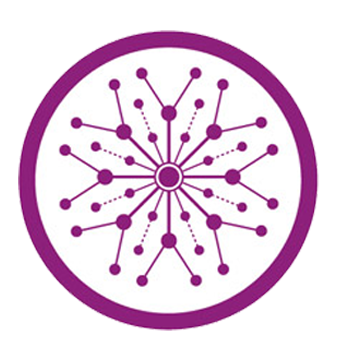
**Final Year Project**

**Session 2020-2024**

A project submitted in partial fulfillment of the degree of

# BS in Software Engineering



Department of Software Engineering

Faculty of Computer Science & Information Technology

The Superior University, Lahore

Fall 2020

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Type (Nature of project) | | | [ ✓ ] **D**evelopment [ ] **R**esearch [ ] **R**&**D** | | |
| Area of specialization | | |  | | |
| FYP ID | | |  | | |
| **Project Group Members** | | | | | |
| Sr.# | Reg. # | Student Name | | Email ID | \*Signature |
| (i) |  | Muhammad Haddi | | bsem-f20-020@superior.edu.pk |  |
| (ii) |  | Ahsan Khalil | | bsem-f20-004@superior.edu.pk |  |
| (iii) |  | Muhammad Usman Khan | | bsem-f19-074@superior.edu.pk |  |

\*The candidates confirm that the work submitted is their own and appropriate credit has been given where reference has been made to work of others

# Plagiarism Free Certificate

This is to certify that, I \_\_\_\_\_\_\_\_ S/D of Muhammad Siddique, group leader of FYP under registration no \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_at Software Engineering Department, The Superior College, Lahore. I declare that my FYP report is checked by my supervisor.

Date:  Name of Group Leader: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_

Name of Supervisor: Dr. ABC Co-Supervisor: Mr. XYZ

Designation: Lecturer Designation: Associate Professor

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

HoD: Dr. Arfan Jaffar

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Project Report**

**Task Assigned System**

**Change Record**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Author(s)** | **Version** | **Date** | **Notes** | **Supervisor’s Signature** |
|  | 1.0 |  | <Original Draft> |  |
|  |  |  | <Changes Based on Feedback from Supervisor> |  |
|  |  |  | <Changes Based on Feedback From Faculty> |  |
|  |  |  | <Added Project Plan> |  |
|  |  |  | <Changes Based on Feedback from Supervisor> |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**APPROVAL**

|  |  |
| --- | --- |
| **Project Supervisor** | |
| Comments: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | |
| Name:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | |
| Date:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |

|  |  |
| --- | --- |
| **Project Manager** | |
| Comments: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |
|  |  |
| Date:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

|  |  |
| --- | --- |
| **Head of the Department** | |
| Comments: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |
|  |  |
| Date:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

# Dedication

This project is dedicated to our respected supervisor, sir Munib Ahmad, who guided us throughout the process with his valuable insights and feedback. We are grateful for his constant support and encouragement, which motivated us to overcome the challenges and achieve our goals. We would also like to thank our team members, Muhammad Haddi, Ahsan Khalil and Muhammad Usman Khan, for their hard work and collaboration. They contributed immensely to the success of this project with their skills and expertise. We appreciate their dedication and professionalism in completing this project on time and with high quality.

# Acknowledgements

This project would not have been possible without the support and guidance of my supervisor, **prof. Munib Ahmad**. He has encouraged me and helped me throughout the project, from the conception to the delivery. I am grateful to my team members**, Ahsan Khalil** and **Muhammad** Usman Khan, for their constructive feedback and cooperation. They have worked hard and efficiently as a team on this project. I also appreciate the help of other people who have assisted me in various aspects of this project. They are (**Muhammad Ahmad** and **Mehak Arshad**). Lastly, I thank Allah for blessing me with the health and ability to complete this project successfully.

# 

# Executive Summary

The task assigned system is a web-based application that allows managers to create, assign, and monitor tasks for their employees. The system aims to improve the efficiency, productivity, and accountability of the workforce by providing a centralized platform for task management. The system also provides features such as task prioritization, deadline reminders, progress tracking, feedback collection, and performance evaluation.

**Main objectives**

* To defeat moment of truth and effort necessary for managers to designate and supervise tasks
* To increase the visibility and transparency of task status and performance
* To enhance the communication and collaboration between managers and employees
* To motivate and reward employees for completing tasks on time and with quality

The task designated system is devised expected user-friendly, secure, and adaptable. Bureaucracy can be achieve from some device accompanying WWW connection and a netting gateway. The system uses encryption and confirmation orders to protect the dossier and solitude of the users. The system can also handle a large number of users and tasks without compromising its performance or functionality.

**Benefits**

* Revised effectiveness and productivity of the trained workers
* Improved quality and consistency of work output
* Improved satisfaction and engagement of managers and employees
* Improved alignment and coordination of organizational goals and strategies

# Table of Contents

[BS in Software Engineering i](#_Toc134079882)

[Plagiarism Free Certificate ii](#_Toc134079883)

[Dedication v](#_Toc134079884)

[Acknowledgements vi](#_Toc134079885)

[Executive Summary vii](#_Toc134079886)

[Table of Contents viii](#_Toc134079887)

[List of Figures x](#_Toc134079888)

[List of Tables xi](#_Toc134079889)

[Chapter 1 1](#_Toc134079890)

[Introduction 1](#_Toc134079891)

[Chapter 2 3](#_Toc134079892)

[Software Requirement Specifications 3](#_Toc134079893)

[1.1. Introduction 4](#_Toc134079895)

[1.1.1. Purpose 4](#_Toc134079896)

[1.1.2. Product Scope 4](#_Toc134079897)

[1.1.3. References 5](#_Toc134079898)

[1.2. Overall Description 5](#_Toc134079899)

[1.2.1. Product Perspective 5](#_Toc134079900)

[1.2.2. Product Functions 5](#_Toc134079901)

