**Project Proposal: Meditation Timer App**

**1. Tech Stack:**

* **Choice:** Python with Flask
* **Reasoning:** Flask provides a simple yet powerful framework for building the back-end of the application. Python’s readability and ease of use will speed up the development process, especially for a straightforward app like a meditation timer. The front-end can be built with basic HTML, CSS, and JavaScript to keep the user interface clean and intuitive.

**2. Focus:**

* **Full-stack Application:** The project will have an evenly focused full-stack approach, with attention to both the front-end user experience and back-end functionality.

**3. Type of Application:**

* **Website:** The meditation timer will be a responsive web application, accessible on both desktop and mobile devices.

**4. Project Goal:**

* **Objective:** The goal of this project is to create a simple, user-friendly timer that assists users in their meditation practice. The app will allow users to set a timer for their sessions, choose from various background sounds, and track their meditation progress over time.

**5. Target Users:**

* **Demographics:** The app is designed for individuals who practice meditation, mindfulness, or yoga. The target users are likely adults aged 18-50, with an interest in mental wellness and stress management.

**6. Data Usage:**

* **Data Requirements:** The app will store user preferences, such as the duration of meditation sessions, chosen background sounds, and session history. This data will be stored securely in a database, likely using SQLite for its simplicity. No sensitive information will be collected, as the app’s functionality focuses on timing and user experience rather than personal data.

**7. Database Schema:**

* **Schema Overview:** The database will include tables for storing user profiles, session history, and sound options. Key relationships will be between the user and their session history to track progress over time.

**8. API Considerations:**

* **Potential Issues:** Since the app does not require external API integration, issues will likely be minimal. However, ensuring the accurate handling of time and sound playback across different devices may require careful testing.

**9. Security Concerns:**

* **Sensitive Information:** While the app will not handle sensitive data, securing user session data and preferences is important. Flask’s built-in security features and HTTPS will be utilized to protect this information.

**10. Functionality:**

* **Features:**
  + Timer setup: Users can select the duration of their meditation session.
  + Background sounds: Users can choose from a set of calming sounds to play during the session.
  + Progress tracking: The app will log the duration and frequency of sessions, providing users with a simple dashboard to view their meditation history.
  + Responsive design: The app will be optimized for both desktop and mobile use.
* **Stretch Goals:**
  + Custom sound uploads: Allow users to upload their own background sounds.
  + Guided meditation: Include a feature for guided meditation sessions with audio instructions.
  + Social sharing: Allow users to share their meditation milestones on social media.

**11. User Flow:**

* **Flow Description:**
  1. **Home Page:** Users can quickly set a timer and start a session.
  2. **Settings Page:** Users customize their session duration, select background sounds, and view their progress.
  3. **Session Page:** A simple interface displays the countdown timer and plays the selected sound.
  4. **Completion:** After the session ends, users are presented with a summary of their session, which is logged in their history.