)
       3. Activity Diagrams
       4. Domain Models
       5. Class Diagrams
    5. Data Models
    6. System Architecture
    7. Code
    8. Test Cases

“GLOBAL SOFTWARE PROJECT MANAGEMENT TOOL” also provides similarity measurement functionality. The system will determine to what extent the idea has been implemented by searching the projects already saved in the repository based on the functional requirements and scope provided by the user. This also leads to reusability of artifacts that have already been implemented. In addition to that, the system acts as a platform for the team leads and the project team members to communicate. The team lead will be able to assign tasks with deadlines and respective documents plus he or she can approve or disapprove the completed tasks. This helps the team lead to monitor the project easily.

Our project consists of following three modules:

1. Project repository
   * Maintain project repository with software artifacts
   * Integration of case tools
2. Project progress monitoring
   * List of tasks
   * Task allocation
   * Task artifacts
   * Monitoring progress
3. Identification of similar and matching artifacts on bases of project scope and functional requirements

This proposed system can also benefit the educational sector as the students and the project supervisors face similar problems while selecting a topic for the Final Year Project (FYP) for their degree. The repositories of the FYP artifacts do already exist but in physical form, if a student is

willing to look upon a project that has been done, he or she may have to go through all the documents available.

### Existing Examples / Solutions

Some existing mobile applications that provide similar functionalities are listed below:

#### BitBucket:



This web application only holds the data of several global users and does not provide any natural language processing techniques to determine how much of the functional requirements and scope of a certain project matches with the given input. And it only provides repositories on a monthly rental basis.

#### DSpace BU Repository:



DSpace is a web application based online repository of fyp for bahria university Islamabad. It is a digital service that collects, preserves, and distributes digital material. It lets project coordinators to evaluate students’ fyp related tasks and the coordinators can also schedule their meetings through it.

#### Final Year Project Management System:

Implemented in Tunku Abdul Rahman University College, Malaysia, The functionality of Final Year Project Management System is to manage students’ final year projects. It assists in managing all the operations of final year project. The Final Year Project Management System consists of 8

features which include User Management, Appointment Making, Students-Supervisor Pairing, Quota & Workload, Project Allocation, Announcement, Assessment and Reports Generation.

### Comparison of Existing Solutions

None of these applications takes functional requirements or scope of the project as an input and compare it to the existing final year projects in the database to see how much of the scope and functional requirements of a similar idea has already been implemented.

The following Table 1.1 shows the comparison between the existing platforms.

*Table 1 comparison of existing solution*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sr.**  **No.** | **Characteristics** |  | **BitBucket** | **DSpace BU Repository** | **Final Year Project Management System** | **Proposed System** |
| **1** | **Repository** |  | ✓ | ✓ | ✓ | ✓ |
| **2** | **Comparison** |  | ❌ | ❌ | ❌ | ✓ |
| **3** | **Category**  **basedsearch projects** |  | ❌ | ❌ | ❌ | ✓ |
| **4** | **Paid** |  | ✓ | ❌ | ❌ | ❌ |

### Business Scope

“Higher education is a $1.9 trillion industry. For comparison, this is how the higher education market compares with other large global markets. Healthcare - $1.7 trillion. Information technology - $3.4 trillion. Transportation - $4.7 trillion.”

The system that we are building is capturing a market that big, we make sure to supply packages

that make it affordable to small educational institutions as well. There is this gap in the educational sector that falls for this system as there is no such platform that provides this functionality is specific.

“Revenue in the Software market is projected to reach US$626.50bn in 2022. The market's largest segment is Enterprise Software with a projected market volume of US$250.30bn in 2022.”

Our Project also reflects in the field of corporate software industry. There are several numbers of software’s and websites that are currently present in the market, but each software holds just one or two of the functionalities that our proposed system offers. Our devised system can be implemented in a form that can also cover the software industry.

### Useful Tools and Technology

In this section, we are describing the tools and technologies that will be used during this project:

#### 1.5.1. Languages used:

Languages used in this project includes, React, html, bootstrap, css, jQuery, angular js, and Asp.net. The following languages are used to make a website and to support these languages following software are used.

#### 1.5.2. File:Visual Studio Code 1.35 icon.svg - Wikimedia Commons Visual studio code:

Visual Studio Code for the Web provides a free, zero-install Microsoft Visual Studio Code experience running entirely in your browser, allowing you to quickly and safely browse source code repositories and make lightweight code changes.

#### FIREBASE:

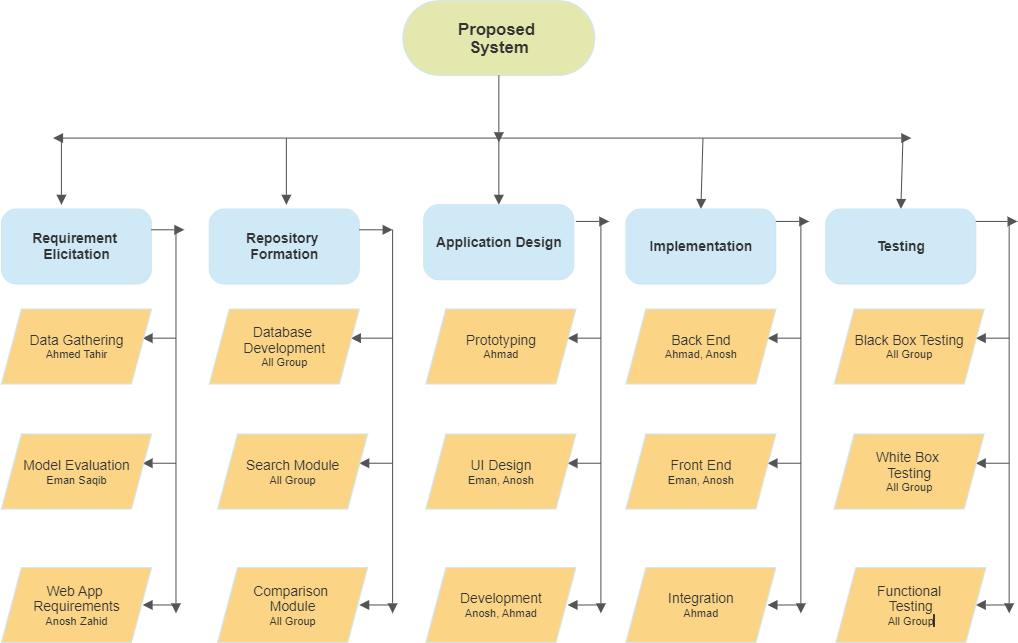
We are using Firebase as database for our project where firebase is an app development platform that helps you build and grow apps and games users love. Backed by Google and trusted by millions of businesses around the world.

#### Platform:

As we talk about the operating system, our project will be web-based application so there will be no constraint with respect to operating system. It will be a web-based platform independent application.

### Project Work Breakdown

In Figure 1 below, work breakdown of the project is shown. The project is broken down into five phases, each phase consisting of its own separate tasks that are allocated to the members of the project according to their expertise.



*Figure 1 work breakdown*

### Project Timeline

In Figure 2, the timeline of this project is shown. All of the tasks that have tobe done are mentioned along with their deadlines and durations.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Timeline** | | | | | | | | | | | | | | | | | | |
|  |  |  |  |  |  | ***Q2 22*** |  |  |  |  | ***Q3 22*** |  |  | ***Q4 22*** |  |  | ***Q1 23*** | |
| ***ID*** | ***Task Name*** |  | ***Start*** | ***Finish*** | ***Duration*** |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  | ***Apr*** | ***Ma*** |  |  | ***Jun*** | ***Jul*** | ***Aug*** | ***Sep*** | ***Oct*** | ***Nov*** |  | ***Dec*** | ***Jan*** | ***Feb*** |
|  |  |  |  |  | **19d** |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **2** | **System Analysis** |  | **4/1/2022** | **4/15/2022** | **11d** |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **3** | **Design & Architecture** |  | **4/18/2022** | **5/27/2022** | **30d** |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **4** | **Implementation** |  | **5/30/2022** | **12/16/2022** | **145d** |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **5** | **Testing** |  | **12/19/2022** | **1/13/2023** | **20d** |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **6** | **Deployment** |  | **1/16/2023** | **1/20/2023** | **5d** |  |  |  |  |  |  |  |  |  |  |  |  |  |

*Figure 2 Project timeline*

***y***

**1 Requirements Management**

**3/15/2022**

**4/8/2022**

## Chapter 2

## Requirement specification and analysis

Requirements analysis is a process of determining user expectations for a new or modified product. These features, called requirements, must be quantifiable, relevant and detailed. In software engineering, such requirements are often called functional specifications. In Chapter 2 we will enlist the functional and non-functional requirements and model functional requirements in the form of use case model.

### Functional Requirement:

Functional requirement define a function of a system or its component. Functional requirement may be calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish.

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. #** | **Functional Requirements** | **Type** | **Status** |
|  | **User Management and Authentication** |  |  |
| 1. | The system administrator shall be able to login to the  System using username and password. | core | completed |
| 2. | The system administrator shall be able to create users  ccount into the system. | core | completed |
| 3 | The system administrator shall be able to activate or deactivate users into the system.(not the ceo) |  | completed |
| 4 | The system administrator shall be able to view all users. |  | completed |
| 5 | Users shall be able to login to the system using their username and password. |  | completed |
| 6 | User shall be able to change their passwords |  | completed |
| 7 | User shall be able to view their profiles. |  | completed |
| 8 | User shall be able to update their profiles. |  | completed |
| 9 | CEO shall be able to view profiles of Project manager, and team members. |  | completed |
| 10 | CEO shall be able to view list of Project manager, and team members. |  | completed |
| 11 | The Project manager shall be able to view profiles of team members. |  | completed |
|  | **Project Management and Monitoring** |  |  |
| 12 | The Project Manager shall be able to create new projects. |  | completed |
| 13 | The Project Manager shall be able to assign team members to different projects. |  | completed |
| 14 | The Project Manager shall be able to view projects. |  | completed |
| 15 | The Project Manager shall be able to delete projects. |  | completed |
| 16 | The Project Manager shall be able to update projects. |  | completed |
| 17 | The Project Manager shall be able to assign tasks to team members. |  | completed |
| 18 | The Project Manager shall be able to view current status of project. |  | completed |
| 19 | The Project Manager shall be able to view current status of the tasks. |  | completed |
| 20 | CEO shall be able to view projects/ and their members. |  | completed |
| 21 | CEO shall be able to view project status. |  | completed |
| 22 | Team Member shall be able to view assigned tasks and deadlines of those tasks. |  | completed |
| 23 | Team member shall be able to submit completed tasks to the Project manager. |  | completed |
| 24 | The Project manager shall be able to approve completed projects. |  | completed |
| 25 | The Project manager shall be able to disapprove completed projects that will be sent back to team member. |  | completed |

### Non-Functional Requirements

A non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. They are contrasted with functional requirements that define specific behavior or functions.