[1.2.3. User Classes and Characteristics 6](#_Toc134079902)

[1.2.4. Operating Environment 6](#_Toc134079903)

[1.3. External Interface Requirements 6](#_Toc134079904)

[1.3.1. User Interfaces 6](#_Toc134079905)

[1.3.2. Software Interfaces 6](#_Toc134079906)

[1.4. System Features 7](#_Toc134079907)

[1.4.1. System Feature 1 7](#_Toc134079908)

[1.4.1.1. Description and Priority 7](#_Toc134079909)

[1.4.1.2. Stimulus/Response Sequences 7](#_Toc134079910)

[1.4.1.3. Functional Requirements 8](#_Toc134079911)

[1.4.2. System Feature 2 8](#_Toc134079912)

[1.4.2.1. Description and Priority 8](#_Toc134079913)

[1.4.2.2. Stimulus/Response Sequences 8](#_Toc134079914)

[1.4.2.3. Functional Requirements 9](#_Toc134079915)

[1.5. Other Nonfunctional Requirements 10](#_Toc134079916)

[1.5.1. Performance Requirements 10](#_Toc134079917)

[1.5.2. Safety Requirements 10](#_Toc134079918)

[1.5.3. Security Requirements 10](#_Toc134079919)

[1.5.4. Software Quality Attributes 10](#_Toc134079920)

[Chapter 3 11](#_Toc134079921)

[Use Case Analysis 11](#_Toc134079922)

[3.1. Use Case Model 12](#_Toc134079923)

[3.2. Use Case Descriptions 12](#_Toc134079924)

[Chapter 4 13](#_Toc134079925)

[System Design 13](#_Toc134079926)

[4.1. Class Diagram 14](#_Toc134079927)

[4.2. Sequence / Collaboration Diagram 15](#_Toc134079928)

[4.3. Activity Diagram 16](#_Toc134079929)

[4.4. Use Case Diagram 18](#_Toc134079930)

[4.5. Data Flow diagram [*only if structured approach is used - Level 0 and 1*] 18](#_Toc134079931)

[Chapter 5 20](#_Toc134079932)

[Implementation 20](#_Toc134079933)

[5.1. Important Flow Control/Pseudo codes 21](#_Toc134079934)

[5.2. Components, Libraries, Web Services and stubs 21](#_Toc134079935)

[5.3. Deployment Environment 21](#_Toc134079936)

[5.4. Tools and Techniques 22](#_Toc134079937)

[5.5. Best Practices / Coding Standards 22](#_Toc134079938)

[5.6. Version Control 22](#_Toc134079939)

[Chapter 6 23](#_Toc134079940)

[Testing and Evaluation 23](#_Toc134079941)

[6.1. Use Case Testing 24](#_Toc134079942)

[6.2. Equivalence partitioning 25](#_Toc134079943)

[6.3. Boundary value analysis 26](#_Toc134079944)

[6.4. Data flow testing 26](#_Toc134079945)

[6.5. Unit testing 27](#_Toc134079946)

[6.6. Integration testing 27](#_Toc134079947)

[6.7. Performance testing 28](#_Toc134079948)

[6.8. Stress Testing 29](#_Toc134079949)

[Chapter 7 30](#_Toc134079950)

[Summary, Conclusion and Future Enhancements 30](#_Toc134079951)

[7.1. Project Summary 31](#_Toc134079952)

[7.2. Achievements and Improvements 31](#_Toc134079953)

[Appendices 32](#_Toc134079954)

[Appendix A: User Manual 33](#_Toc134079955)

# List of Figures

Figure 1 Basic Diagram 5

Figure 2 Class diagram 15

Figure 3 Sequence diagram 16

Figure 4 Activity Diagram 17

Figure 5 Use Case Diagram 18

Figure 6 Flowchart Diagram 19

# List of Tables

1.1 label of first table of first chapter 6

1.2 label of second table of first chapter 7

2.1 label of first table of second chapter 14

2.2 label of second table of second chapter 22

2.3 label of third table of second chapter 26

5.1 label of first table of fifth chapter 49

5.2 label of second table of fifth chapter 49

Chapter 1

Introduction

**Chapter 1:** Introduction

In this project, we developed test cases for a Task assigned System and executed them throughout the Software Testing Life Cycle (STLC). The test cases included basic functionality testing such as login, forget password, project and task creation, Calculating the Progress of the Project on the basis of close tasks and new tasks, task time, adding and removing the members from the project, The task page allows users to view open tasks and their status, which can be new, progress, resolved, or close. Users can create a task for a specific project, and it will be assigned the "New" status by default. Members of the project can view and change the status of the task and reassign the task to another member. Each task has a description. We followed the STLC phases.

Chapter 2

Software Requirement Specifications

**Chapter 2:** Software Requirement Specifications



## Introduction

## Purpose

The purpose of this app is to provide functionality to manage software projects divided in several tasks it has the following features such as login, forget password, project and task creation, Calculating the Progress of the Project based on close closing and new tasks, task time, adding and removing the members from the project, The task page have all the lists of New, Progress, Resolve and close allows users to see tasks lists and can open tasks, which can be new, progress, resolved, or close. Users can create a tasks for a specific project, and it will be assigned the "New" status by default. Members of the project can view and change the status of the task and reassign the task to another member. Each task has a description.

## Product Scope

The software defines the process or phase of the (SDLC) software development Life cycles and the main purpose of this software is to complete the projects using this software more effectively, fast, and within a team. This is best for those company performed big project regularly this software can complete their project.

## References

There is no reference to any web pages and etc.

## Overall Description

## Product Perspective

The software contains the phase of SDLC Software development life cycles. These phase includes planning, analysis, design, development, testing, implementation and maintenance.

