Mini Project Report: To Do List

_____

1. Project Details

•   Title: To Do List Application

•   Submitted by: Harsh Bhardwaj

•   Roll Number: 2300290120104

•   Course: B.Tech, Computer Science

•   Section : 3B

_____

2. Table of Contents

_____

3. Introduction

This mini project aims to create a simple and functional To Do List web application. The goal of the project is to help users manage their tasks effectively by providing features to add, update, delete, and mark tasks as completed.

_____

4. Objectives

- Develop a user-friendly interface for managing tasks.

- Enable users to add, edit, delete, and track the progress of tasks.

- Ensure responsive design for usability across devices.

_____

5. Tools & Technologies Used

- Language: HTML, CSS, JavaScript

- Framework: None (Pure Frontend)

- Database: Local Storage (Browser-based)

- Others: None

_____

6. Methodology

User Requirements

1. A user should be able to add tasks with descriptions.

2. Tasks should be displayed in a list.

3. Users should be able to mark tasks as complete or delete them.

System Architecture

- Frontend: HTML/CSS for structure and style, JavaScript for functionality.

- Backend: No backend, as data is stored in the browser's Local Storage.

_____

7. Features of the Application

- Add Task: Users can add new tasks to their list.

- Delete Task: Users can remove tasks once completed or if no longer needed.

- Mark Task as Complete: Tasks can be marked complete visually (using styles).

_____

8. Screenshots

_____
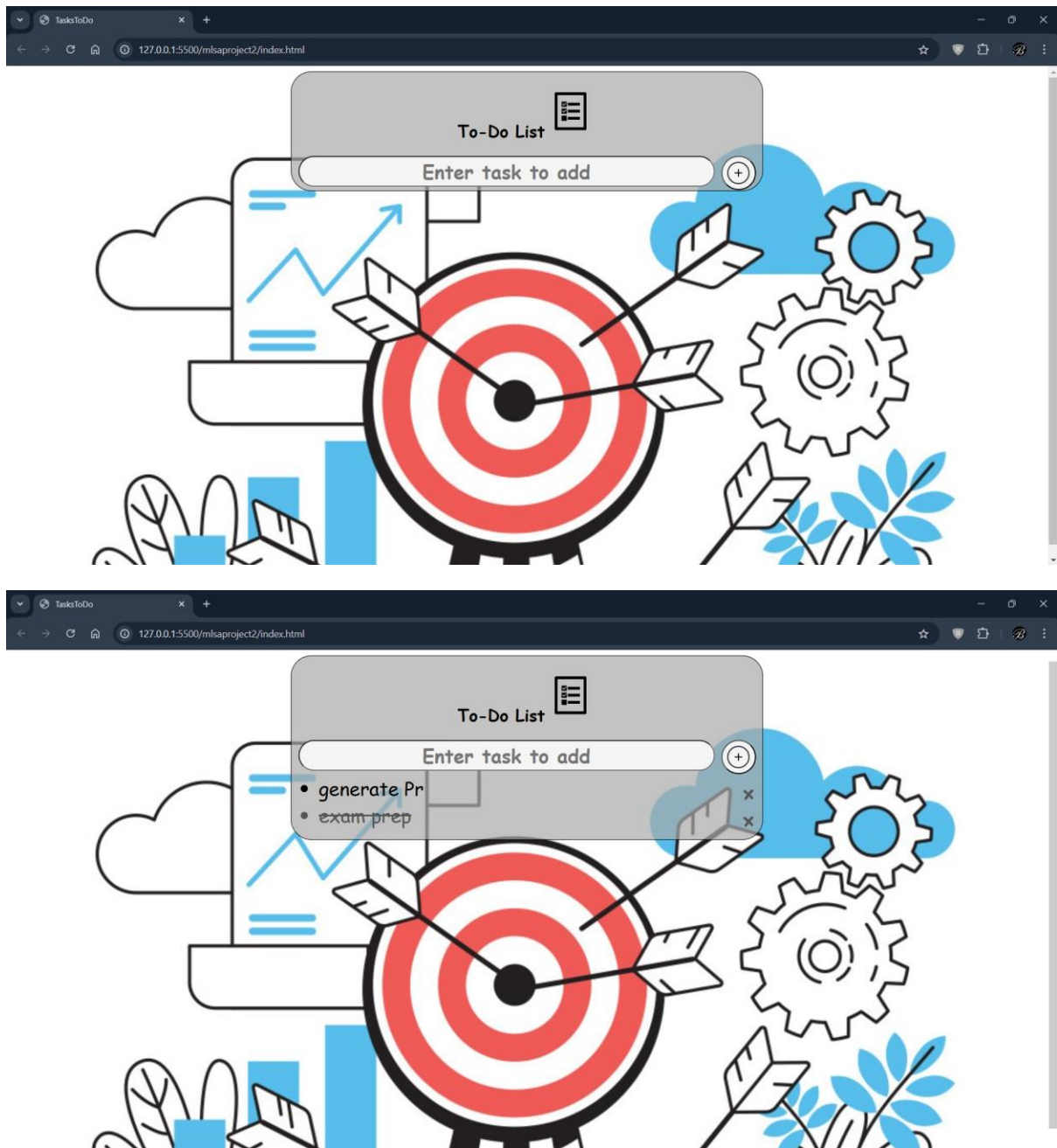
9. Testing

The application was tested across different browsers (Chrome, Firefox) and devices (desktop). It performed smoothly, and tasks were properly stored in the browser's Local Storage.

———————————————————————————

10. Conclusion

The To Do List project helped in understanding the basic concepts of web development, including HTML for structure, CSS for styling, and JavaScript for functionality. It demonstrated how front-end technologies can be used to build interactive applications.

———————————————————————————

11. Future Scope

•   Add functionality to edit tasks.

•   Implement categories or priorities for tasks.

•   Connect to a database for storing tasks persistently across sessions.

———————————————————————————

12. References

•   W3Schools for HTML, CSS, and JavaScript documentation.

•   Mozilla Developer Network (MDN) for advanced JavaScript techniques.