

+

Backend development requirements

Share

Website Layout-
Sahil.PDF
1,001 lines

PDF

U

bhaiu eska backend k bara m bta ktya kya chahiye or kya kya bnanaa h b

Bilkul bhai, main tumhe iss complete platform ke backend architecture ka breakdown deta hoon. Yeh ek complex multi-service platform hai toh main systematic tareeke se samjhata hoon:

Core Backend Requirements

1. Technology Stack (Recommended)

- Backend Framework: Node.js (Express/NestJS) ya Python (Django/FastAPI)
- Database: PostgreSQL (relational data) + MongoDB (documents/PDFs)
- File Storage: AWS S3 / Cloudflare R2
- Payment Gateway: Razorpay API
- Communication:
 - Email: SendGrid / AWS SES
 - WhatsApp: Twilio / Interakt API
 - SMS: Twilio / MSG91
- Calendar: Google Calendar API
- Video Calls: Zoom API / Google Meet API
- Authentication: JWT + OAuth

2. Database Schema (Main Tables)

Users Table

- user_id (PK)
- name, email, phone
- password_hash
- dob, tob, pob
- profile_photo_url
- lagna, moon_sign, nakshatra
- current_mahadasha, antardasha
- wallet_balance, coins
- referral_code
- created_at, updated_at

Orders Table

- order_id (PK)
- user_id (FK)
- service_type (energy/astrology/vastu/manifestation/material)
- service_name
- status (new/processing/delivered/completed)
- payment_id
- amount
- form_data (JSON)
- expected_delivery_date
- created_at, delivered_at

Reports/PDFs Table

- report_id (PK)
- order_id (FK)
- user_id (FK)
- report_type
- file_url (S3 link)
- uploaded_at
- viewed_at

Remedies Table

- remedy_id (PK)
- user_id (FK)
- remedy_type (donation/mantra/gemstone/vastu/affirmation)
- instructions (TEXT)
- duration_days
- start_date
- completion_status
- feedback
- reminder_enabled

Calls/Consultations Table

- call_id (PK)
- user_id (FK)
- order_id (FK)

- scheduled_datetime
- duration_minutes
- meeting_link
- status (scheduled/completed/missed/rescheduled)
- call_summary (TEXT)
- recording_url

Transactions Table

- transaction_id (PK)
- user_id (FK)
- order_id (FK)
- payment_gateway_id
- amount
- status (success/failed/pending)
- payment_method
- created_at

Material Program Clients Table

- client_id (PK)
- user_id (FK) / company_id (FK)
- client_type (individual/business)
- company_name (if business)
- industry
- team_members (JSON - for business)
- goals (TEXT)
- progress_level
- quarterly_strategy_due_date

Forms/Submissions Table

- submission_id (PK)
- user_id (FK)
- order_id (FK)
- form_type
- form_data (JSON)
- uploaded_files (array of URLs)
- submitted_at

Notifications Table

- notification_id (PK)
- user_id (FK)
- type (email/whatsapp/sms/push)
- message
- sent_at
- read_at

3. Key API Endpoints Required

User APIs

POST /api/auth/register
POST /api/auth/login
GET /api/user/profile
PUT /api/user/profile
GET /api/user/dashboard
GET /api/user/birth-chart

Services/Orders APIs

GET /api/services (all services list)
GET /api/services/:category
POST /api/orders/create
GET /api/orders/:orderId
GET /api/orders/user/:userId (all user orders)
PUT /api/orders/:orderId/status

Payment APIs

POST /api/payments/create-order (Razorpay)
POST /api/payments/verify
POST /api/payments/refund
GET /api/payments/history

Reports/PDFs APIs

POST /api/reports/upload (admin)
GET /api/reports/:reportId
GET /api/reports/user/:userId
DELETE /api/reports/:reportId

Remedies APIs

POST /api/remedies/create (admin assigns to user)
GET /api/remedies/user/:userId
PUT /api/remedies/:remedyId/status
POST /api/remedies/:remedyId/feedback