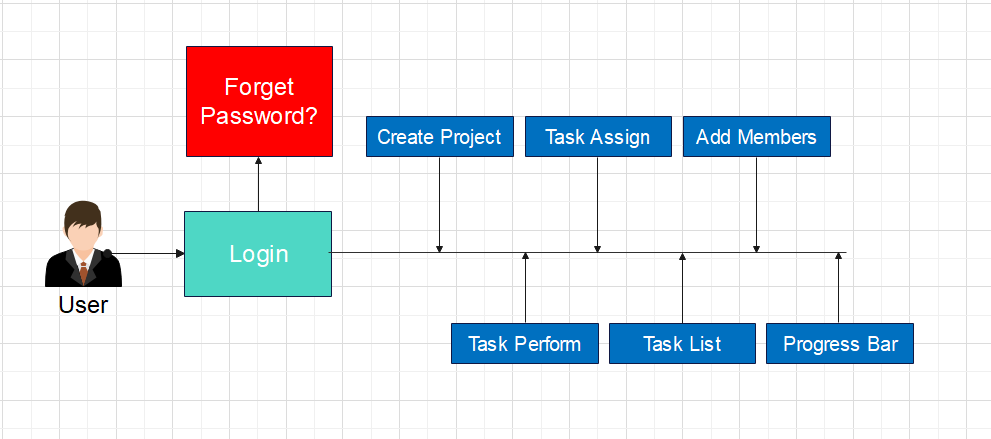


Figure 1 Basic Diagram

## Product Functions

* **Login and forget password:** Users can sign in to the app using their credentials or reset their password if they forget it.
* **Project and task creation:** Users can create new projects and tasks within the app and assign them to different members.
* **Project progress calculation:** The app calculates the progress of each project based on the number and status of the tasks associated with it.
* **Member management:** Users can add or remove members from a project and assign them different roles and permissions.
* **Task status and description:** Users can view the status and description of each task in a project and change them as needed. The possible statuses are new, progress, resolved, or close.

## User Classes and Characteristics

This product include classes that are Login, Project and Task create, Task Assigned, there are some edited based classes which can be access by only project manager , there some other access by only who’s has done the task that are Task completion , View Task and some other basic functionality which task place in these classes.

## Operating Environment

This software environment in which the software is operating is a website-based application. The technology we are utilizing is the latest. There is various component such as login, forget password, project and task creation, Calculating the Progress of the Project based on close closing and new tasks, task time, and adding and removing the members from the project, The task page allows users to view open tasks and their status, which can be new, progress, resolved, or close and etc.

## External Interface Requirements

## 

## User Interfaces

This app allows users to manage projects and tasks efficiently and collaboratively. It has features such as login, forget password, project and task creation, progress calculation, member management, and task status tracking. The user interface (UI) design of this app focuses on making the interface easy to access, understand, and use. The UI consists of graphical elements such as buttons, text, pictures, and tables.

## Software Interfaces

The software Interfaces of this app are the tools and languages that enable communication between the app and other applications, systems, or devices. For example, the app may use an API (application programming interface) to exchange data and functionality with external third-party services, such as payment processing or email verification. The app may also use a hardware interface to interact with the device's features, such as camera, microphone, or touch screen. The software Interfaces define the rules and protocols for these interactions, ensuring compatibility and security.

## System Features

The system contains of the following features such as login, forget password, project and task creation, Calculating the Progress of the Project based on close closing and new tasks, task time, adding and removing the members from the project, The task page allows users to view open tasks and their status, which can be new, progress, resolved, or close. Users can create a task for a specific project, and it will be assigned the "New" status by default. Members of the project can view and change the status of the task and reassign the task to another member. Each task has a description.

## System Feature 1

There is Login, Project Create, Task Create and members to the Project.

## Description and Priority

The login is the main feature of the software which is high (9), Project Create is also high (9), Task Create is medium (6), and add members is medium (6). All these feature contains also contain validations.

## Stimulus/Response Sequences

1. User can login the system, when user enter the email and password on the login page, which is serve as the stimulus, the system verify the user credential and login, which is the response.
2. Once login, User can create the project, by enter **“Project Create”** button. The action serve as stimulus, the system displays about where user enter detail about the new project, such as project title, description, start date and end date. Once user create the project, the system responds with confirmation massage. Which is the response.
3. Once Project Create, user can add the task against that project, by clicking the button **“Add Task or Create Task”**, The action serve as stimulus, the system displays the forms where us fill detail of the new task, such as name description and due date. Upon the completion of the form, the system response by adding task to the project and displaying it to user’s task list.
4. Once Project create and Add Task is done, the user add the members to the project, who will have access to the project’s task and other details, the action serve as a stimulus, by clicking on the **“add member”** button. The system responds by presenting a form for the user to enter the email address of the member by a wish to add. Upon completion of the form, the system responds by sending an invitation to the specified email address, inviting the member to join the project.

## Functional Requirements

REQ-SF1-1: The user login the system, and the user enters the email and password, in the case email and password are incorrect system must send the error message to the specific user.

REQ-SF1-2: The user can create the project, but all required fields must be filled in, such as project title, description, start date and end date, if one of these is missed send a message which is the response by the system.

REQ-SF1-3: Users can create new tasks, but all required fields must be completed. If any required fields are missed, the system will throw an error.

REQ-SF1-4: Projects must have at least three members at the beginning, and more can be added as necessary.

## System Feature 2

The Forget password, remove member, delete a project, calculate the progress and Reassign the task.

## Description and Priority

In the case of password forget, add forget password medium (7), the remove the Member medium (9) delete the project high (9). Calculate the progress task in the basic of close and new task medium (7). The reassign the task medium (6).

