

Task Management System

MCA213-MINI PROJECT

By

Yash Chauhan

3123015010115

Mini Project work carried out at

Department of Management

Faculty of Commerce and Management



VISHWAKARMA UNIVERSITY

PUNE

Second Semester 2023-2024

Task Management System

MCA213-MINI PROJECT

By

Yash Chauhan

3123015010115

Mini Project work carried out at

Department of Management

Faculty of Commerce and Management

**Submitted in partial fulfilment of Master Computer Application (MCA)
Programme**

Under the Supervision of

Dr. Yogesh Desale, Assistant Professor.



VISHWAKARMA UNIVERSITY

PUNE

Second Semester 2023-2024

CERTIFICATE

This is to certify that the mini project entitled **task management system** and submitted by **Yash Chauhan** having SRN. **31231418** for the partial fulfilment of the requirements of Master Computer Application (MCA) degree of Vishwakarma University, Pune, embodies the Bonafede work done by him/her under my supervision.

Signature of the Supervisor

Place :

Date :

Checklist of items for the Mini Project Report

This checklist is to be attached as the last page of the report.

This checklist is to be duly completed, verified and signed by the student.

1.	Is the final report neatly formatted with all the elements required for a technical Report?	Yes
2.	Is the Cover page in proper format as provide format	Yes
3.	Is the Title page (Inner cover page) in proper format?	Yes
4.	(a) Is the Certificate from the Supervisor in proper format? (b) Has it been signed by the Supervisor?	Yes Yes
5.	Is the Abstract included in the report properly written within one page? Have the technical keywords been specified properly?	Yes Yes
6.	Is the title of your report appropriate? The title should be adequately descriptive, precise and must reflect scope of the actual work done. Uncommon abbreviations / Acronyms should not be used in the title	Yes
7.	Have you included the List of abbreviations / Acronyms?	Yes
8.	Does the Report contain a summary of the literature survey?	Yes
9.	Does the Table of Contents include page numbers? (i). Are the Pages numbered properly? (Ch. 1 should start on Page # 1) (ii). Are the Figures numbered properly? (Figure Numbers and Figure Titles should be at the bottom of the figures) (iii). Are the Tables numbered properly? (Table Numbers and Table Titles should be at the top of the tables) (iv). Are the Captions for the Figures and Tables proper? (v). Are the Appendices numbered properly? Are their titles appropriate	Yes Yes Yes Yes Yes Yes
10.	Is the conclusion of the Report based on discussion of the work?	Yes
11.	Are References or Bibliography given at the end of the Report? Have the References been cited properly inside the text of the Report? Are all the references cited in the body of the report	Yes Yes Yes
12.	Is the report format and content according to the guidelines? The report should not be a mere printout of a Power Point Presentation, or a user manual. Source code of software need not be included in the report.	Yes

Declaration by Student:

I certify that I have properly verified all the items in this checklist and ensure that the report is in proper format as specified in the course handout.

Signature of the Student

Place:

Date:

Name: Yash Chauhan

SRN No.: 31231418

Mini Project Completion Certificate

MCA Programme

Batch: 2023-2025

This is to certify that Mr. Yash Chauhan student of Master of Computer Application (MCA) Programme under Faculty of Commerce and Management at Vishwakarma University, Pune, has completed the Mini Project titled Task Management System as per university guidelines for the two-year full-time MCA programme.

Faculty Guide (Name):

Faculty Guide (Signature):

ACKNOWLEDGEMENTS

This project would not have been possible without the support of many people. I would like to acknowledge and extend my heartfelt gratitude to the following persons who have made the completion of this project possible.

My Supervisor, **Dr. Yogesh Desale** for his full support and guidance throughout this project. My additional examiner **Prof. Gauri Kulkarni** for reviewing project and helping me in design.

I also thank to Vishwakarma University, Pune team for step by step updating regarding project with nice taxila support.

My team members who have helped me during various phases of the project. Last but not the least; I would like to express my love and gratitude to my beloved family, for their understanding & motivation, through the duration of this project.

ABSTRACT

An effective and efficient 'Task Management System' includes principal of “**Getting Things Done**”. It is the process of managing a task through its life cycle. It involves planning, testing, tracking, and reporting. Task management system can help either individual achieve goals, or groups of individuals collaborate and share knowledge for the accomplishment of collective goals.

Task management system may form part of project management and process management and can serve as the foundation for efficient workflow in an organization. Project managers adhering to task-oriented management have a detailed and up-to-date project schedule, and are usually good at directing team members and moving the project forward.

The purpose of our project titled, “Task Management System” is to automates the process by streamlining the handling of the work tasks requests thereby reducing the manual intervention. It is an automated support tool to be used by administrative staff of various departments. It eliminates manual request; assignment of work and the users can request any work tasks online.

Keyword: User- Friendly, Dynamic Questioning, Security, Dynamic Result.

Signature of the Student

Name: Bapat Neeraj

Date: 25/04/2024

Place: Pune

Signature of the Supervisor

Name: Prof. Yogesh Desale

Date: 25/04/2024

Place: Pune

INDEX

Chapter 1	Introduction	1
1.1	Introduction	1
1.2	Background	1
Chapter 2	System Analysis.....	2
2.1	Objective and Scope:.....	2
2.2	Definition	2
2.3	Requirements Gathering:.....	2
2.3.1	Functional Requirements	2
2.3.2	Non-Functional Requirements	2
Chapter 3	Software and Hardware Requirements	3
3.1	Software Specifications.....	3
3.2	Hardware Specifications	3
Chapter 4	Selected Software	4
4.1	PHP?.....	4
4.2	HTML5.....	4
4.3	Xampp	6
Chapter 5	System Design	7
5.1	UML Diagrams	7
5.1.1	Use case diagram:	7
5.2	Workflow Diagram for Web Application	10
5.3	Flow of the data dictionary.....	10
Chapter 6	System Implementation	11
6.1	Plan of implementation	11
6.2	Requirements Analysis:.....	11
6.3	Simple Code	12

Chapter 7	System Testing	15
7.1	Unit testing	15
	Testing Methodology.	15
	Type of Testing.	15
	Testing Tools.....	15
Chapter 9	Sample Screen	16
9.1	Login Page.....	16
9.2	CEO Dashboard.....	16
9.3	Project Page.....	17
9.4	Create Project Page	17
9.5	Add Employee Page	18
9.6	Manager Dashboard	18
9.7	Create Task Page.....	19
9.8	Employee Dashboard	19
Chapter 10	Conclusion	20
Chapter 11	Reference	21

Chapter 1 Introduction

1.1 Introduction

- The system starts with the creating CEO/Manger/Employee profile. Then the CEO assign project (with its name) to manager and manger allocate the project part as a task (with its name) to employer/employee team.
- Employee can complete the task with its daily updating and within time limit which can assign by the manger or CEO.
- Employee can submit/send the completed task (work can upload by pdf, docx etc.) to manager. Manager review the task on his/her desktop which will send by the employee. If manger can tick/checked the task as a done then the manager can send the work to CEO for review (CEO side).
- otherwise, the manger can reassign the task with the comment (regarding the task problems) and with new time limit.
- There are three users of the system.
 - CEO.
 - Manager.
 - Employee.