*Table 2 nonfunctional requirements*

|  |  |  |
| --- | --- | --- |
| **SR.** | **NON FUNCTIONAL REQUIRMENT** | **CATEGORY** |
| 1 | The system shall lock itself for 60 seconds in case of invalid inputs by the user. | Efficiency |
| 2 | Users shall be able to log into the system only by using their correct login credentials. | Usability |
| 3 | System administer shall be able to reset users’ password in case they fail to preserve  it. | Security |
| 4 | The system shall offer functionality for data backup. | Maintainability |
| 5 | The system shall be available at all times. | Efficiency |
| 6 | System shall have online connectivity 24/7. | Availability |
| 7 | The web application shall have help pop-ups built into the interface to help the users. | Availability |
| 8 | System shall be able to bear workload of more than 10,000 active users. | Availability |
| 9 | System shall be able to recover in 20 to 30 seconds in case of application crash. |  |
| 10 | Website shall take approximately 5 to 10 seconds in searching projects in repositories and giving desired output. | Efficiency |
| 11 | System shall be highly responsive to user inputs. | Efficiency |

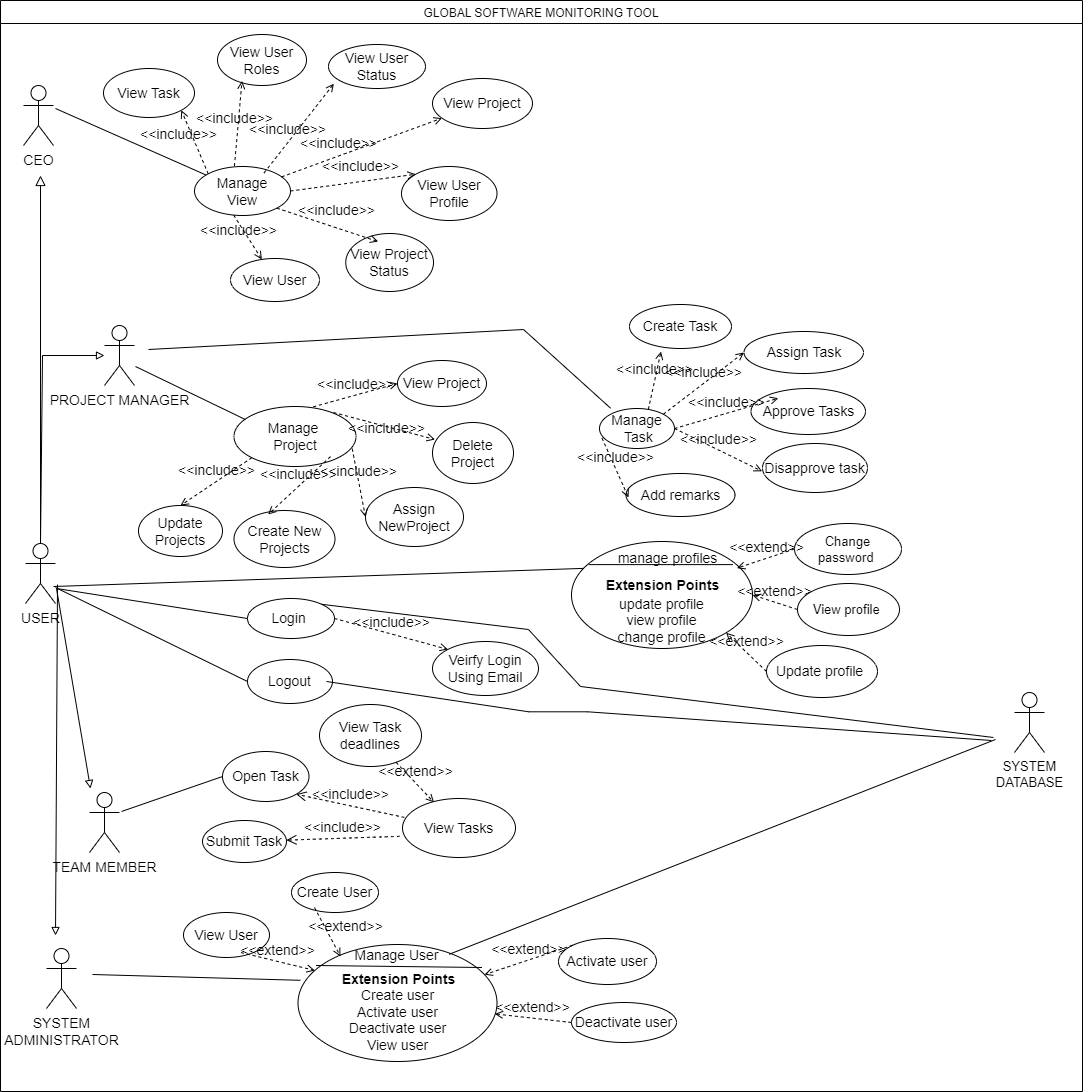
### Selected Functional Requirement

Following is the list of the requirements selected for the current iteration.

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. #** | **Functional Requirements** | **Type** | **Status** |
|  | **User Management and Authentication** |  |  |
| 1. | The system administrator shall be able to login to the  system using username and password. | core | completed |
| 2. | The system administrator shall be able to create users  account into the system. | core | completed |
| 3 | The system administrator shall be able to activate or deactivate users into the system.(not the ceo) |  | completed |
| 4 | The system administrator shall be able to view all users. |  | completed |
| 5 | Users shall be able to login to the system using their username and password. |  | completed |
| 6 | User shall be able to change their passwords |  | completed |
| 7 | User shall be able to view their profiles. |  | completed |
| 8 | User shall be able to update their profiles. |  | completed |
| 9 | CEO shall be able to view profiles of Project manager, and team members. |  | completed |
| 10 | CEO shall be able to view list of Project manager, and team members. |  | completed |
| 11 | The Project manager shall be able to view profiles of team members. |  | completed |
|  | **Project Management and Monitoring** |  |  |
| 12 | The Project Manager shall be able to create new projects. |  | completed |
| 13 | The Project Manager shall be able to assign team members to different projects. |  | completed |
| 14 | The Project Manager shall be able to view projects. |  | completed |
| 15 | The Project Manager shall be able to delete projects. |  | completed |
| 16 | The Project Manager shall be able to update projects. |  | completed |
| 17 | The Project Manager shall be able to assign tasks to team members. |  | completed |
| 18 | The Project Manager shall be able to view current status of project. |  | completed |
| 19 | The Project Manager shall be able to view current status of the tasks. |  | completed |
| 20 | CEO shall be able to view projects/ and their members. |  | completed |
| 21 | CEO shall be able to view project status. |  | completed |
| 22 | Team Member shall be able to view assigned tasks and deadlines of those tasks. |  | completed |
| 23 | Team member shall be able to submit completed tasks to the Project manager. |  | completed |
| 24 | The Project manager shall be able to approve completed tasks. |  | completed |
| 25 | The Project manager shall be able to disapprove completed tasks that will be sent back to team member. |  | completed |

### System Use Case Modeling

A use case is a list of actions or event steps, typically defining the interactions between a role (known in the Global Software Project Management Tool as an actor) and a system, to achieve a goal. The actor can be a human or other external system, use cases are shown in the following the figure.



*Figure 3 system use case diagram*

#### 2.5. Use Cases

Usecase-1: The system administrator shall be able to login to the system using username and password.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | UC.1 | | | |
| Use Case Name | Login System Administrator | | | |
| Created by: | Emaan Saqib | Last updated by: | | - |
| Created date: | 11/10/2022 | Last revision date: | | 18/10/2022 |
| Actor: | System Administrator | | | |
| Description : | The System Administrator will login into the system using his/her credentials i.e., username and password. | | | |
| Trigger: | Click Login Button | | | |
| Preconditions: | 1. The login page must be displayed. 2. The system administrator must be registered. 3. The system administrator has an active and working account. | | | |
| Post conditions: | 1. The system shows a successful login message on the same page. The system redirects the user to system administrator dashboard. | | | |
| Normal flow: | Actor | | System | |
| **1**. The system administrator clicks on login button.   1. The system administrator enters his/her username and password. 2. The system administrator presses the login button. | | **2**. The System directs user to log inform to fill in his credentials i.e., username and password.  **5**. The System validates the username and password from the database and redirects user to system administrator dashboard. | |
| Alternative flows: | There is no specific alternate flow to this use case. | | | |
| Exceptions: | Exception (wrong username or password) is raised in case of wrong credential entry by the system administrator. | | | |

Usecase-2: The system administrator shall be able to create users account into the system.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | UC.2 | | | |
| Use Case Name | Create User Account | | | |
| Created by: | Emaan Saqib | Last updated by: | | - |
| Created date: | 11/10/2022 | Last revision date: | | 18/10/2022 |
| Actor: | System Administrator | | | |
| Description : | The System Administrator shall be able to create user accounts using User ID, their role, and their basic information such as their first and last names, email, field of expertise | | | |
| Trigger: | Add User Button | | | |
| Preconditions: | The system administrator must be logged in. | | | |
| Post conditions: | 1. The system shows a user successfully created message. 2. The system sends username and password to the user via email. 3. The system adds the user added in the users list. | | | |
| Normal flow: | Actor | | System | |
| **1**. The system administrator clicks on add user button.   1. The system administrator enters the User ID, role, and first and last names, email, field of expertise. 2. The system administrator clicks on submit button on form. | | 1. The System directs user to create user page. 2. The System opens a form to fill out all the information required for creating a new user i.e., User ID, role, and first and last names, email, field of expertise.   **6** .The System generates a random 6 digit password against the entered member id.   1. The System sends user ID and password to the created via email. 2. The System shows (successfully added user) message. | |
| Alternative flows: | There is no specific alternate flow to this use case. | | | |

|  |  |
| --- | --- |
| Exceptions: | The system administrator misses out any one of the information boxes then the system raises an alert message to fill out all the information fields before submitting the form. |

Usecase-3: The system administrator shall be able to activate or deactivate users into the system.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | UC.3 | | | |
| Use Case Name | Activate or Deactivate User Account | | | |
| Created by: | Emaan Saqib | Last updated by: | | - |
| Created date: | 11/10/2022 | Last revision date: | | 18/10/2022 |
| Actor: | System Administrator | | | |
| Description : | The System Administrator shall be able to Activate or Deactivate the user accounts. | | | |
| Trigger: | Activate User Button/Deactivate User Button | | | |
| Preconditions: | User should be active or deactivate. | | | |
| Post conditions: | User status will be changed | | | |
| Normal flow: | Actor | | System | |
| **1**. The system administrator clicks on Activate User button if the user is deactivated or The system administrator clicks on Deactivate User button if the user is activated. | | **2**. The System will change the status of the user and display on the screen. | |
| Alternative flows: | There is no specific alternate flow to this use case. | | | |
| Exceptions: | N/A | | | |

Usecase-4: The system administrator shall be able to view all users

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | UC.4 | | | |
| Use Case Name | View Users | | | |
| Created by: | Emaan Saqib | Last updated by: | | - |
| Created date: | 11/10/2022 | Last revision date: | | 18/10/2022 |
| Actor: | System Administrator | | | |
| Description : | The System Administrator shall be able to all users that are registered into the system by system administrator. | | | |
| Trigger: | Users button | | | |
| Preconditions: | 1. The system administrator must be logged in. 2. Users should be registered. | | | |
| Post conditions: | 1. Users will be shown to system administrator. | | | |
| Normal flow: | Actor | | System | |
| **1**. The system administrator clicks on Users button. | | 1. The System directs admin to all users page. 2. The System will show all users that are registered to the system. | |
| Alternative flows: | There is no specific alternate flow to this use case. | | | |
| Exceptions: | N/A | | | |

Usecase-5: Users shall be able to login to the system using their username and password.