## Stimulus/Response Sequences

1. In the case of forgetting a password, we are adding a **“forget password"** button, before the login button, by clicking on the **“forget password”** button, the action serves as stimulus, the system displays the form, where we have enter the user detail such as email, the send the link to mail, you can reset your password again.
2. In this case project member is not doing well or not responds to any task, the member who are working to the project will be found in project member option, where project member list will be displayed by clicking on the particular member form will displayed in system responds, we can delete that particular member by force, by enter the member detail, such as member email.
3. The user can delete the project, in the case of a mistaken project created, privacy and security concerns, this option is located at the setting of the project for every project, by clicking on **“delete project”** in the response system displayed forms, such as project name and project creator name. The project will be deleted.
4. The Projects task calculation progress by showing with a graph, the project progress bar in basic of close task and new task, the formula is calculated **Progress=(close task)/(close task+ new Task)**. For example (4) / (4+6) is equal to 0.4\*100 is 40%.
5. The Tester detect the bug in the feature, you have to reassign the task, if the task is to reassign to any user it should be written as task reassign and how many tasks have been assigned to the user.

## Functional Requirements

REQ-SF2-1: The user can add the email in the field, In this case, you can reset your passwords, the case of an empty field system throw an error to the user field must not be empty.

REQ-SF2-2: The user can add the email in the field, In case of an empty field system throw an error to the user field must not be empty. In the case of entering the email in, the field you can delete the user to that particular project.

REQ-SF2-3: All fields must be required to fill, in the case of an empty system must throw an error to the user, if the user enters the project name and creator name you can successfully delete the project.

REQ-SF2-4: When task is completed progress must be updated and in the case of new task progress bar also updated.

REQ-SP2-5: When the task is done by the developer, it should be displayed in specific, this task has been reassigned by the specific tester to the developer, and number of reassign task.

## Other Nonfunctional Requirements

## Performance Requirements

When users perform this functionality, each requirement must be performed as soon as possible. The user login is performed within 5 seconds in the case of clicking on the login button. When the user creates the project after a while created project cannot dissipate from the software, and the required field fills according to the title must be 50 words and the description must be 500 words it can increase in the case of a big project and start time and dead time for created project minimum 15 days duration, it could be increased, the task assigned, and reassigned must be performed well to the related components must well such progress bar must display the right status and the progress bar also change in the case of real-time submission of task. All lists of tasks must display.

## Safety Requirements

This Login feature allows users to safely access the Task assigned system by providing their credentials. The forget password feature allow the user recover the password in the case of forget their passwords. This feature allow create the new project and task within the system. List of the task only view by member accounts. User can read the detail related task when we open the specific task. This feature allow the user add member into the particular project.

## Security Requirements

If User login in to system the system check the confidential of the user and Authentication that user is part admin group and project team member, project creator can change the setting of the project such as delete the project and project privacy add member task assign to any member means edit and update part no normal member able to performed such activities.

## Software Quality Attributes

The Software may include usability, reliability, correctness, adaptability, robustness, maintainability, and testability. The design of the system is easy to use, each function performed without any error all system feature is error free, it easy to learn for every user, system is in stable state every type of input, system must upgrade able in the case of add new feature in future. Each system of subsystem is work properly.

Chapter 3

Use Case Analysis

**Chapter 3:** System Analysis

The system must performed login, forget password, project and task creation, Calculating the Progress of the Project based on close closing and new tasks, task time, adding and removing the members from the project, The task page allows users to view open tasks and their status, which can be new, progress, resolved, or close. Users can create a task for a specific project, and it will be assigned the "New" status by default. Members of the project can view and change the status of the task and reassign the task to another member. Each task has a description.

## Use Case Model

**System:** Task Assign System

**Use case label:** Login and Task Management.

**Description:** This use case describes how user log in to the system and manage tasks.

**Actors:** User

**Precondition:** The User must have account for the system.

**Basic Flow:**

1. The accesses the login page of the system.
2. The system prompt user to enter their login credential
3. The user enter their login credential and submit them
4. The system authenticates the user and logs them in.
5. The user can create tasks, assign tasks to team members, and add members to the project.
6. The user can view their assigned tasks, perform tasks, and check the progress of their tasks using the progress bar.

## Use Case Descriptions

There is two type of user admin and team members, Login into the system, the system authenticate the user, admin can performed Task Create, Task Assign and add member etc. the team member can perform these function performed the given task, check the list given task and also check the progress bar.

Chapter 4

System Design

**Chapter 4:** System Design

This app is a project management system that allows users to create and manage tasks for different projects. Login and forget password functionality, project and task creation, progress calculation based on task status, adding and removing project members, task page with status and description. The front-end web application that communicates with the back-end using HTTP and REST. The back-end server that handles requests from the front-end and performs business logic. The relational database that stores data about projects, tasks, users, and other entities. The message queue that enables asynchronous communication between the front-end and the back-endA cache that improves the performance and scalability of the system by storing frequently accessed data.

## Class Diagram

The user Login into system check permission system gives the user following. We are using class diagram check it. We inheritance to check the permission of the user.

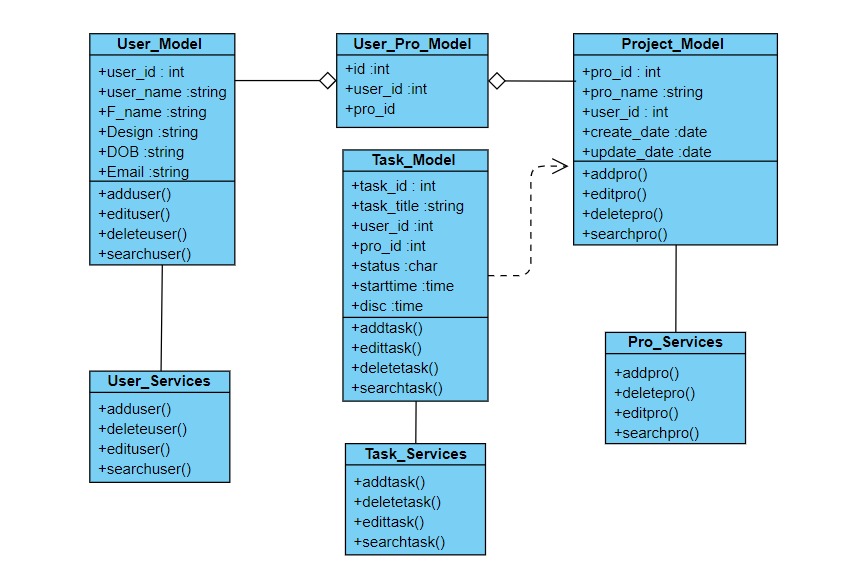


Figure 2 Class diagram

## Sequence / Collaboration Diagram

The user Login into system gives the user following. We are using sequence diagram check it. User has to manage project, manage task, add, save and delete the list of task to any user.

