MELBOURNE 2026

From Bots to Brains

Patterns for Building Intelligent AI Agents

Rakesh Lakshminarayana

https://github.com/rakeshL4

Rakesh Lakshminarayana

Principal AI Consultant
Agile Insights

https://github.com/rakeshL4

https://binarytrails.com/

https://www.linkedin.com/in/rakeshL

MELBOURNE 2026

PROUDLY SPONSORED BY

DIAMOND SPONSOR

pinch

PLATINUM SPONSORS

Octopus Deploy

netwealth

GOLD SPONSORS

Buildkite

Fabric

Microsoft

twilio

umbraco

Vercel

SILVER SPONSORS

aws

Stile

NDC { Sydney }

WIFI BY

Luminary

NETWORKING BREAKFAST

Kaleida

CHILDCARE BY

Non Zero Sum

COFFEE CARTS BY

REA Group

Drawboard

MELBOURNE 2026

PLAY BINGO AND YOU COULD WIN!

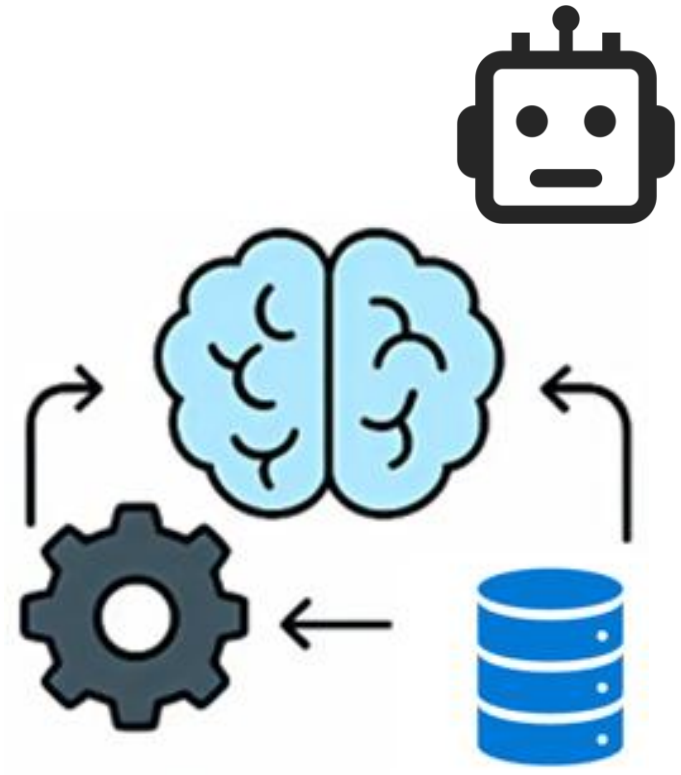
Collect stamps from each sponsor and community booth to complete your bingo card. Drop your card in the prize draw box for a chance to win!

Travel Assistant using Microsoft Agent Framework

AI Agents Workshop

- Remember Me - Personalization
- Remember Everything - Memory
- Give It Superpowers - Tools
- Human-in-the-Loop - Approval workflows
- Specialist Team - Multi-agent collaboration

What is an Agentic Application



Reason & Plan
Break the goal into clear, executable steps.

Remember
Maintain context and memory across interactions

Decide
Evaluate options and choose the next best action.

Act
Use tools, find information, and complete tasks

(Problem Space)

