

Cayson Yim

I Topic Selection and Capabilities

The inspiration for this demo stemmed from my experience with helping my mom start her floral business. I saw how critical a smooth delivery process is for both customer satisfaction and successful business operations. That insight led me to explore utilizing Inngest's Realtime API for providing realtime delivery status updates and intuitive interactions with a chatbot.

The core capabilities of the demo include:

- Realtime delivery status updates
- OpenAI powered chatbot

II Implementation Details

Inngest Functions

track-delivery: In order to allow users to monitor each ongoing delivery, status updates are published via this Inngest function. When a user books a delivery, it sends the "delivery/delivery.requested" event, triggering this function. It publishes delivery status updates to the appropriate deliveryChannel using the delivery's unique id.

communicate-with-chatbot: To allow users to experience an intuitive interaction with the chatbot, its responses are streamed directly to the user interface. When a user sends their prompt to the chatbot, it sends the "ai/ai.requested" event, triggering this function. It processes the user's prompt and using the OpenAI API, generates a response that streams to the user's userChannel.

Channels

userChannel: Each user is subscribed to a dedicated userChannel (for demo purposes we are dealing with one user). This channel has a topic, "ai", which streams responses from the OpenAI API back to the user.

deliveryChannel: Each delivery subscribes to a dedicated deliveryChannel provided by their unique id. Delivery statuses are streamed to the user interface through the channel's "status" topic, allowing users to monitor their deliveries.

UI

The UI is defined in `page.tsx`. This file renders the application's interface and manages its state via React's `useState`.