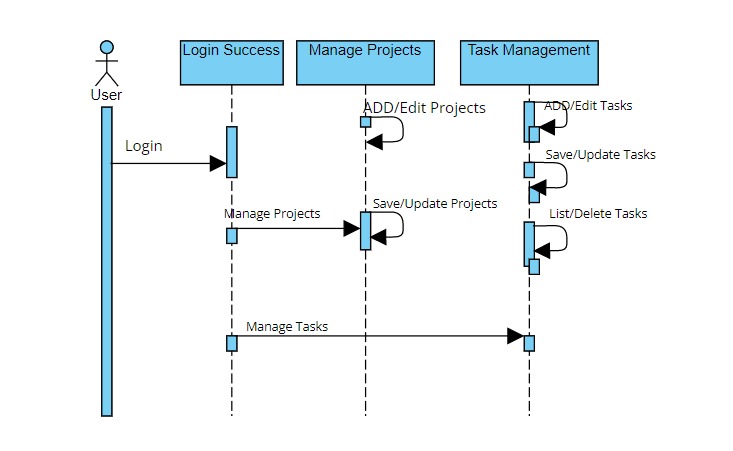


Figure 3 Sequence diagram

## Activity Diagram

The user Login into system gives the user following. We are using Activity diagram check it. Which user can performed Manage Project, Create Project Manage Task, Assign Task and Submit Project.

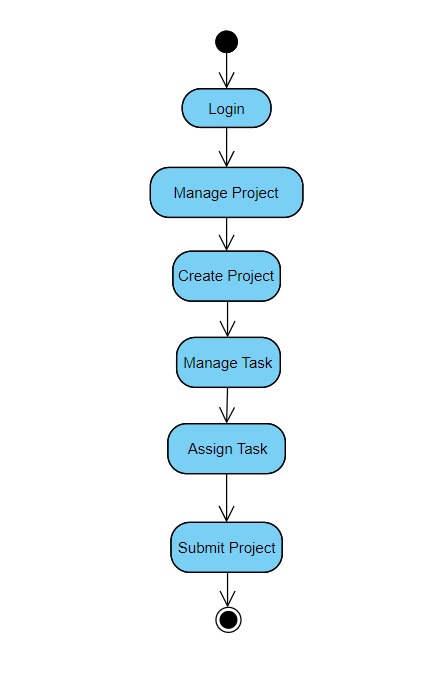


Figure 4 Activity Diagram

## Use Case Diagram

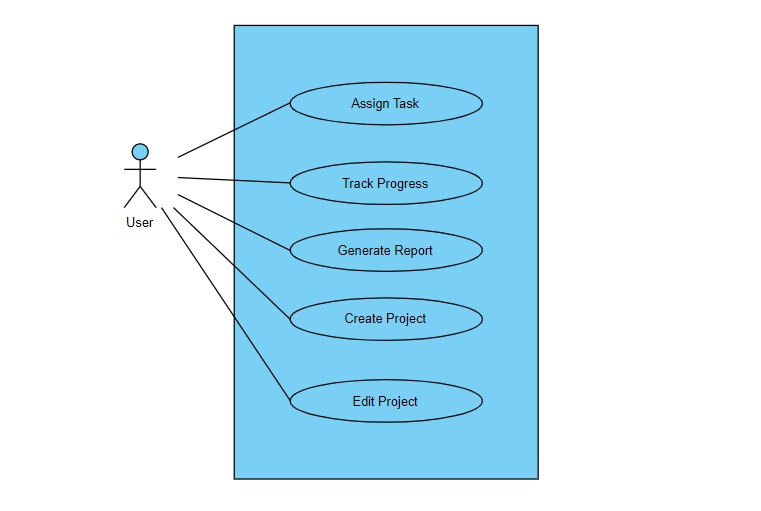
Every User Can perform every functionality, the user who performed assign task Track ,generate report, Edit Project where we user own this project or part of the project. 

Figure 5 Use Case Diagram

## Data Flow diagram [*only if structured approach is used - Level 0 and 1*]

The user Login into system gives the user following. We are using Data flow diagram check it. We manage the project, assign the task, create the project Task progress contain two possibilities Edit and Project Complete, with project complete submit.

