**Project Proposal: Latter-Day Saints Chapel Locator**

**Overview:**

Chapel locator is a JavaScript application designed to help users explore information about various church building. The application interacts with the DOM of an HTML document to display chapel details dynamically. Users can filter chapel based on location.

**Objectives:**

Create an interactive JavaScript application hosted and rendered by GitHub Pages.

Utilize DOM manipulation to dynamically update the UI based on user interactions.

Implement basic CSS styles to enhance the visual presentation of the application.

Employ conditional branching to handle different user actions and scenarios.

Utilize array methods for processing temple data.

Use template literals extensively for string interpolation and building UI components.

Fetch external data using the Fetch API, consuming JSON data from a third-party API or a custom data source.

Features:

**Chapel Locator:** Display a list of church building fetched from an external JSON data source. Each chapel will include its name, image, and location.

**Filtering:** Allow users to filter church building based on location, and other criteria using a dropdown menu.

**Dynamic UI Update:** Dynamically update the chapel list based on the selected filter option without requiring a page refresh.

**Responsive Design:** Ensure the application is responsive and accessible across different devices and screen sizes.

**Technology:**

JavaScript: For application logic, DOM manipulation, and data fetching.

HTML: For structuring the application's user interface.

CSS: For styling and enhancing the visual presentation.

Fetch API: To consume external JSON data from a third-party API or a custom data source.

GitHub Pages: For hosting and rendering the application.

**The Process:**

**Stage 1:** Project Planning and Setup

Define project requirements, objectives, and features.

Set up GitHub repository and project structure.

Research and select external data sources/APIs.

**Stage 2:** Application Development

Implement DOM manipulation for displaying chapel data.

Integrate Fetch API to retrieve chapel information from the chosen data source.

Implement filtering functionality based on user input.

Develop responsive UI design and basic CSS styling.

**Stage 3:** Testing and Debugging

Conduct comprehensive testing to identify and fix bugs and issues.

Ensure cross-browser compatibility and responsiveness across devices.

Test data fetching and filtering mechanisms to ensure data accuracy.

**Stage 4:** Documentation and Deployment

Write detailed documentation covering project overview, features, and usage instructions.

Deploy the application on GitHub Pages and ensure seamless rendering.

Finalize project presentation and prepare for project submission.

**Conclusion:**

Latter-Day Saints Chapel Locator aims to provide users with an intuitive and interactive platform to explore information about chapel worldwide. Through dynamic data fetching, filtering capabilities, and responsive design, the application offers a seamless and engaging user experience. By leveraging modern web technologies and best practices, Chapel Locator seeks to deliver a valuable resource for individuals interested in learning about chapels across different regions.