

PROBLEM

Coordination is eating the workday. Professionals lose hours every day to repetitive scheduling and communication by checking calendars, drafting routine emails, and chasing follow-ups across disconnected tools. There's no unified, conversational way to handle both calendar and email in plain language.

Existing solutions often require navigating separate apps, manually checking availability, and switching context between email composition and event creation reducing productivity for executives and teams.

CURRENT SYSTEM

Today's tools (Google Calendar, Gmail, Outlook) operate in isolation. Users must manually cross-reference availability, switch between apps, and draft emails without contextual assistance.

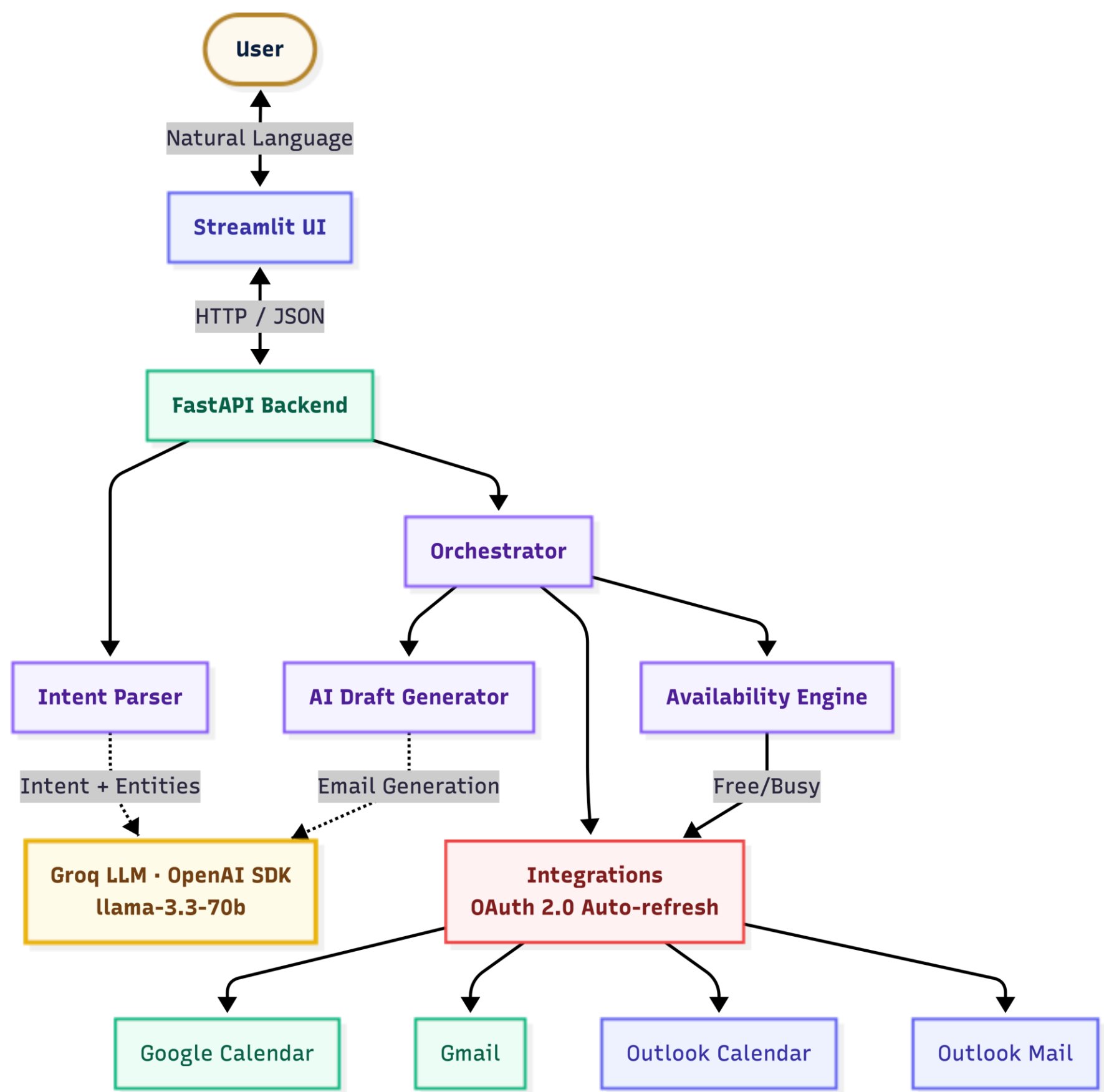
No system interprets a single request like *"Reply to my latest email saying I'm available at 2pm and create the meeting"* and executes all steps automatically through integrated services.