1.2 Background

- The background for the system stems from the need for efficient task management within organizational hierarchies. With three user roles—CEO, Manager, and Employee—the system streamlines task delegation, tracking, and completion processes. It facilitates clear assignment of projects from CEO to Manager, who then distributes tasks to employees. Employees update task progress and submit completed work, which managers review before sending to the CEO for final approval. This digital solution aims to enhance communication, efficiency, and accountability throughout the organization's workflow.

Chapter 2 System Analysis

2.1 Objective and Scope:

- The main objective of a Task Management System is to provide hands-on-solution for the task handling.
- TMS is web application to provide functionality of Project/Task handling smoothly and speedily without any obstacle within hierarchy of employee in a company.

2.2 Definition

- TMS is web application/mobile application to provide functionality of Project/Task handling smoothly and speedily without any obstacle within hierarchy of employee in a company.

2.3 Requirements Gathering:

2.3.1 Functional Requirements

- Registration:
- Manage Project/Task:
- Assign Project/Task:
- Tracking of Project/Task:
- Manage Team:
- Submit Project/Task:
- View Dashboard:
- Add Document:
- User wise Dashboard:

2.3.2 Non-Functional Requirements

- Security
- Performance
- Reliability
- Usability

Chapter 3 Software and Hardware Requirements

3.1 Software Specifications

Software's that are used for implementation is as follows:

- Language: HTML5, CSS, JavaScript, PHP
- Database: MySQL (PhpMyAdmin)
- Operating System: Windows 7, Windows 8, Windows 10 and above.

3.2 Hardware Specifications

The minimum hardware requirements for running the project are like below

- Processor: Multi-core processors (e.g., quad-core or higher) for handling concurrent userrequests.
- RAM: 4 GB
- Hard Disk: 8gb

Chapter 4 Selected Software

4.1 PHP?

PHP (Hypertext Preprocessor) is a widely used open-source server-side scripting language primarily used for web development. While PHP is not inherently focused on UI development like React.js, it plays a crucial role in backend development and server-side logic. Here are some key features and concepts associated with PHP:

- **Server-Side Scripting:** PHP is primarily used for server-side scripting, meaning it runs on a web server to generate dynamic web pages. It can interact with databases, manage sessions, handle form submissions, and perform various server-side tasks.
- **Embedded in HTML:** One of PHP's defining features is its ability to be embedded directly into HTML code. This allows developers to mix PHP code with HTML seamlessly, making it easier to generate dynamic content within web pages.
- **Extensive Database Support:** PHP has extensive support for interacting with databases, including MySQL, PostgreSQL, SQLite, and many others. This makes it well-suited for building database-driven web applications and managing data effectively.

4.2 HTML5

HTML, which stands for HyperText Markup Language, is the standard markup language for creating web pages. It provides the structure and semantics of content on the web. HTML is essential for web development and is used in conjunction with Cascading Style Sheets (CSS) and JavaScript to build interactive and visually appealing websites. Here are some key aspects of HTML:

HTML Document Structure:

An HTML document is structured with various elements, each serving a specific purpose. The basic structure includes the `<!DOCTYPE html>` declaration, the `<html>` element, the `<head>` section (containing metadata), and the `<body>` section (containing the content).

HTML Elements:

HTML elements are the building blocks of a web page. They are defined by tags, such as `<p>` for paragraphs, `<h1>` for headers, `<a>` for links, `` for images, and many others. Elements can have attributes that provide additional information.

Attributes:

HTML elements can have attributes that provide additional information or configuration. Attributes are included within the opening tag of an element and are written as name-value pairs. For example, the `<a>` element may have an `href` attribute specifying the link destination.

Headings and Paragraphs:

HTML provides heading elements `<h1>` to `<h6>` for defining headings of different levels. Paragraphs are created using the `<p>` element.

Links:

Links are created using the `<a>` (anchor) element. The `href` attribute is used to specify the URL to which the link points. Links can navigate to other pages, resources, or sections within the same page.

Images:

Images are inserted using the `` element. The `src` attribute specifies the path to the image file. Additional attributes, such as `alt` for alternative text, can provide more information about the image.

Lists:

HTML supports ordered lists `` (numbered) and unordered lists `` (bulleted). List items are defined using the `` element.

Forms:

Forms are created with the `<form>` element and can include input elements like text fields (`<input type="text">`), checkboxes (`<input type="checkbox">`), radio buttons (`<input type="radio">`), and more. The form's data is typically submitted to a server for processing.

Tables:

Tables are constructed using the `<table>` element, with rows defined by `<tr>` and cells by `<td>` (data cell) or `<th>` (header cell). Tables are commonly used to organize and display tabular data.

4.3 Xampp

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages.

MySQL is an open-source relational database management system (RDBMS) that is a central component of the XAMPP stack. It offers several advantages and features:

- **Ease of Installation:** XAMPP provides a straightforward installation process, allowing users to set up a local web server environment quickly and easily on their operating system.
- **Cross-Platform Compatibility:** XAMPP is compatible with major operating systems including Windows, Linux, and macOS, providing developers with a consistent environment regardless of their platform.
- **Apache Web Server:** XAMPP includes the Apache HTTP Server, a powerful and widely used web server software. Apache supports various features such as virtual hosting, URL rewriting, and SSL/TLS encryption.
- **MariaDB Database:** MariaDB, a community-developed fork of MySQL, is included as the default database in XAMPP. It offers similar functionality to MySQL, including support for SQL queries, transactions, and indexing.
- **PHP Interpreter:** XAMPP includes the PHP interpreter, allowing developers to write server-side scripts to generate dynamic web content. PHP is a popular scripting language for web development, known for its ease of use and broad community support.
- **Perl Interpreter:** XAMPP also includes the Perl interpreter, enabling developers to write CGI scripts and command-line utilities in Perl. Perl is a versatile programming language commonly used for text processing and system administration tasks.
- **phpMyAdmin:** XAMPP comes with phpMyAdmin, a web-based administration tool for managing MySQL databases. phpMyAdmin provides a user-friendly interface for performing tasks such as database creation, table management, and SQL query execution.

Chapter 5 System Design

5.1 UML Diagrams

Unified Modelling Language or UML Diagrams are used to represent the system diagrammatically.