Non-Deterministic

Context & Memory

Tool Orchestration

Multi-Agent Coordination

Safety & Guardrails

Human-in-the-Loop

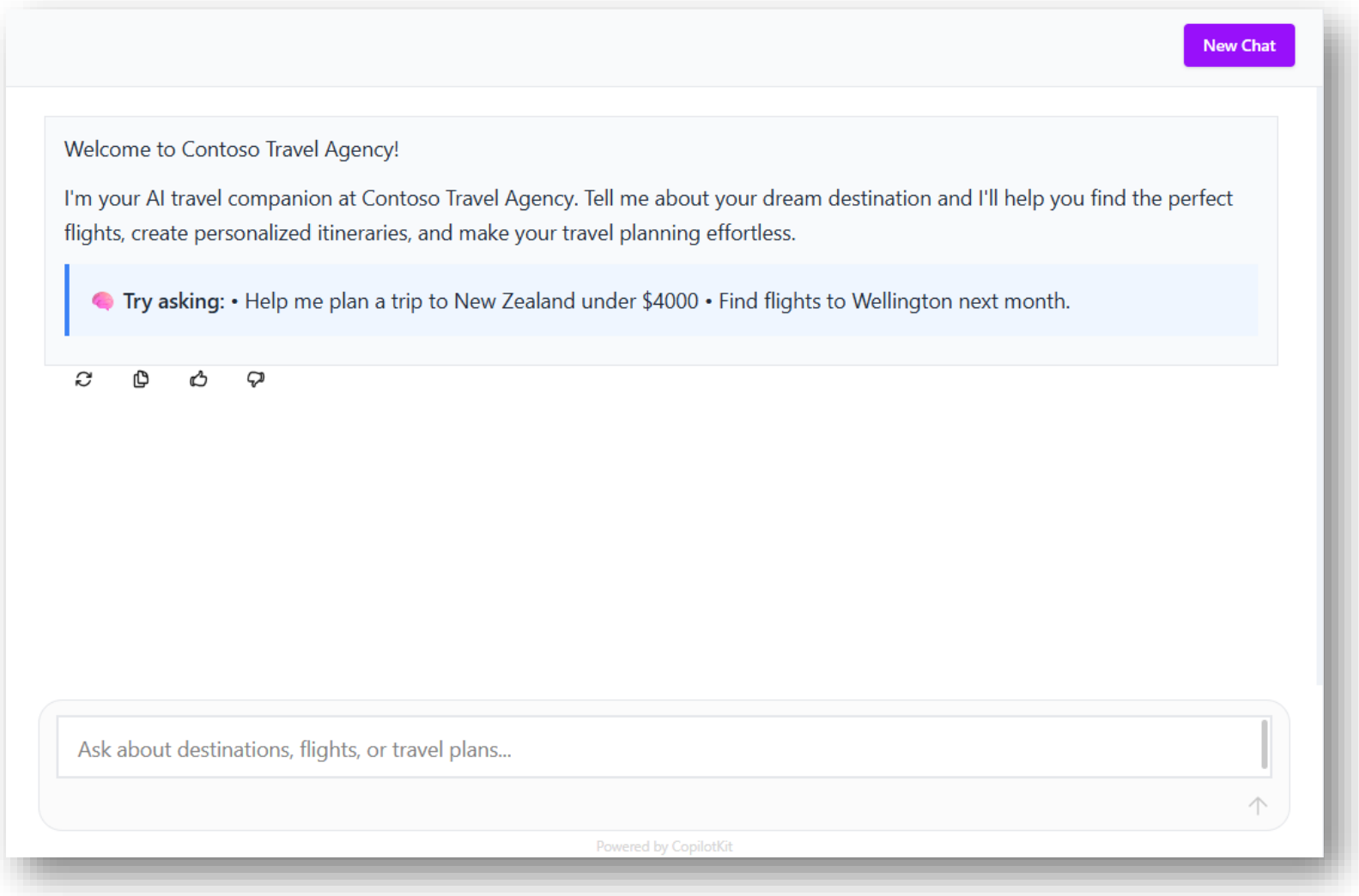
Observability

Cost & Tokens

Testing & Evaluation

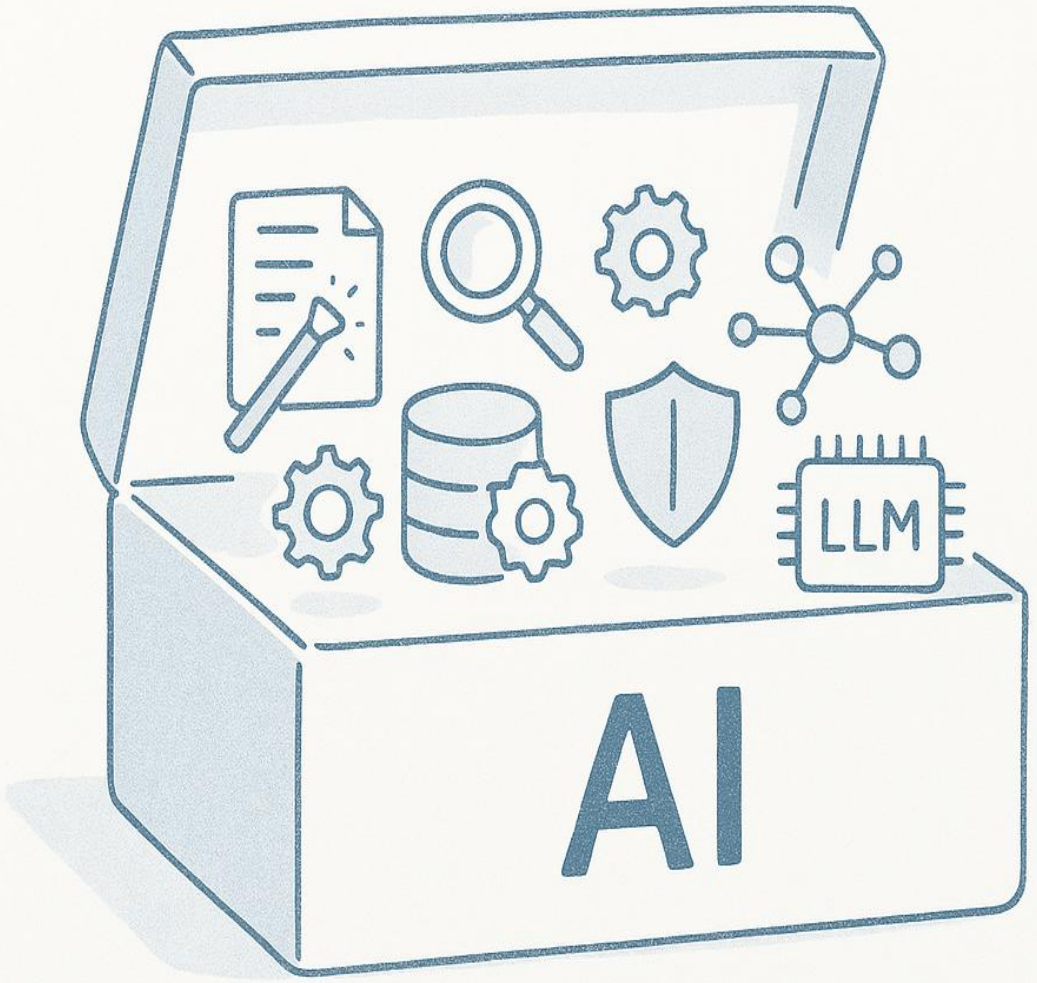
Travel Assistant

- [x] Context & Memory
- [x] Tool Orchestration
- [x] Human-in-the-Loop
- [x] Multi-Agent Coordination
- [x] Observability



