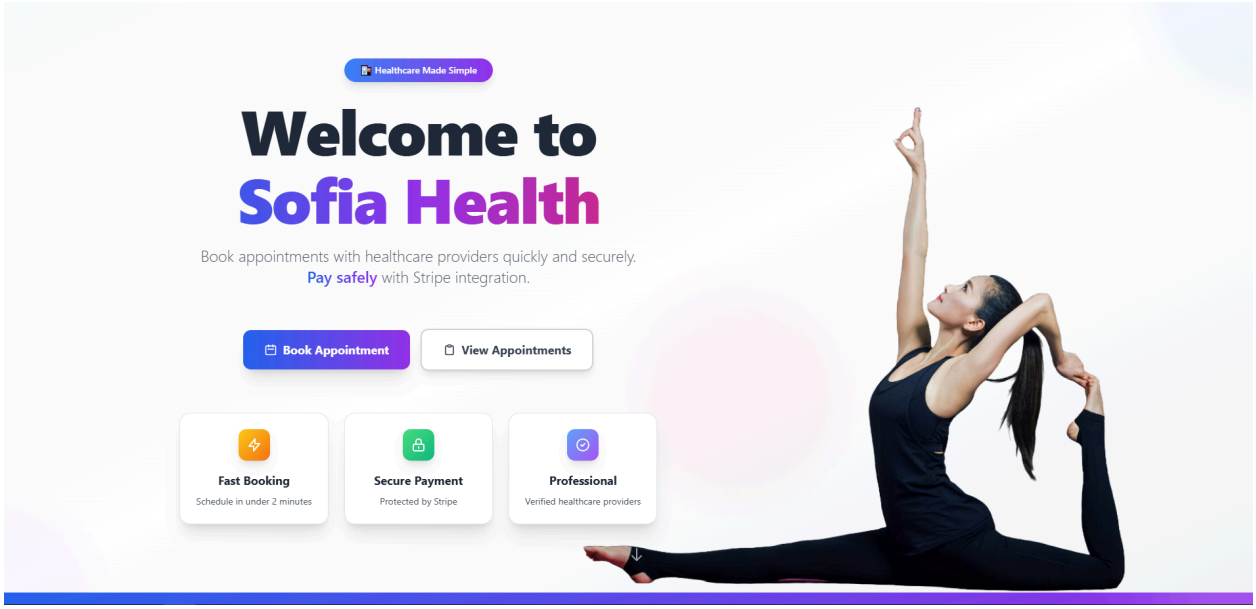
Django Demo Project for

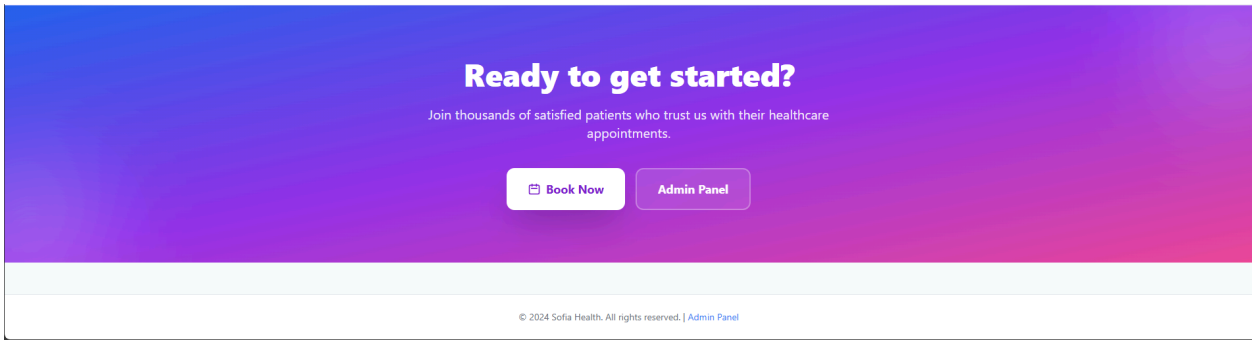
Sofia Health

