PROJECT - SYSTEM TASK

1. General Description

This application allows users to add, edit, mark as completed, or remove tasks. The tasks are stored in LocalStorage, so they are preserved even after the browser is closed. The interface will be simple and responsive, with additional options for filtering and sorting tasks.

---

2. Main Features

- **Add Tasks:** Users can add new tasks with a title and description.
- **Edit Tasks:** Tasks can be edited to change the title and description.
- **Mark as Completed:** Users can mark a task as completed. The task's status will be updated in LocalStorage and visually reflected in the interface.
- **Remove Tasks:** Users can remove any task.
- **LocalStorage Persistence:** Tasks will be saved in the browser's LocalStorage, and the list will be reloaded when the page is reopened.
- **Filters:** Users can filter tasks to display only completed, pending, or all tasks.
- **Sorting:** Tasks can be sorted by creation date or completion date.

---

3. Technical Specifications

3.1. HTML

The basic structure of the application will include:

- **Task Addition Form:**
  - Field for the task title.
  - Field for the task description.
  - Button to add a task.
- **Task List:** Area where tasks will be displayed, with options to edit, remove, and mark them as completed.
- **Filter and Sorting Buttons:** Filters to show all tasks, completed tasks, or pending tasks, along with sorting options.

3.2. CSS

The interface should be styled to be responsive and intuitive. **Flexbox** or **Grid** should be used for layout.

- **Task Styles:**
    - Visual distinction between completed and pending tasks (e.g., a different line style or color).
    - Visual feedback when hovering over the edit, remove, and complete buttons.

3.3. JavaScript

The application will be highly interactive through DOM manipulation and event handling.

3.3.1. Add Task

- Create a task object with title, description, status (completed/not completed), and timestamps (creation and completion).
- Store the task in LocalStorage.
- Display the task in the interface.

3.3.2. Edit Task

- When clicking “edit,” populate the title and description fields with the existing task data.
- Update the task in LocalStorage after editing.

3.3.3. Mark as Completed

- Change the task status (completed or not completed).
- Update the task in LocalStorage.

3.3.4. Remove Task

- Remove the task from LocalStorage.
- Update the interface.

3.3.5. Load Tasks from LocalStorage

- On page load, read tasks from LocalStorage and display them in the interface.

3.3.6. Filters and Sorting

- Apply filters to display all tasks, only completed tasks, or only pending tasks.
- Sort tasks by creation or completion date.

---

4. Extras

4.1. Filters

Implement three buttons to filter the task display:

- **All Tasks.**
- **Completed Tasks Only.**
- **Pending Tasks Only.**

4.2. Sorting

Add a dropdown that allows sorting tasks by:

- **Creation Date.**
- **Completion Date.**

4.3. Responsiveness

Use Media Queries to ensure the layout adapts well to different screen sizes (desktop, tablet, and mobile).