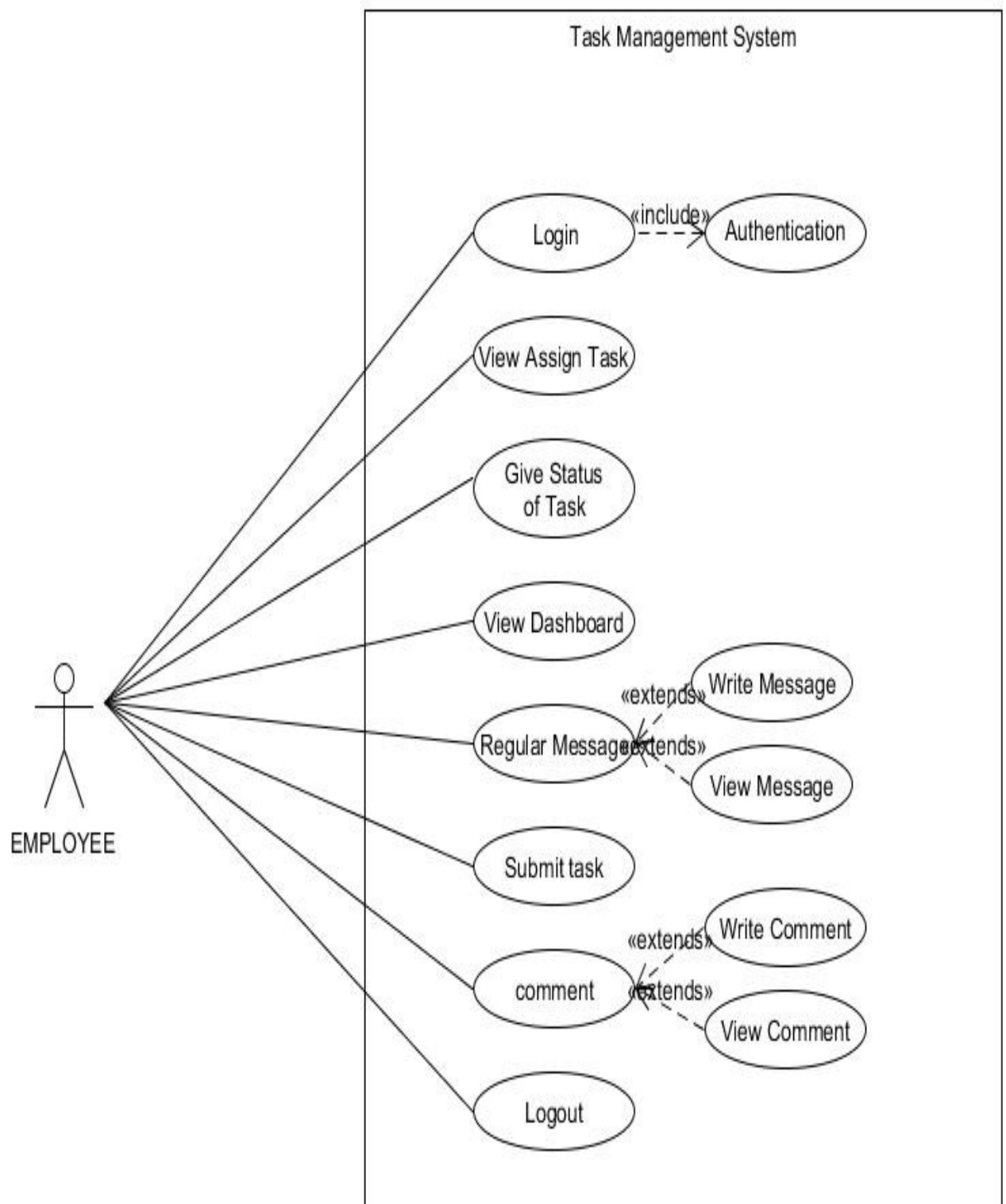
5.1.1 Use case diagram:

Use case diagrams are drawn to represent the functionality of the system.