DEMO

Travel Assistant



Key Highlights of the Travel Assistant

Personalized Experience
Remembers budget, dates, and preferences

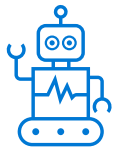
Conversation Memory
Recalls past interactions beyond token limits

Tool Integration
Structured tool calls to take real actions

Key Highlights of the Travel Assistant

Human in the loop (HITL)

User approval for critical actions



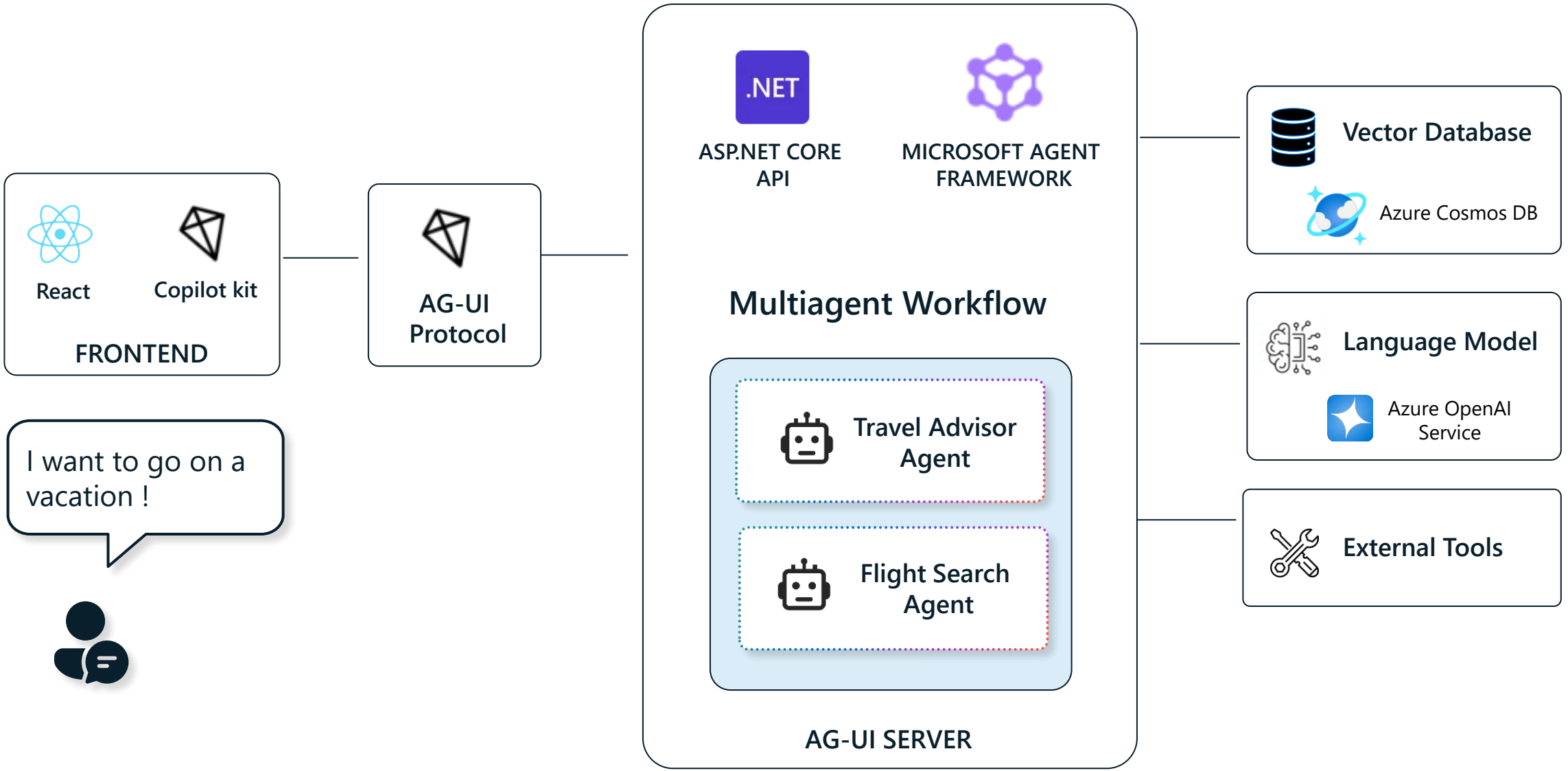
Multi Agent Workflows

Specialized agents collaborating to complete complex tasks

Observability

Full trace of conversations and tool calls

Demo: Travel Assistant



Microsoft Agent Framework

Unified SDK

One SDK for

single & multi-agent systems

Open Standards

MCP (tools)

A2A (agent-to-agent)

AG-UI (frontend)

Pluggable Architecture

Extend LLM providers, memory stores,

and tool adapters

Hosting Options

Deploy as OpenAI compatible HTTP endpoints

Distributed Observability

End-to-end tracing with OpenTelemetry

LAB

Travel Assistant

AI Agents Workshop

Setting Up Your Environment for the Workshop

Prerequisites

- **GitHub Account**: If you don't have one yet, sign up on GitHub.
- **Azure Subscription**: Sign up for a free Azure account.