REQUIREMENTS

- Natural language interface for scheduling and email tasks
- Intent detection and entity extraction from free-form text
- Real-time conflict detection against live calendar data
- Alternative time slot suggestions when conflicts are found
- AI-generated email drafts and replies with tone control
- Secure OAuth 2.0 for both Google and Microsoft Outlook
- Combined workflows: reply + book, draft + schedule
- Transparent reasoning via debug inspection panel

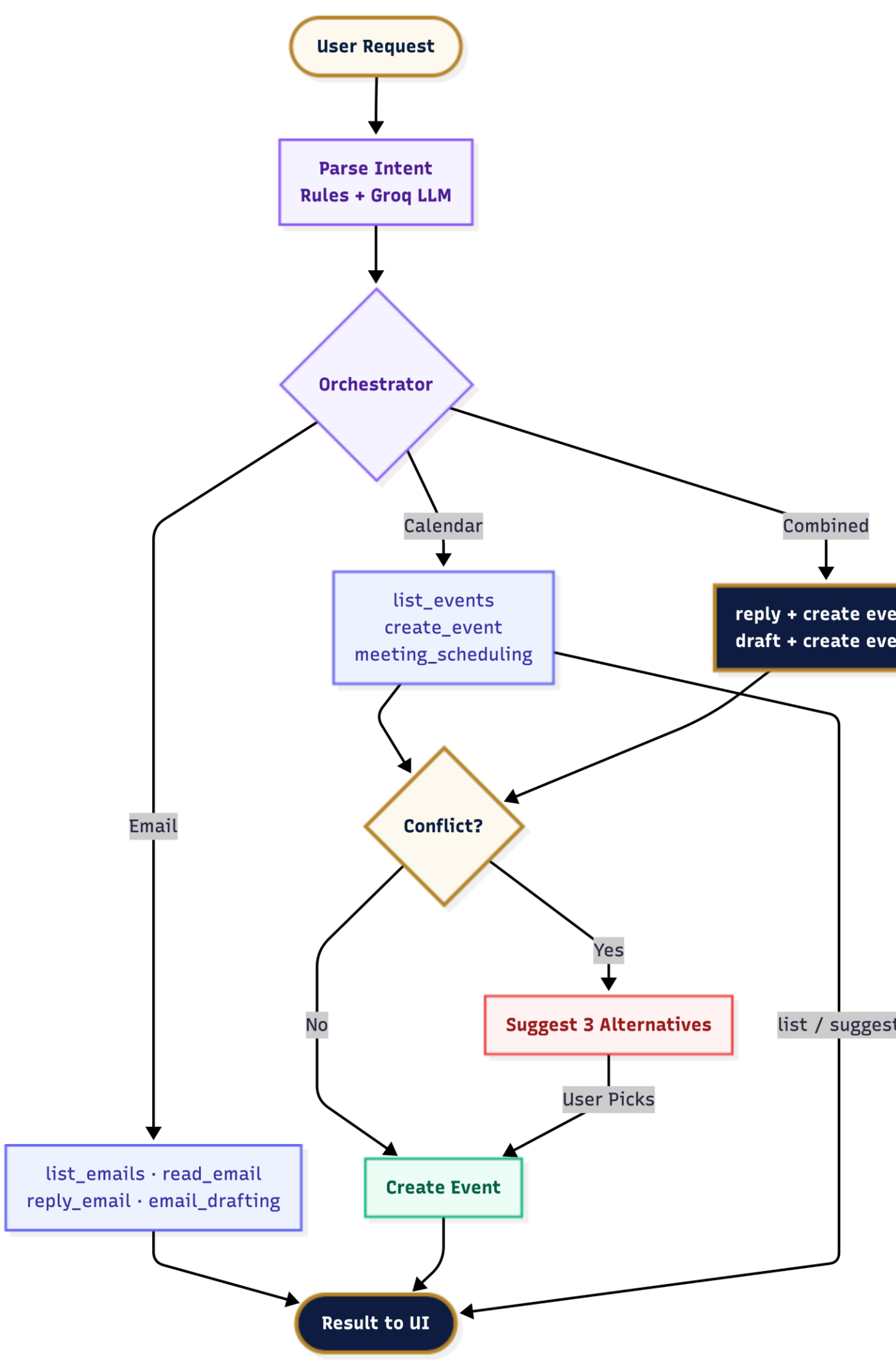
SYSTEM ARCHITECTURE

ExecAI compresses hours of coordination into a short conversation.