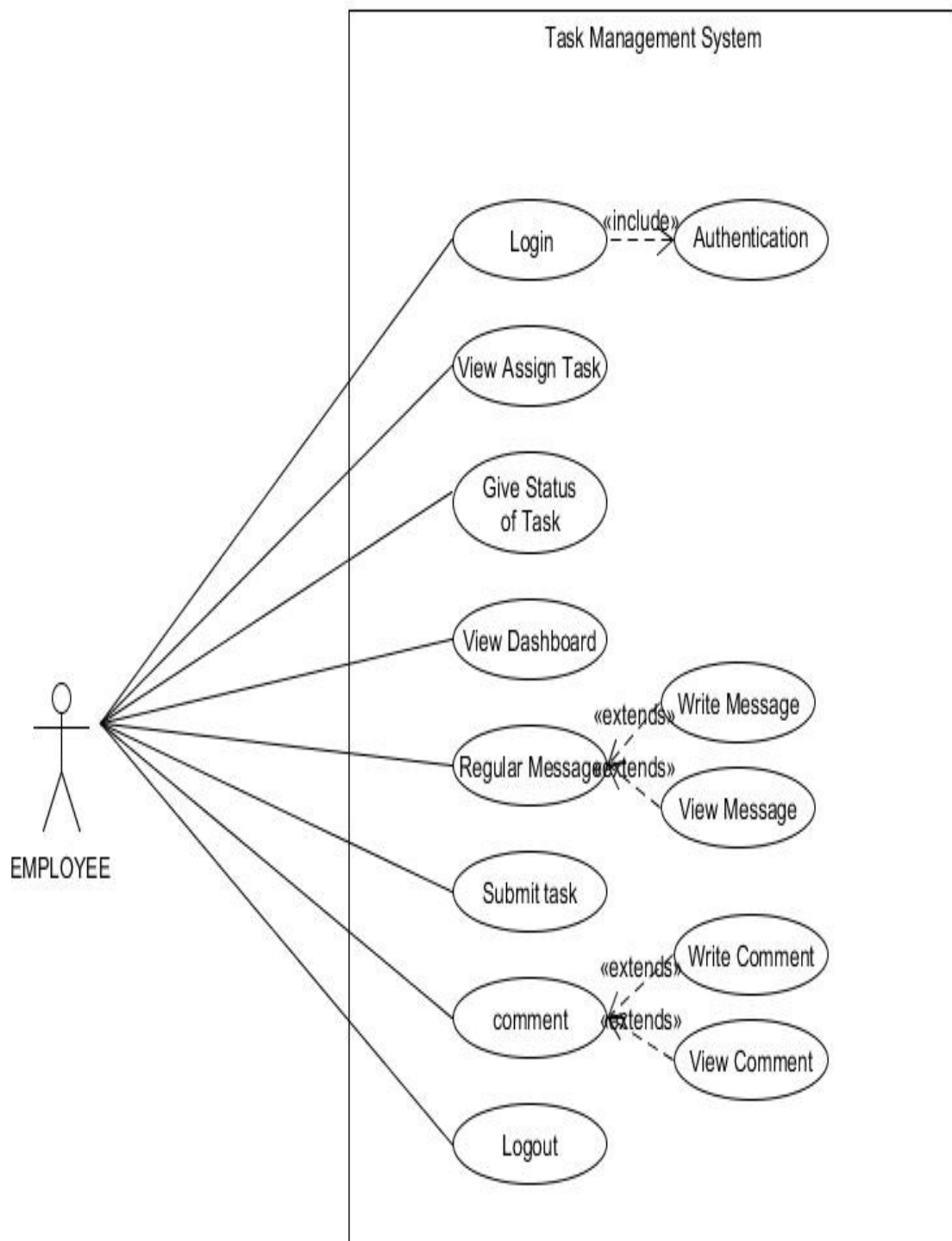
➤ CEO / Admin



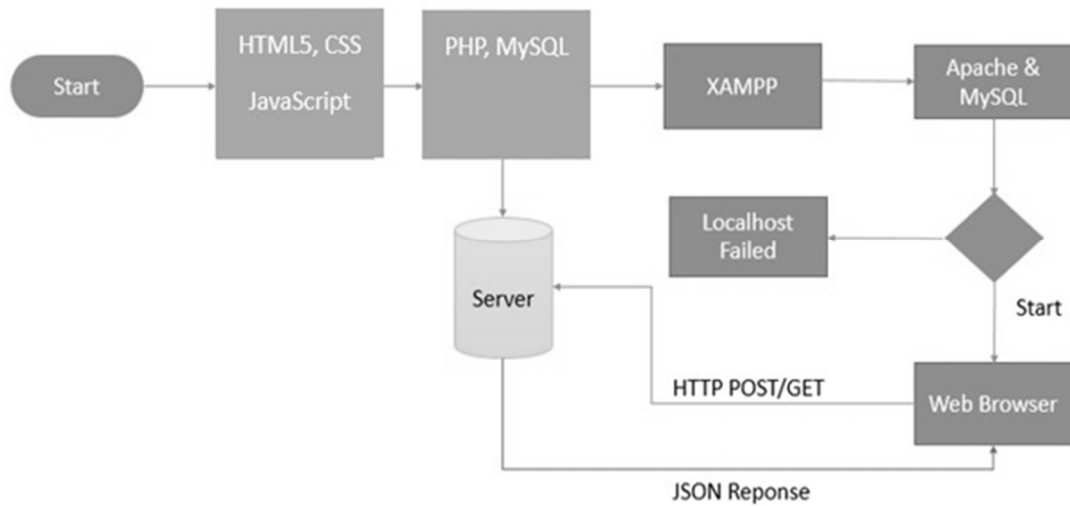
➤ Manager (Employee)



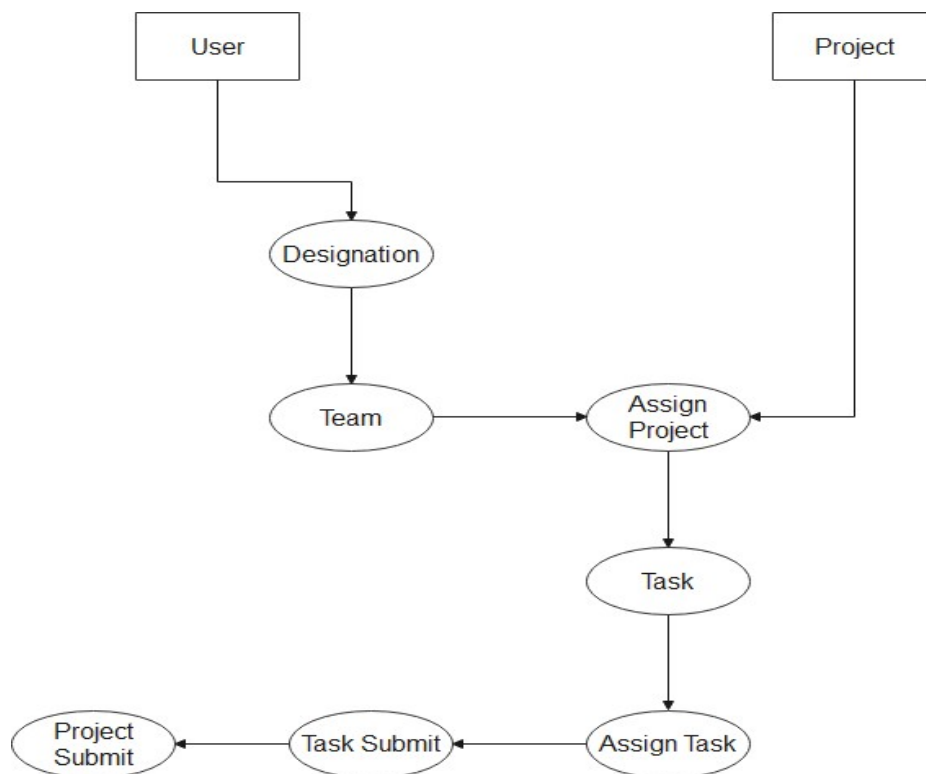
➤ Employee



5.2 Workflow Diagram for Web Application



5.3 Flow of the data dictionary



Chapter 6 System Implementation

6.1 Plan of implementation

System Development Life Cycle (SDLC) Model

Waterfall model:

The Waterfall Model follows a linear, step-by-step approach, where each phase must be completed before moving on to the next. This methodology is well-suited for projects with well-defined and stable requirements, where changes are expected to be minimal once the project has started.

6.2 Requirements Analysis:

Objective: Gather and document detailed requirements for the Task Management web app.

Activities: Engage with stakeholders to define the scope, functionalities, and constraints. Document all requirements thoroughly.

System Design:

Objective: Create a detailed system design based on the gathered requirements.

Activities: Design the system architecture, modules, and components. Develop specifications for hardware, software, and network requirements.

Implementation:

Objective: Code the Task Management web app based on the detailed system design.

Activities: Start coding individual modules and components. Conduct unit testing for each module to ensure they meet the specifications.

Integration:

Objective: Combine all developed modules to create the complete Task Management system.

Activities: Integrate individual modules, ensuring they work together seamlessly. Conduct integration testing to identify and resolve any issues.

Testing:

Objective: Conduct thorough testing to verify the system meets specified requirements

Activities: Perform system testing to validate the entire Task Management web app. Identify and fix defects or issues. Conduct user acceptance testing (UAT) with stakeholders.

Deployment:

Objective: Deploy the Task Management web app to the production environment.

Activities: Transfer the finalized system to the production servers. Conduct final checks and tests before making the app available to users. Train end-users and provide documentation.

Maintenance:

Objective: Address any issues or bugs that arise post-deployment and provide ongoing support.