Setup Source Code Repository

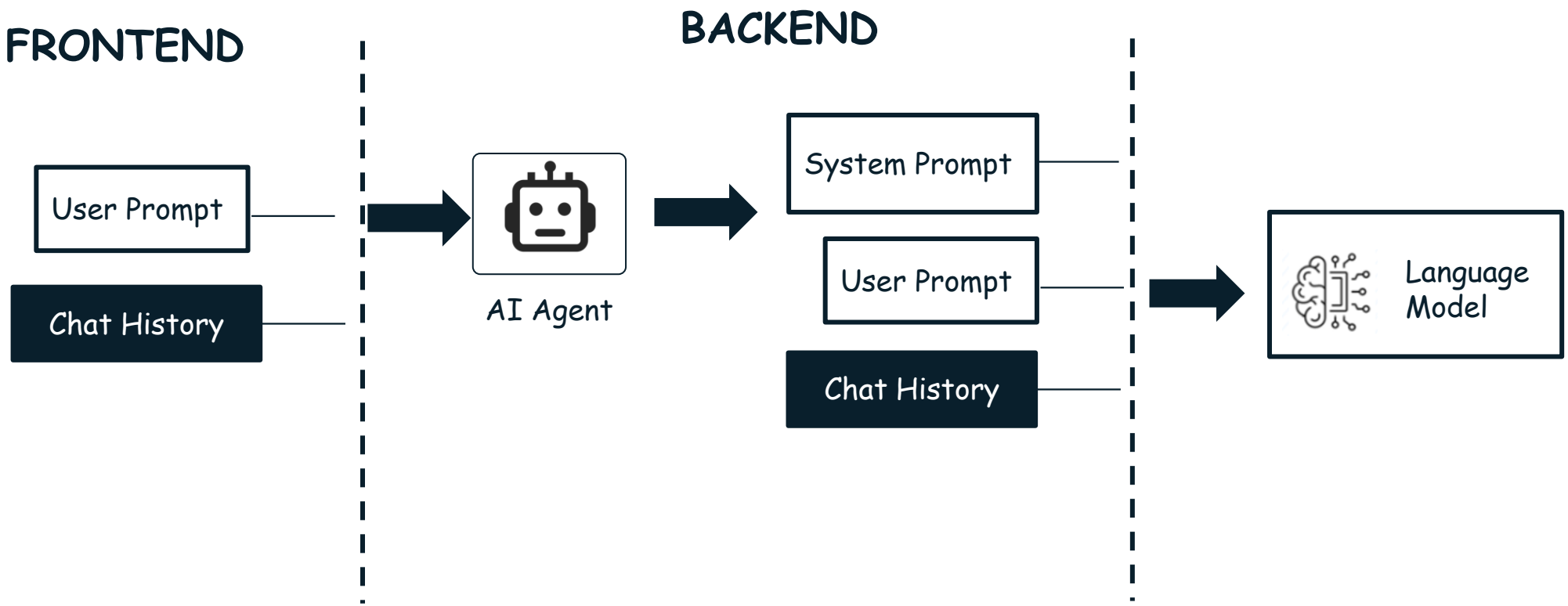
1. From your browser, navigate to the agent-builder-workshop repository on GitHub. This repository has all the code and resources for the workshop.
2. Fork this repository to your own GitHub account.
   FORK REPO
3. The recommended way to work through this workshop is with **GitHub Codespaces**, which provides a ready-to-use environment with all required tools.
   Alternatively, you can use a Visual Studio Code to run the workshop locally.

   Using GitHub Codespaces: Once you've forked the repository, navigate to your forked repository on GitHub and click the green **Code** button, then select the **Codespaces** tab and click **Create codespace on main**.

