**PanditSeva**

**Final Tech Stack for Pandit/Priest Booking System**

**1. Frontend (Web)**

* **Framework**: React.js (latest stable version)
* **UI Library**: Material UI + TailwindCSS (for flexibility + speed)
* **State Management**: Redux Toolkit
* **Routing**: React Router v6
* **Calendar**: FullCalendar.js (customized to show service + priest name in slots)
* **Notifications**: Toastify for in-app alerts

**2. Mobile App**

* **Framework**: React Native (Expo for fast builds)
* **UI Kit**: Native Base / React Native Paper
* **Navigation**: React Navigation
* **Push Notifications**: Firebase Cloud Messaging (FCM)
* **Calendar View**: react-native-calendars

**3. Backend**

* **Framework**: ASP.NET Core 8 Web API
* **Language**: C#
* **API Design**: REST (JSON)
* **Authentication**: ASP.NET Identity + JWT
* **Payment Gateway Integration**:
  + Stripe for international
  + Razor pay for India

**4. Database**

* **Type**: Microsoft SQL Server
* **ORM**: Entity Framework Core 8
* **Hosting**: Azure SQL Database
* **Key Tables**:
  + Users (roles: User, Priest, Admin)
  + Services
  + Bookings
  + Payments
  + Priest Availability
  + Notifications

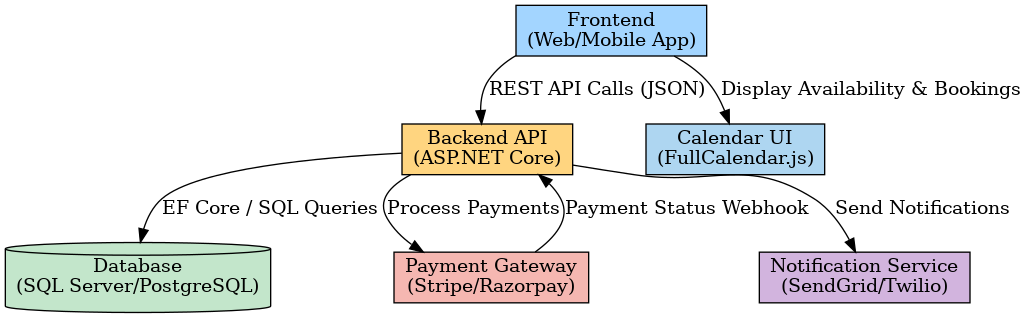
**5. DevOps & Hosting**

* **Backend Hosting**: Azure App Service
* **Frontend Hosting**: Azure Static Web Apps
* **Mobile App Deployment**:
  + Android: Google Play Store
  + iOS: Apple App Store
* **CI/CD**: GitHub Actions for automated deployment
* **Logging/Monitoring**: Serilog + Azure Application Insights

**6. Extras**

* **Email Notifications**: SendGrid (Azure integration)
* **SMS Notifications**: Twilio
* **File Storage** (for receipts, images): Azure Blob Storage
* **Testing**: xUnit for backend, Jest for frontend

**Architecture Diagram**



**1. System Overview**

A web and mobile platform allowing users to:

* Search and book a priest for religious services.
* Select a date and optionally time.
* Choose from a list of services.
* Optionally request a specific priest.
* Pay online to confirm booking.
* Receive notifications and receipts.

**2. Core Features**

**User Side**

1. **Service Selection** – Choose service type (e.g., Puja, Wedding, etc.).
2. **Date & Time Picker** – Integrated calendar view.
3. **Priest Selection** – Choose "Any Available" or a specific priest.
4. **Live Availability Check** – Based on priest calendars.
5. **Booking Confirmation & Payment** – Full or partial payment.
6. **Notifications** – Email/SMS/App push notifications for status updates.
7. **Receipt Generation** – PDF/Email receipt.

**Priest Side**

1. **Profile Management** – Name, bio, services offered, contact info.
2. **Calendar Management** – Mark availability/unavailability.
3. **Booking Requests** – Accept or reject requests.
4. **Payment Tracking** – See confirmed and pending payments.

**3. Booking Flow**

1. **User selects service + date/time**
2. **Request sent to matching priests**
3. **First priest to accept → booking reserved**
4. **User makes payment**
5. **Booking confirmation sent to both**
6. **Event added to both calendars**

**WorkFlow Diagram**