AGENT DECISION FLOW

The orchestrator routes each intent through conflict-aware, provider-agnostic workflows:



IMPLEMENTATION

- 1 **Intent Parser** — Hybrid NLP: rule-based regex + OpenAI(primary) , with Groq fallback. Extracts 12+ entity types: title, timeframe, attendees, tone, body hint, duration.
- 2 **Orchestrator** — Decision engine mapping 9 intents to provider-agnostic actions. Handles conflict detection, alternative slots, and atomic combined workflows.
- 3 **Availability Engine** — Queries real free/busy data via Google Calendar or Microsoft Graph. 30-min slots within 9am–5pm, ranks top 3 conflict-free.
- 4 **AI Draft Generator** — OpenAI LLM generates context-aware emails, with Groq fallback and template fallback. For replies, injects real thread content (subject, sender, body). Template fallback.
- 5 **Integrations** — OAuth 2.0 with auto-refresh for Google & Microsoft. Calendar CRUD, Gmail drafts/send, Outlook via Graph API.

EXAMPLE INTERACTIONS

EVENT CREATION

"Schedule a budget review with **johanahuertas24@gmail.com** tomorrow at 2pm for 45 minutes"
→ Parses intent → Checks conflicts → Creates calendar event → Sends invite

CONFLICT DETECTION

"Schedule a meeting with **johanahuertas24@gmail.com** tomorrow at 3pm"
→ Detects busy slot → Suggests alternative times → User confirms new slot

REPLY + BOOK MEETING

"Reply to my latest email from **johanahuertas24@gmail.com** saying I'm available at 2pm and create the meeting"
→ Reads email → Generates reply → Books calendar event in one flow

EMAIL DRAFTING

"Draft a professional email to **prof.smith@fiu.edu** requesting an extension"
→ Generates AI email draft → Applies tone → Ready to send

KEY RESULTS

9

Intents

2

Providers

3

Alt. Slots

12+

Entities

Technology Stack

Python

FastAPI

Streamlit

Pydantic

Groq LLM

Google Calendar API

Gmail API

OAuth 2.0

Microsoft Graph

Open AI

VERIFICATION & TESTING

- Intent Accuracy:** 9 intents across 40+ NL variations , rule-based and LLM parsers
- Integration Testing:** E2E for Google & Outlook event creation, conflict detection, drafts
- OAuth Flows:** Token refresh, expiry, re-auth for both Google and Microsoft
- Conflict Engine:** Real busy blocks with correct alternative slot ranking
- Combined Workflows:** reply+create and draft schedule , atomic with fallback
- AI Drafts:** Contextual replies with real email thread + template fallback

SUMMARY & FUTURE WORK

ExecAI turns natural language into real calendar and email actions across Google and Outlook by replacing manual coordination with a single conversation. It demonstrates hybrid intent parsing (rules + LLM), orchestration-first design, conflict-aware scheduling, and a provider-agnostic integration layer. The system contributes: hybrid intent parsing (rules + LLM), orchestration-first design, conflict-aware scheduling, AI-generated communication, and a provider-agnostic integration layer via OAuth 2.0.

Future Work:

- Multi-step task planning and user preference memory
- Multi-user auth and shared team calendars
- Cloud deployment (AWS / GCP) for production
- Advanced NLU and multi-turn dialog

REFERENCES

- Google Calendar API — developers.google.com/calendar
- Gmail API — developers.google.com/gmail
- Microsoft Graph API — learn.microsoft.com/graph
- FastAPI — fastapi.tiangolo.com
- Streamlit — docs.streamlit.io
- Groq LLM API — console.groq.com/docs
- OAuth 2.0 — oauth.net/2
- Pydantic — docs.pydantic.dev
- OpenAI API — platform.openai.com/docs

ACKNOWLEDGEMENT

We thank our instructor and faculty advisor for their guidance and mentorship throughout the senior design project.