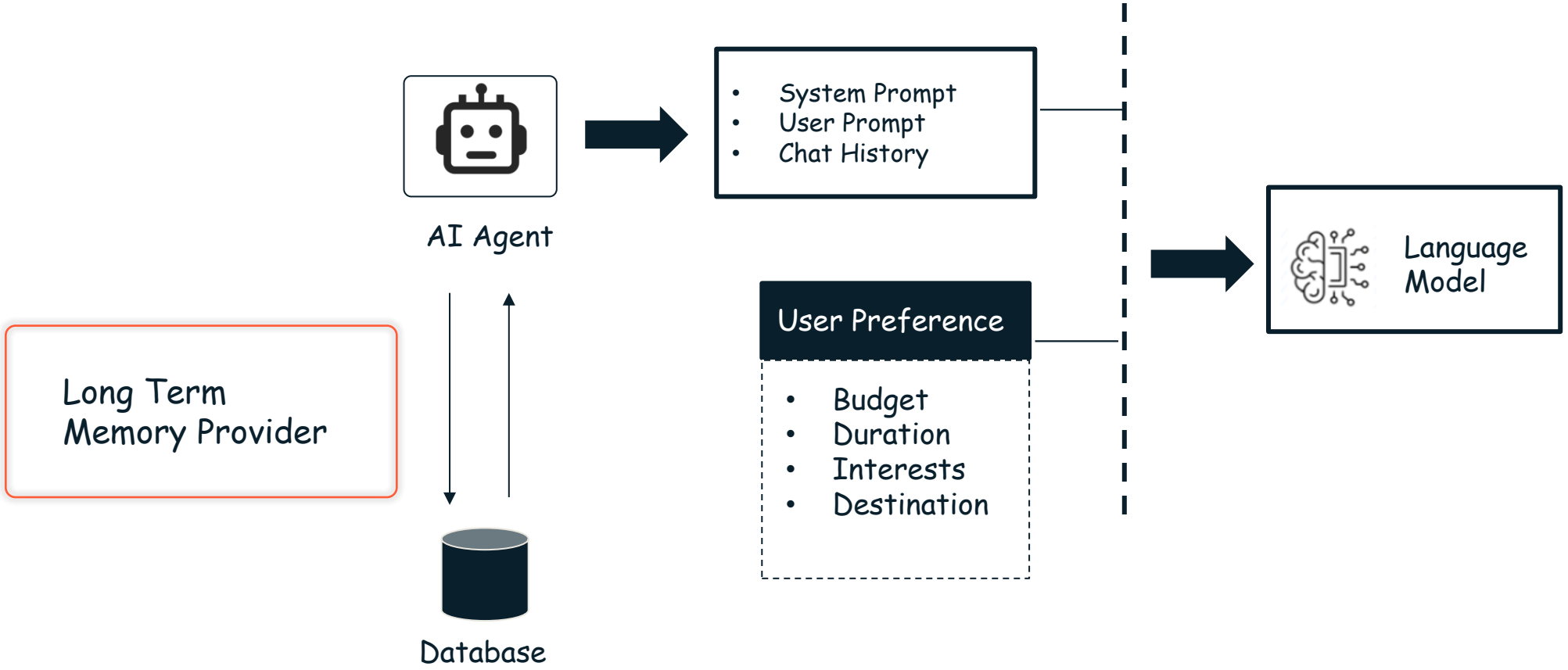
   The Codespace will be pre-configured with all the necessary dependencies and tools to run the labs.

   ⚠ **It may take a few minutes for the Codespace to be created and all dependencies to be installed.**

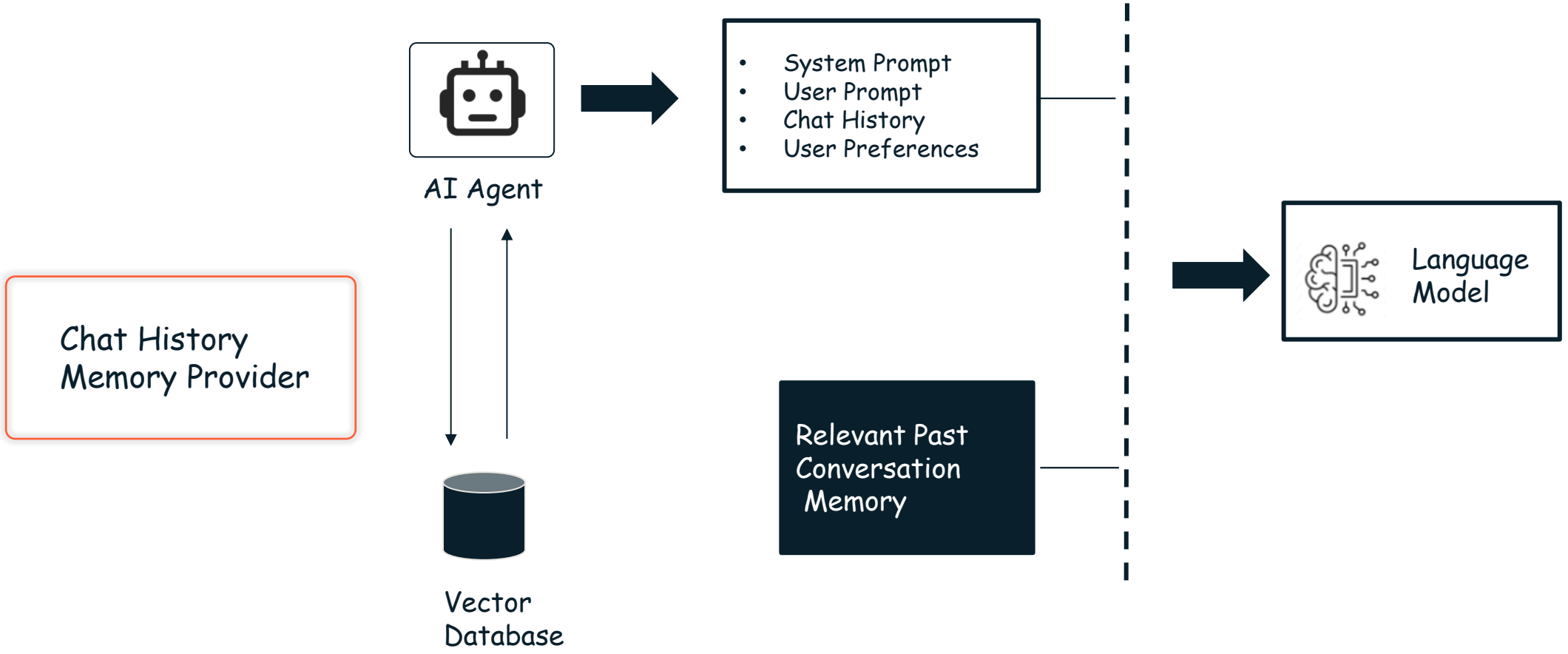
Travel Assistant – Basic Setup



Personalized Experience using Long-Term Memory



Recall Past Conversations using Episodic Memory



How Agents Remember

Short-Term Memory

- Remembers information within a single session
- Maintains conversation context
- Enables coherent responses