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC.5 | | |
| Use Case Name | Login User | | |
| Created by: | Emaan Saqib | Last updated by: | - |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | |  |
| Created date: | 11/10/2022 | Last revision date: | | 18/10/2022 |
| Actor: | CEO / Project Manager / Team Lead / Team Member | | | |
| Description : | The User will login into the system using his/her credentials i.e., username and  password. | | | |
| Trigger: | Click Login Button | | | |
| Preconditions: | 1. The login page must be displayed. 2. The user (Project Manager / Team Lead / Team Member) must be added by the system administrator. 3. The system has sent the credentials to the user via email. 4. The user (Project Manager / Team Lead / Team Member) must have an active and working account. | | | |
| Post conditions: | 1. The system shows a successful login message on the same page. 2. The system redirects the user to the member dashboard. | | | |
| Normal flow: | Actor | | System | |
| **1**. The User (Project Manager / Team Lead / Team Member) clicks on login button.   1. The User (Project Manager / Team Lead / Team Member) enters his/her username and password. 2. The User (Project Manager / Team Lead / Team Member) clicks the login button. | | **2**. The System directs user (Project Manager / Team Lead / Team Member) to login form.  **5**. The System validates the username and password from the database and redirects user (Project Manager / Team Lead / Team Member) to member dashboard. | |
| Alternative flows: | There is no specific alternate flow to this use case. | | | |
| Exceptions: | Exception (wrong username or password) is raised in case of wrong credential  entry by the user (Project Manager / Team Lead / Team Member). | | | |

Usecase-6: User shall be able to change their passwords.

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC.6 | | |
| Use Case Name | Change User’s Password | | |
| Created by: | Emaan Saqib | Last updated by: | - |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | |  |
| Created date: | 11/10/2022 | Last revision date: | | 18/10/2022 |
| Actor: | CEO / Project Manager / Team Lead / Team Member | | | |
| Description : | The user (Project Manager / Team Lead / Team Member) will change his  account password. | | | |
| Trigger: | Click Change Password Button | | | |
| Preconditions: | 1. The system must have online connectivity. 2. The user (Project Manager / Team Lead / Team Member) must be logged in. 3. The user (Project Manager / Team Lead / Team Member) must be on account settings. | | | |
| Post conditions: | 1. The system shows a “password successfully changed” message on the same page. 2. The system redirects the user (Project Manager / Team Lead / Team Member) to the login page. | | | |
| Normal flow: | Actor | | System | |
| **1**. The user (Project Manager / Team Lead / Team Member) clicks on change password button.   1. The user (Project Manager / Team Lead / Team Member) enters existing password, new password, confirm new password. 2. The user (Project Manager / Team Lead / Team Member) presses the submit button. | | **2**. The System directs user (Project Manager / Team Lead / Team Member) to change password form   1. The System validates if the user (Project Manager / Team Lead / Team Member) has entered correct existing password and new password matches confirm new password. 2. The system then displays “password successfully changed” message and redirects user (Project Manager / Team Lead / Team Member) to the home page. | |
| Alternative flows: | The user can click on forgot password on the login page and change his password from there with the same procedure. | | | |
| Exceptions: | Exception (Enter correct existing password/ Enter matching new password) is raised in case the user (Project Manager / Team Lead / Team Member) enters wrong existing password or wrong matching password. | | | |

Usecase-7: User shall be able to view their profiles.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | UC.7 | | | |
| Use Case Name | User viewing Profile | | | |
| Created by: | Emaan Saqib | Last updated by: | | - |
| Created date: | 11/10/2022 | Last revision date: | | 18/10/2022 |
| Actor: | CEO / Project Manager / Team Lead / Team Member | | | |
| Description : | The user (Project Manager / Team Lead / Team Member) will view their user profile. | | | |
| Trigger: | Click View profile Button | | | |
| Preconditions: | 1. The system must have online connectivity. 2. The user (Project Manager / Team Lead / Team Member) must be logged in. 3. The user (Project Manager / Team Lead / Team Member) must have his home page displayed on the screen. | | | |
| Post conditions: | The system shows a graphical table with all the details of the user (Project Manager / Team Lead / Team Member) on the screen. | | | |
| Normal flow: | Actor | | System | |
| **1**. The user (Project Manager / Team Lead / Team Member) clicks on profile button.  **3**. The user clicks on view profile | | **2**. The System shows a dropdown menu of the options of profile.  **4**. The System displays user (Project Manager / Team Lead / Team Member) a graphical table with his/her details. | |
| Alternative flows: | There is no alternative flow to this use case. | | | |
| Exceptions: | N/A | | | |

Usecase-8: User shall be able to update their profiles.

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC.8 | | |
| Use Case Name | Update User Profile | | |
| Created by: | Emaan Saqib | Last updated by: | - |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | |  |
| Created date: | 11/10/2022 | Last revision date: | | 18/10/2022 |
| Actor: | CEO / Project Manager / Team Lead / Team Member | | | |
| Description : | The user (CEO/ Project Manager / Team Lead / Team Member) will update  information in their user profile. | | | |
| Trigger: | Click Update profile Button | | | |
| Preconditions: | 1. The system must have online connectivity. 2. The user (Project Manager / Team Lead / Team Member) must be logged in. 3. The user (Project Manager / Team Lead / Team Member) must have his home page displayed on the screen. | | | |
| Post conditions: | 1. The system shows a “profile successfully updated” message on the same page. 2. The system redirects the user (Project Manager / Team Lead / Team Member) to the profile page. | | | |
| Normal flow: | Actor | | System | |
| **1**. The user (Project Manager / Team Lead / Team Member) clicks on profile button.  **3**. The User clicks on update profile button.  **5**. The user (Project Manager / Team Lead / Team Member) updates the fields as required and presses the submit button. | | **2**. The System shows a dropdown menu of the options of profile.  **4**. The System directs user (Project Manager / Team Lead / Team Member) to profile information form.   1. The System updates the user profile in the database. 2. The system then displays “profile successfully updated” message and redirects user (Project Manager / Team Lead / Team Member) to the home page. | |
| Alternative flows: | There is no alternative flow to this use case. | | | |
| Exceptions: | Exception (fill all the information fields) is raised in case the user (Project Manager / Team Lead / Team Member) leaves one of the information field boxes empty and clicks on submit button. | | | |

Usecase-9: CEO shall be able to view profiles of Project manager, team lead, and team members.

Use Case ID: UC.9

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | | | |
| Use Case Name | CEO view profiles | | | |
| Created by: | Emaan Saqib | Last updated by: | | - |
| Created date: | 11/10/2022 | Last revision date: | | 18/10/2022 |
| Actor: | CEO | | | |
| Description : | CEO shall be able to view profiles of Project manager, team lead, and team members. | | | |
| Trigger: | Click View profile Button | | | |
| Preconditions: | 1. The system must have online connectivity. 2. CEO must be logged in. 3. The CEOmust be on view team page. | | | |
| Post conditions: | The system shows user profile information page. | | | |
| Normal flow: | Actor | | System | |
| **1**. CEO clicks on profile button. | | **2**. The System shows the profile of the user. | |
| Alternative flows: | There is no alternative flow to this use case. | | | |
| Exceptions: | N/A | | | |

Usecase-10: CEO shall be able to view list of Project manager, team lead, and team members.

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC.10 | | |
| Use Case Name | CEO view profiles | | |
| Created by: | Emaan Saqib | Last updated by: | - |
| Created date: | 11/10/2022 | Last revision date: | 18/10/2022 |
| Actor: | CEO | | |

|  |  |  |
| --- | --- | --- |
|  |  | |
| Description : | CEO shall be able to view profiles of Project manager, team lead, and team members. | |
| Trigger: | Click View profile Button | |
| Preconditions: | 1. The system must have online connectivity. 2. CEO must be logged in. 3. The CEO must be on view team page. | |
| Post conditions: | The system shows user profile information page. | |
| Normal flow: | Actor | System |
| 1. CEO clicks on the user’s button. 2. CEO clicks on project management button. | **3**. The System shows all the projects information to the user. |
| Alternative flows: | There is no alternative flow to this use case. | |
| Exceptions: | N/A | |

Usecase-11: The Project manager shall be able to view profiles of team lead and team member.

