Front-End UI/UX Mini Project

To-Do List

1. Title Page

- **Project Title**: *To-Do List*
- **Submitted By**:
  - *Team Members- 1)Jinkala Shahanaz 2)Priyavarseinee Anthiyur Somasundaram*
  - *Roll Number 1)2462086 2)2462127*
  - *College-E-mailid 1) jinkala.shahanaz@btech.christuniversity.in 2)priyavarseineeanthiyur@btech.christuniversity.in*
- **Course**: *e.g., UI/UX Design Fundamentals*
- **Instructor Name**: Mr *Dhiraj*
- **Institution**: *e.g., Christ University*
- **Date of Submission**: 2*6/09/2025*

2. Abstract

This project focuses on the design and development of a **front-end To-Do List web application** with a pastel rainbow theme. The core purpose of the project is to build a **lightweight yet functional task manager** that enables users to manage daily activities efficiently while maintaining a pleasant and engaging user experience.

The project incorporates **task creation, editing, deletion, filtering, and completion tracking**. Additionally, data persistence is achieved using the **Local Storage API**, ensuring that tasks are saved and accessible even after a page refresh or browser restart. The UI has been carefully designed with **pastel sticky-note colors**, hover effects, rounded cards, and badges to indicate task status.

Core technologies used include **HTML5 for structure, CSS3 and Bootstrap 5 for styling and responsiveness, and jQuery for interactivity**. The final outcome is a responsive, interactive, and visually appealing to-do list that reflects a practical understanding of **UI/UX principles, user-centered design, and front-end web development practices**.

3. Objectives

The main objectives of this project are:

1. **Design & Aesthetics**: To design a visually pleasing and pastel-themed user interface that makes task management enjoyable.
2. **Core Functionality**: To provide users with the ability to add, edit, delete, and mark tasks as completed.
3. **Filtering**: To allow filtering of tasks into “Active”, “Completed”, and “All” categories for better organization.
4. **Persistence**: To ensure task data is not lost on page reload by storing it using Local Storage.
5. **User Experience**: To create a responsive, accessible, and simple interface that works seamlessly across devices.
6. **Learning Goal**: To gain hands-on experience in applying front-end technologies and practicing modern UI/UX principles.

4. Scope of the Project

The scope of the Pastel Rainbow To-Do List project is limited to **front-end functionality** without server-side integration. The features include:

- Adding new tasks dynamically.
- Editing tasks inline without page reload.
- Deleting tasks individually.
- Marking tasks as completed with strikethrough formatting.
- Filtering tasks based on status (All, Active, Completed).
- Clearing either completed tasks or all tasks at once.
- Persisting data using Local Storage.

Exclusions: The application does not include multi-user support, authentication, or cloud-based synchronization. It is not integrated with backend services or databases. However, it is scalable enough to be extended with these features in the future.

5. Tools & Technologies Used

Tool/Technology	Purpose
HTML5	Provides semantic structure to the app including input fields, buttons, and list containers.
CSS3	Adds styling, pastel colors, hover effects, shadows, and overall visual hierarchy.
Bootstrap 5	Ensures responsiveness with grid layout and prebuilt utility classes.
jQuery	Handles DOM manipulation, event handling, and interactivity for add/edit/delete.
JavaScript (Local Storage API)	Ensures persistent storage of tasks between sessions.
VS Code	Used as the development IDE.
Chrome DevTools	Debugging and testing during development.

6. HTML Structure Overview

- The HTML structure follows a **clean and semantic layout**:
- A `<div>` container (Bootstrap-based) for responsiveness.
- An `<input>` field with an associated “Add” button for task creation.
- A `<select>` dropdown and buttons for filtering and clearing tasks.
- An `<ul>` (unordered list) that dynamically renders all tasks as `<li>` elements.
- Each `<li>` includes:
    - A checkbox for completion.
    - A span showing the task text.
    - A badge to display “Active” or “Done”.
    - Edit and Delete buttons.

7. CSS Styling Strategy

The **CSS** focuses on making the app visually attractive while keeping it functional:

- **Pastel Rainbow Colors**: Applied sticky-note themed backgrounds rotating across tasks.
- **Rounded Corners & Shadows**: Cards and list items have smooth edges and soft shadows for a modern look.
- **Hover Effects**: Scaling and shadow enhancements for interactive feel.
- **Typography**: Poppins font for clean, modern readability.
- **Custom Buttons**: Add button styled green, Edit in blue, Delete in red for instant recognition.
- **Badges**: Yellow for Active tasks and Green for Completed tasks to improve visibility.

8. Key Features

Feature	Description
Task Management	Users can add, edit, delete, and mark tasks as completed.
Filtering Options	Active, Completed, or All tasks view.
Clear Functions	Option to clear only completed tasks or all tasks entirely.
Status Indicators	Badges clearly show the state of each task.
Persistent Storage	Local Storage ensures data is not lost on refresh.
Responsive Design	Works seamlessly on mobile, tablet, and desktop.