Activities: Monitor the system for performance and issues. Address any bugs or problems promptly. Provide ongoing support and maintenance as needed.

6.3 Simple Code

```
<?php
session_start();
include_once 'db_Connection.php';
if (isset($_POST['login'])) {
    $r = ['role'];
    $email = $_POST['email'];
    $id = ['ID'];
    $name = ['Name'];
    // $login = ['user'];
    // md5($password = $_POST['password']);
    $password = $_POST['password'];
```

```
$add_rec = "SELECT * FROM registration WHERE Email = " . $email . " AND password  
= " . $password . ""'; // query for the login
```

```
$qryReturn = mysqli_query($conn, $add_rec) or die("Error in login " .  
mysqli_error($conn));
```

```
while ($row = mysqli_fetch_assoc($qryReturn)) {
```

```
    $id = $row["ID"];
```

```
    $r = $row["role"];
```

```
    $email = $row["Email"];
```

```
    $name = $row['Name'];
```

```
}
```

```
if (mysqli_num_rows($qryReturn) > 0) { //validate the query
```

```
    echo "<center> Login sucessfully</center>";
```

```
} else {
```

```
    echo $conn->error;
```

```
}
```

```
$_SESSION['id'] = $id;
```

```
$_SESSION['email'] = $email;
```

```
$_SESSION['Name'] = $name;
```

```
$_SESSION['user'] = $r;
```

```
if ($r == 'C') {
```

```
    $_SESSION['user'] = "CEO";
```

```
} elseif ($r == 'M') {
```

```
    $_SESSION['user'] = "Manager";
```

```
} else if ($r == 'E') {
```

```
    $_SESSION['user'] = "Employee";
```

```

    }

    $login = $_SESSION['user'];

    if ($login == "CEO") {

        header("Location:Admin_Dashboard.php");

    } else if ($login == "Manager") {

        header("Location:Manager-dashboard.php");

    } else if ($login == "Employee") {

        header("Location:Employee_dashboard.php");

    }

}

?>

```

Database Connection Code

```

<?php

//main connection file for both admin & front end

$servername = "localhost"; //server

$username = "root"; //username

$password = ""; //password

$dbname = "managent"; //database

$db = mysqli_connect($servername, $username, $password, $dbname);

// Check connection

if (!$db) {    //checking connection to DB

    die("Connection failed: " . mysqli_connect_error());

}

?>

```


Chapter 7 System Testing

7.1 Unit testing

Unit testing is a crucial aspect of the software development process, including when developing an eCommerce web app using the Waterfall Model or any other development methodology.

Unit testing involves testing individual units or components of a software application in isolation to ensure they perform as intended.

Testing Methodology.

Our testing approach followed a comprehensive strategy, encompassing unit testing, integration testing, and system testing. Test-driven development (TDD) practices were implemented to ensure that each component met its requirements before integration.

Type of Testing.

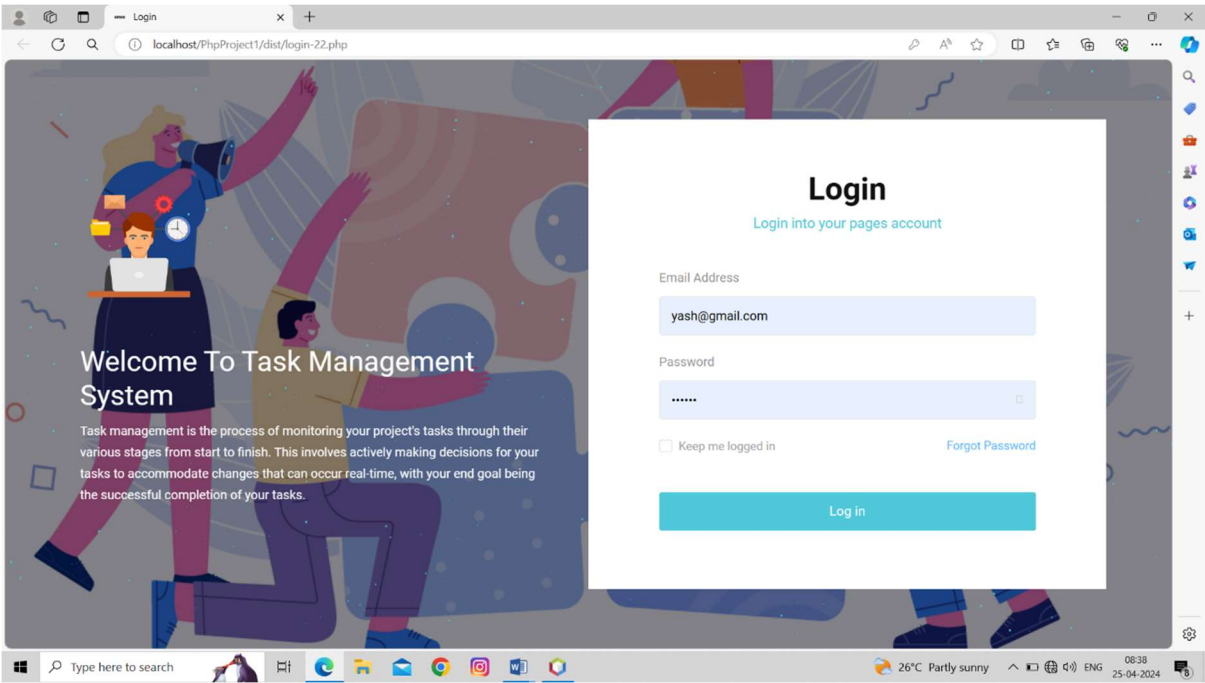
- **Unit Testing:** Ensured individual components functioned as intended.
- **Integration Testing:** Verified the proper collaboration of integrated components.
- **System Testing:** Validated the end-to-end functionality of the entire E-commerce site.
- **User Acceptance Testing (UAT):** Involved stakeholders to confirm the system's compliance with business requirements.

Testing Tools.

