Brief Notes on Key Prompt Engineering Decisions

This document outlines the core prompt engineering decisions for the TripSmith travel assistant, focusing on architectural choices, LLM behavior management, and personalization.

---

1. Conversational Architecture

Multi-Stage Prompt Pipeline: The system uses a modular, multi-stage prompt architecture instead of a single monolithic prompt. This pipeline includes:

- **System Prompt:** Establishes the assistant's persona and core constraints.
- **Router Prompt:** Uses few-shot classification for intent detection and entity extraction.
- **Context-Aware Prompts:** Dynamically adapt to conversation history and user-provided information.
- **Tool Integration Prompts:** Guide the LLM to incorporate external data naturally.
- **Rationale:** This separation allows for specialized optimization at each stage, leading to a more predictable and coherent conversation flow.

Chain-of-Thought: A hidden scaffold guides the LLM's internal reasoning process (e.g., inputs → tools → factors → response). The LLM's thought process remains private to the user to avoid conversational clutter while ensuring systematic decision-making.

---

2. LLM Model Selection and Factual Integrity

Model Choice: The project was developed and tested on both Groq and Ollama. While Groq offered superior performance and speed, it was found to have a higher tendency for factual hallucination. For this reason, **Ollama was chosen for the final implementation due to its more conservative behavior, which led to fewer fabricated responses.** This decision prioritized trustworthiness and reliability over raw speed.

Hallucination Prevention: The system is engineered to function as a data interpreter, not a knowledge generator. This is achieved through:

- **Explicit Data-Only Policy:** Prompts explicitly forbid the use of any information not sourced directly from a tool.

- **Transparent Limitation Handling:** The assistant is prompted to gracefully acknowledge data gaps (e.g., "I couldn't find specific events in...") rather than inventing plausible-sounding facts.
- **Self-Checking Mechanisms:** A validation step is integrated directly into the generation prompts, where the LLM is instructed to verify that every fact in its response can be attributed to a tool output before finalizing its answer.

---

3. User Context and Refinement

Anti-Repetition Logic: A detection system was developed to distinguish between genuine user requests for refinement (e.g., adding constraints like "solo traveler") and simple repetitions. When refinement is detected, a context-aware prompt is triggered, instructing the LLM to build upon the previous recommendations rather than generating a new, redundant response.

Progressive Conversation: The prompts are designed to guide the conversation progressively, starting with general recommendations and escalating to refined suggestions, specific planning, and logistics. This approach mimics a natural, helpful interaction and prevents the assistant from overwhelming the user with too much detail at once.

---

4. Personalization and Error Handling

Regional Intelligence: The system incorporates a personalization layer for Israeli travelers. This includes using Tel Aviv as the default origin for budget calculations and a hierarchical cost model to provide more accurate estimates for different countries and regions. This approach provides practical utility that generic travel advice would lack.

Graceful Degradation: Error recovery is built directly into the prompts. The system can acknowledge tool failures and offer alternative suggestions, ensuring the conversation remains helpful and on-track even when technical components do not provide the requested data.