9. Challenges Faced & Solutions

Challenge	Solution
Data loss on refresh	Implemented Local Storage API to save and load tasks automatically.
UI becoming cluttered with multiple actions	Used Bootstrap utilities and grouped action buttons neatly.
Inline editing breaking layout	Used a hidden input field toggled dynamically during edit mode.
Hover effects inconsistent across devices	Simplified CSS animations for mobile compatibility.
Maintaining multiple features without breaking code	Used modular functions and jQuery event delegation.

10. Outcome

The outcome of the project is a **functional, attractive, and user-friendly To-Do List application**. It successfully integrates all intended features such as adding, editing, deleting, and filtering tasks. The pastel rainbow theme creates a cheerful environment that makes productivity enjoyable.

Key outcomes include:

- Strengthened front-end coding skills.
- Clear understanding of **state management with Local Storage**.
- Hands-on application of **UI/UX principles**.
- Development of a lightweight but effective productivity tool.

11. Future Enhancements

To make the application more robust, the following enhancements are proposed:

1. **Drag-and-Drop Reordering** of tasks for better flexibility.
2. **Priority Levels** with color-coded tags (High/Medium/Low).
3. **Search Functionality** to quickly find tasks.
4. **Due Dates & Reminders** for task deadlines.
5. **Dark Mode Toggle** for accessibility.
6. **Cloud Syncing** with Firebase or a backend database.
7. **Gamification** with streaks or achievement badges for completing tasks.