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC.11 | | |
| Use Case Name | Project Manager view profiles | | |
| Created by: | Emaan Saqib | Last updated by: | - |
| Created date: | 11/10/2022 | Last revision date: | 18/10/2022 |
| Actor: | Project Manager | | |
| Description : | Project manager shall be able to view profiles of team lead, and team members. | | |
| Trigger: | Click View profile Button | | |
| Preconditions: | 1. The system must have online connectivity. | | |

|  |  |  |
| --- | --- | --- |
|  | 1. The Project Manager must be logged in. 2. The Project Manager must be on view team page. | |
| Post conditions: | The system shows user profile information page. | |
| Normal flow: | Actor | System |
| 1. Project Manager clicks on the user’s button. 2. Project Manager clicks on profile button. | **3**. The System shows the profile of the user. |
| Alternative flows: | There is no alternative flow to this use case. | |
| Exceptions: | N/A | |

Usecase-12: Team leads shall be able view their team member’s profile.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | UC.12 | | | |
| Use Case Name | Team lead view profiles | | | |
| Created by: | Emaan Saqib | Last updated by: | | - |
| Created date: | 11/10/2022 | Last revision date: | | 18/10/2022 |
| Actor: | Team lead | | | |
| Description : | Team leads shall be able view their team member’s profile. | | | |
| Trigger: | Click View profile Button | | | |
| Preconditions: | 1. The system must have online connectivity. 2. The team lead must be logged in. 3. The team lead must be on view team page. | | | |
| Post conditions: | The system shows user profile information page. | | | |
| Normal flow: | Actor | | System | |
| **1.** Team lead clicks on the user’s  button. | | **3**. The System shows the profile of the  user. | |

|  |  |  |
| --- | --- | --- |
|  | **2**. Team lead clicks on profile button. |  |
| Alternative flows: | There is no alternative flow to this use case. | |
| Exceptions: | N/A | |

Usecase-13: The Project Manager shall be able to create new projects.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | UC.13 | | | |
| Use Case Name | Create New Project | | | |
| Created by: | Emaan Saqib | Last updated by: | | - |
| Created date: | 11/10/2022 | Last revision date: | | 18/10/2022 |
| Actor: | Project Manager | | | |
| Description : | The Project Manager shall be able to create new projects. | | | |
| Trigger: | Add Project Button | | | |
| Preconditions: | The Project Manager must be logged in. | | | |
| Post conditions: | 1. The system shows a user successfully created message. 2. The system sends username and password to the user via email. 3. The system adds the user added in the users list. | | | |
| Normal flow: | Actor | | System | |
| **1**. The Project Manager clicks on add project button.   1. The Project Manager enters the project name, description, start date, end date, functional requirements and members. 2. The system administrator clicks on submit button on form. | | 1. The System directs user to create project page. 2. The System displays the form of new project.   **6** .The System create the project in the database. | |

|  |  |
| --- | --- |
| Alternative flows: | There is no specific alternate flow to this use case. |
| Exceptions: | The Project manager misses out any one of the information boxes then the  system raises an alert message to fill out all the information fields before submitting the form. |

Usecase-14: The Project Manager shall be able to view projects.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | UC.14 | | | |
| Use Case Name | Project Manager view projects | | | |
| Created by: | Emaan Saqib | Last updated by: | | - |
| Created date: | 11/10/2022 | Last revision date: | | 18/10/2022 |
| Actor: | Project Manager | | | |
| Description : | The Project Manager shall be able to view projects | | | |
| Trigger: | Project management button | | | |
| Preconditions: | 1. The system must have online connectivity. 2. The Project Manager must be logged in. | | | |
| Post conditions: | The system shows projects. | | | |
| Normal flow: | Actor | | System | |
| 1. Project Manager clicks on the project management button. 2. Project Manager clicks on view projects. | | **3**. The System shows the projects list. | |
| Alternative flows: | There is no alternative flow to this use case. | | | |
| Exceptions: | N/A | | | |

Usecase-15: The Project Manager shall be able to delete projects.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | UC.15 | | | |
| Use Case Name | Project Manager delete projects | | | |
| Created by: | Emaan Saqib | Last updated by: | | - |
| Created date: | 11/10/2022 | Last revision date: | | 18/10/2022 |
| Actor: | Project Manager | | | |
| Description : | The Project Manager shall be able to delete projects | | | |
| Trigger: | Project management button | | | |
| Preconditions: | 1. The system must have online connectivity. 2. The Project Manager must be logged in. | | | |
| Post conditions: | The system deletes the projects. | | | |
| Normal flow: | Actor | | System | |
| **1.** Project Manager clicks on the project management button.  **3**. Project Manager clicks on delete project button. | | **2**. The System shows the projects list.  **4.** The System will delete projects from the database. | |
| Alternative flows: | There is no alternative flow to this use case. | | | |
| Exceptions: | N/A | | | |

Usecase-16: The Project Manager shall be able to update projects.

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC.16 | | |
| Use Case Name | Update Projects | | |
| Created by: | Emaan Saqib | Last updated by: | - |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Created date: | 11/10/2022 | Last revision date: | | 18/10/2022 |
| Actor: | Project Manager | | | |
| Description : | The Project Manager shall be able to update projects | | | |
| Trigger: | Update Button | | | |
| Preconditions: | 1. The system must have online connectivity. 2. The Project Manager must be logged in. | | | |
| Post conditions: | 1. The system shows “updated successfully” message. 2. The system updates the projects in the database. | | | |
| Normal flow: | Actor | | System | |
| **1.** Project Manager clicks on the project management button.   1. Project Manager clicks on Update Button. 2. Project manager will enter the new information. 3. Project manager will click on save button. | | **2**. The System shows the projects list.  **6.** The System updates the project in the database. | |
| Alternative flows: | There is no alternative flow to this use case. | | | |
| Exceptions: | N/A | | | |

Usecase-17: The Project Manager shall be able to create teams for project.

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC.17 | | |
| Use Case Name | Create Teams | | |
| Created by: | Emaan Saqib | Last updated by: | - |
| Created date: | 11/10/2022 | Last revision date: | 18/10/2022 |
| Actor: | Project Manager | | |

|  |  |  |
| --- | --- | --- |
| Description : | The Project Manager shall be able to create teams for project. | |
| Trigger: | Create Team Button | |
| Preconditions: | 1. The system must have online connectivity. 2. The Project Manager must be logged in. | |
| Post conditions: | Teams will be created. | |
| Normal flow: | Actor | System |
| 1. Project Manager clicks on the Team management button. 2. Project manager will click on create team button. 3. Project manager fill the information. 4. Project manager will click on save button. | **3**. The System shows the create team form.  **6.** The System creates team in the database. |
| Alternative flows: | There is no alternative flow to this use case. | |
| Exceptions: | N/A | |

Usecase-18: The Project Manager shall be able to update or delete teams of projects.

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC.18 | | |
| Use Case Name | Update Teams or Delete Teams | | |
| Created by: | Emaan Saqib | Last updated by: | - |
| Created date: | 11/10/2022 | Last revision date: | 18/10/2022 |
| Actor: | Project Manager | | |
| Description : | The Project Manager shall be able to update or delete teams of projects. | | |
| Trigger: | Update Button and Delete Button | | |
| Preconditions: | 1. The system must have online connectivity. | | |

|  |  |  |
| --- | --- | --- |
|  | 2. The Project Manager must be logged in. | |
| Post conditions: | Teams will be updated or deleted. | |
| Normal flow: | Actor | System |
| **1.** Project Manager clicks on the Team management button.  **3**. Project manager will click on update or delete team button. | 1. The System shows teams list. 2. The System will update or delete the project from the database. |
| Alternative flows: | There is no alternative flow to this use case. | |
| Exceptions: | N/A | |

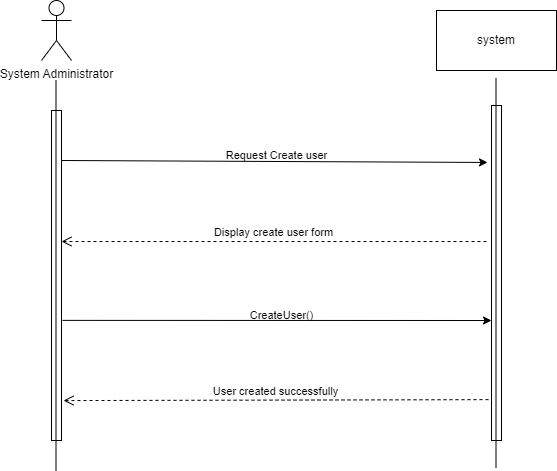
Usecase-19:

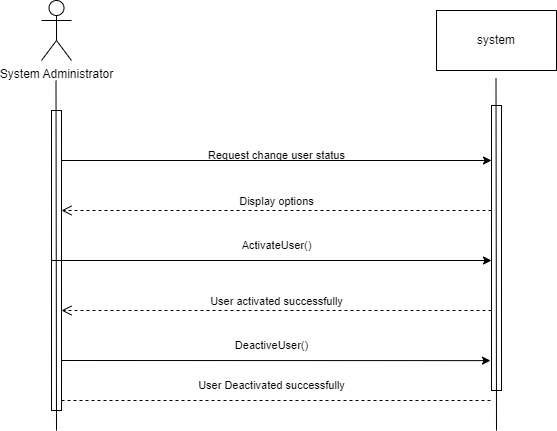
The Project Manager shall be able to view teams.

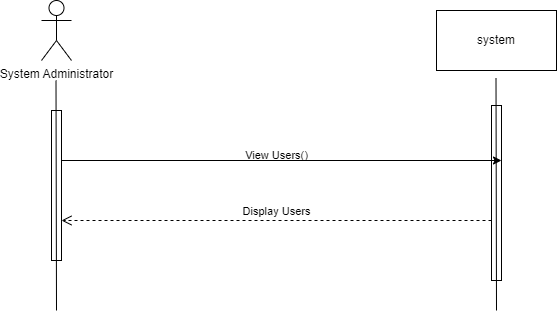
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | UC.19 | | | |
| Use Case Name | View Teams | | | |
| Created by: | Emaan Saqib | Last updated by: | | - |
| Created date: | 11/10/2022 | Last revision date: | | 18/10/2022 |
| Actor: | Project Manager | | | |
| Description : | The Project Manager shall be able to view teams. | | | |
| Trigger: | View Button | | | |
| Preconditions: | 1. The system must have online connectivity. 2. The Project Manager must be logged in. | | | |
| Post conditions: | Teams will be shown. | | | |
| Normal flow: | Actor | | System | |
| **1.** Project Manager clicks on the  Team management button. | | **2.** The System shows teams list. | |

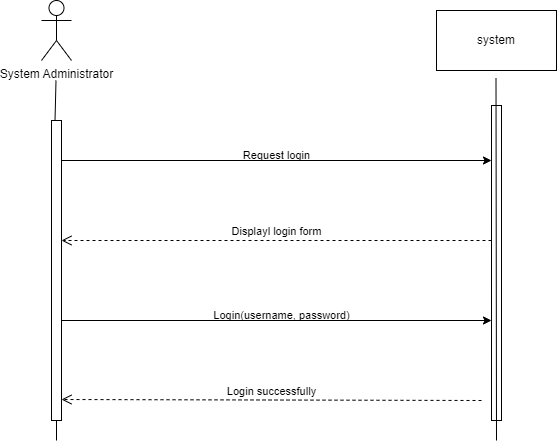
|  |  |  |
| --- | --- | --- |
|  |  | **3.** The System will display all the teams created. |
| Alternative flows: | There is no alternative flow to this use case. | |
| Exceptions: | N/A | |

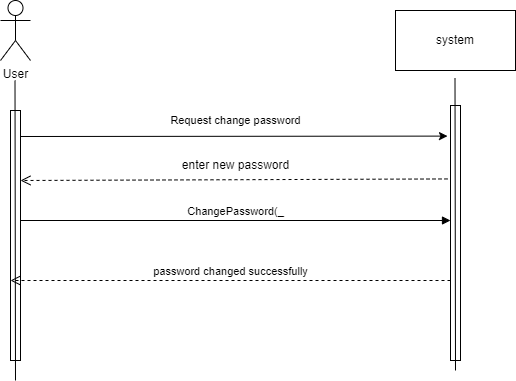
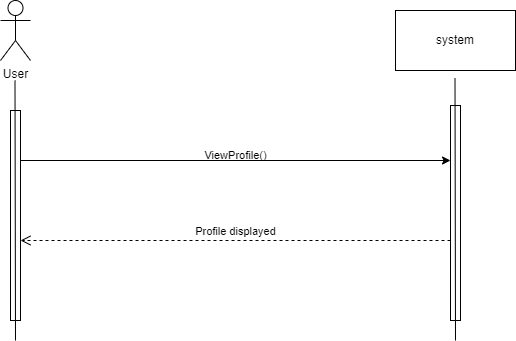
### System Sequence diagram

