

Luan A. Tobias
Author

Dr. Emanuel Santos
Instructor/Teacher

ABSTRACT

The goal of this project was to create an AI assistant chat bot capable of interacting with the user's google calendar in order to assist with schedule related tasks. The project's functionality proved to be useful and reliable enough that an API was later developed in order to host the project on the cloud and support multiple frontend variations.

EVENT CREATION EXAMPLE

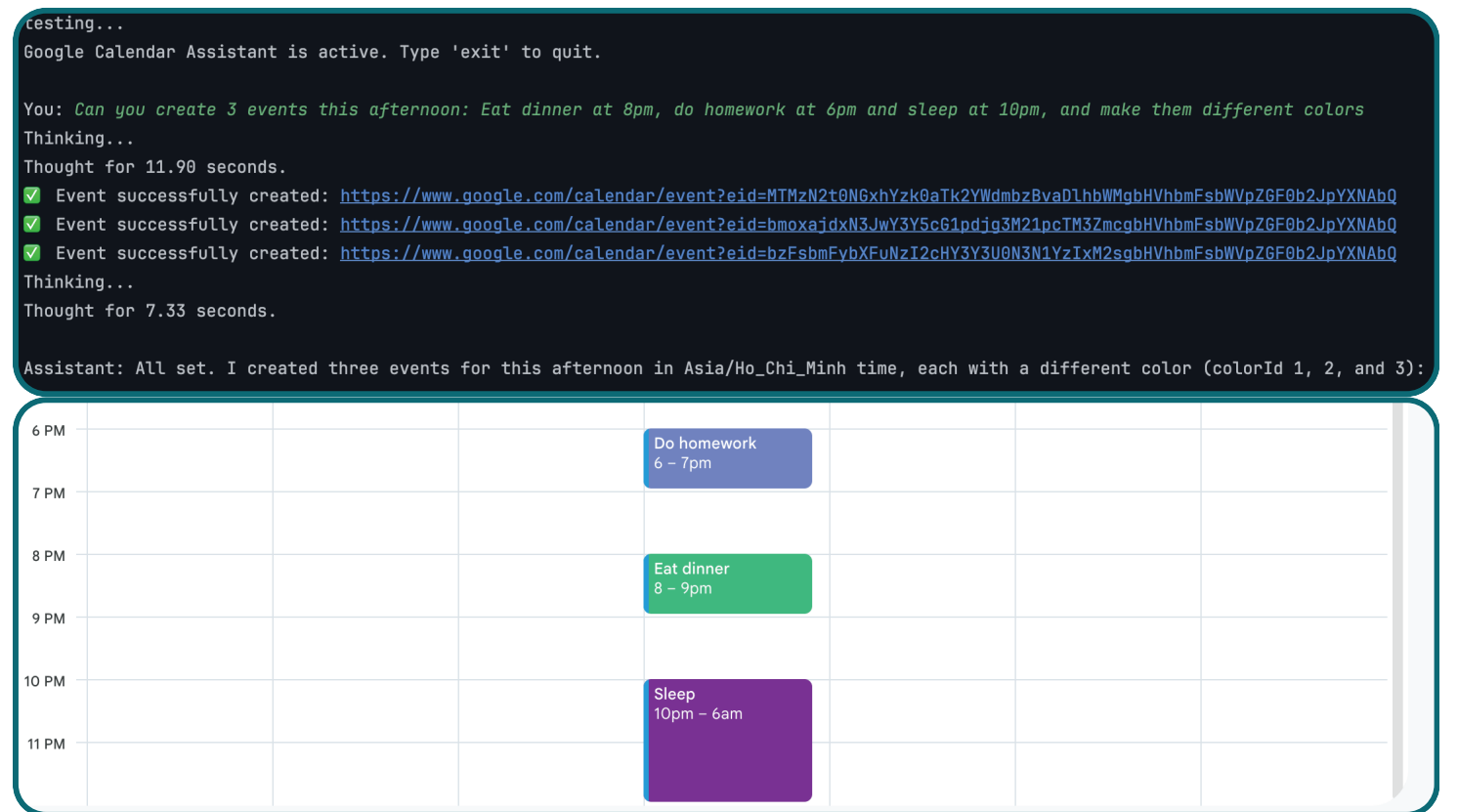


Fig. 1 – Agent testing for creating events into a google calendar

API WORKFLOW

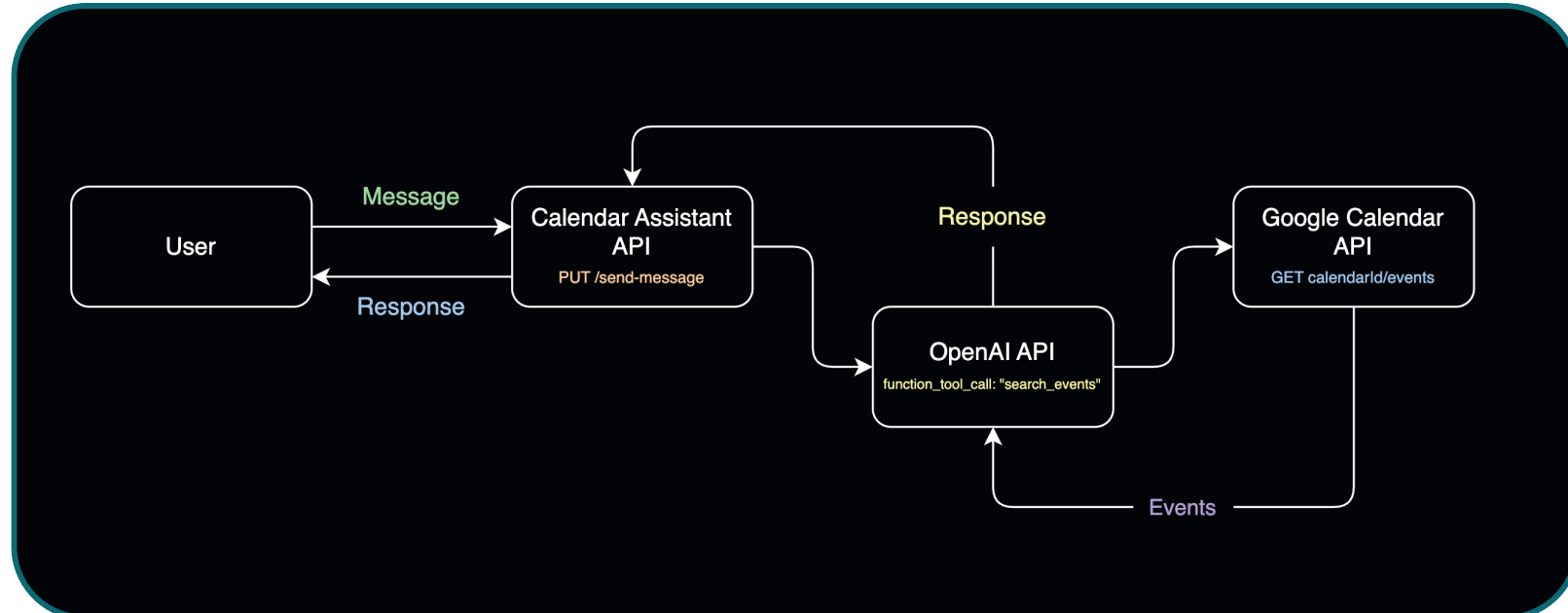


Fig. 2 – API calls flowchart

API FUNCTIONALITY

The **Calendar Assistant API** serves as a bridge between the LLM model and Google Calendar, executing the necessary functions requested by the model. It relays the function's output to the model and properly formats the model's response before passing it back to the user.