Calls/Consultations APIs

POST /api/calls/schedule
GET /api/calls/user/:userId
PUT /api/calls/:callId/reschedule
POST /api/calls/:callId/summary (admin)

Communication APIs

POST /api/notifications/send-email
POST /api/notifications/send-whatsapp
POST /api/notifications/send-sms
GET /api/notifications/user/:userId

Admin Panel APIs

GET /api/admin/dashboard-stats
GET /api/admin/orders?status=pending
GET /api/admin/clients
GET /api/admin/client/:userId/profile
POST /api/admin/broadcast
PUT /api/admin/orders/:orderId/upload-report
POST /api/admin/remedies/assign
GET /api/admin/analytics

Material Program APIs

GET /api/material/clients
GET /api/material/client/:clientId
POST /api/material/strategy-upload
GET /api/material/tasks/:clientId
POST /api/material/tasks/:taskId/comment

Forms Builder APIs

POST /api/forms/create (dynamic form creation)
GET /api/forms/:formId
POST /api/forms/:formId/submit

4. Backend Features to Build

A. File Upload System

- Multi-file upload support
- PDF generation from data
- Automatic email delivery on upload
- S3/Cloud storage integration

B. Payment Integration

- Razorpay payment gateway
- Webhook handlers for payment status
- Refund management
- Coupon/discount code system
- Wallet/credits system

C. Communication System

- **Email Templates:** Order confirmation, PDF delivery, remedy reminders, feedback requests
- **WhatsApp API:** Order updates, remedy alerts, follow-ups
- **SMS:** Critical alerts
- **Push Notifications:** In-app alerts

D. Scheduling System

- Google Calendar integration
- Call booking with timezone handling
- Auto-send meeting links
- Reminder emails/WhatsApp 24 hours before

E. Reminder/Automation System

- Cron jobs for:
 - Remedy reminders
 - Delivery deadline alerts
 - Follow-up nudges (every 3 months)
 - Missed call alerts
 - Feedback requests

F. Admin Upload Portal

- Drag-and-drop PDF upload
- Auto-tag to user/order

- Bulk upload capability
- Delivery mode selection

G. Analytics Engine

- Revenue tracking
- Service-wise performance
- User engagement metrics
- Conversion funnel analysis
- Remedy completion rates

H. Suggestion/Recommendation Engine

- Based on user's chart + past orders
- Dasha-based suggestions
- Auto-suggest next service
- Flag upcoming transits

I. Access Control System

- Role-based access (User/Admin/Material Client)
- Program-specific content visibility
- Family account linking

J. Backup System

- Daily automated backups
- Change history logs
- Admin activity logs

5. Third-Party Integrations Needed

1. **Razorpay** - Payments
2. **Twilio/Interakt** - WhatsApp API
3. **SendGrid/AWS SES** - Email
4. **MSG91** - SMS
5. **AWS S3/Cloudflare R2** - File storage
6. **Google Calendar API** - Scheduling
7. **Zoom/Google Meet API** - Video calls
8. **Firebase** - Push notifications (optional)

6. Security Requirements

- JWT authentication
- Password hashing (bcrypt)
- HTTPS only
- Input validation & sanitization
- Rate limiting on APIs
- CORS configuration
- Secure file upload validation
- Payment webhook verification
- Admin panel 2FA (recommended)

7. Development Priority

Phase 1:

- User authentication
- Service catalog
- Order creation & payment
- Basic admin panel (order management)

Phase 2:

- PDF upload & delivery
- Email/WhatsApp notifications
- Remedies system
- Call scheduling

Phase 3:

- Material program features
- Analytics dashboard
- Suggestion engine
- Forms builder

Phase 4:

- Advanced features (gamification, referrals, etc.)

Bhai yeh complete backend architecture hai. Agar kisi specific part mein deep dive chahiye

toh batao, main detailed breakdown de dunga!

ⓘ Claude does not have the ability to run the code it generates yet.



📄 🗑️ 💬 Retry ▾

Claude can make mistakes. Please double-check responses.

Reply to Claude...



Sonnet 4.5 ▾