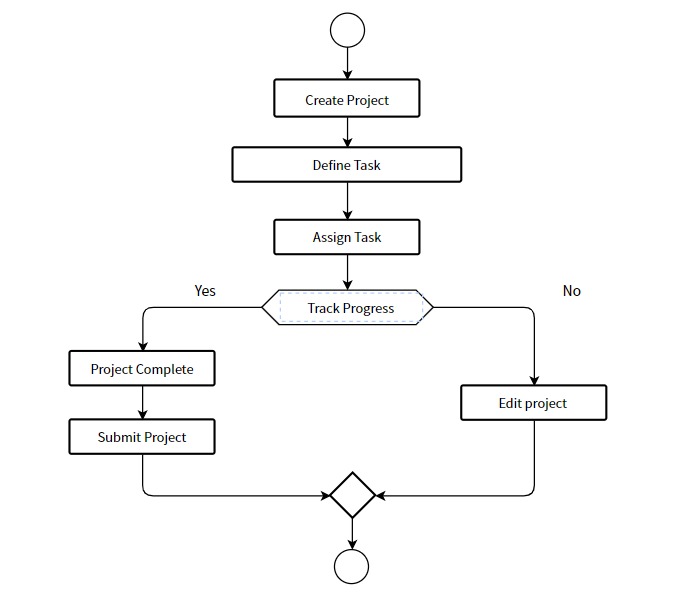


Figure 6 Flowchart Diagram

Chapter 5

Testing and Evaluation

**Chapter 6:** Testing and Evaluation

## Use Case Testing

1. Login to the system with a valid username and password.
2. Select a project from the list of projects or create a new one.
3. Click on the "Tasks" tab to access the task page.
4. Verify that the task page displays all the open tasks for the selected project, along with their status and assignee.
5. Click on the "Create Task" button to create a new task for the project.
6. Enter a title and a description for the task and click on "Save".
7. Verify that the new task is added to the task page with the "New" status and assigned to the current user.
8. Click on the task title to view its details.
9. Verify that the task details page shows the title, description, status, assignee, and comments of the task.
10. Change the status of the task to "Progress" and click on "Update".
11. Verify that the task status is updated to "Progress" on both the task details page and the task page.
12. Reassign the task to another member of the project and click on "Update".
13. Verify that the task assignee is updated to the selected member on both the task details page and the task page.
14. Add a comment to the task and click on "Post".
15. Verify that the comment is added to the task details page and visible to all members of the project.
16. Repeat steps 8-15 for other tasks with different statuses and assignees.
17. Close some tasks by changing their status to "Close" and verify that they are removed from the task page.
18. Logout from the system.

**The expected results for this use case are:**

1. The system allows users to create, view, update, reassign, comment, and close tasks for a project.
2. The system displays all open tasks for a project on the task page, along with their status and assignee.
3. The system updates the status and assignee of a task on both the task details page and the task page when they are changed by a user.
4. The system adds comments to a task on its details page and makes them visible to all members of the project.
5. The system removes closed tasks from the task page.

**The test data for this use case are:**

1. A valid username and password for logging in to the system.
2. A list of projects with different members and tasks.
3. A title and a description for each new task created.
4. A status (new, progress, resolved, or close) for each updated task.
5. An assignee (a member of the project) for each reassigned task.
6. A comment for each commented task.

## Equivalence partitioning

1. Partition the inputs into valid and invalid username and password combinations. Test one valid and one invalid input from each partition.
2. Partition the projects into existing and new ones. Test one existing and one new project from each partition.
3. Partition the tasks into open and closed ones. Test one open and one closed task from each partition.
4. Partition the task status into new, progress, and close. Test one task with each status from each partition.
5. Partition the task assignee into current user and other project members. Test one task assigned to each assignee from each partition.
6. Partition the task comments into empty and non-empty ones. Test one task with each comment type from each partition.

## Boundary value analysis

1. Create a task with a valid description and verify that it is assigned the "New" status and displayed on the task page.
2. Create a task with an empty description and verify that it is rejected with an appropriate error message.
3. Create a task with a very long description (e.g. more than 500 characters) and verify that it is rejected with an appropriate error message.
4. Change the status of a task from "New" to "Progress" and verify that it is updated on the task page.
5. Change the status of a task from "Progress" to "Resolved" and verify that it is updated on the task page.
6. Change the status of a task from "Resolved" to "Close" and verify that it is updated on the task page.
7. Change the status of a task from "Close" to any other status and verify that it is rejected with an appropriate error message.
8. Reassign a task to another member of the project and verify that it is updated on the task page.
9. Reassign a task to a non-member of the project and verify that it is rejected with an appropriate error message.
10. Reassign a task to oneself and verify that it is rejected with an appropriate error message.

## Data flow testing

Data flow testing is a technique of software testing that analyzes the paths of data through a program. It focuses on the variables and their definitions and uses in different parts of the code. Data flow testing can help identify anomalies, such as uninitialized variables, dead code, or infinite loops. Data flow testing can be applied at different levels of testing, such as unit testing, integration testing, or system testing.

## Unit testing

Unit testing is a software development technique that involves writing and running small pieces of code to verify the functionality of individual units or components of a larger system. Unit tests are usually automated and isolated from external dependencies, such as databases, network resources, or user interfaces. Unit testing helps developers to find and fix bugs early in the development process, as well as to improve the quality, reliability, and maintainability of the code.

## Integration testing

Integration testing is a type of software testing that verifies the interaction and functionality of different components or modules of an application. In this testing, we check how the application works as a whole by combining the individual units and testing them together. Integration testing can be done at different levels, such as system, subsystem, or interface level.

**Features:**

1. **Login**: We need to test if the user can log in successfully with valid credentials and if the application rejects invalid or unauthorized credentials. We also need to test if the user can reset their password if they forget it and if the password reset process is secure and reliable.
2. **Project and task creation**: We need to test if the user can create a new project and add relevant details, such as name, description, deadline, and members. We also need to test if the user can create tasks for a project and assign them to different members. We need to check if the task creation process is consistent and error-free.
3. **Calculating the progress of the project**: We need to test if the application can calculate and display the progress of the project based on the number and status of the tasks. We need to check if the progress calculation is accurate and updated in real-time.
4. **Task time**: We need to test if the user can track the time spent on each task and if the application can record and report the task time correctly. We need to check if the task time feature is compatible with different devices and browsers.
5. **Adding and removing members from the project:** We need to test if the user can add or remove members from a project and if the application reflects the changes accordingly. We also need to test if the user can manage the permissions and roles of the members and if the application enforces them properly.
6. **Task page:** We need to test if the user can view and filter the tasks by their status, such as new, progress, resolved, or closed. We also need to test if the user can open a task and view its details, such as description, assignee, comments, attachments, etc. We need to check if the user can change the status of a task or reassign it to another member and if the application updates the task information correctly.

## Performance testing

Performance testing is a type of software testing that evaluates how a system performs under a specific workload. It is not about finding software bugs or defects, but about identifying and eliminating performance bottlenecks. Performance testing can measure various aspects of a system, such as speed, response time, stability, reliability, scalability, and resource usage.

