Front-End UI/UX Mini Project

TO-DO LIST

1. Team Members

- Nevan Miranda
  246120
  nevan.miranda@btech.christuniversity.in

- Y Arrin Nischal Paul
  2462174
  Yandapalli.arrin@btech.christuniversity.in

- Reyan V Rinoj
  2462136
  reyan.v@btech.christuniversity.in

Course: UI/UX Design Fundamentals

Instructor Name: Dhiraj Alate

Institution: Christ (Deemed to be University)

Date of Submission: 26/09/2025

2. Abstract

This project focuses on designing and developing a To-Do List Application that allows users to manage their daily tasks effectively. The application provides features for adding, editing, deleting, and marking tasks as completed. Additionally, it offers task filtering options to help users focus on pending or completed tasks.

The primary objective is to create an interactive, responsive, and user-friendly task management system using modern front-end technologies such as HTML, CSS, JavaScript, Bootstrap, and jQuery. The final outcome is a productivity tool that helps users organize their workflow efficiently across different devices.

3. Objectives

- Enable users to add, edit, and delete tasks.
- Allow marking tasks as completed to track progress.
- Provide filtering options for better task organization (e.g., completed, pending).
- Ensure interactivity with smooth UI updates using JavaScript/jQuery.
- Design a responsive interface that works across desktops, tablets, and mobile devices.

4. Scope of the Project

- Covers task management on the front end only (no backend or database integration).
- Users can manage tasks dynamically during their session.
- Provides filtering options for improved productivity.
- Focused on clean UI/UX principles and mobile-first responsive design.
- Can be extended later with backend integration for permanent storage.

5. Tools & Technologies Used

Tool/Technology	Purpose
HTML5	Markup and content structure
CSS3	Styling and layout management
JavaScript	Interactivity and form validation
Bootstrap	Responsive grid system and UI components
JQuery	Simplified DOM manipulation and effects
VS Code	Code editor
Chrome DevTools	Testing and debugging

6. HTML Structure Overview

- Used semantic HTML tags: <header>, <nav>, <main>, <section>, <footer>.
- Sections included: Task Input Form, Task List, Filters, and Completed Tasks.
- List items dynamically generated with JavaScript for each task.

7. CSS Styling Strategy

- External CSS with modular, well-structured styles.
- Bootstrap used for responsive grid layouts.
- Flexbox for feature alignment and testimonials.
- Media queries for fine-tuning design across breakpoints.

8. Key Features

Feature	Description
Mark Task as Completed	Users can toggle tasks between completed and pending states.
Task Management	Add, edit, and delete tasks dynamically.
Interactivity	Real-time updates using JavaScript and jQuery.
Filter Tasks	View all tasks, only completed tasks, or only pending tasks.
Responsive Design	Works seamlessly on desktop, tablet, and mobile devices.

9. Challenges Faced & Solutions

Challenge	Solution
Handling task updates dynamically	Used JavaScript event listeners and DOM manipulation
Keeping UI responsive across devices	Implemented Bootstrap grid and custom media queries
Managing completed vs pending tasks	Added conditional rendering with filtering buttons
Providing smooth interactivity	Used jQuery for quick DOM updates and animations

10. Outcome

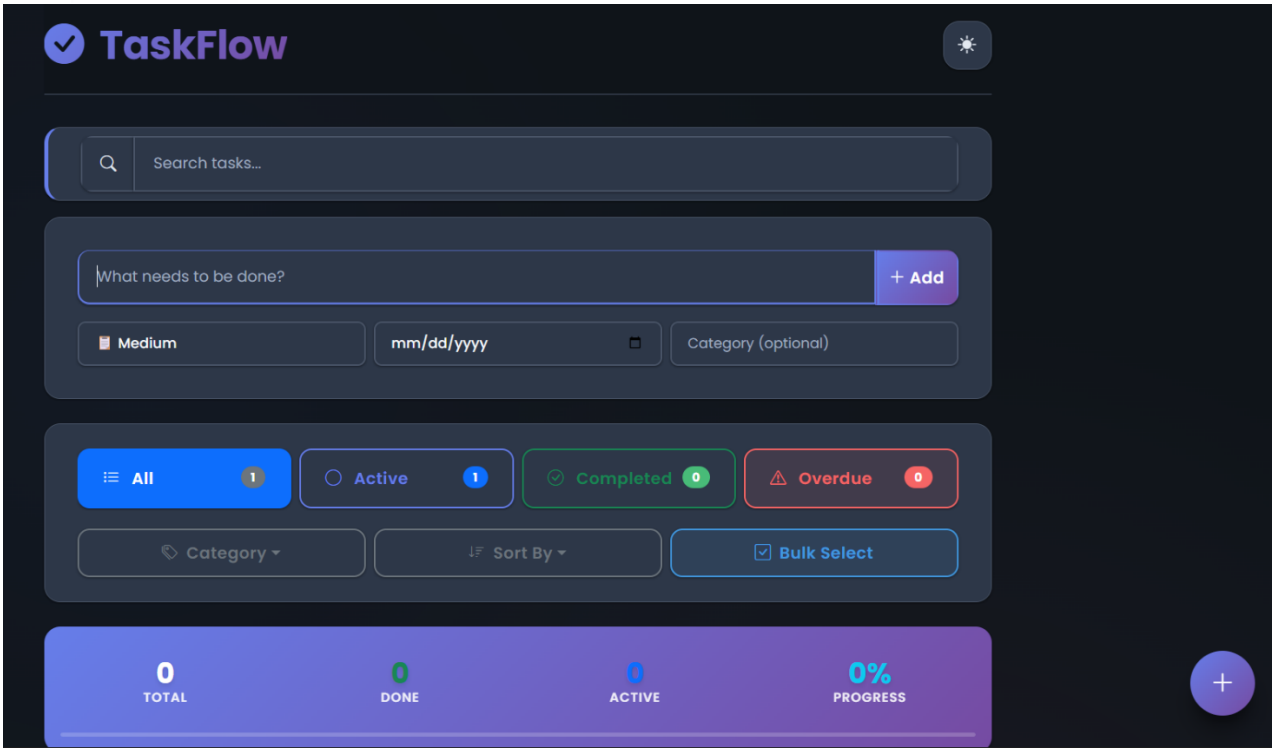
- Successfully developed a functional and interactive To-Do List Application.
- Implemented task management, filtering, and completion tracking effectively.
- Designed a clean and responsive UI accessible across devices.
- Gained hands-on experience with JavaScript interactivity and Bootstrap responsiveness.

11. Future Enhancements

- Add backend/database integration (e.g., Firebase, MySQL) to store tasks permanently.
- Enable user authentication for personalized task management.
- Add drag-and-drop task reordering for better usability.
- Implement reminders/notifications for due tasks.
- Provide dark mode for better accessibility.

12. Sample Code

13. Screenshots of Final Output



11. Conclusion

The To-Do List Application demonstrates how front-end technologies can be applied to build a practical productivity tool. By combining HTML, CSS, JavaScript, Bootstrap, and jQuery, the application ensures interactivity, task filtering, and responsive design. This project enhanced understanding of event-driven programming, DOM manipulation, and responsive UI development, while also highlighting potential future improvements through backend integration and advanced features.

12. References

- L&T LMS : https://learn.lntedutech.com/Landing/MyCourse