

Backend Development (Node.js)

Duration: 6-8 weeks

The backend development will focus on building a robust and scalable API to support the functionalities of the apnaPandit application. The tasks include:

1. User and Pandit Registration and Authentication

- **User Registration:** Implement user registration with fields for basic details such as name, email, password, and mobile number.
- **Pandit Registration:** Include additional fields for Pandits like qualifications, experience, and area of expertise.
- **Mobile Number Verification:** Integrate SMS OTP verification to validate user and Pandit phone numbers during registration. (If Require)

2. Profile Management

- **User Profile Management:** Allow users to update their profile details, view booking history, and manage their preferences.
- **Pandit Profile Management:** Enable Pandits to update their profiles, upload photos and videos, and manage their availability status.

3. Search and Filtering Functionality for Pandits

- **Basic Search:** Allow users to search for Pandits based on name, qualifications, or experience.
- **Location-Based Search:** Implement a feature to find Pandits within a specified radius from the user's current location, using map and list views.
- **Advanced Filtering:** Provide filters for various criteria like availability, rating or specific services offered.

4. Booking System

- **Booking Management:** Develop a system for users to book Pandits for pujas, including selecting available time slots and specifying details.
- **Scheduling:** Implement scheduling features for both immediate and future bookings.
- **Confirmations and Cancellations:** Manage booking confirmations, cancellations, and notifications for both users and Pandits.
- **Puja Samagari Management:** Allow users to customize and rent items required for pujas, with details on availability based on location and stock.

5. Reviews and Ratings System

- **Post-Puja Reviews:** Enable users to provide feedback and rate Pandits after puja completion.
- **Review Moderation:** Develop tools for moderating reviews and handling any reported issues.

6. Notification System

- **Booking Notifications:** Notify users and Pandits about new bookings, cancellations, and reminders.
- **Live Puja Notifications:** Alert users when a Pandit starts a live puja session.

7. Payment Gateway Integration

- **Subscription Payments for Pandits:** Implement a payment system for Pandit subscriptions, including various plans and payment methods.
- **Secure Transactions:** Ensure secure handling of payment data and compliance with relevant standards.

8. Admin Dashboard and Functionalities

- **User and Pandit Management:** Provide tools for the admin to view, edit, and manage users and Pandits, including approvals and account status changes.
- **Booking and Payment Monitoring:** Track bookings, payments, and overall system metrics.
- **Content Management:** Allow the admin to manage puja categories, Samagari lists, advertisements, and other platform content.

Frontend Development (React Native)

Duration: 8-10 weeks

The frontend development will focus on creating an intuitive and responsive user interface for the apnaPandit mobile application. The tasks include:

1. UI/UX Design Implementation

- **User-Centric Design:** Implement designs that are user-friendly, visually appealing, and consistent across all devices.
- **Responsive Layouts:** Ensure the application is fully responsive and works seamlessly on various screen sizes.

2. User Registration and Profile Management Interfaces

- **Registration and Login:** Create user interfaces for registration and login, including mobile number verification.
- **Profile Management:** Allow users to view and edit their profiles, view booking history, and manage preferences.

3. Pandit Profile Viewing and Interaction

- **Profile Display:** Design detailed Pandit profiles showcasing qualifications, experience, photos, videos, and user reviews.
- **Follow and Interaction:** Enable users to follow Pandits, receive updates, and interact through live puja sessions.

4. Search and Filtering Interfaces

- **Search Bar and Filters:** Implement a search bar with filters for criteria like location, ratings, and services offered.
- **Map and List Views:** Provide options to view Pandits in map or list formats for easier browsing.

5. Booking Flow and Confirmation Screens

- **Booking Interface:** Develop an intuitive booking process, including selecting time slots and customizing puja details.
- **Confirmation and Tracking:** Display booking confirmations, tracking, and notifications for users and Pandits.

6. Review and Rating Submission

- **Post-Puja Feedback:** Create interfaces for users to submit reviews and ratings after puja completion.
- **Review Display:** Show user reviews on Pandit profiles with options for sorting and filtering.

7. Samagari Customization and Rental Process

- **Samagari Selection:** Design a system for selecting and customizing Samagari items for pujas.
- **Rental Management:** Allow users to view rental availability, prices, and confirm rentals.

8. Live Puja Streaming and Notification System

- **Live Streaming Interface:** Enable live streaming of pujas with options for users to interact and engage.
- **Notification Management:** Implement notifications for live sessions, bookings, and other updates.

9. Admin Interfaces (if applicable)

- **Admin Dashboard:** Create a user-friendly admin dashboard for managing users, Pandits, bookings, and content.
- **Analytics and Reporting:** Provide tools for viewing analytics, reports, and system metrics.