



COLLEGE CODE : 9628

**COLLEGE NAME: UNIVERSITY COLLEGE OF ENGINEERING
NAGERCOIL.**

DEPARTMENT :COMPUTER SCIENCE AND ENGINEERING

STUDENT-ID : 3791FE0EE3ABCB99BA4438D4CDA7A1E8

REGISTER NO: 962823104078

DATE : 08-09-2025

COMPLETED THE PROJECT NAMED AS :

-THEME CUSTOMISER IN WORDPRESS

SUBMITTED BY ,

NAME : MOHAMMED RASIDH M

MOBIL NO : 6381482861

PROBLEM DEFINITION AND DESIGN THINKING

TITLE : Theme customizer in WordPress

Problem Statement:

The WordPress Theme Customizer enables users to preview and modify website appearance, but it often provides limited options and inconsistent live previews across themes. Non-technical users struggle with advanced customization, while developers face challenges in extending its functionality. This leads to difficulties in creating fully personalized, responsive, and visually appealing websites. Hence, there is a need for an improved Theme Customizer that offers a user-friendly interface, accurate real-time previews, and extended customization features to meet the needs of both users and developers.

Target Audience :

- **Non-technical WordPress users** (bloggers, small business owners, and content creators) who want to easily customize their websites without coding.
- **Web developers and designers** who require flexible tools and APIs to extend theme customization options.
- **Organizations and businesses** looking to build professional, responsive websites with minimal effort.
- **Educational learners and students** exploring WordPress customization as part of web development project.

Objectives:

- To design a user-friendly Theme Customizer that allows non-technical users to modify website appearance without coding.
- To provide accurate real-time previews of customization changes for better decision-making.
- To extend customization options such as typography, layouts, and responsive design controls.
- To create developer-friendly APIs for adding and managing custom controls in themes.
- To enhance the overall user experience by ensuring consistency across different WordPress themes.

Design Thinking Approach:

- **Empathize** – Understand the needs of non-technical users who want simple customization and developers who require extensibility. Gather feedback from bloggers, small businesses, and web designers about challenges with the current Theme Customizer.
- **Define** – Clearly define the problem: users struggle with limited customization options, inconsistent previews, and complex settings, making it hard to build personalized and responsive websites.
- **Ideate** – Brainstorm solutions such as adding advanced but user-friendly customization controls (colors, typography, layouts), ensuring accurate live previews, and developing APIs for developers to extend features.
- **Prototype** – Create a functional mock-up of the improved Theme Customizer with enhanced UI/UX, real-time preview capabilities, and extended customization options.
- **Test** – Collect feedback from end-users and developers by testing the prototype on various themes, refine features based on usability issues, and improve until it meets both technical and non-technical user needs.

Test:

- **Usability Testing** – Ensure that non-technical users can easily navigate the Theme Customizer and make changes without confusion.
- **Functionality Testing** – Verify that all customization controls (colors, fonts, layouts, widgets) work correctly and changes are applied as expected.
- **Live Preview Accuracy** – Check that real-time previews accurately reflect changes across different devices (desktop, tablet, mobile).
- **Compatibility Testing** – Ensure the customizer works smoothly across various WordPress themes and plugins.
- **Performance Testing** – Evaluate the responsiveness of the customizer interface when making multiple changes or using advanced controls.
- **Developer API Testing** – Validate that developers can add custom controls or extend functionality without errors.

- **Accessibility Testing** – Confirm that controls are accessible to all users, including keyboard navigation, screen readers, and tooltips

Testing Goals:

- Ensure a seamless, user-friendly customization experience for non-technical users.
- Verify reliability and accuracy of real-time preview features.
- Confirm extended customization options and developer APIs function as intended.
- Identify and fix usability or compatibility issues before deployment.