



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## Report File FULL STACK

**Student Name:** Sukhjinder Singh

**UID:** 23BAI70078

**Branch:** BE-AIT-CSE

**Section/Group:** 23AIT-KRG-G2

**Semester:** 5th

**Subject Code:** 23CSP-339

**Subject Name:** Full Stack

---

### Aim

To build an admin dashboard with dynamic theming using CSS Grid and JavaScript.

### Objectives

1. Create grid-based layout
2. Implement dark/light mode toggle
3. Design responsive sidebar
4. Add data visualization placeholders
5. Persist theme preference using localStorage

### Hardware/Software Requirements

#### Category Requirements

**Hardware** i5 CPU, 8GB RAM

**Software** Chrome DevTools, JSON Server (mock API)

### About the Experiment

#### Advanced Concepts:

- CSS Grid template areas
- prefers-color-scheme media query
- JavaScript classList manipulation
- localStorage API

### Code Implementation

#### HTML:

```
<div class="dashboard" data-theme="light">  
  <aside class="sidebar">Sidebar Menu</aside>
```

```
<main class="content">
  <div class="card">Dashboard Content</div>
  <button id="theme-toggle">Toggle Theme</button>
</main>
</div>
```

## CSS:

```
:root {
  --bg-color: #ffffff;
  --text-color: #000000;
}
```

```
[data-theme="dark"] {
  --bg-color: #1a1a1a;
  --text-color: #f0f0f0;
}
```

```
.dashboard {
  display: grid;
  grid-template-columns: 250px 1fr;
  grid-template-areas: "sidebar content";
  background-color: var(--bg-color);
  color: var(--text-color);
  min-height: 100vh;
}
```

```
.sidebar {
  grid-area: sidebar;
  padding: 1rem;
  background-color: #ccc;
}
```

```
.content {
  grid-area: content;
  padding: 1rem;
}
```

```
@media (max-width: 768px) {
  .dashboard {
    grid-template-columns: 1fr;
    grid-template-areas:
      "sidebar"
      "content";
  }
}
```

```
}  
}
```

### JavaScript:

```
const toggle = document.getElementById('theme-toggle');  
const dashboard = document.querySelector('.dashboard');
```

```
toggle.addEventListener('click', () => {  
  const newTheme = dashboard.dataset.theme === 'light' ? 'dark' : 'light';  
  dashboard.dataset.theme = newTheme;  
  localStorage.setItem('theme', newTheme);  
});
```

```
window.addEventListener('DOMContentLoaded', () => {  
  const savedTheme = localStorage.getItem('theme');  
  if (savedTheme) {  
    dashboard.dataset.theme = savedTheme;  
  }  
});
```

### Output Screenshots

1. Desktop view with sidebar
2. Mobile responsive layout
3. Dark mode activation
4. Theme persistence on refresh



Sidebar Menu

Dashboard Content

Toggle Theme