

PROJECT 1: Report on To-Do List Application

LALIT: 2462516

Email id: lalit.kumar@btech.christuniversity.in

CHARUSHREE: 2462509

Email id: charushree.s@btech.university.in

AARVISHA: 2462501

Email id: aarvisha.yadav@btech.christuniversity.in

COURSE: UI/UX DESIGN FUNDAMENTALS

INSTRUCTOR NAME : DHIRAJ ALATE

INSTITUTION : CHRIST (DEEMED TO BE) UNIVERSITY, KENGERI CAMPUS

DATE OF SUBMISSION : 26/09/2025

Title

To-Do List Application using HTML, CSS, Bootstrap, and JavaScript (jQuery)

Objective

The objective of this project is to design a simple **To-Do List Application** that allows users to:

- Add new tasks.
- Mark tasks as completed.
- Edit existing tasks.
- Delete tasks.
- Filter tasks based on status (All, Completed, Active).

Software Requirements

- Any Web Browser (Google Chrome, Edge, Firefox).
- Code Editor (VS Code, Sublime Text, Notepad++).
- Internet connection (for Bootstrap & jQuery CDN).

Algorithm / Steps

1. Start with HTML structure to create the web page.
2. Add an input field and button for entering new tasks.
3. Display tasks inside a list ().
4. Use buttons for editing and deleting tasks.
5. Add a checkbox to mark tasks as completed.
6. Use jQuery functions to handle user actions like:

- Adding tasks.
 - Deleting tasks.
 - Marking tasks complete.
 - Editing tasks.
 - Filtering tasks.
7. End the program.

Code Explanation

1. HTML Structure

- Defines the page layout, input field, filter dropdown, and task list.

2. CSS Styling

- Adds background color and a `.completed` class to strike through finished tasks.

3. Bootstrap

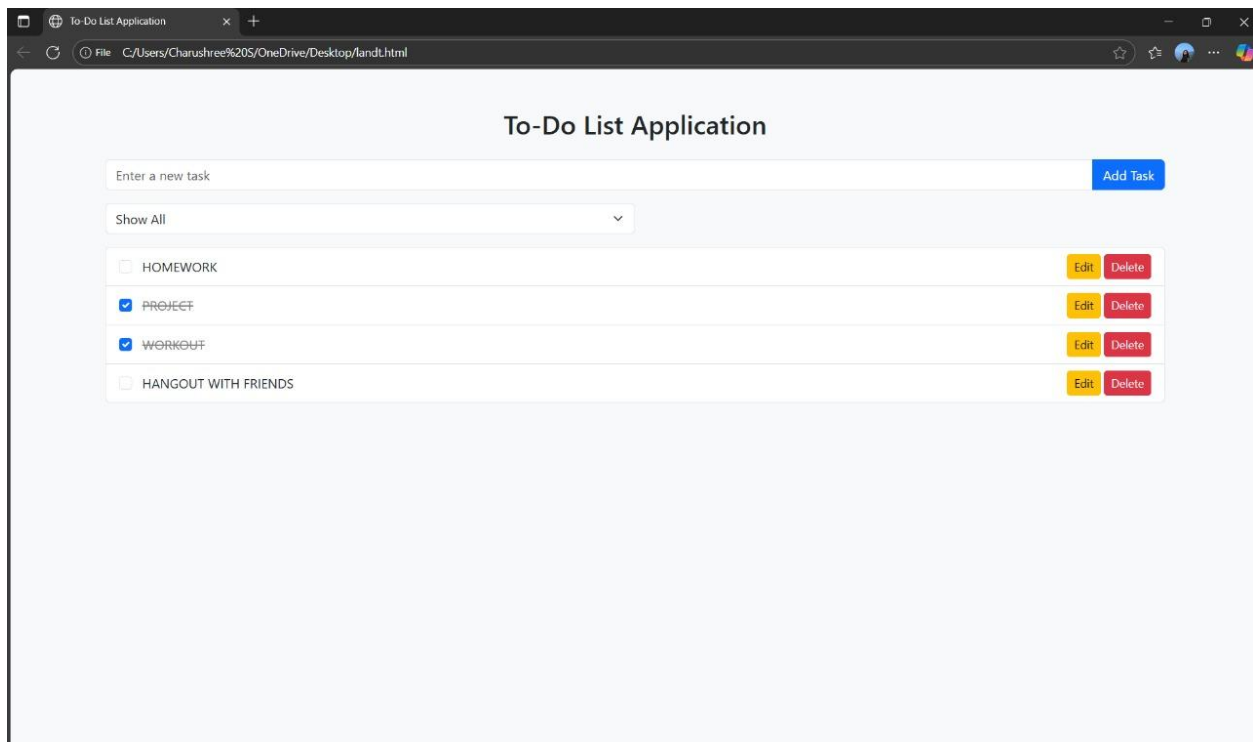
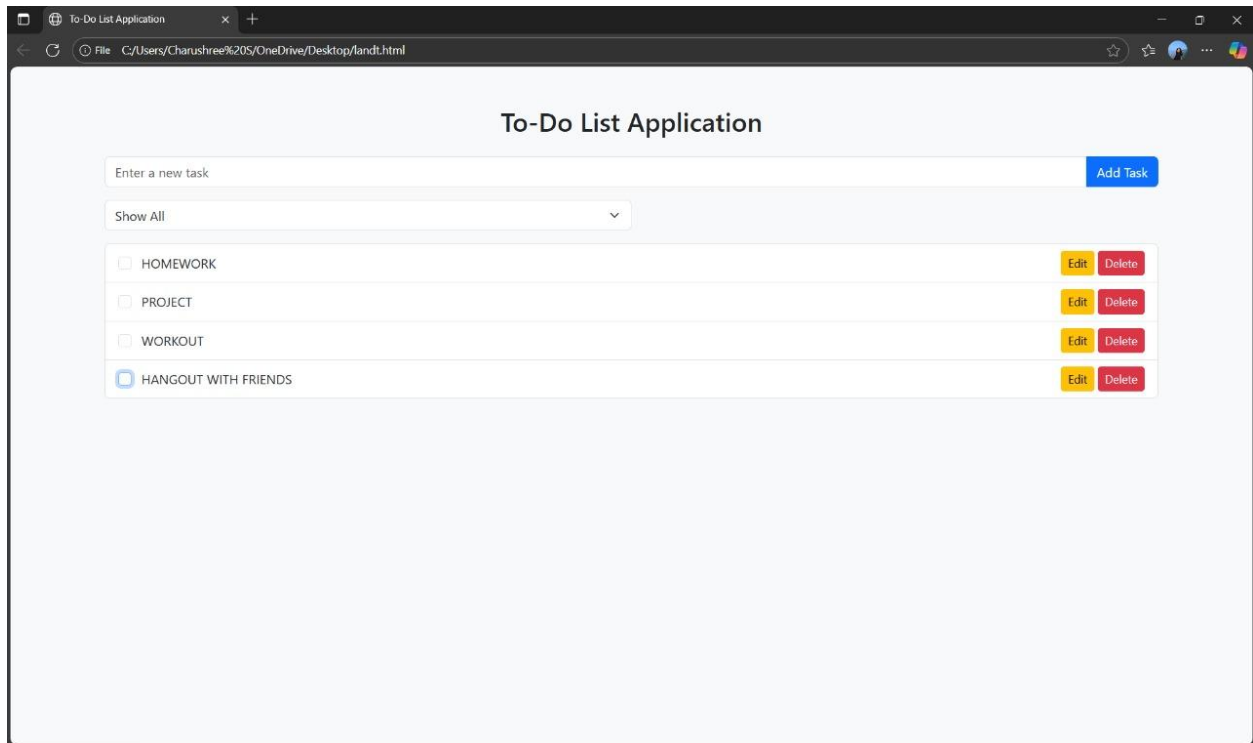
- Provides responsive design and button styles.

