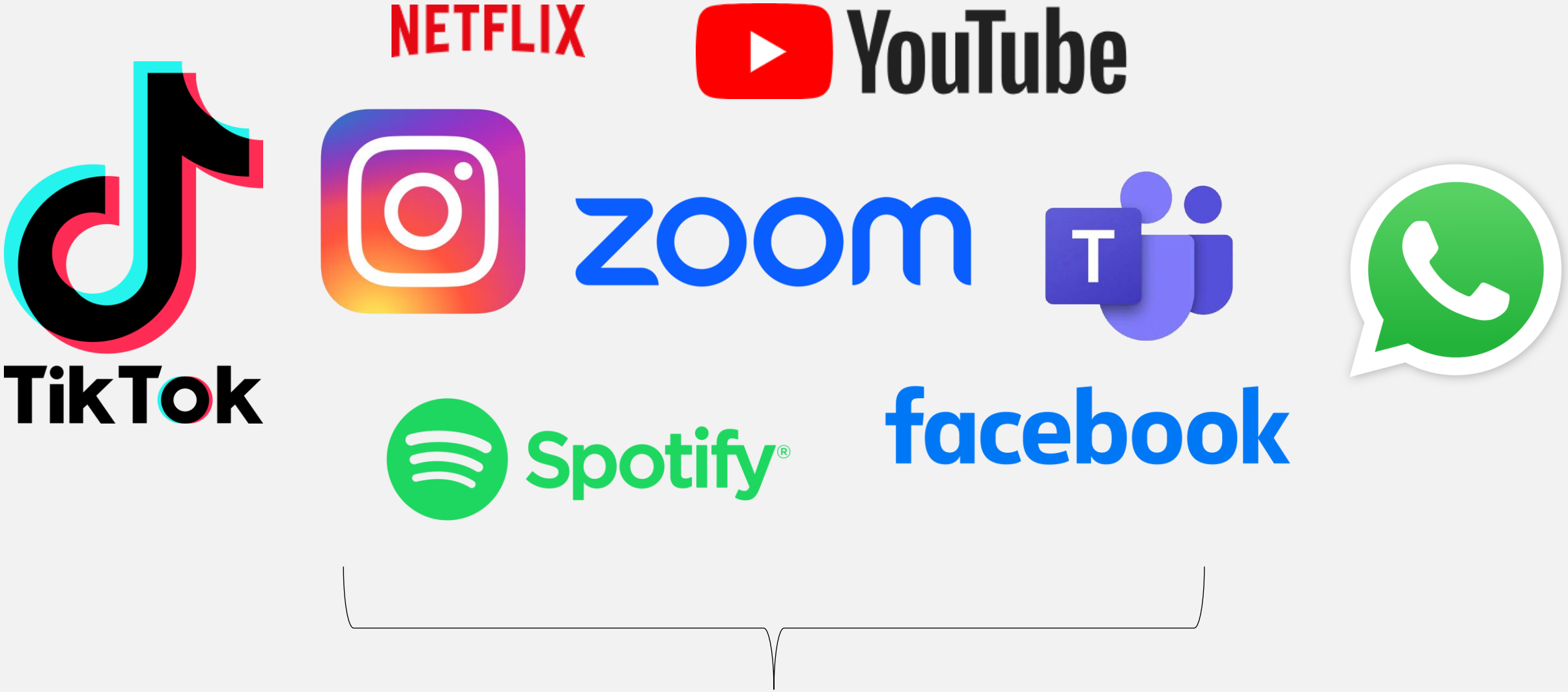
Build your own flight tracker with Semantic Kernel ✈️