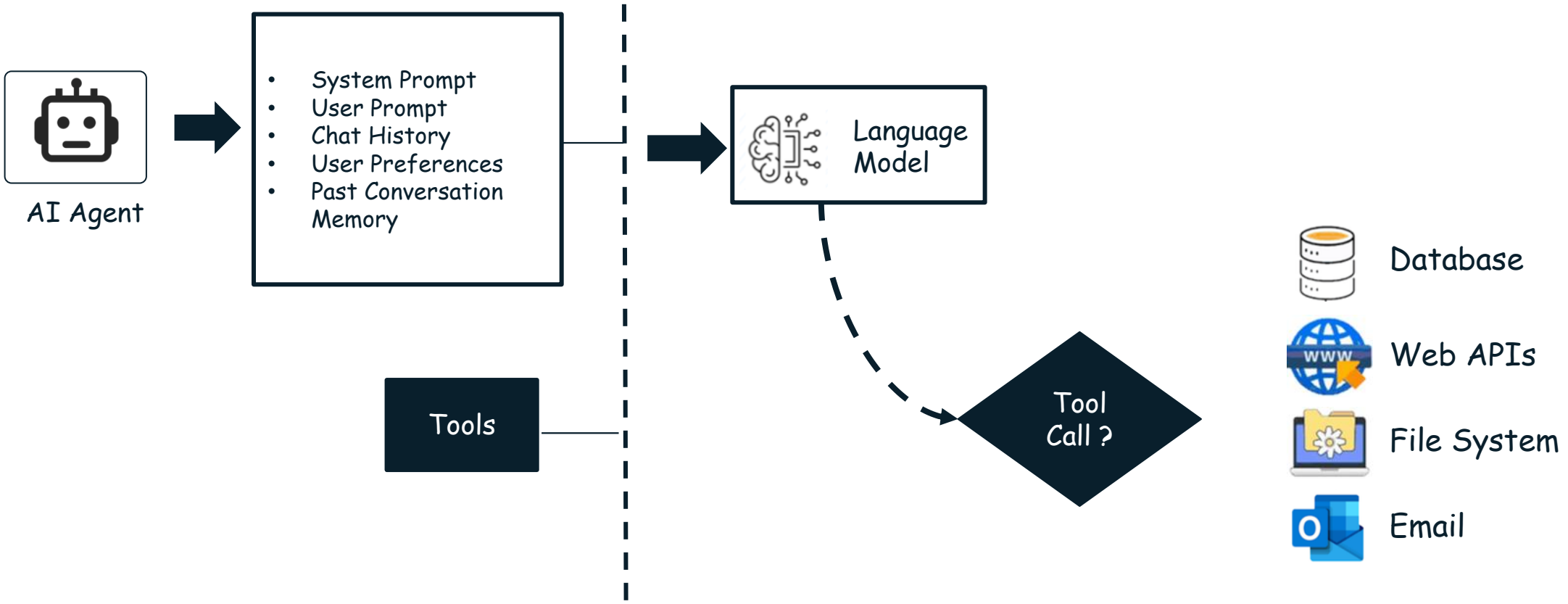
Long-Term Memory

- Persists information across sessions
- Stores user preferences
- Supports personalization over time

Episodic memory

- Enables context-aware responses based on past events
- Captures specific events or interactions