The functions available for the model to call are predefined in a JSON file, and divided into two categories: Reading & Writing

- Reading: retrieves data from a calendar (eg. list_events, search_events)
- Writing: sends data to a calendar (eg. create_event, update_event, delete_event)

As reading & writing to a private calendar requires authentication from the user, the current API version only includes reading from public calendars to avoid security concerns.

PROJECT DEMO

To test the project demo scan this QR code. Scroll down to **Messages**, click on **/send-message** → **Try it out**, scroll down to the Schema box until you see this:

```

Edit Value | Schema
{
  "chat": [],
  "chat_history_limit": 3,
  "message": "When is the next School Assembly?"
}

```

You can change the text in between the quotation marks to ask the agent anything about Concordia related events (this version only has access to the Concordia Calendar)



API SWAGGER UI

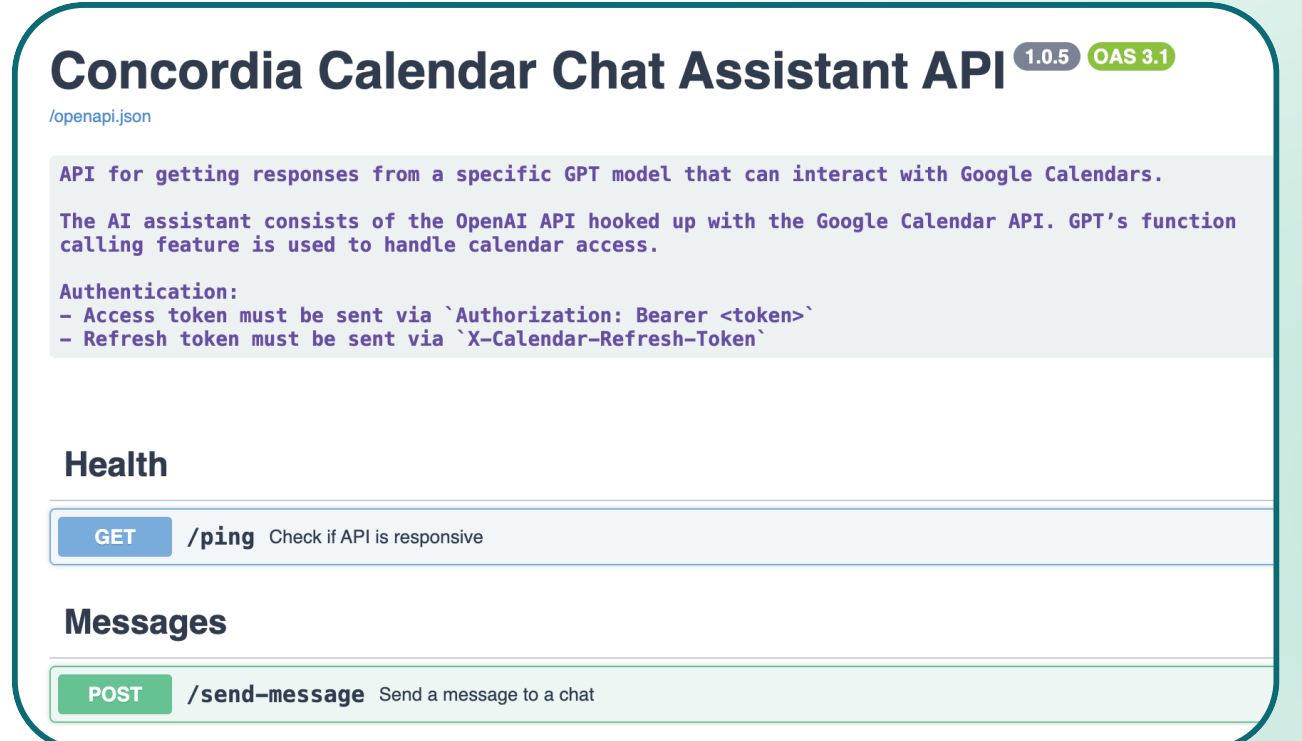


Fig. 3 – API UI for testing endpoints