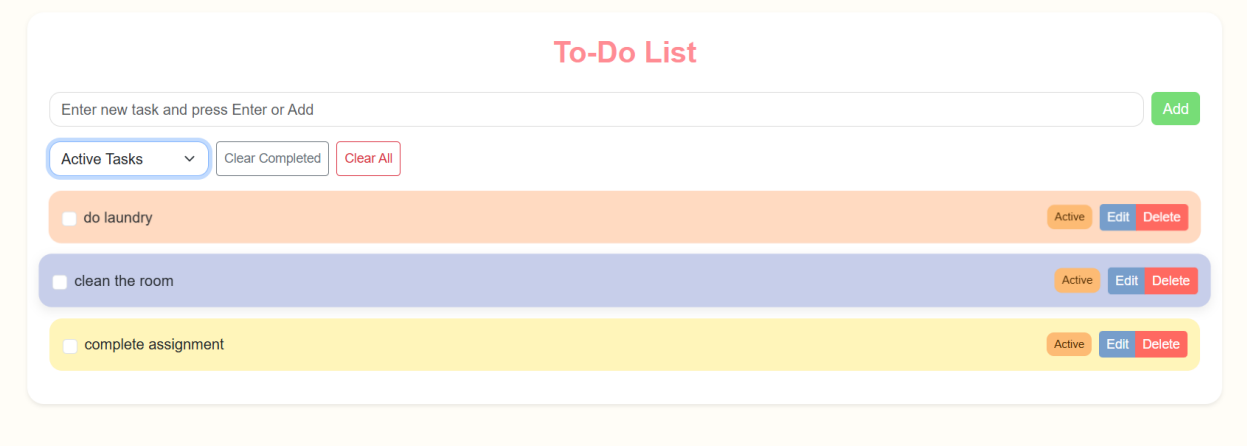
12. Sample Code

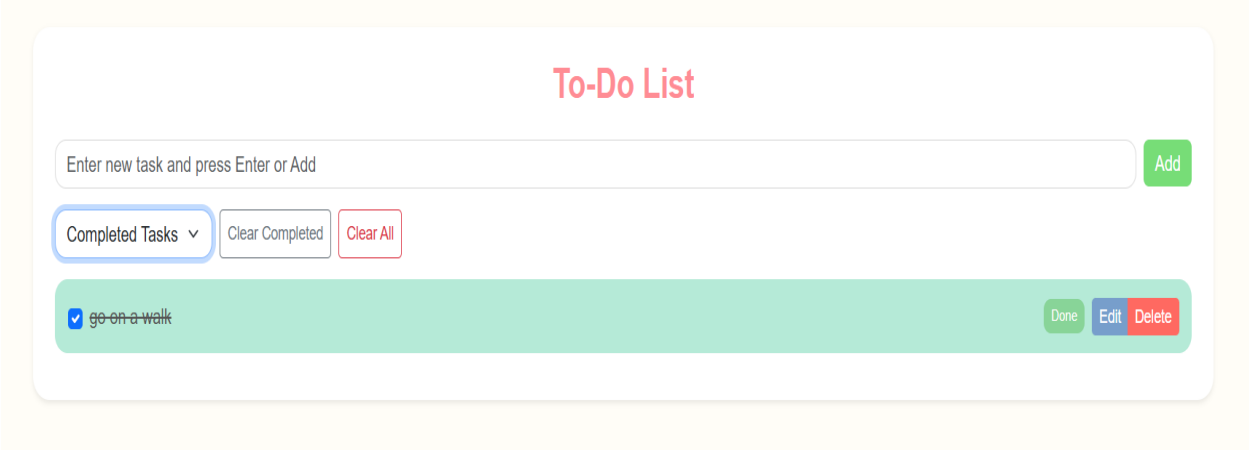
todolist.html > html > head

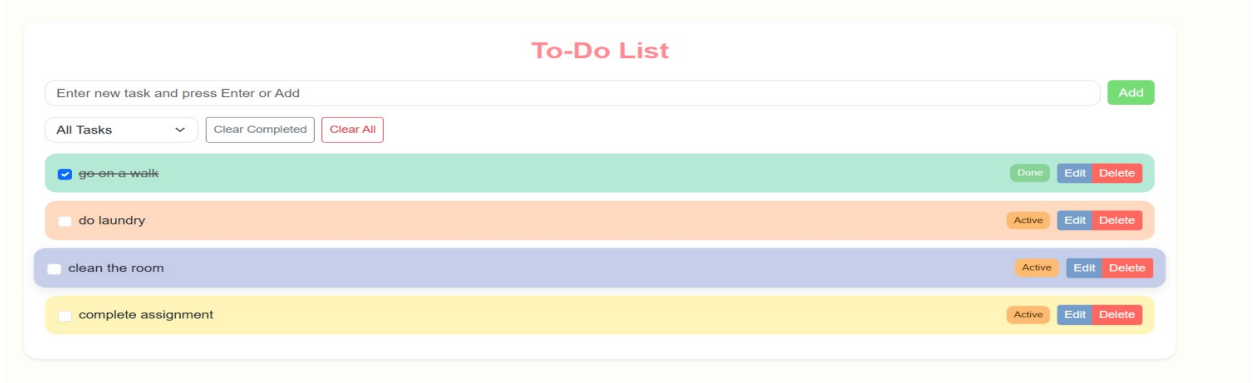
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width,initial-scale=1" />
  <title>Pastel Rainbow To-Do List</title>

  <!-- Bootstrap 5 -->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet"
  <!-- jQuery -->
  <script src="https://code.jquery.com/jquery-3.7.1.min.js"></script>

  <style>
    body {
      background: #fffdf7;
      min-height: 100vh;
      font-family: "Poppins", sans-serif;
    }
    .card {
      border: none;
      border-radius: 18px;
      background: #ffffff;
      box-shadow: 0 6px 20px rgba(0,0,0,0.06);
    }
    .completed {
      text-decoration: line-through;
      color: #555;
    }
    .list-group-item {
      border: none;
      margin-bottom: 12px;
      border-radius: 14px !important;
      padding: 14px;
      font-weight: 500;
      font-size: 1rem;
      transition: all 0.2s ease;
    }
```

```
<body>
  <div class="container py-5">
    <div class="card shadow-sm p-4">
      <h2 class="text-center mb-4 fw-bold" style="color:#ff8c94;"> To-Do List</h2>

      <div class="row g-2 mb-3">
        <div class="col">
          <input id="taskInput" class="form-control" placeholder="Enter new task and press Enter or Add">
        </div>
        <div class="col-auto">
          <button id="addTaskBtn" class="btn btn-primary">Add</button>
        </div>
      </div>

      <div class="mb-3 d-flex flex-wrap gap-2">
        <select id="filterTasks" class="form-select w-auto">
          <option value="all">All Tasks</option>
          <option value="active">Active Tasks</option>
          <option value="completed">Completed Tasks</option>
        </select>
        <button id="clearCompleted" class="btn btn-outline-secondary btn-sm">Clear Completed</button>
        <button id="clearAll" class="btn btn-outline-danger btn-sm">Clear All</button>
      </div>

      <ul id="taskList" class="list-group"></ul>
    </div>
  </div>
```

13. Screenshots of Final Output







14. Conclusion

The **Pastel Rainbow To-Do List project** demonstrates how even a simple productivity application can be enhanced with **UI/UX principles, responsive design, and interactivity**. The project successfully achieved its objectives by combining core technologies like HTML, CSS, Bootstrap, jQuery, and Local Storage.

This project was not only a technical implementation but also a **learning journey**. It improved problem-solving skills, enhanced creativity in UI design, and gave hands-on practice with event-driven programming.

In conclusion, the project highlights the value of **front-end development** in solving real-world productivity challenges. It stands as a strong example of how design and functionality can go hand in hand to create applications that are both useful and enjoyable.

15. References

- W3Schools: https://www.w3schools.com
- MDN Web Docs: https://developer.mozilla.org
- Bootstrap Documentation: https://getbootstrap.com
- jQuery Documentation: https://api.jquery.com
- FreeCodeCamp Tutorials: https://www.freecodecamp.org
- L&T LMS (Christ University Resource): https://learn.lntedutech.com