Tool Integration



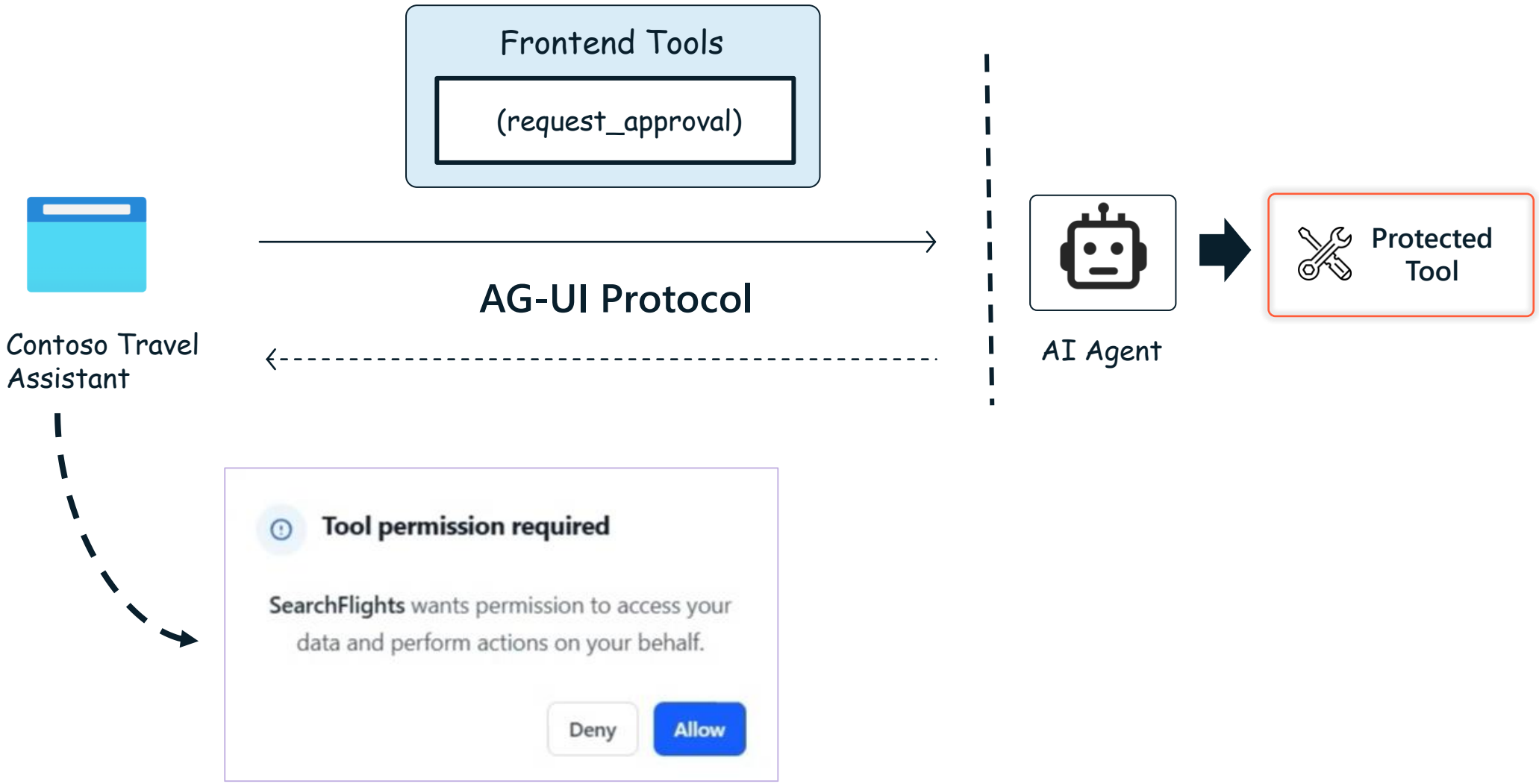
Agent – User Interaction (AG-UI) Protocol

Event-based protocol that enables dynamic communication between agentic **frontends** and **backends**.

Why Agentic Apps Need AG-UI

- Stream results in real time
- Shared, persistent state
- Structured + unstructured I/O
- Pause, approve, retry
- Trace reasoning & tools

Human In The Loop Approval (HITL) Workflow



Why Multi-Agent Applications

Distributed Intelligence

- Eliminate single-agent bottlenecks
- Improve reliability and response quality
- Support more tools and capabilities

Scalability

- Add new specialized agents easily
- Scale horizontally without redesign
- Support evolving business needs

Modular Architecture

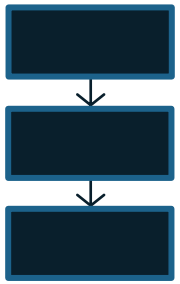
- Update agents independently
- Swap capabilities without disruption
- Accelerate iteration and innovation

Parallel Execution

- Reduce overall processing time
- Improve responsiveness and throughput

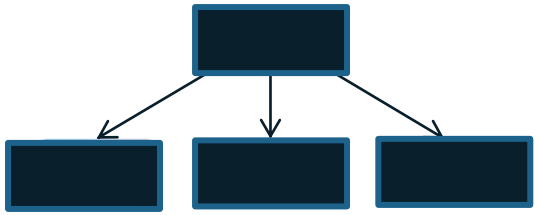
Workflow Orchestration Patterns

Sequential



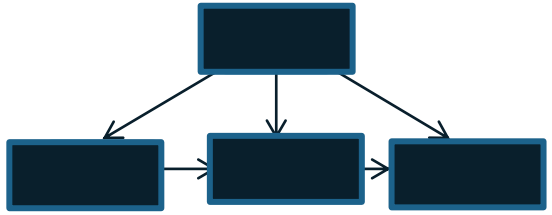
Agents perform tasks one after another in a fixed order.