Developed By Auria Ahmad

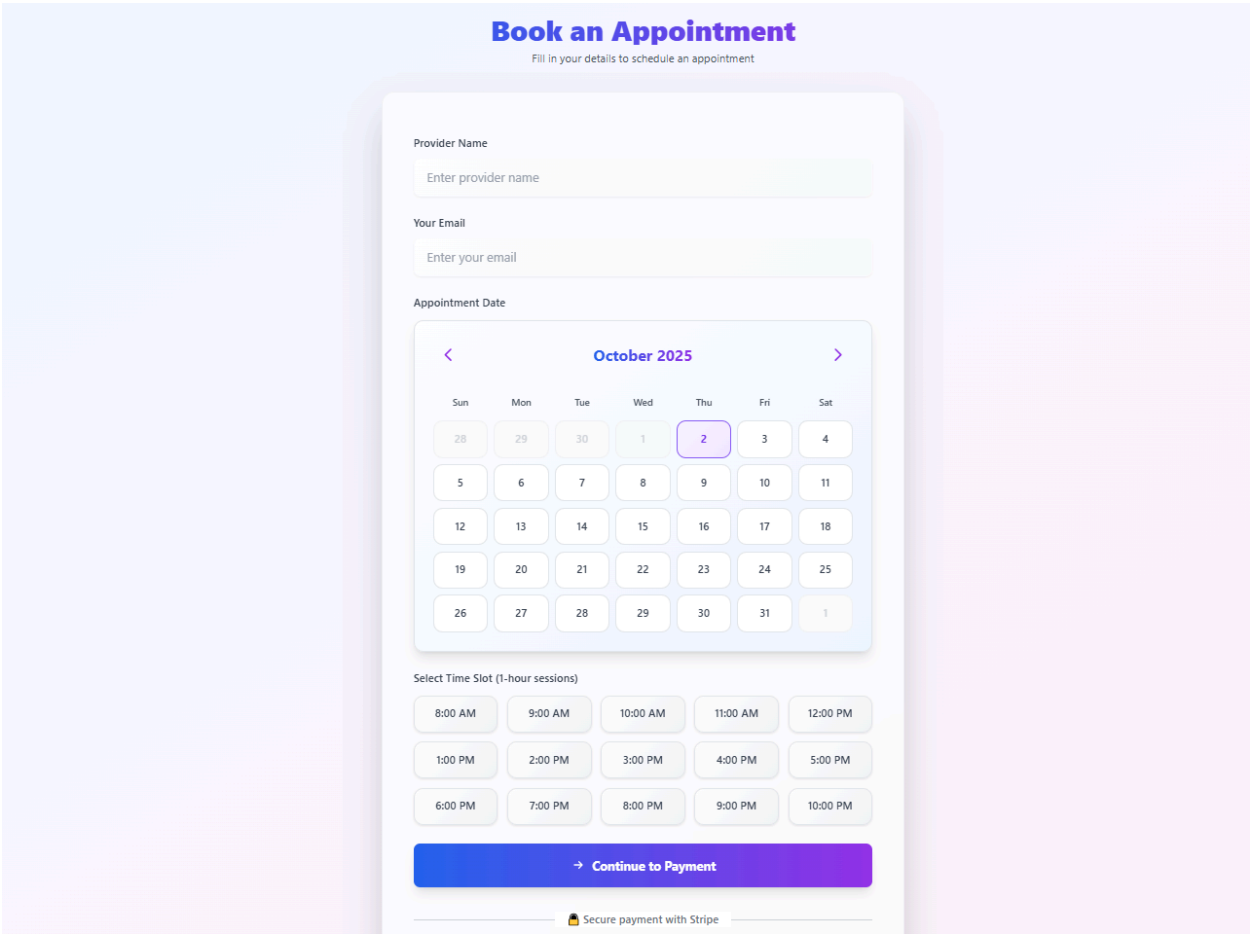
Part 1: Core Skills (Hands-On) Visualization

