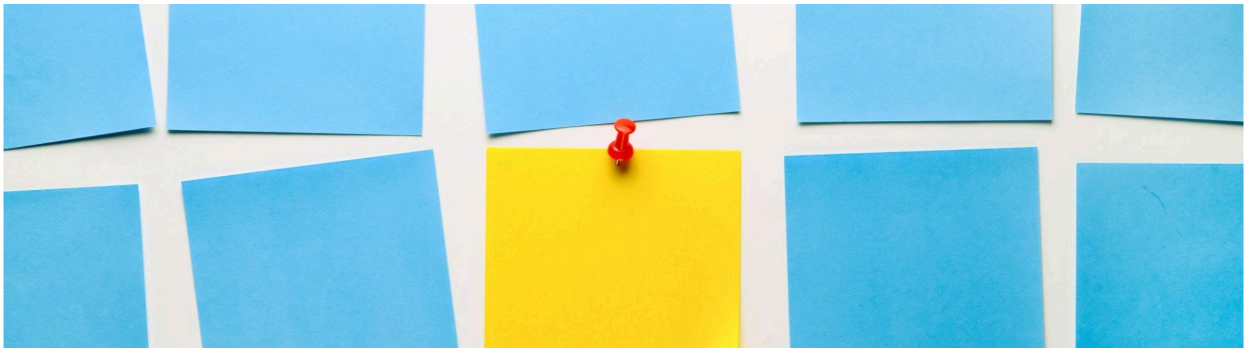


Project brief

Status In progress ▾

Timing Jan 23, 2026 to Jan 26, 2026

Owners Salman Birajdar

Overview

Acme Dental is a simple dental practice that offers routine check-up appointments. The clinic operates with a single dentist and uses Calendly to manage its appointment scheduling. Below are the key details you'll need to know about the clinic's operations.

Clinic Information:
Service Type: Dental Check-up (only service offered)

Appointment Duration: 30 minutes
Staff: Only 1 dentist
Scheduling System: Calendly

- Calendar URL: Provided in the email.
- Calendly Token: Provided in the email.
- Use the Calendly API to check availability and manage bookings

Objectives

Build an AI agent that allow users to book dental check-up appointments with a fictitious Acme Dental clinic through natural language conversation in a chat interface.

Project objectives

Functional Requirements

1. Create New Bookings
    - Greet users and understand their booking intent
    - Check available appointment slots via the Calendly calendar
    - Present available time slots to users
    - Help users select a suitable time slot
    - Collect necessary patient information (i.e full name and email address)
    - Create the booking through the Calendly API
    - Provide confirmation with appointment details (date, time, duration)
2. Reschedule Existing Bookings
    - Allow users to reschedule their existing appointments
    - Identify the booking to update
    - Retrieve current appointment details
    - Check availability for the new requested time slot
    - Reschedule the booking through the Calendly API
    - Provide updated confirmation with new appointment details
3. Cancel Bookings
    - Allow users to cancel their appointments
    - Identify the booking to cancel
    - Retrieve current appointment details
    - Process cancellation through the Calendly API
    - Provide cancellation confirmation
4. Answer FAQs from the Knowledge Base

    - A document containing the clinic's knowledge base (KB) will be provided
    - Extract and process information from the document to build a searchable knowledge base
    - Answer frequently asked questions about the clinic using information from the KB

Non-Functional Requirements

- Implement the agent using LangGraph (https://docs.langchain.com/oss/python/langgraph/overview)
- You are free to choose the LLM model(s) or a combination of models you consider most appropriate for this task
- Be aware that API integrations (e.g., Calendly) may be unreliable or experience delays
- Document your architectural decisions

Strategy

Target audience

- Users are trying to book an appointment for Acme Dental
- Doctor using the Calendly app to view the bookings

Measurement

Success Metrics:

- All the functional requirements above are met
- CSAT > 95 % (This is the customer satisfaction metric)
- Task completion rate ~ 100 %
- Intent recognition rate (How accurately the model understands the customer intention)
- Conversation efficiency (Time taken from conversation start to completion of a task)
- Latency Target: < 1.5sec (Time taken by agent to give a response)
- Reliability ~ 100%

🗋 File 🗋 File

Edge Cases:

Edge Case	Risk	Mitigation
Same user books multiple times	Double appointments	Check availability window limits
Race condition - two users book same slot	One fails	Handle 409 conflict, offer next available slot
User abandons mid-conversation	Incomplete data	Don't persist partial bookings, session timeout cleanup
User in different timezone	Books wrong time	Detect timezone from browser, confirm with user

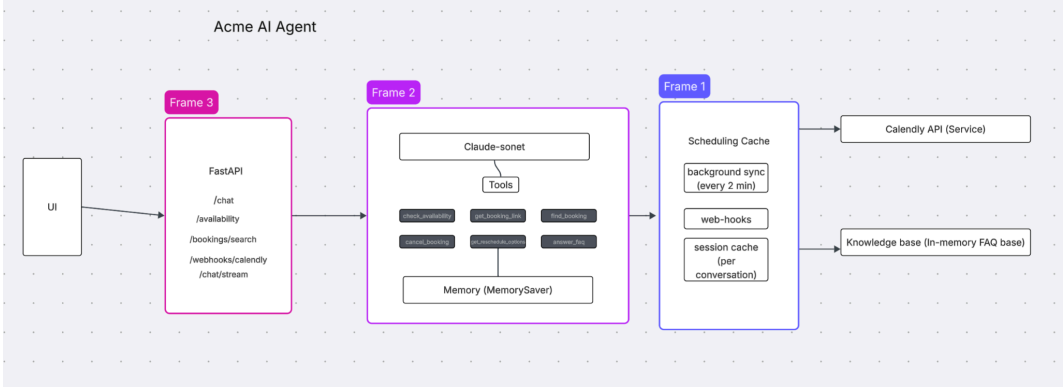
Bottlenecks

- LLM latency (1-3 sec typical)
    - Streaming responses, smaller model for simple queries
- Calendly API latency
    - Aggressive caching (poll status update)

Scalability

- In-memory Cache
    - Possibly move to DB
- Single Destist calendar
- Conversation in memory
    - Possibly move conversation to DB

Design:



Milestones

Date	Milestone	Description	Expected outcome
🗓 Date	Milestone 1	Add LLM service using Langraph and Claude-sonet	LLM returning responses
🗓 Date	Milestone 2	Integrate an agent with the LLM, along with the tools definition	LLM reasoning well with requests and assigning tools for action
🗓 Date	Milestone 3	Implement scheduling cache and web-hooks for Calendly	In memory cache updated every 2 mins to keep requests in sync with the user
🗓 Date	Milestone 4	Implement FastAPI - Set up endpoints for chat and webhooks	API endpoints returning responses
🗓 Date	Milestone 5	Implement a lightweight UI (Vite -Typescript/React)	UI can send requests and receive responses
🗓 Date	Milestone 6	Deploy on Railway	App available on production with a URL to access it

Date	Milestone	Description	Expected outcome
🗓 Date	Milestone 7 (Stretch)	Add SSE streamer to reduce latency on response times	Stream responses as they come

Open questions

- 1. What happens if the Calendly calendar is manually edited? (Webhook handles new bookings, but what about manual time blocks?)
- 2. Should we support multiple languages? (Current implementation is English-only)
- 3. What's the cancellation policy? (How close to appointment time can users cancel?)
- 4. Should we send confirmation emails? (Currently relying on Calendly's built-in emails)
- 5. How long should conversation history persist? (Current: in-memory, lost on restart)
- 6. What's the maximum booking window? (How far in advance can users book?)

Resources

Deployed at: https://acme-dental-front-end-production.up.railway.app/
Repo at: https://github.com/salbirajdar/acme-dental/tree/main

Readme: https://github.com/salbirajdar/acme-dental/blob/main/README.md