

HTML CODE:

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Task Manager</title>

  <link rel="stylesheet" href="styles.css">

</head>

<body>

  <div class="container">

    <h1>Task Manager</h1>

    <div class="task-input">

      <input type="text" id="new-task" placeholder="Add a new task...">

      <button id="add-task">Add Task</button>

    </div>

    <ul id="task-list"></ul>

  </div>

  <script src="app.js"></script>

</body>

</html>
```

CSS CODE:

```
/* styles.css */

Body {

  Font-family: Arial, sans-serif;

  Background-color: #f4f4f4;
```

```
    Display: flex;
    Justify-content: center;
    Align-items: center;
    Height: 100vh;
    Margin: 0;
}
```

```
.container {
    Background: white;
    Padding: 20px;
    Border-radius: 5px;
    Box-shadow: 0 0 10px rgba(0,0,0,0.1);
    Width: 100%;
    Max-width: 400px;
}
```

```
H1 {
    Text-align: center;
}
```

```
.task-input {
    Display: flex;
    Justify-content: space-between;
}
```

```
.task-input input {
```

```
Flex: 1;
Padding: 10px;
Border: 1px solid #ccc;
Border-radius: 5px 0 0 5px;
}
```

```
.task-input button {
  Padding: 10px;
  Border: none;
  Background: #28a745;
  Color: white;
  Border-radius: 0 5px 5px 0;
  Cursor: pointer;
}
```

```
.task-input button:hover {
  Background: #218838;
}
```

```
ul {
  List-style: none;
  Padding: 0;
}
```

```
li {
  Background: #fff;
```

```
Margin: 5px 0;
Padding: 10px;
Border: 1px solid #ddd;
Border-radius: 5px;
Display: flex;
Justify-content: space-between;
Align-items: center;
}
```

```
li.completed {
  text-decoration: line-through;
}
```

```
Li button {
  Background: transparent;
  Border: none;
  Cursor: pointer;
}
```

```
Li button.edit {
  Color: #007bff;
}
```

```
Li button.delete {
  Color: #dc3545;
}
```

```
Li button.complete {  
  Color: #28a745;  
}
```

JAVASCRIPT CODE:

```
// app.js
```

```
Document.addEventListener('DOMContentLoaded', () => {  
  Const taskInput = document.getElementById('new-task');  
  Const addTaskBtn = document.getElementById('add-task');  
  Const taskList = document.getElementById('task-list');
```

```
  Let tasks = JSON.parse(localStorage.getItem('tasks')) || [];
```

```
  Const saveTasks = () => {  
    localStorage.setItem('tasks', JSON.stringify(tasks));  
  };
```

```
  Const renderTasks = () => {  
    taskList.innerHTML = '';  
    tasks.forEach((task, index) => {  
      const li = document.createElement('li');  
      li.className = task.completed ? 'completed' : '';  
      li.innerHTML = `  
        <span>${task.text}</span>  
        <div>  
          <button class="complete">${task.completed ? 'Undo' : 'Complete'}</button>
```

```
<button class="edit">Edit</button>

<button class="delete">Delete</button>

</div>

`;
```

```
li.querySelector('.complete').addEventListener('click', () => {

  tasks[index].completed = !tasks[index].completed;

  saveTasks();

  renderTasks();

});
```

```
li.querySelector('.edit').addEventListener('click', () => {

  const newText = prompt('Edit task', task.text);

  if (newText) {

    tasks[index].text = newText;

    saveTasks();

    renderTasks();

  }

});
```

```
li.querySelector('.delete').addEventListener('click', () => {

  tasks.splice(index, 1);

  saveTasks();

  renderTasks();

});
```

```
    taskList.appendChild(li);  
  });  
};
```

```
addTaskBtn.addEventListener('click', () => {  
  const taskText = taskInput.value.trim();  
  if (taskText) {  
    tasks.push({ text: taskText, completed: false });  
    taskInput.value = '';  
    saveTasks();  
    renderTasks();  
  }  
});
```

```
renderTasks();  
});
```