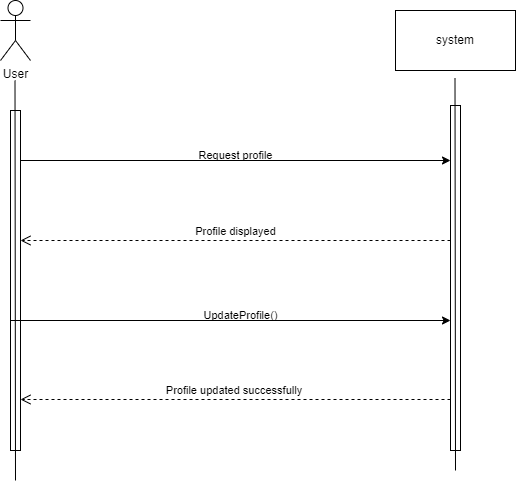


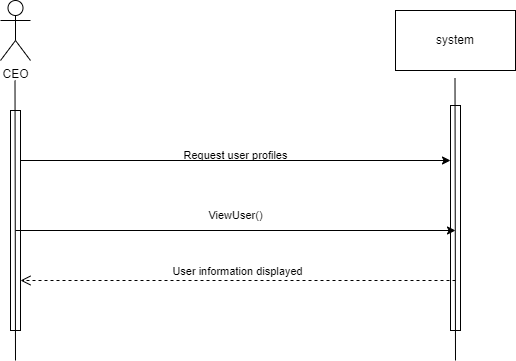
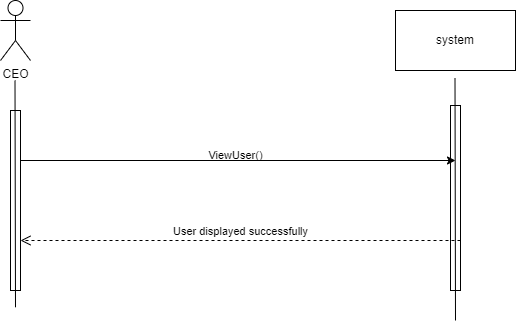