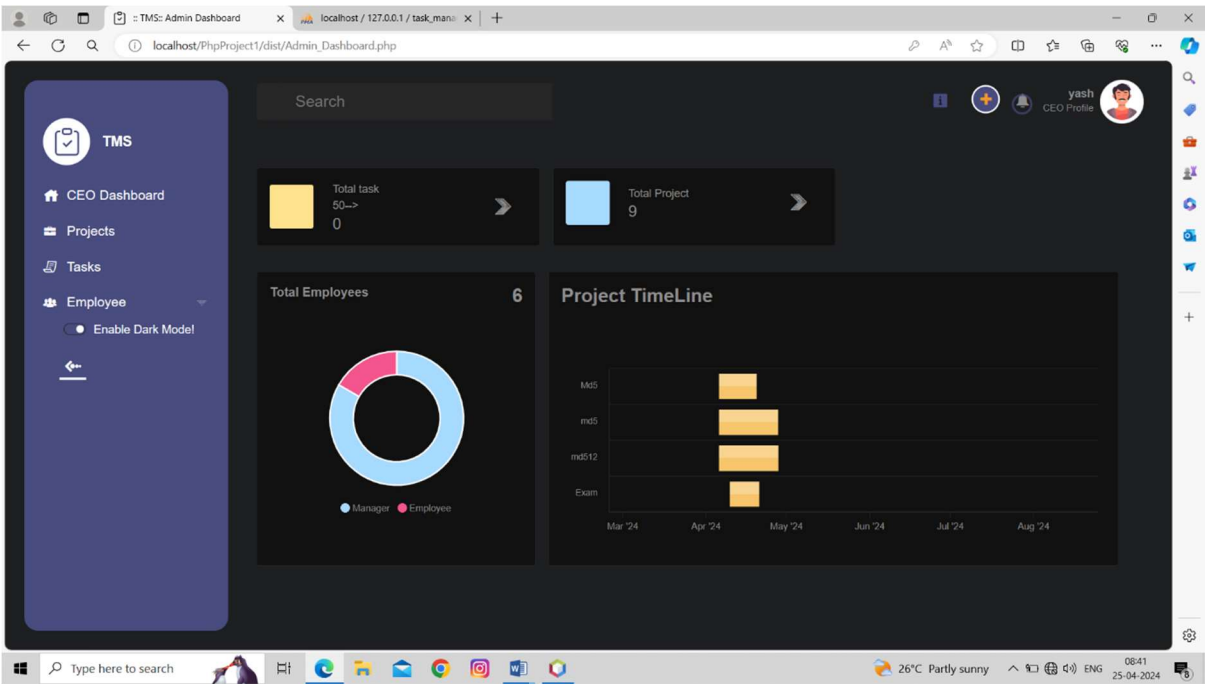
- **Jest and Enzyme:** For front-end unit testing.
- **Thunder Client:** For API testing and validation.
- **Selenium:** For automated browser testing. (**Used in System**)
- **Jenkins:** For continuous integration and automated testing.

