PROJECT REPORT

Task Manager Web Application

-Each user has their own tasks.

-Includes deadlines and status(pending, completed).

---

1. Title Page

Project Title: Task Manager Web Application
Project Type: Minor Project
Internship Domain: Web Development
Organization: Naviotech Solution
Submitted By: *ISHANT LINGWAL*

---

2. Abstract

This project presents the development of a Task Manager Web Application designed to help users manage their daily tasks efficiently. The application allows users to register, log in, and maintain a personal list of tasks with deadlines and status tracking. Each user can add new tasks, mark tasks as completed, and delete tasks as required. The system is developed using basic web technologies and follows a client-server architecture. This project helps in understanding full-stack web development concepts and practical implementation of CRUD operations.

---

3. Introduction

Task management is an important activity in both personal and professional life. With increasing workloads, it becomes necessary to organize tasks efficiently. The Task Manager Web Application provides a simple and user-friendly platform for managing tasks digitally.

This project is developed as a minor project during a web development internship to gain hands-on experience in frontend and backend integration.

4. Objectives

The main objectives of this project are:

- To understand the basics of full-stack web development
- To implement user authentication using username and password
- To perform CRUD (Create, Read, Update, Delete) operations
- To develop a user-based task management system
- To gain experience in handling backend logic and data storage

---

5. Tools and Technologies Used

Frontend Technologies

- HTML for structuring web pages
- CSS for designing and styling the interface
- JavaScript for client-side logic and interaction

Backend Technologies

- Node.js for server-side execution

Database

- JSON file-based storage for storing user and task data

---

6. System Architecture

The Task Manager Web Application follows a client-server architecture.
The frontend interacts with the backend using HTTP requests. The backend processes these requests and stores the data in a JSON file. Each user has a separate task list, ensuring user-specific data handling.

---

7. Implementation

User Authentication

Users can register using a username and password. During login, credentials are verified, and successful login allows access to the task dashboard.

Task Management

After logging in, users can:

- Add tasks with deadlines
- View their task list
- Mark tasks as completed
- Delete tasks

Usernames are handled in a case-insensitive manner to avoid duplication issues. Proper error handling is implemented to ensure smooth operation.

---

8. Data Handling

The application uses a JSON file to store data. Each user object contains a list of tasks, where each task includes:

- Task title
- Deadline
- Status (Pending or Completed)

This lightweight data storage approach is suitable for beginner-level projects.

---

9. Results

The Task Manager Web Application successfully performs all required operations. Users can manage their tasks efficiently through a clean and professional interface. The backend functions correctly, and data persists properly during the application runtime.

---

10. Challenges Faced

Some challenges encountered during development include:

- Handling case-sensitive username comparison
- Preventing backend crashes due to invalid or empty JSON files
- Debugging frontend and backend communication
- Fixing routing and server startup issues

These challenges were resolved through systematic debugging and code optimization.

11. Conclusion

The Task Manager Web Application fulfills all the requirements of a minor internship project. It demonstrates a clear understanding of frontend and backend development, user authentication, and task management. This project helped in gaining practical knowledge of web development concepts and improved problem-solving skills.

12. Future Scope

- Adding password encryption
- Integrating a database such as MySQL or MongoDB
- Implementing task priorities
- Adding user profile management
- Making the application fully responsive

End of Report