4. jQuery Functions

- **Add Task:** Appends new tasks to the list.
- **Delete Task:** Removes a selected task.
- **Mark as Completed:** Toggles strike-through using checkbox.
- **Edit Task:** Prompts the user to change task text.
- **Filter Tasks:** Shows All / Completed / Active tasks.

Output

When the program is run in a browser:



PROJECT 2: Report on Customer Feedback Form

Title

Customer Feedback Form using HTML, CSS, Bootstrap, and jQuery

Objective

The aim of this project is to design a simple **Customer Feedback Form** where users can:

- Rate a service using star ratings.
- Provide written comments.
- View all submitted feedback.
- Delete any feedback if needed.

This project helps in understanding how **forms, event handling, and dynamic updates** work in web development.

Software Requirements

- Web Browser (Google Chrome, Edge, or Firefox).
- Code Editor (VS Code, Sublime Text, Notepad++).

- Internet connection (for Bootstrap, Bootstrap Icons, and jQuery).

Algorithm / Steps

1. Create an HTML page with form structure.
2. Add **star icons** for rating and store the value in a hidden field.
3. Provide a **textarea** for entering comments.
4. Add a **Submit button** to submit feedback.
5. On clicking stars, update the rating visually and store the value.
6. On form submission:
 - Validate that both rating and comments are entered.
 - Create a feedback entry (rating + comment).
 - Display the feedback under the “All Feedback” section.
 - Reset the form.
7. Allow the user to **delete feedback** dynamically.
8. If no feedback exists, display a message "No feedback yet."

Code Explanation

1. HTML Section

- form: Contains star rating and comments box.
- div#feedbackList: Stores all submitted feedbacks.

2. CSS Section

- Styles stars to appear large and gold when selected.

3. Bootstrap & Icons

- Used for layout, buttons, and star icons.

4. jQuery Script

- **Star Rating:** Clicking a star highlights it and all stars before it.
- **Form Submission:**
 - Collects rating and comments.
 - Adds a feedback item with stars and text.
- **Delete Feedback:** Removes selected feedback.
- Shows “No feedback yet” if list becomes empty.

Output

When this program runs in a browser:

The image shows two parts of a web application. The top part is a 'Customer Feedback Form' with a title, a 'Rate our Service:' label, five empty star icons, a 'Your Comments:' label, a text input field, and a blue 'Submit Feedback' button. The bottom part is an 'All Feedback' list showing three entries. Each entry displays a star rating (3/5, 5/5, and 4/5 respectively), the comment text, and a red 'Delete' button.

| Customer Feedback Form | |
|----------------------------------|----------------------|
| Rate our Service: | ☆☆☆☆☆ |
| Your Comments: | <input type="text"/> |
| <button>Submit Feedback</button> | |

| All Feedback | |
|--|-------------------------|
| ☆☆☆ (3/5) very good place, i loved it ❤️ | <button>Delete</button> |
| ☆☆☆☆☆ (5/5) food over here is just WOW | <button>Delete</button> |
| ☆☆☆☆ (4/5) Would be happy to come again and again | <button>Delete</button> |

Conclusion

This project demonstrates how to create an **interactive feedback system** using HTML, CSS, Bootstrap, and jQuery. It shows the use of **form handling, event-driven programming, DOM manipulation, and dynamic updates** in a webpage. This type of project can be extended to real-world applications such as **customer service portals, online shopping sites, and review systems**.