Concurrent



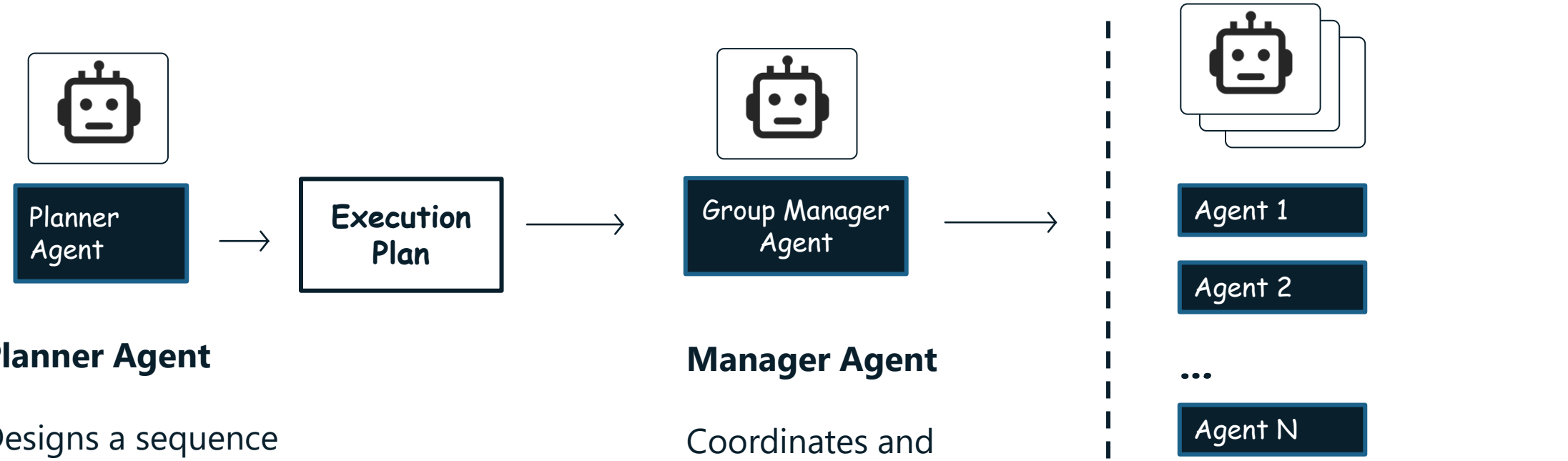
Multiple agents work in parallel on different parts of a task.

Handoff



One agent completes its part and then passes control or data to another agent.

Magentic Workflow Orchestration



Planner Agent

Designs a sequence of actions to achieve a goal

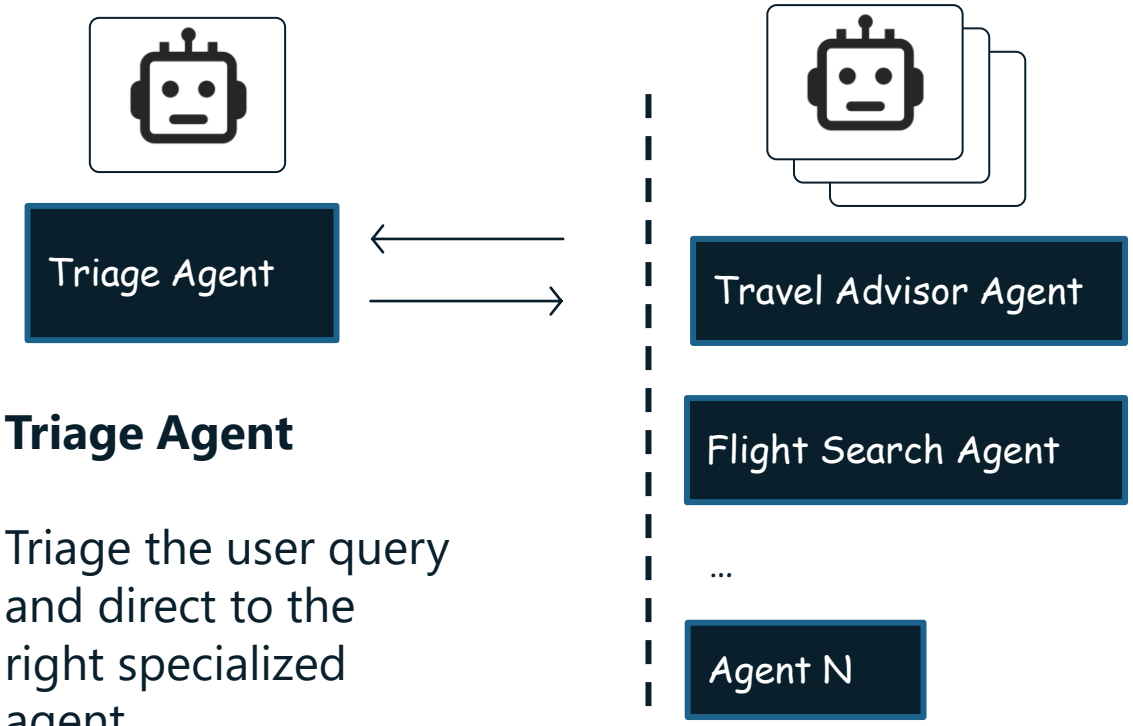
Manager Agent

Coordinates and manages a team of agents

Multiple Specialized Agents

Each agent designed to perform specific tasks efficiently

Travel Assistant – Workflow Orchestration



Triage Agent

Triage the user query and direct to the right specialized agent

Multiple Specialized Agents

Each agent designed to perform specific tasks efficiently

Handoff Workflow Pattern

Next steps

1. Learn more about Microsoft Agent Framework
   Microsoft Agent Framework Overview

2. Build and deploy Agents on Microsoft Azure
   Develop AI agents on Azure

3. AI Agent sample code using Microsoft Agent Framework
   Agent Framework .NET Samples

4. Learn more about Agent–User Interaction protocol
   AG UI Documentation

MELBOURNE 2026

THANK YOU !

Let's stay connected

AI Agent Workshop