Chapter 9 Sample Screen

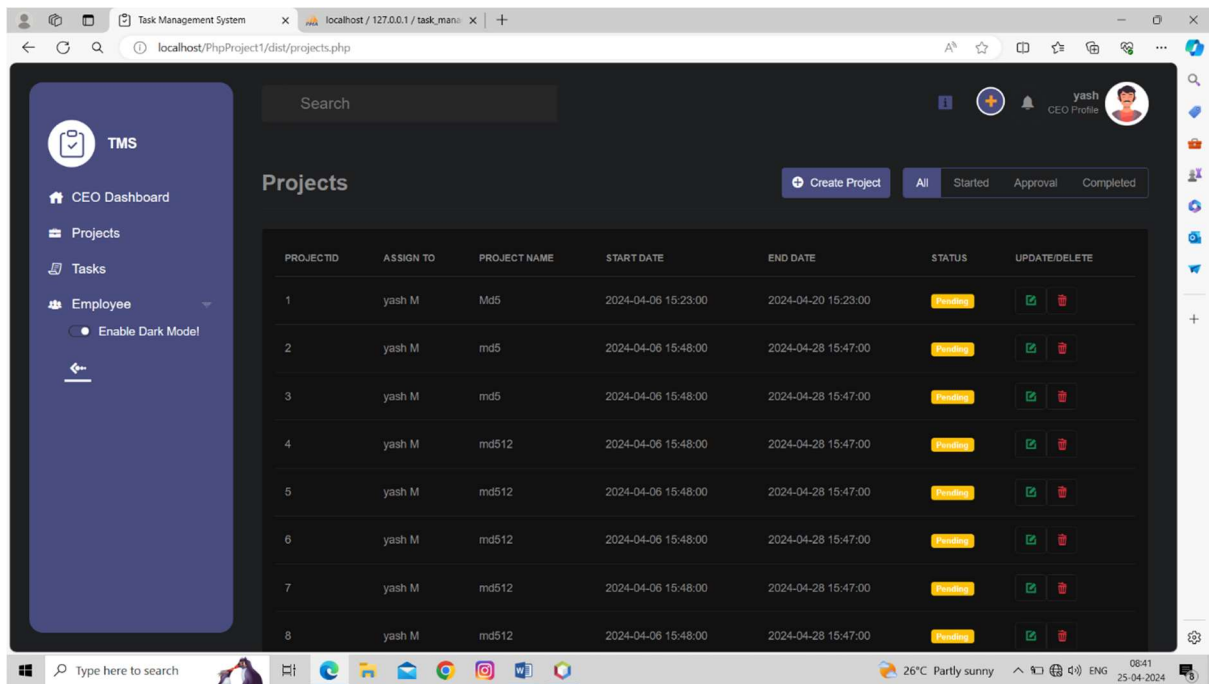
9.1 Login Page



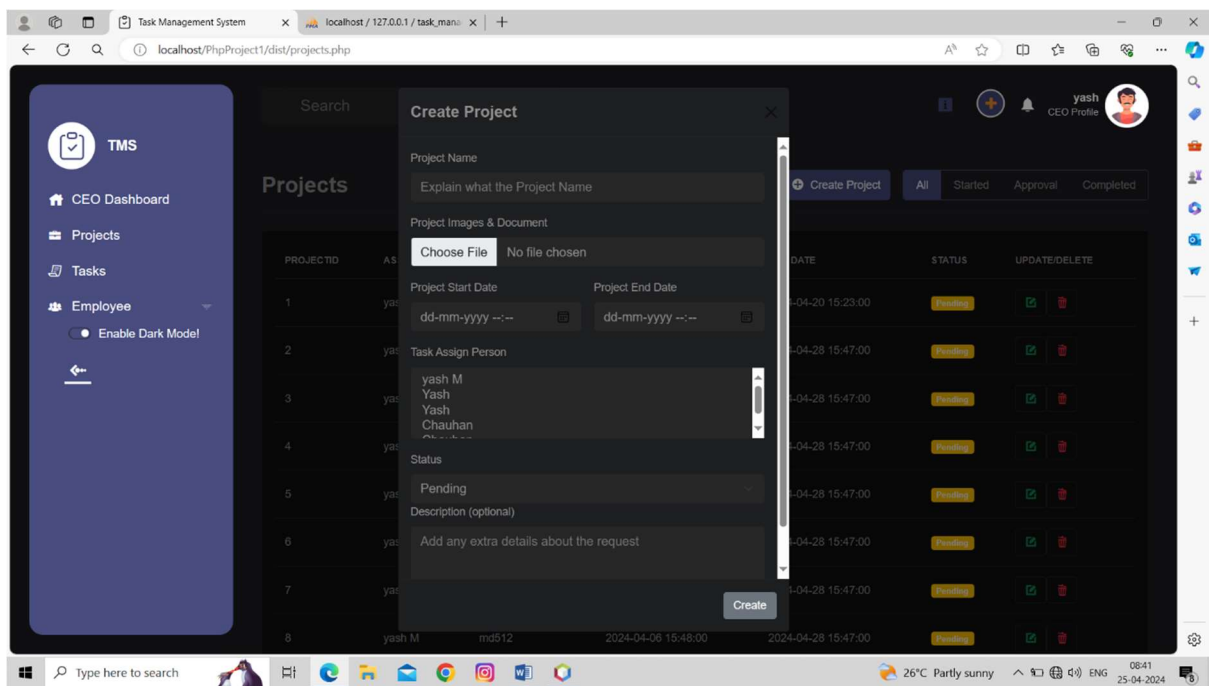
9.2 CEO Dashboard



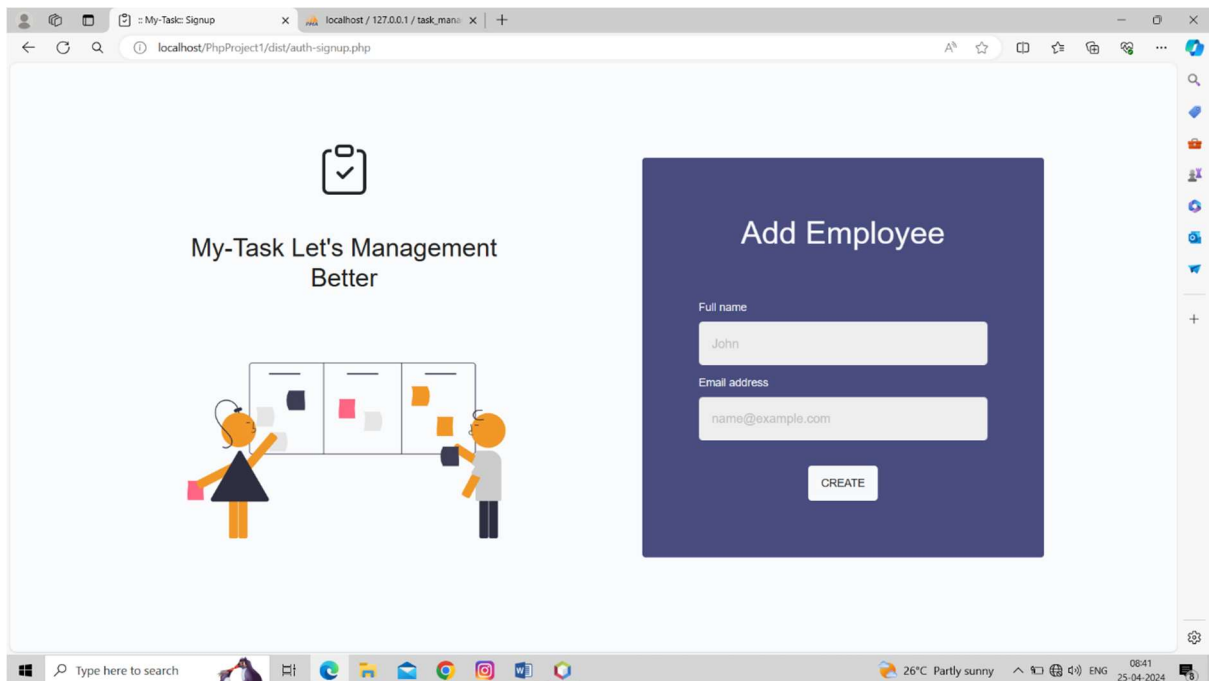
9.3 Project Page



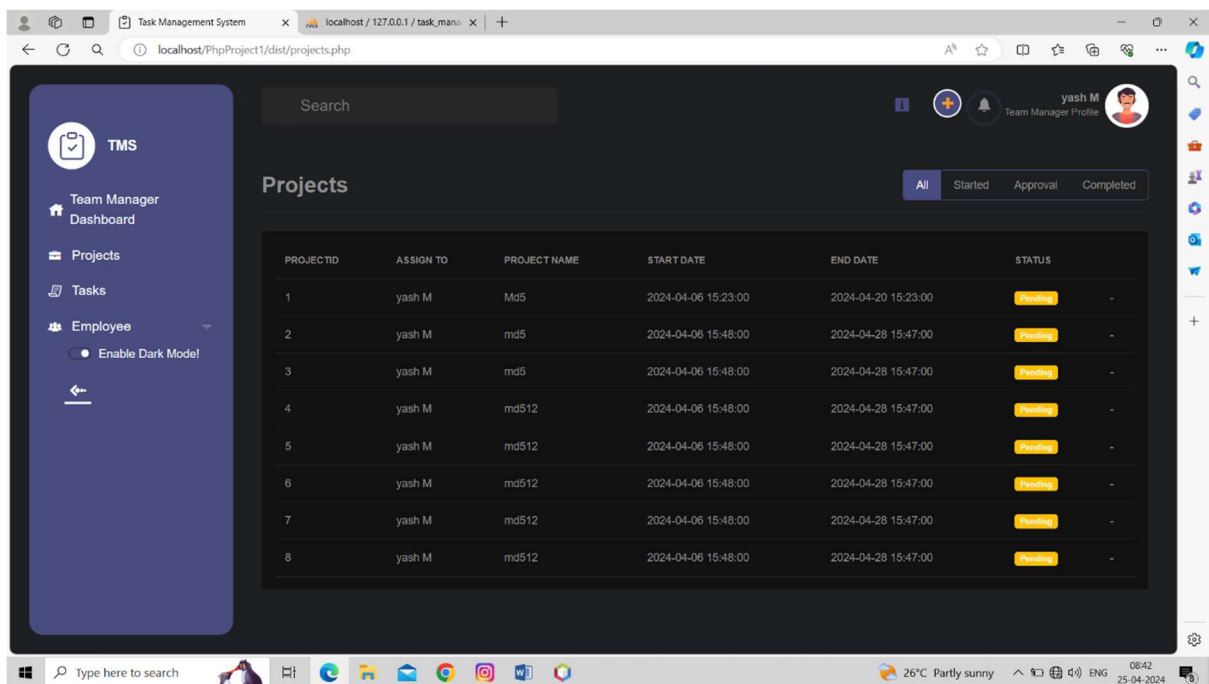
9.4 Create Project Page



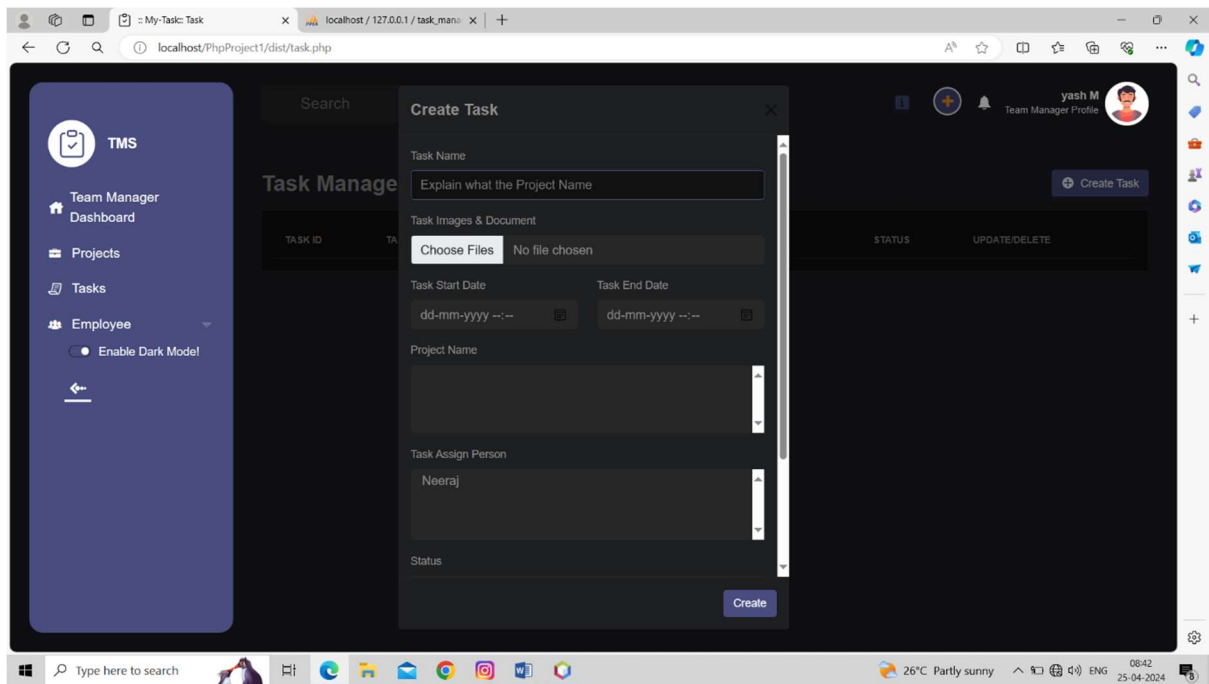
9.5 Add Employee Page



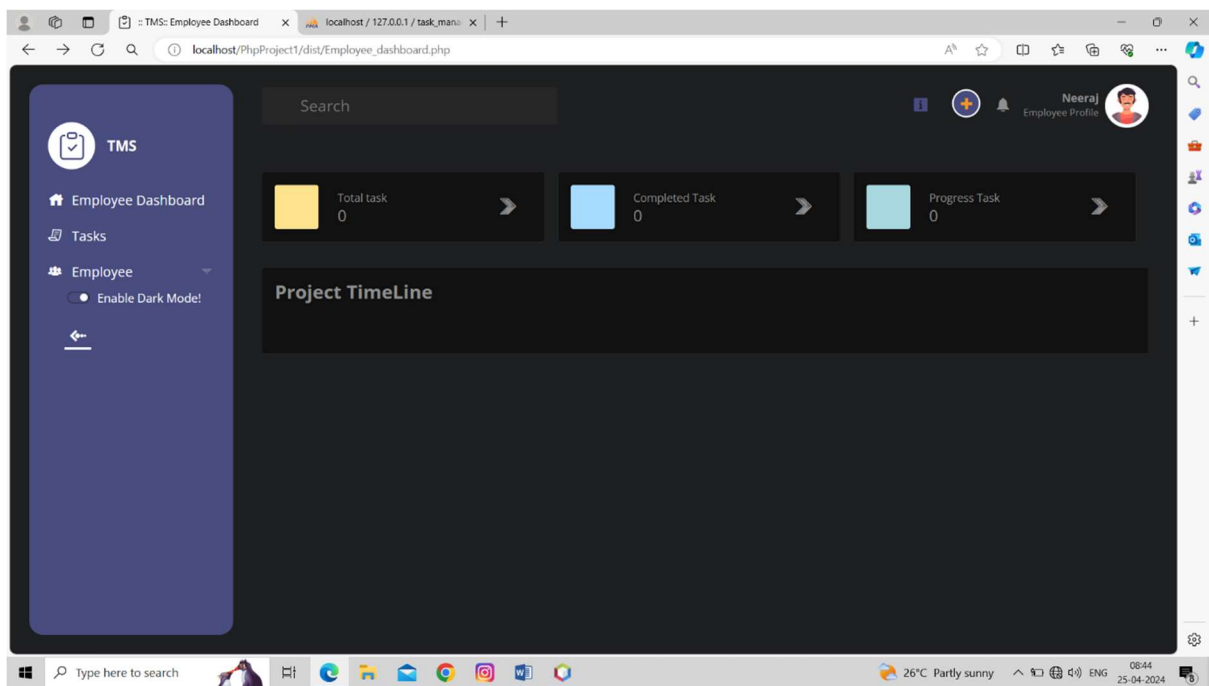
9.6 Manager Dashboard



9.7 Create Task Page



9.8 Employee Dashboard



Chapter 10 Conclusion

In conclusion, the development of a Task Management website is a complex and multifaceted process that requires careful planning, strategic decision-making, and meticulous execution. Whether following a traditional Waterfall Model or a more iterative approach, certain key considerations and activities are essential for success.

The market research phase sets the foundation by understanding the target audience, analyzing competitors, and identifying market trends. This information informs the conceptualization and planning stage, where the purpose, features, and goals of the Task Management website are defined. Careful selection of the technology stack ensures the chosen tools align with scalability, security, and integration requirements.

The design phase is critical for creating an attractive and user-friendly interface, involving wireframing, prototyping, and finalizing the visual design and branding elements. The subsequent development phase brings the design to life, implementing features such as project listings, user authentication, and payment processing.

Testing plays a vital role in ensuring the Task Management website functions correctly and meets quality standards. From unit testing individual components to system testing and user acceptance testing, identifying and resolving issues is integral before deployment. The deployment phase involves making the website accessible to users, including server configurations, training, and documentation.

Ultimately, a successful Task Management website requires a holistic approach that considers not only the technical aspects of development but also the user experience, market dynamics, and ongoing maintenance. Flexibility and responsiveness to changing market conditions and user expectations are key to ensuring the long-term success and sustainability of an Task Management website.

Chapter 11 Reference

- www.stackoverflow.com/
- <https://www.tutorialspoint.com/>
- www.w3schools.com
- www.google.com/
- <https://transno.com/>
- <https://www.sintelly.com/>
- <https://brilliant.org/>
- <https://deepstash.com/>
- <https://medium.com/>
- <https://www.selenium.dev/selenium-ide/>