Home Page

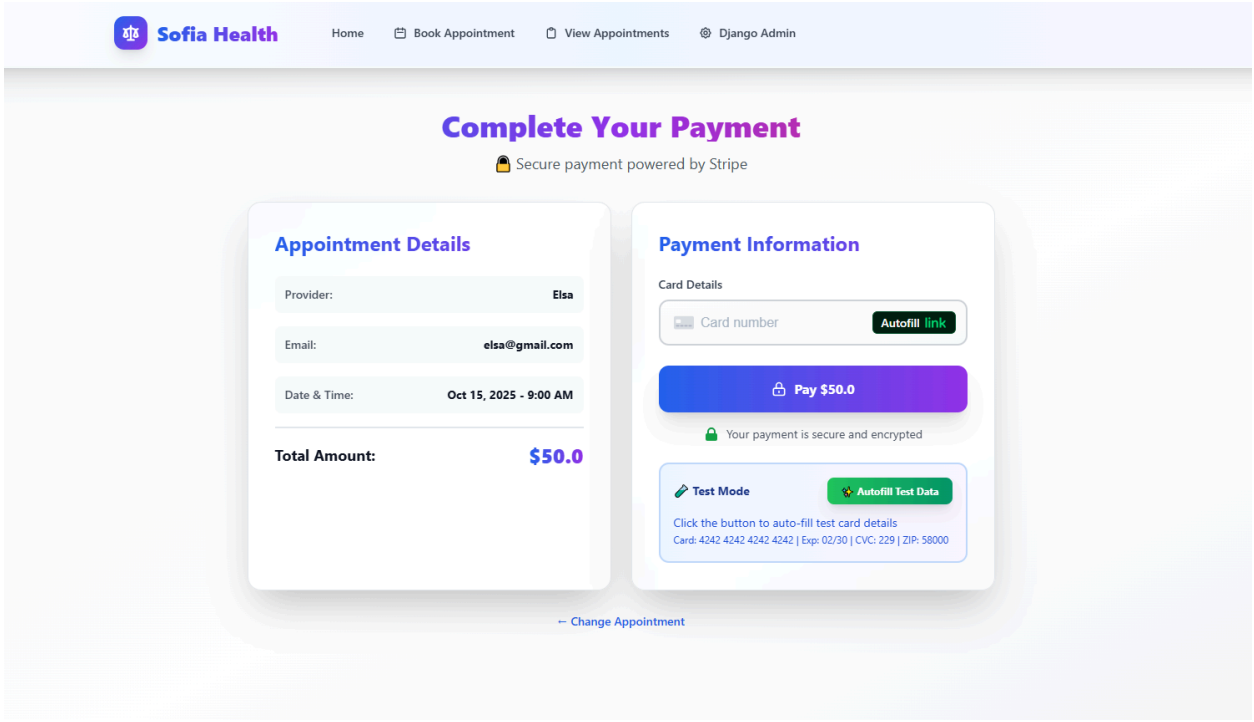




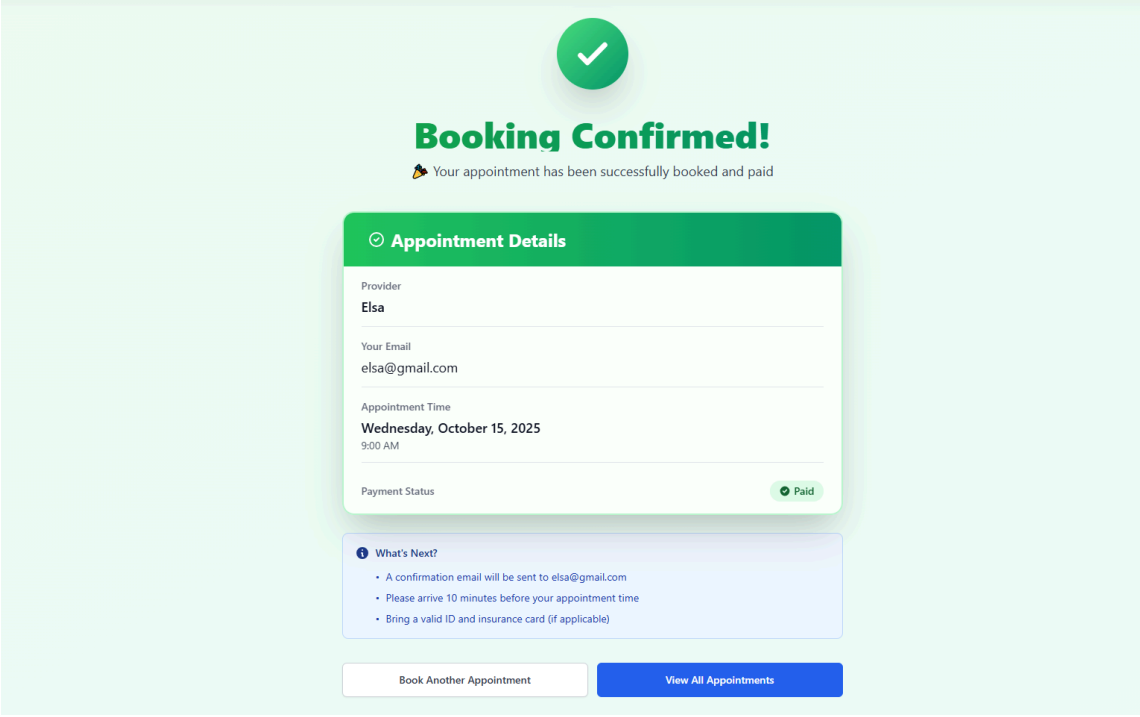
Appointment Booking Page



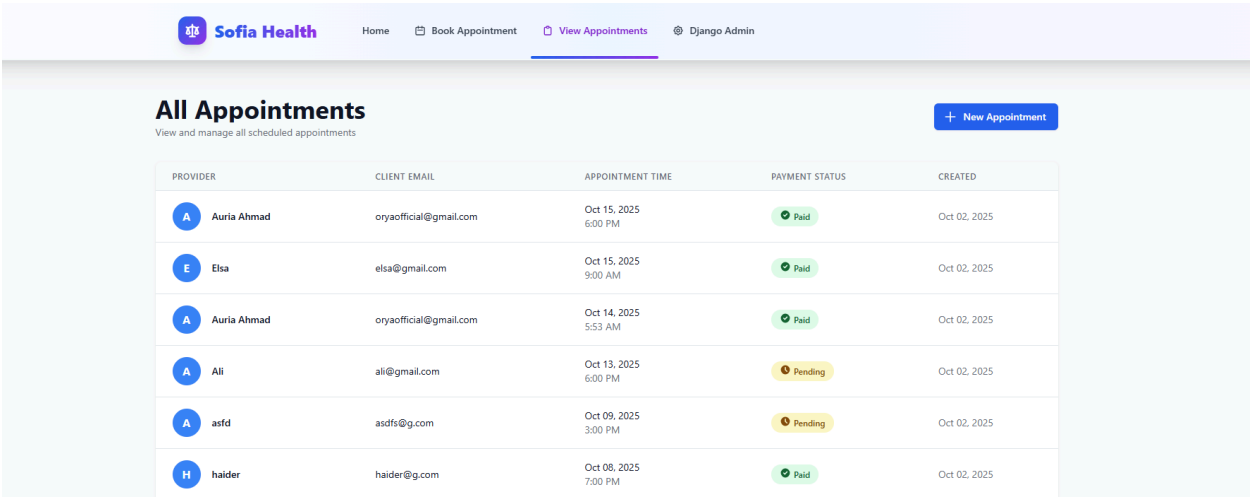
Payment Page



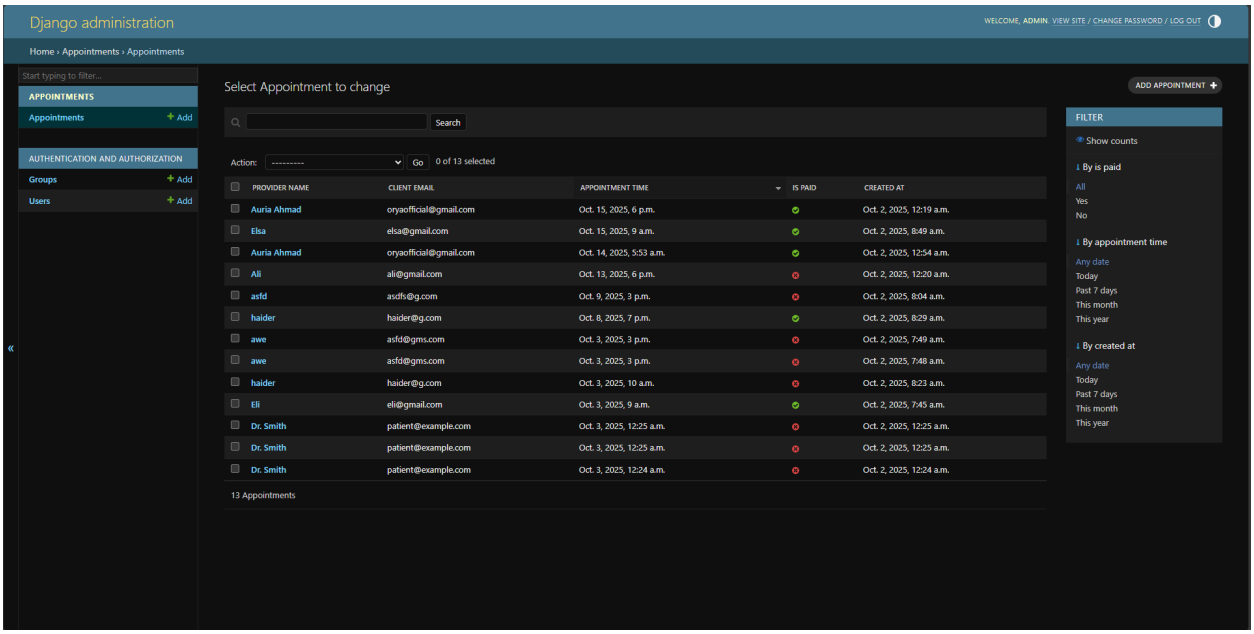
Payment Confirmation



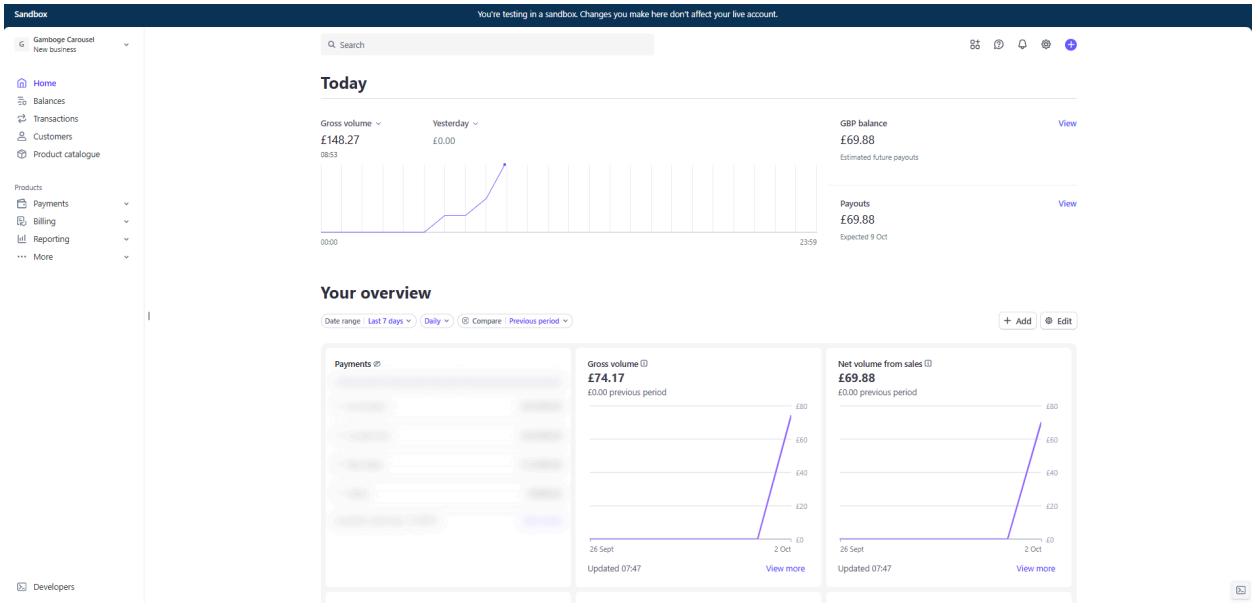
All Appointments Page



Django Admin



Stripe Test Payment Dashboard



Part 2: Problem-Solving & Communication

If this appointment feature had to handle thousands of bookings per day, what one or two changes would you make first?

The first change I would make is adding database indexes on frequently queried fields like appointment_time, client_email, and is_paid. This provides immediate performance gains with minimal code changes - just a few lines in the model's Meta class. The second change would be implementing Redis caching for frequently accessed data such as available time slots and provider information. These two optimizations address the most common bottlenecks (slow database queries and repeated computations) before requiring major architectural changes like moving to microservices.

2. How would you communicate progress and blockers to a non-technical founder?

I would use a structured, transparent approach with daily quick Slack updates (30 seconds) covering what I completed, what I'm working on, and any blockers, plus a detailed weekly summary showing accomplishments and upcoming work. When communicating blockers, I'd explain the business impact first (e.g., "We can't launch payments yet because Stripe needs to approve our account"), then offer 2-3 solutions with clear trade-offs and specific timelines rather than vague estimates. I'd also provide short video demos of working features and use visual progress indicators (like "3 out of 5 features complete") instead of technical jargon. This builds trust through transparency while keeping communication concise and actionable for decision-making.

Thank you