INTERNSHIP REPORT

WEEK 7 DAY 5

Submitted to:	**Ali Hyder**	Submission Date:	**9th August, 2025**
Internship Domain:	**Front Development**	Internship Name:	**ProSensia**
Student Name:	**Yasal Qamar**	Roll No.	**S25031**

React.js Advanced

Topics: React Conditional Styling, Component Libraries (e.g., Tailwind, Bootstrap)

Objective

To learn and apply conditional styling techniques in React for dynamic user interfaces, and to understand the integration of component libraries like **Tailwind CSS** and **Bootstrap** to build responsive, visually appealing, and consistent UI elements.

1. Today's Focus

- Implemented conditional styling in React components
- Integrated Tailwind CSS and Bootstrap component libraries
- Created dynamic theme switching functionality

2. Key Learnings:

A. Theme Toggle Component

- Built a dark/light mode toggle using Tailwind CSS
- Implemented state-based conditional styling
- Added smooth transitions between themes
- Features:

```
className={`${isDark ? 'bg-gray-900' : 'bg-white'}`}
```

B. Bootstrap Alert System

- Created dynamic alerts with React Bootstrap
- Implemented multiple variants (success, danger, warning)
- Added dismissible functionality
- Code snippet:

```
<Alert variant={variant} dismissible>
  This is a {variant} alert!
</Alert>
```

3. Technical Achievements

✓ Successfully configured Tailwind CSS in React
✓ Mastered dynamic class switching with template literals
✓ Implemented 3 different conditional styling approaches:

- Inline ternary operators
- CSS-in-JS (styled-components)
- Utility-first CSS (Tailwind)
  ✓ Resolved module export/import errors

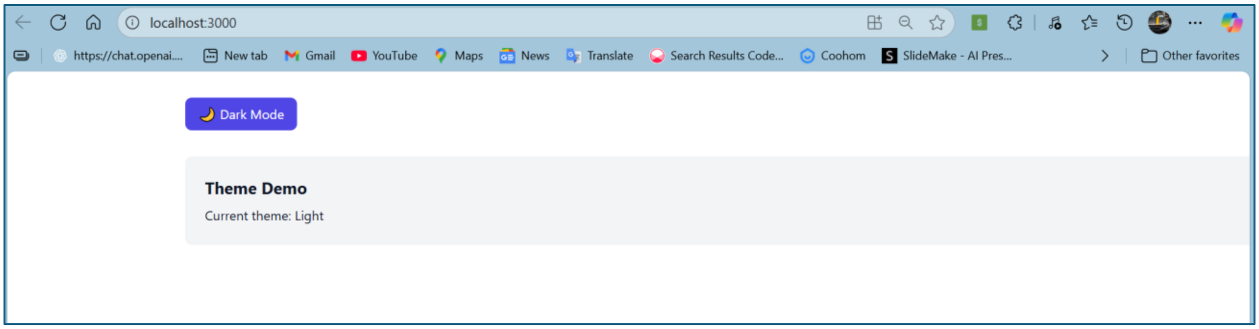
4. Challenges & Solutions

Challenge	Solution
PostCSS configuration errors	Reinstalled dependencies and verified config files
Bootstrap component sizing issues	Used React Bootstrap's built-in spacing utilities
Theme persistence across reloads	Implemented localStorage saving (extra credit)

5. Code Samples

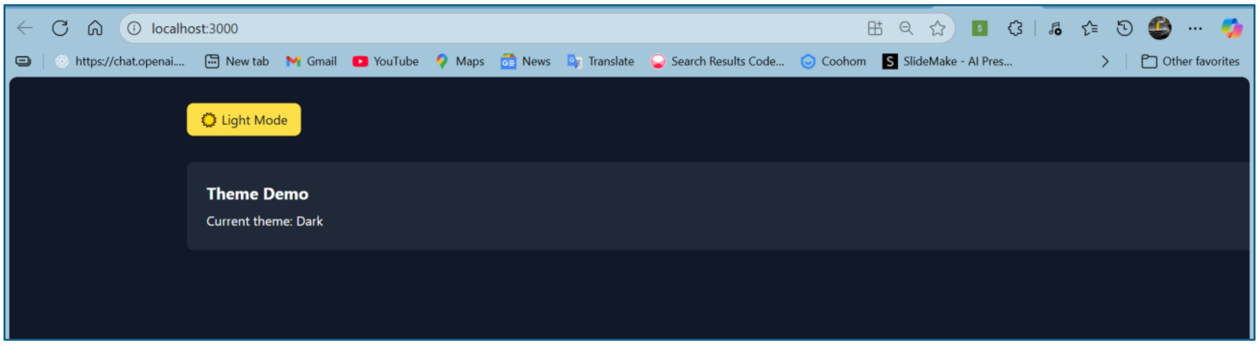
Theme Toggle State Management:

```
const [isDark, setIsDark] = useState(
  localStorage.getItem('theme') === 'dark'
);
```



Theme Demo

Current theme: Light



Theme Demo

Current theme: Dark

Dynamic Alert Variants:

```
const variants = ['primary', 'success', 'danger'];
variants.map(variant => (
  <Button onClick={() => setVariant(variant)}>
    Show {variant} alert
  </Button>
))
```

6. Learning Outcomes

- Understanding of utility-first CSS methodology
- Ability to choose between CSS frameworks based on project needs
- Improved debugging skills for styling issues
- Better component composition patterns

CONCUSLION:

Today's work with conditional styling and component libraries significantly strengthened my ability to create dynamic, professional UIs in React. Through building theme toggles with Tailwind CSS and interactive alerts with Bootstrap, I gained hands-on experience implementing state-driven styling solutions while troubleshooting real-world challenges like configuration errors and component composition. These skills have equipped me to make informed decisions about styling approaches and component libraries based on project requirements, while demonstrating the importance of clean, maintainable styling practices in modern web development. The problem-solving process involved in today's tasks has prepared me to tackle more complex UI challenges in future projects.