**Common types of performance testing are:**

1. Load testing: It measures how the system performs as the workload increases within the normal range of conditions.
2. Stress testing: It measures how the system performs beyond the normal range of conditions and identifies the breaking point of the system.
3. Spike testing: It measures how the system handles sudden and repeated increases in the workload.
4. Endurance testing: It measures how the system performs over a long period of time with a normal workload.
5. Scalability testing: It measures how the system adapts to changes in the workload or resources.
6. Volume testing: It measures how the system handles large amounts of data.

Performance testing can help developers to diagnose and fix performance issues before they affect the end-users. It can also help stakeholders to evaluate whether the system meets the performance requirements and expectations.

One of the examples of performance testing is testing a web application that supports various features such as login, forget password, project and task creation, calculating the progress of the project based on closing and new tasks, task time, adding and removing members from the project, etc. The performance test can simulate different scenarios and user behaviors to measure how the web application responds and performs under different conditions. For instance, the performance test can measure

1. How long does it take for a user to log in or reset their password?
2. How many concurrent users can the web application handle without affecting its speed or stability?
3. How does the web application cope with high volumes of data or transactions?
4. How does the web application scale up or down when more or fewer resources are available?
5. How does the web application recover from failures or errors?

The performance test can use various tools and metrics to collect and analyze data from the web application. Some of the common tools are JMeter, LoadRunner, Gatling, etc. Some of the common metrics are response time, throughput, latency, error rate, CPU usage, memory usage, etc. The performance test can generate reports and graphs to visualize and compare the results of different tests.

Performance testing is essential to software quality assurance and should be planned and executed throughout the software development life cycle. Performance testing can help to ensure that the software delivers a fast, stable, reliable, and scalable user experience.

## Stress Testing

Stress Testing is a type of software testing that evaluates the performance and reliability of a system under extreme conditions. It involves subjecting the system to high levels of load, traffic, data volume, or concurrency to identify any potential bottlenecks or failures. Stress Testing can help ensure that the system can handle peak demand and maintain its functionality and stability.

In this project, we will conduct Stress Testing on various features of the system, such as login, forget password, project and task creation, calculating the progress of the project based on closed and new tasks, task time, adding and removing the members from the project, and the task page. We will use different tools and techniques to simulate high load scenarios and measure the system's response time, throughput, error rate, memory usage, CPU utilization, and other metrics. We will also analyze the results and identify any areas for improvement or optimization.

# Chapter 7

# Summary, Conclusion and Future Enhancements

**Chapter 7:** Summary, Conclusion & Future Enhancements

## Project Summary

The web application is a tool for project management that enables users to create, assign, and track tasks for different projects. Users can log in, reset their password, and create new projects. Each project has a name, a description, and a list of members who can access it. Users can also add or remove members from a project at any time. The progress of the project is calculated based on the number and status of the tasks. The web application has a task page where users can view all the open tasks for a project and their status, which can be new, progress, resolved, or close. Users can create new tasks by providing a name and a description. They can also change the status of a task or reassign it to another member. The web application helps users to organize their work and collaborate with others on projects.

## Achievements and Improvements

The web application is a tool for project management that enables users to create, assign, and. track tasks for different projects. It has the following achievements and improvements. It allows users to log in, reset their password, and create new projects with ease and security. It enables users to add a name, a description, and a list of members for each project, and to modify them at any time. It calculates the progress of the project based on the number and status of the tasks, and displays it on a dashboard. It has a task page where users can view all the open tasks for a project and their status, which can be new, progress, resolved, or close. It lets users create new tasks by providing a name and a description, and to change the status or reassign them to another member as needed. It helps users to organize their work and collaborate with others on projects effectively and efficiently.

## Critical Review

The task management system is a web application that allows users to create and manage projects and tasks. The system has several features, such as login, forget password, project and task creation, calculating the progress of the project based on closed and new tasks, task time, adding and removing the members from the project, etc. The task page has all the lists of new, progress, resolve and close tasks, and allows users to see tasks lists and can open tasks, which can be new, progress, resolved, or close. Users can create a task for a specific project, and it will be assigned the "New" status by default. Members of the project can view and change the status of the task and reassign the task to another member. Each task has a description.

The system is designed to facilitate collaboration and communication among project members and to improve the efficiency and quality of work. However, the system also has some limitations and areas for improvement. This review will evaluate the system based on the following criteria: usability, functionality, reliability, security, and scalability.

## Lessons Learnt

The project involved developing a web application that allows users to manage their projects and tasks.

**Main features**

1. Login and forget password functionality
2. Project and task creation and deletion
3. Calculating the progress of the project based on closed and new tasks
4. Task time tracking and reporting
5. Adding and removing members from the project
6. Viewing tasks by status (new, progress, resolved, close)
7. Opening and editing tasks with descriptions

The project was completed on time and within budget. The web application met the requirements and expectations of the client and the users. The project team worked well together and communicated effectively. The project manager provided clear guidance and support throughout the project.

1. Problem: The web application was not tested for cross-browser compatibility.
2. Impact: Some users reported issues with the web application's functionality and appearance on different browsers.
3. Recommendation: Perform cross-browser testing before deploying the web application to ensure a consistent user experience.
4. Problem: The web application did not have a backup system.
5. Impact: The web application was vulnerable to data loss or corruption in case of a server failure or a cyberattack.
6. Recommendation: Implement a backup system that regularly saves the web application's data to a secure location.
7. Problem: The web application did not have a user feedback mechanism.
8. Impact: The project team did not receive any feedback from the users about the web application's performance, usability, or satisfaction.
9. Recommendation: Incorporate a user feedback mechanism in the web application, such as a survey, a rating system, or a comment section, to collect user feedback and suggestions for improvement.