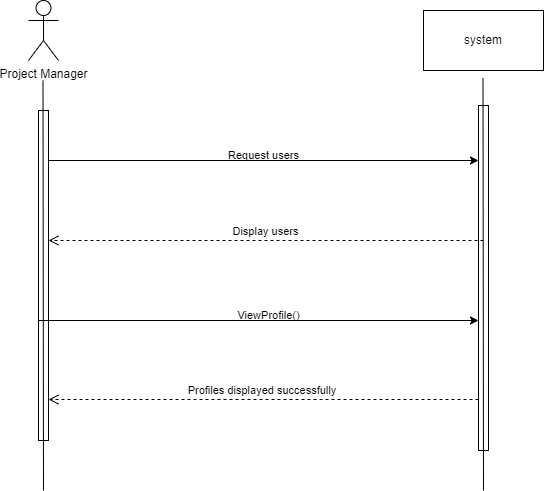


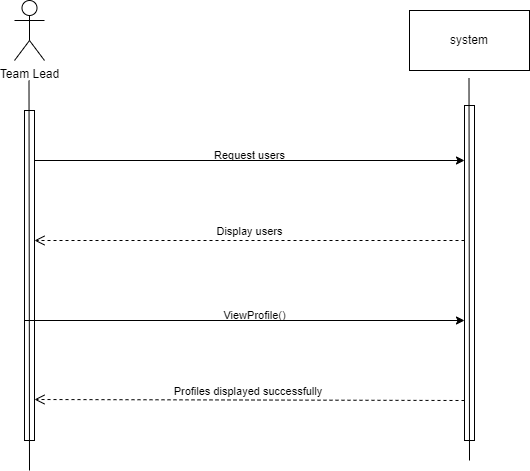


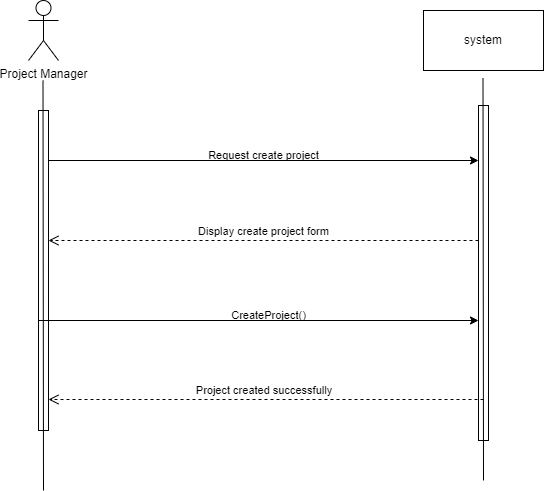


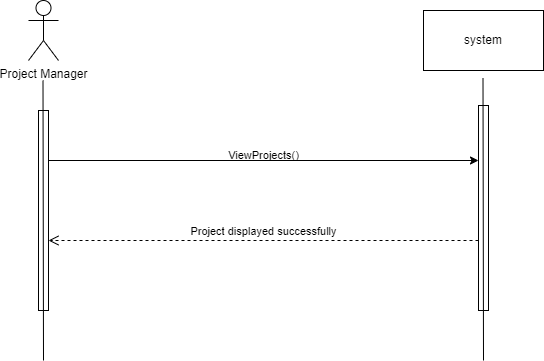


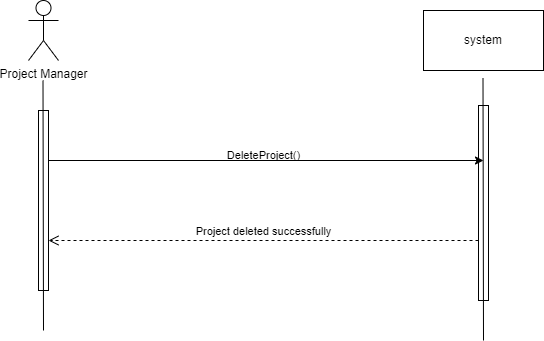


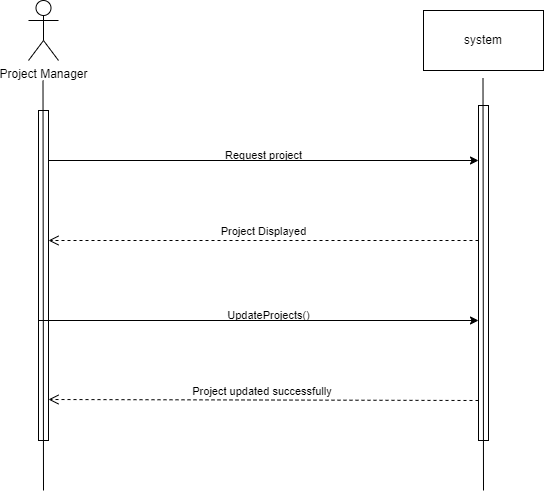


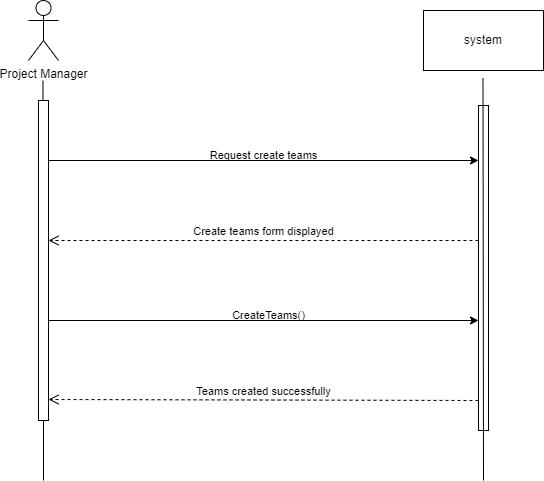


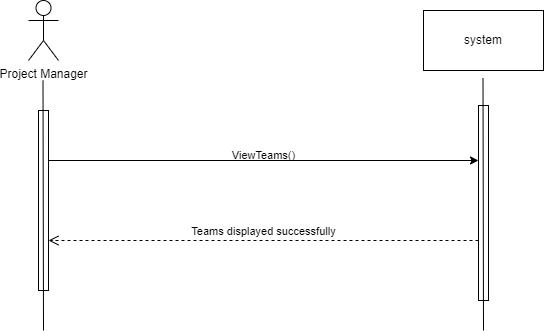


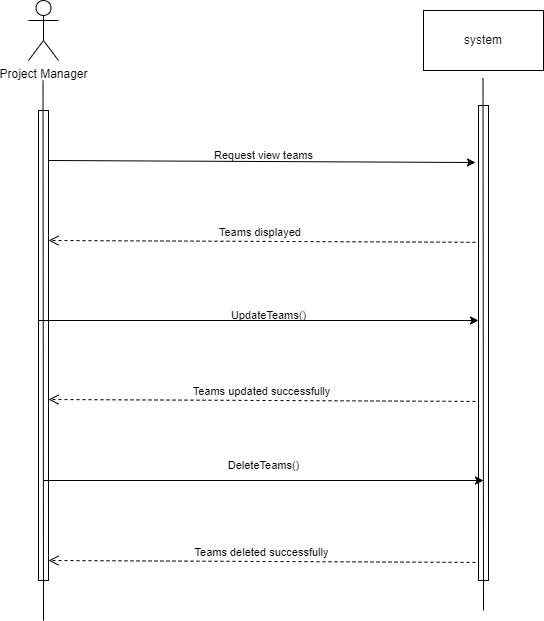


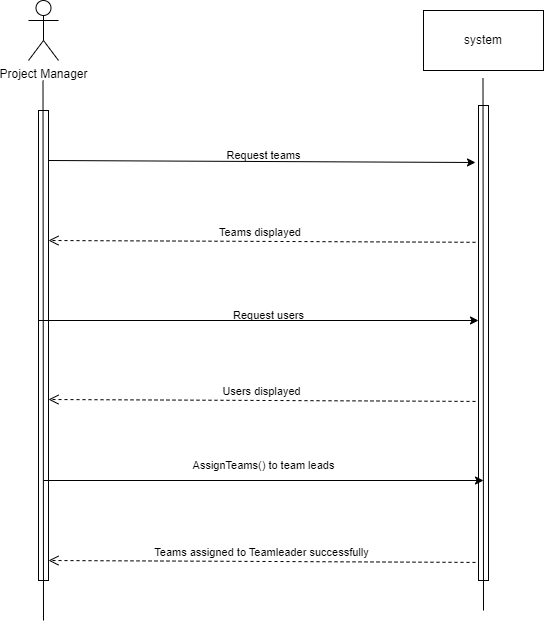


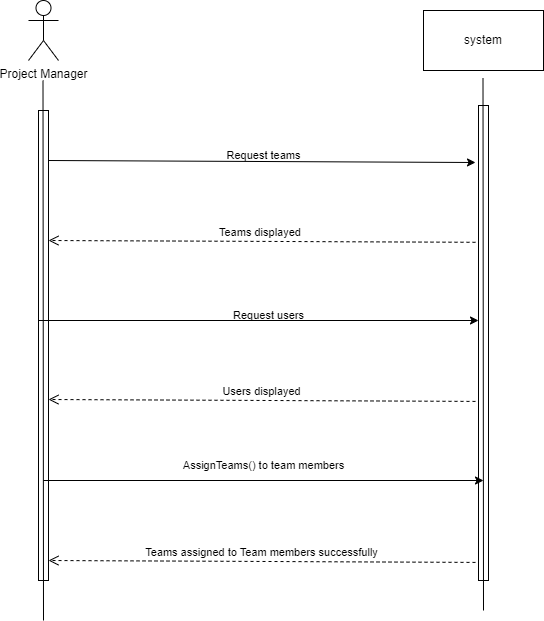


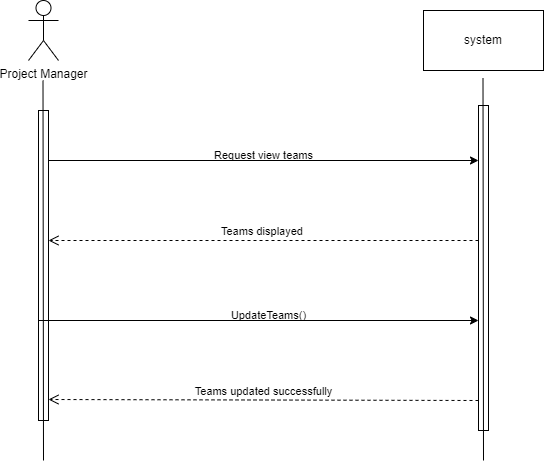


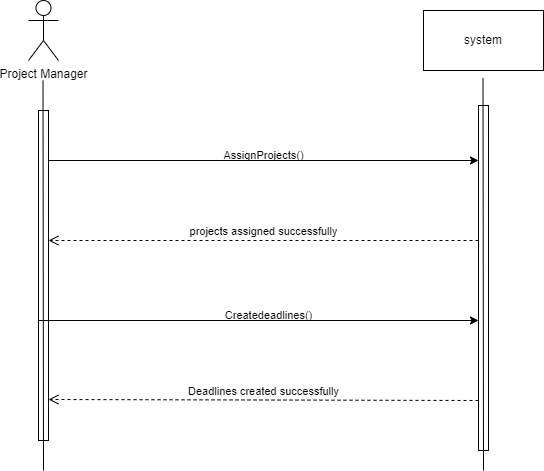


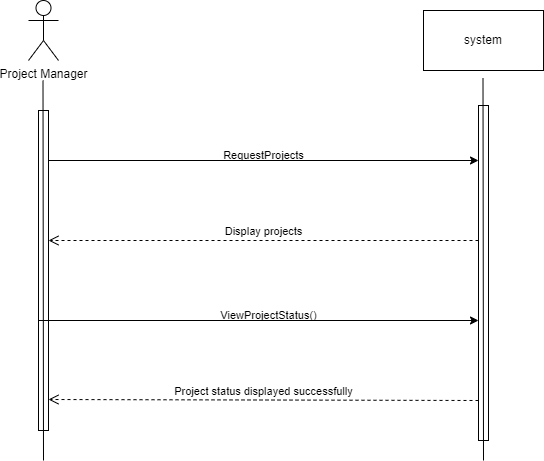


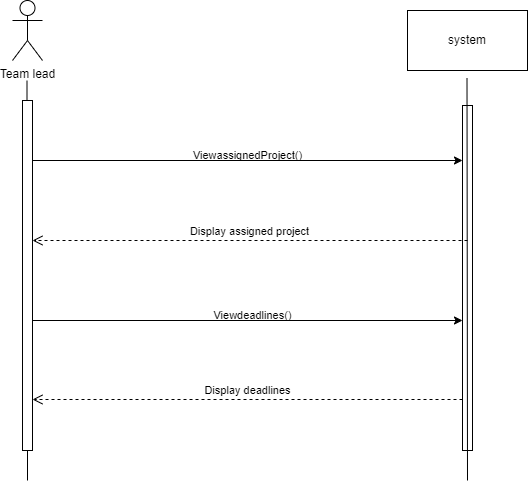


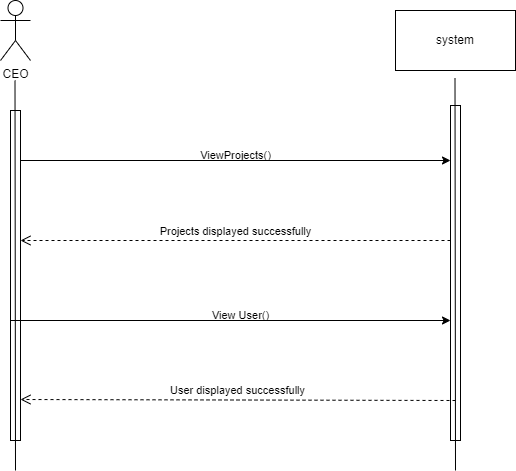


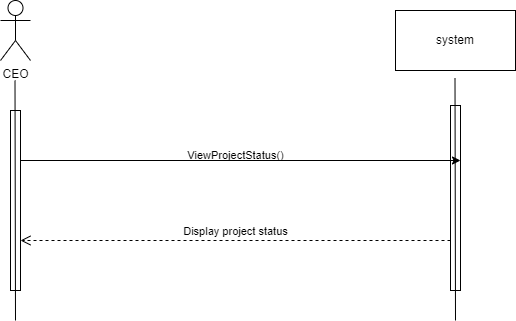
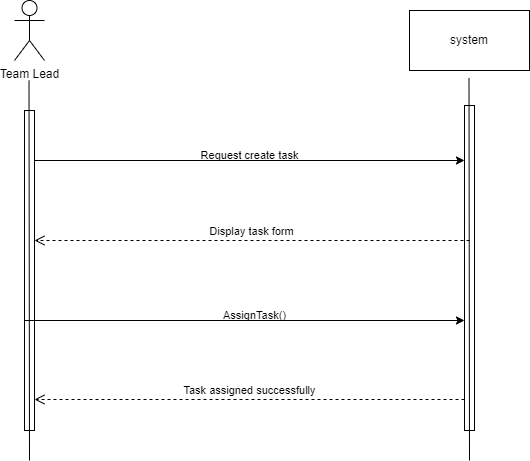


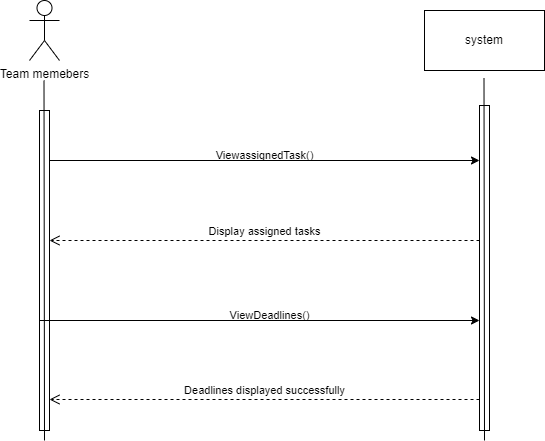


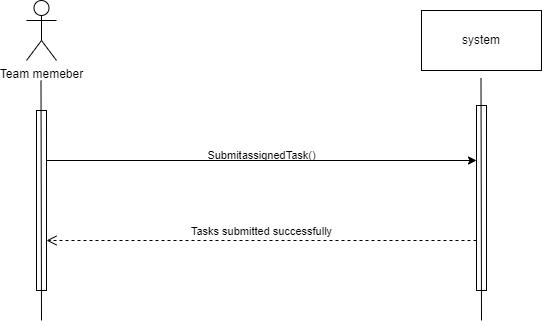
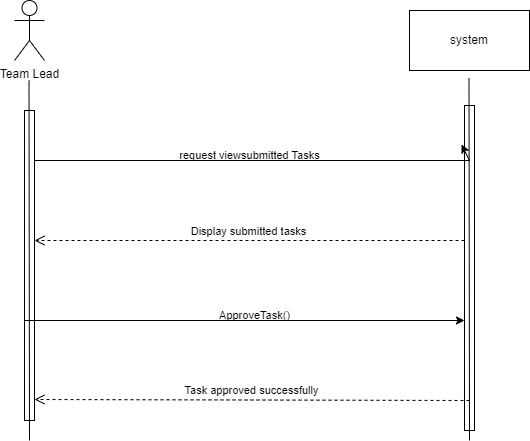


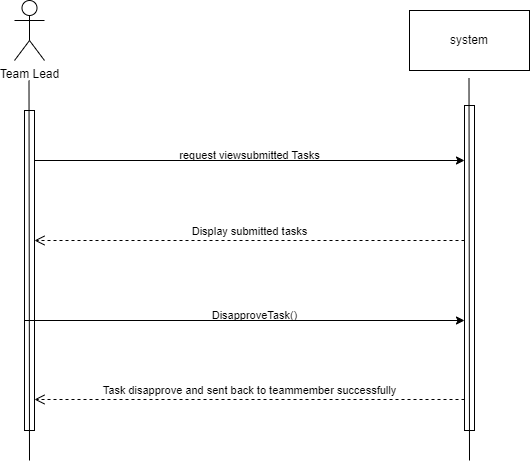
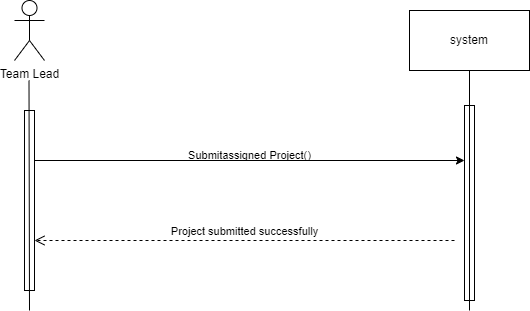


