

Stage 3: Technical Documentation – Imagine Project

1. User Stories and Mockups

Must Have:

- As a tourist, I want to register/login so that I can book guides.
- As a guide, I want to register so that I can offer my services.
- As an admin, I want to approve/reject guides so that only verified guides appear.
- As a tourist, I want to browse places so that I can explore options before booking.

Should Have:

- As a tourist, I want to leave reviews so that others benefit from my feedback.

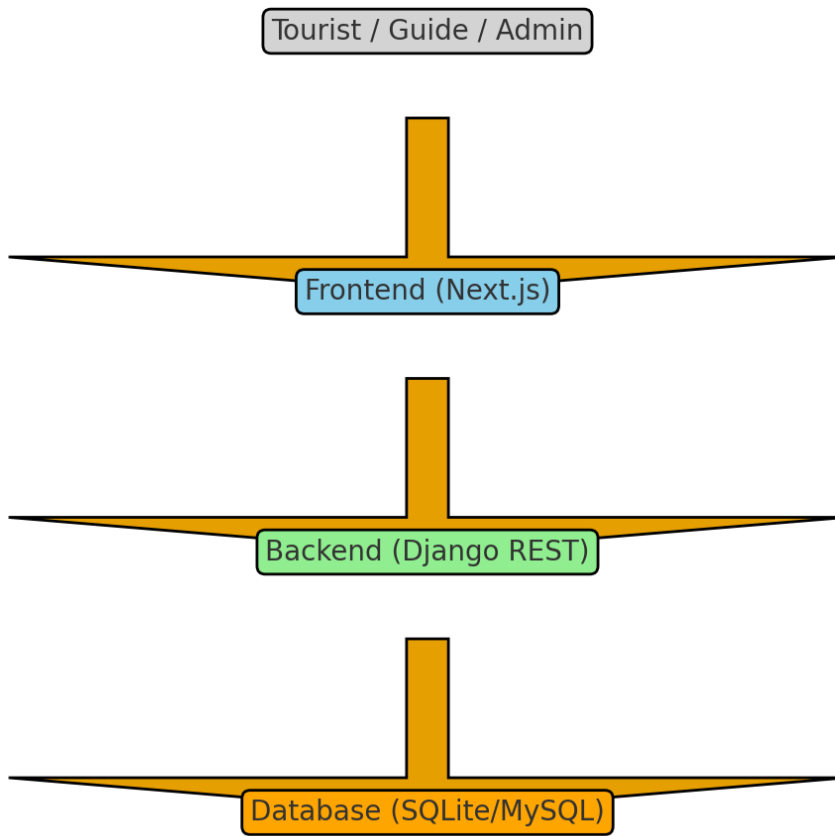
Could Have:

- As a guide, I want to upload multiple images/videos.

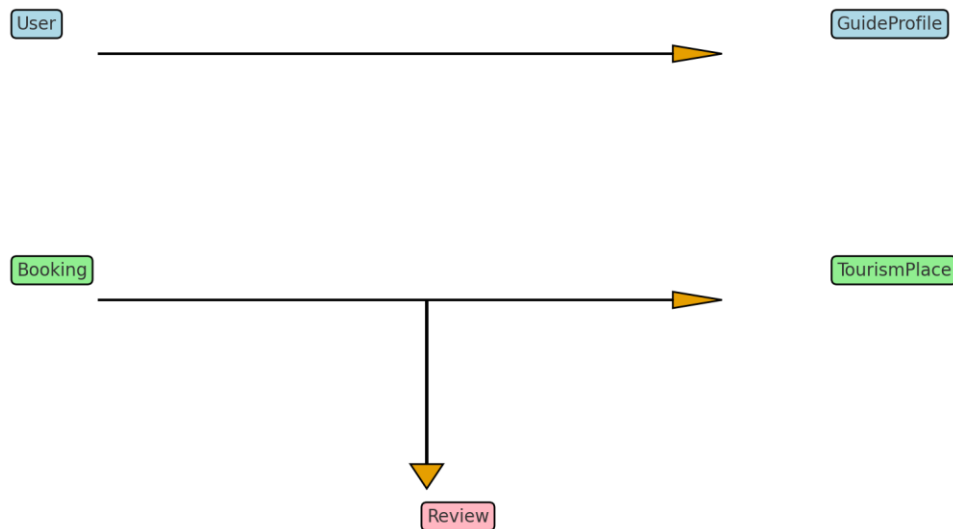
Won't Have:

- Real-time chat (not in MVP).

2. System Architecture

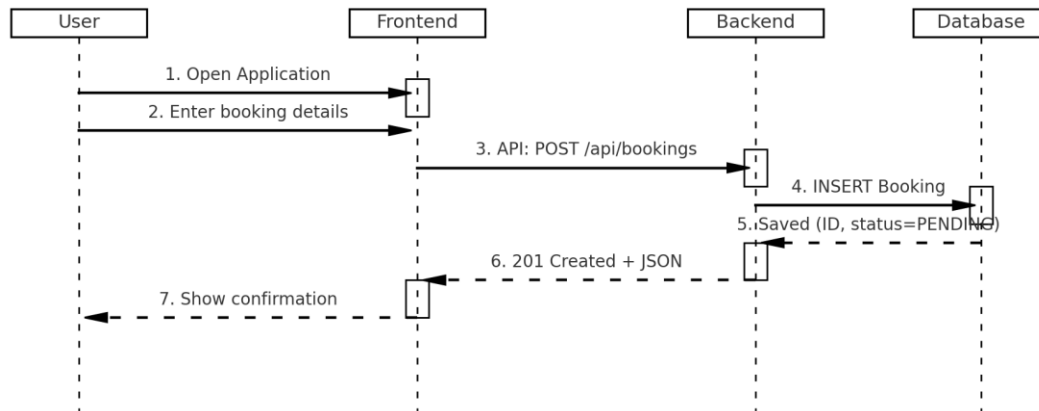


3. Database ER Diagram



4. Sequence Diagram (Booking Flow) – UML Styled

Formatted to match the provided example: lifelines, activation bars, and numbered messages. The image is sized to avoid splitting.



5. API Specifications

- POST /api/auth/register {name,email,password,role}
- POST /api/auth/token {email,password}
- GET /api/places
- GET /api/guides
- POST /api/bookings {guide_id,place_id,start_datetime}
- POST /api/reviews {booking_id,rating,comment}

6. SCM and QA Strategies

SCM: GitHub repo with branches main/dev/feature. Pull requests required.

QA: Unit tests (Django), frontend Jest tests, Postman for integration.

7. Technical Justifications

Next.js for modern UI, Django REST for robust backend, SQLite for dev, MySQL for production. JWT for stateless auth.