## Future Enhancements/Recommendations

The project report describes the main features and functionalities of the web application developed for managing tasks and projects. The web application allows users to perform various operations such as login, forget password, project and task creation, calculating the progress of the project based on closed and new tasks, task time, adding and removing the members from the project, etc. The task page has all the lists of new, progress, resolve and close tasks and allows users to see tasks lists and can open tasks, which can be new, progress, resolved, or close. Users can create a task for a specific project, and it will be assigned the "New" status by default. Members of the project can view and change the status of the task and reassign the task to another member. Each task has a description.

Some of the future enhancements/recommendations for improving the web application are:

- Implementing a notification system to alert users about new or updated tasks

- Adding a comment feature to allow users to communicate and collaborate on tasks

- Integrating a calendar view to display tasks by due date and priority

- Incorporating a feedback mechanism to collect user satisfaction and suggestions

- Enhancing the security and performance of the web application

Appendices

# Appendix A: User Manual

The User Manual for the software includes instructions on login, forget password, project and task creation, and tracking project progress. It also covers adding and removing members from the project, viewing and changing the status of tasks, and creating new tasks with descriptions. The Task page provides lists for New, Progress, Resolved, and Closed tasks, which users can open and reassign.

**Appendix A: Task Assign System**

This appendix focuses on the task assign system within the software, providing additional guidance and details on how tasks are assigned, managed, and tracked. It covers the steps for assigning tasks to team members, modifying task statuses, and monitoring task progress. The appendix also includes information on creating tasks with descriptions, as well as opening and reassigning tasks within the New, Progress, Resolved, and Closed lists. It serves as a supplementary resource to the User Manual, offering users a comprehensive understanding of the task assignment functionality and its associated workflows within the software.

1. **Task Management and Project Progress Tracking**

This first-level heading encompasses the main sections or topics related to task management, project progress tracking, and related functionalities. It covers login and forget password procedures, project and task creation, as well as the process of adding and removing members from the project. It also includes instructions on viewing and changing the status of tasks and creating new tasks with descriptions. Additionally, it mentions the Task page that provides lists for New, Progress, Resolved, and Closed tasks, allowing users to open and reassign tasks.

* + 1. **Task Management and Project Progress Tracking**

1. **Login and forget password:** Users can log in to the application using their email and password. If they forget their password, they can request a reset link to their email.
2. **Project and task creation**: Users can create new projects and assign them a name, a description, and a deadline. Users can also create tasks for each project and assign them a name, a description, a priority, and an estimated time.
3. **Calculating the progress of the project based on closing and new tasks:** The application calculates the percentage of completion of each project based on the number of closed and new tasks. The application also shows the remaining time and the overdue time for each project.
4. **Task time**: Users can track the time they spend on each task by starting and stopping a timer. The application records the total time spent on each task and shows it on the task page.
5. Adding and removing members from the project: Users can invite other users to join their projects by sending them an email invitation. Users can also remove members from their projects if they are no longer needed.
6. **Task page**: The task page has all the lists of new, progress, resolve, and close tasks. Users can see the tasks lists and can open tasks, which can be new, progress, resolved, or close. Users can change the status of the task and reassign the task to another member. Each task has a description.
   * + 1. **Task Management and Project Progress Tracking**

Under this heading, you can provide detailed information about the various aspects of task management and collaboration within the software system.

1. Task Time Tracking: Explain how users can track the time spent on tasks, either manually or using built-in timers or tracking mechanisms.
2. Member Management: Describe how project members are added and removed, including instructions on assigning roles or permissions to different members.
3. Task Status and Progress Calculation: Elaborate on how the project's overall progress is calculated based on the status of tasks, such as the ratio of closed tasks to open tasks or the percentage of tasks in the "Resolved" status.
4. Task Page Functionality: Detail the features and functionality of the task page, including the visibility of task lists (New, Progress, Resolved, Closed) and the ability for users to open tasks for further action.
5. Task Assignment and Reassignment: Provide instructions on how tasks can be assigned to specific members, how members can view and change task statuses, and how tasks can be reassigned to other members when necessary.
6. Task Descriptions: Emphasize the importance of task descriptions and how they serve as a means to provide relevant information and context for each task.

Reference and Bibliography

1. (Choudhary et al., 2019)
2. (Delli Bovi et al., 2021)
3. (Gonzalez Munoz et al., 2023)
4. (Liu et al., 2020)
5. (Zhang et al., 2018)
6. Choudhary, M., Perry, H. B., & Solomon, R. (2019). Effectiveness of a Census-Based Management Information System for Guiding Polio Eradication and Routine Immunization Activities: Evidence from the CORE Group Polio Project in Uttar Pradesh, India. *Am J Trop Med Hyg*, *101*(4\_Suppl), 33-44. <https://doi.org/10.4269/ajtmh.18-0935>
7. Delli Bovi, A. P., Manco Cesari, G., Rocco, M. C., Di Michele, L., Rimauro, I., Lugiero, A., Mottola, S., De Anseris, A. G. E., Nazzaro, L., Massa, G., & Vajro, P. (2021). Healthy Lifestyle Management of Pediatric Obesity with a Hybrid System of Customized Mobile Technology: The PediaFit Pilot Project. *Nutrients*, *13*(2). <https://doi.org/10.3390/nu13020631>
8. Gonzalez Munoz, J. L., Garcia-Agua Soler, N., & Garcia Ruiz, A. J. (2023). Study Protocol on Social Return on Investment (SROI) Project of the Surgical Waiting List Management System. *Healthcare (Basel)*, *11*(6). <https://doi.org/10.3390/healthcare11060825>
9. Liu, H., He, Y., Hu, Q., Guo, J., & Luo, L. (2020). Risk management system and intelligent decision-making for prefabricated building project under deep learning modified teaching-learning-based optimization. *PLoS One*, *15*(7), e0235980. <https://doi.org/10.1371/journal.pone.0235980>
10. Zhang, B., Ye, Y. W., Shen, X. Z., Mei, G., & Wang, H. X. (2018). Design and implementation of levee project information management system based on WebGIS. *R Soc Open Sci*, *5*(7), 180625. <https://doi.org/10.1098/rsos.180625>