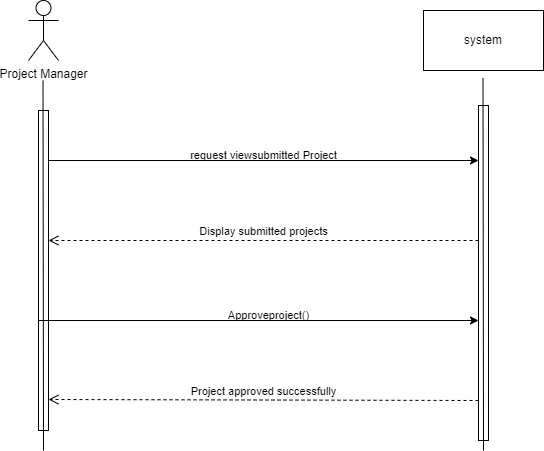


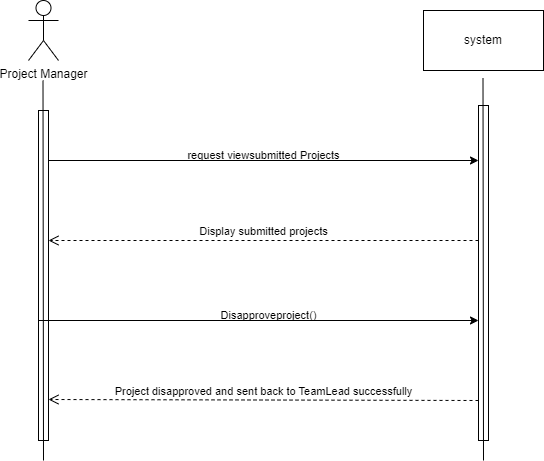






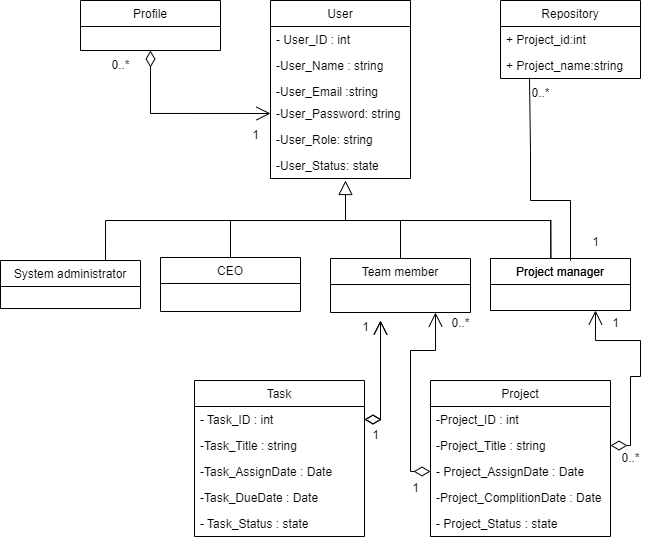






### Domain Model

The basic concepts of the domain are User, Profile, Repository, Artifacts, System administrator, Team Lead, Team member, Project manager, Task and Project are shown in the following figure representing domain model.



***Figure 2.28 domain model***

## Chapter 3

## System Design

The purpose of this chapter is to provide information that is complementary to the development phase. Without an adequate design, that delivers required function as well as quality attributes, the project will fail. However, communicating architecture to its stakeholders is as important a job as creating it in the first place.

### Layer Definition

###### Presentation Layer:

Occupies the top level and displays information related to services available on a website. This tier communicates with other tiers by sending results to the browser and other tiers in the network.

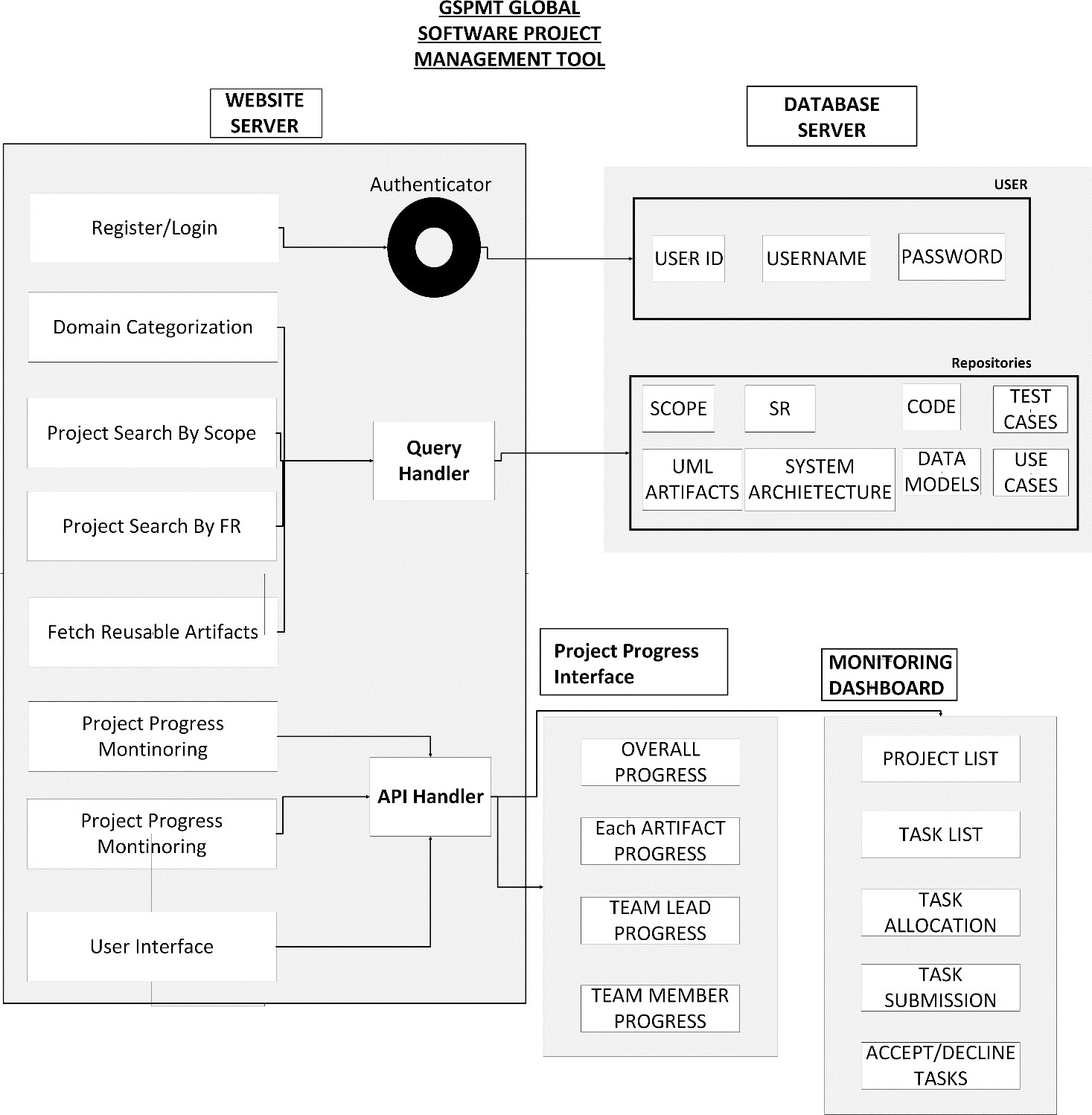
###### Business Logic Layer:

Application Layer also called the middle tier, logic tier, business logic or logic tier, this tier is pulled from the presentation tier. It controls application functionality by performing detailed processing.

###### Database Layer:

Database layer includes database servers where information is stored and retrieved. Data in this tier is kept independent of application servers or business logic.

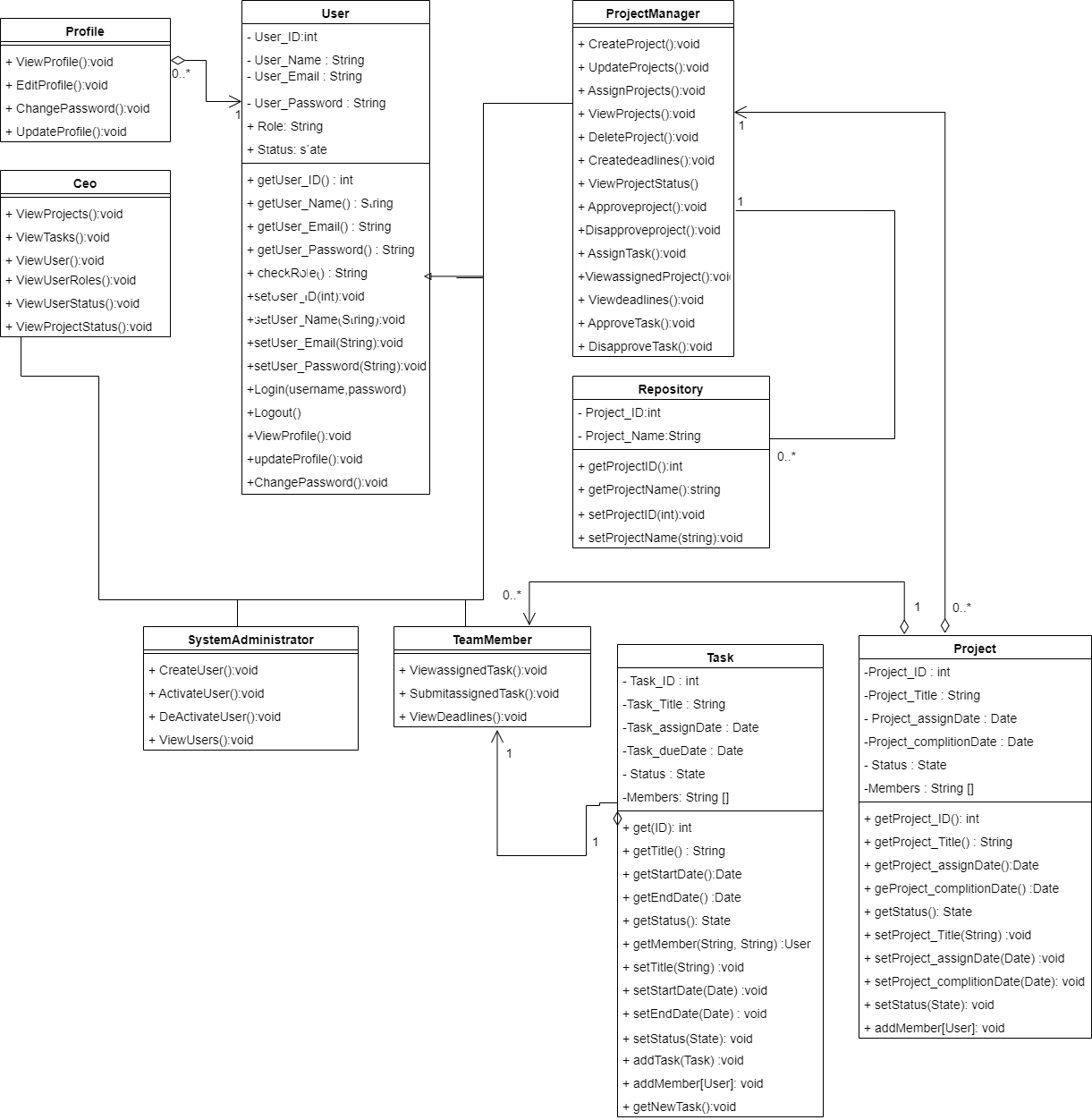
### Software Architecture

Software architecture is described as the organization or structure of a system, where the system represents a collection of components that accomplish a specific function or set of functions. Below is the architecture diagram of the system:

***Figure 4.1 architecture diagram***

### Class Diagram

The class diagram describes the attributes and operations of a class and the constraints imposed on the system. The class diagrams are widely used in the modeling of object oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages:



***Figure 3.2 class diagram***

### Sequence Diagram

Sequence Diagram model the flow of logic within your system in a visual manner enabling you both to document and validate your logic, and are commonly used for both analysis and design purposes

###### 1.5.1. System Administrator

* + 1. **Project manager**

***Figure 2.2 admin***

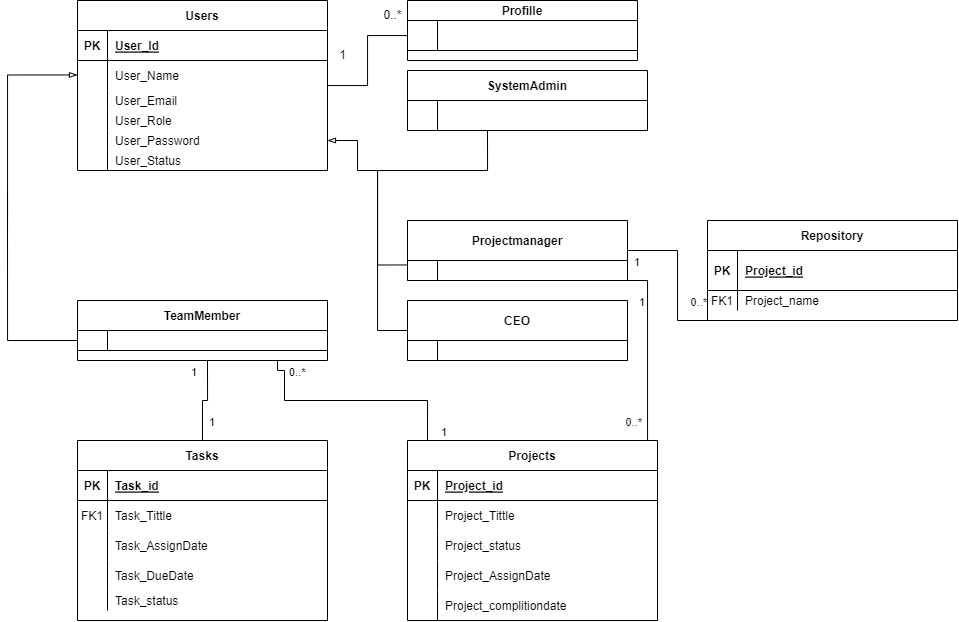
###### Team lead

***Figure 2.3 login (admin)***

* + 1. **Team member**

### Database Schema

***Figure 2.4 Team lead***



### User Interface Design

# Chapter 4

## Software Development

**4.1. Coding Standard:**

We are using all the coding standards. The naming convention we are using are camel case in which the name of class is noun and first letter of class name is capital. The names of method are verb and first letter are small.

We also use indentation properly in every code. Due to indentation our coding is in proper format.

### Development Environment

Tools that we are using:

* + - Studio code
    - Firebase
    1. **Studio code:**

I am developing a web application in REACT JS using Visual Studio Code because it provides good platform and interface. Studio code is beneficial for faster programming and testing the application.

##### Firebase:

Firebase is Google-backed application development software that enables developers to develop IOS, Android and Web apps. Firebase provides tools for tracking analytics, reporting and fixing app crashes, creating marketing and product experiment.

### Software Description

Main modules of our project are:

* + 1. User authentication and management
    2. Project management and monitoring

1. User authentication and management:
   1. Create User:
   2. Login User:
   3. Deactivate User:
   4. Activate User:
   5. View Profile:
   6. Update Profile:
2. Project management and monitoring
   1. Create Project
   2. Delete Project
   3. Update Project
   4. Create Team
   5. Delete Team

# Chapter 5

## Software Testing

This chapter provides a description about the adopted testing procedure. This includes the selected testing methodology, test suite and the test results of the developed software.

##### Testing Methodology

After implementation, the process flow manager is tested for functional errors. We are going to do **Black Box Testing** (by passing random selected values and mapping it against the expected output in a normal flow), Unit and Integration Testing which is the testing of the functional requirements implemented in our system without regard to code. The test cases are done manually without the use of any tool.

##### Test Cases:

A test case is a specification of the inputs, execution conditions, testing procedure and expected results that define a single test to be executed to achieve a particular software testing objective. This section contains both positive and negative test cases.

###### User created by Admin:

|  |  |
| --- | --- |
| *Date:* 11 Nov 2022 |  |
| *System: Global software Monitoring tool* |  |
| *Objective: user registered by admin* | *Test ID:* 1 |
| *Version :*1 | *Test Type:* black box testing |
| *Input :*  *User name : Emaan saqib Email:* [*Emaansaqib@gmail.com*](mailto:Emaansaqib@gmail.com) *Password: Emaan123*  *User status: Active Roles : Team member* | |
| *Expected Result*: User registered successfully. | |
| *Actual result*: Passed | |

* + 1. **Registering same user again by Admin:**

|  |  |
| --- | --- |
| *Date:* 11 Nov 2022 |  |
| *System: Global software Monitoring tool* |  |
| *Objective: Admin registered same user by admin* | *Test ID:* 2 |
| *Version :*1 | *Test Type:* black box testing |
| *Input :*  *User name : Emaan saqib Email:* [*Emaansaqib@gmail.com*](mailto:Emaansaqib@gmail.com) *Password: Emaan123*  *User status: Active Roles : Team member*  *Alert: user already registered.* | |
| *Expected Result*: User registered unsuccessfully. | |
| *Actual result*: Failed | |

###### Login:

|  |  |
| --- | --- |
| *Date:* 11 Nov 2022 |  |
| *System: Global software Monitoring tool* |  |
| *Objective: Login* | *Test ID:* 3 |
| *Version :*1 | *Test Type:* black box testing |
| *Input :*  *Email:* [*Emaansaqib@gmail.com*](mailto:Emaansaqib@gmail.com) *Password: Emaan123* | |
| *Expected Result*: user is login successfully. | |
| *Actual result*: passed | |
| *Date:* 11 Nov 2022 |  |
| *System: Global software Monitoring tool* |  |
| *Objective: Login* | *Test ID:* 4 |
| *Version :*1 | *Test Type:* black box testing |

|  |
| --- |
| *Input :*  *Email:* [*Emaan@gmail.com*](mailto:Emaan@gmail.com) *Password: Emaan123*  *Alert: entered email\password is incorrect.* |
| *Expected Result*: user is login successfully. |
| *Actual result*: passed |

* + 1. **Edit Registered User by Admin:**

|  |  |
| --- | --- |
| *Date:* 11 Nov 2022 |  |
| *System: Global software Monitoring tool* |  |
| *Objective: edit registered user by admin* | *Test ID:* 5 |
| *Version :*1 | *Test Type:* black box  testing |
| *Input :*  *User name : Emaan saqib Email:* [*Emaansaqib@gmail.com*](mailto:Emaansaqib@gmail.com) *Password: Emaan123*  *User status: Active Roles : Team member* | |
| *Expected Result*: Registered user edited successfully. | |
| *Actual result*: Passed | |

###### Set User Status by Admin:

|  |  |
| --- | --- |
| *Date:* 11 Nov 2022 |  |
| *System: Global software Monitoring tool* |  |
| *Objective: set user status by admin* | *Test ID:* 6 |
| *Version :*1 | *Test Type:* black box  testing |
| *Input :*  *User status: Active/Disabled* | |
| *Expected Result*: User Status set successfully. | |

|  |
| --- |
| *Actual result*: Passed |

* + 1. **Reset Password by All Users:**

|  |  |
| --- | --- |
| *Date:* 11 Nov 2022 |  |
| *System: Global software Monitoring tool* |  |
| *Objective: reset password by all user* | *Test ID:* 7 |
| *Version :*1 | *Test Type:* black box  testing |
| *Input :*  *Password: Emaan224 Confirm Password: Emaan224* | |
| *Expected Result*: Reset Password by all Users successfully. | |
| *Actual result*: Fail (Less than 8 characters) | |

|  |  |
| --- | --- |
| *Date:* 11 Nov 2022 |  |
| *System: Global software Monitoring tool* |  |
| *Objective: reset password by all user* | *Test ID:* 8 |
| *Version :*1 | *Test Type:* black box  testing |
| *Input :*  *Password: Emaan2246778 Confirm Password: Emaan224* | |
| *Expected Result*: Reset Password by all Users successfully. | |
| *Actual result*: Fail (Password not matched) | |

|  |  |
| --- | --- |
| *Date:* 11 Nov 2022 |  |
| *System: Global software Monitoring tool* |  |
| *Objective: reset password by all user* | *Test ID:* 9 |
| *Version :*1 | *Test Type:* black box  testing |

|  |
| --- |
| *Input :*  *Password: Emaan2246778 Confirm Password: Emaan2246778* |
| *Expected Result*: Reset Password by all Users successfully. |
| *Actual result*: Passed |

###### Add project:

|  |  |
| --- | --- |
| *Date:* 11 Nov 2022 |  |
| *System: Global software Monitoring tool* |  |
| *Objective: Add project* | *Test ID:* 6 |
| *Version :*1 | *Test Type:* black box testing |
| *Input :*  *Project name : management*  *Description : project is related to management Start date : 5 sep,22*  *End date: 10 Nov,22 Requirements: members ,tool Team members : Humaira, Emaan* | |
| *Expected Result*: project added successfully. | |
| *Actual result*: passed | |

###### Add Task:

|  |  |
| --- | --- |
| *Date:* 11 Nov 2022 |  |
| *System: Global software Monitoring tool* |  |
| *Objective: Add Task* | *Test ID:* 7 |
| *Version :*1 | *Test Type:* black box testing |
| *Input :*  *Project name : management*  *Description : project is related to management*  *Task name : design domain model*  *Description : design the domain model of the project assigned*  *Start date : 5 sep,22*  *End date: 10 Nov,22 Requirements: members ,tool Team members : Emaan* | |
| *Expected Result*: project added successfully. | |
| *Actual result*: passed | |