Arafat Tehsin

Senior Manager – Platform & Digital Engineering – EY

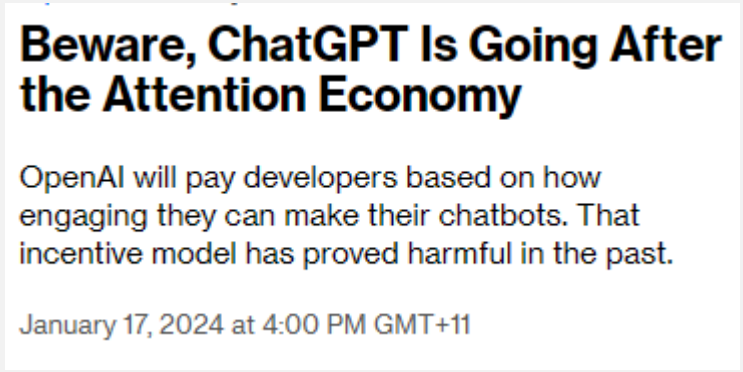


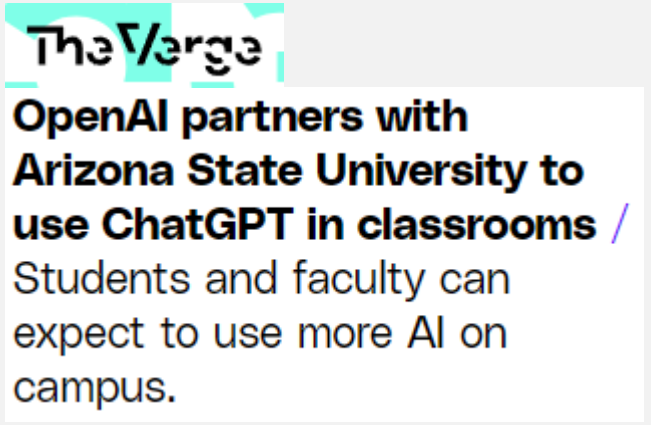
Infused with Artificial Intelligence

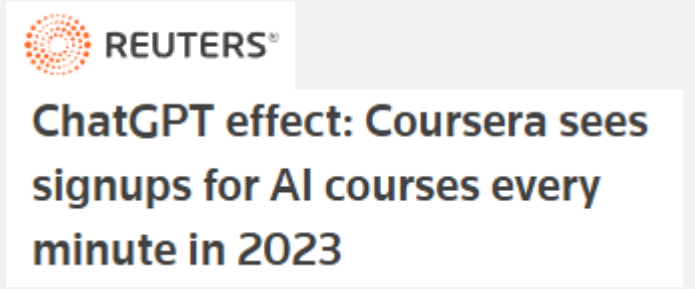
AI First Apps

- Artificial Intelligence as their core technology and functionality
  - Designed for better user experience
  - Use machine learning algorithms to learn from user behaviors over time
  - Automate repetitive tasks, allowing humans to focus on important things
  - Help users make informed decisions by providing insights and predictions
  - Overall experience

ChatGPT revolutionised the internet





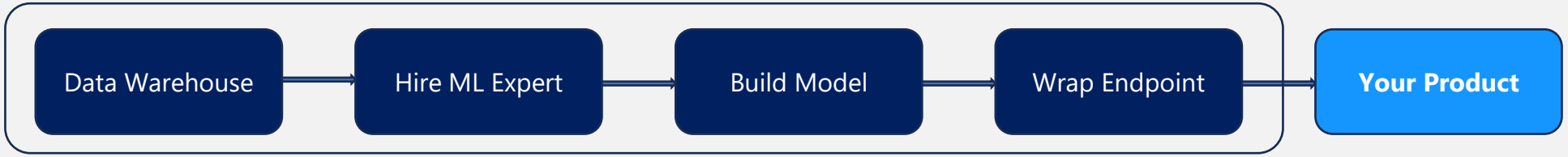


Unprecedented growth

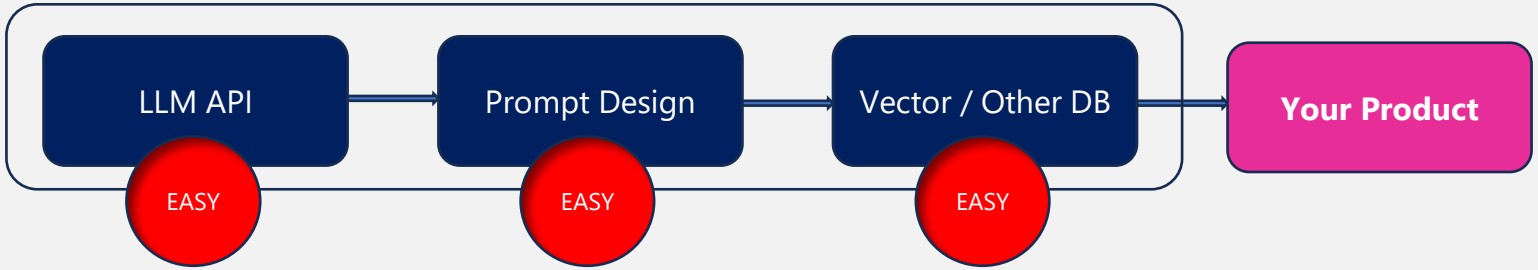
Time to reach 100M users

AI Then and Now 🤔

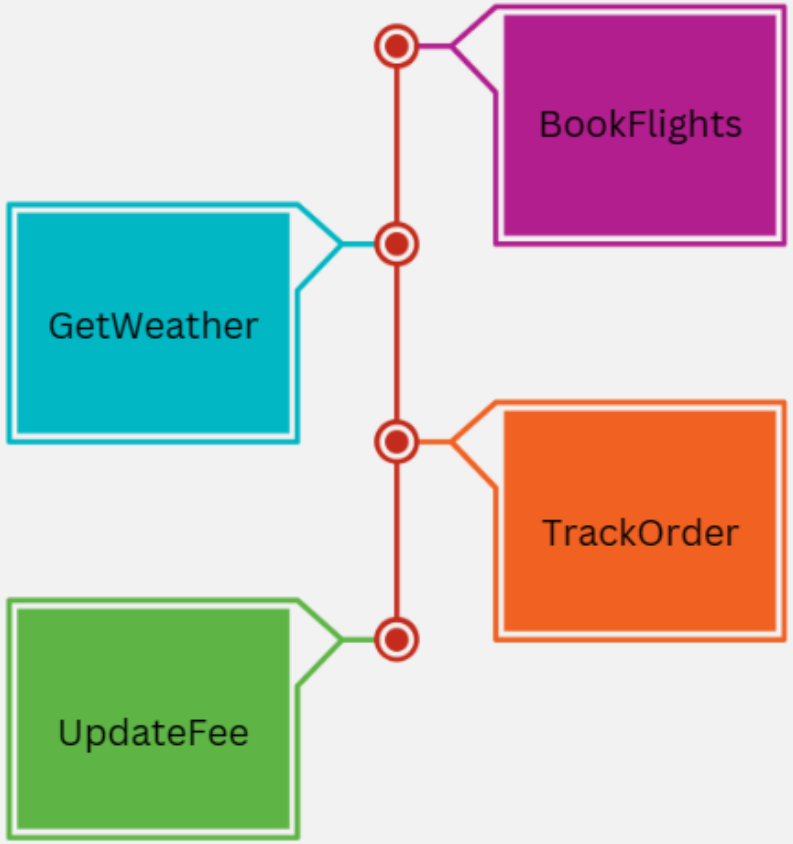
🧠 Traditional Machine Learning / AI Development



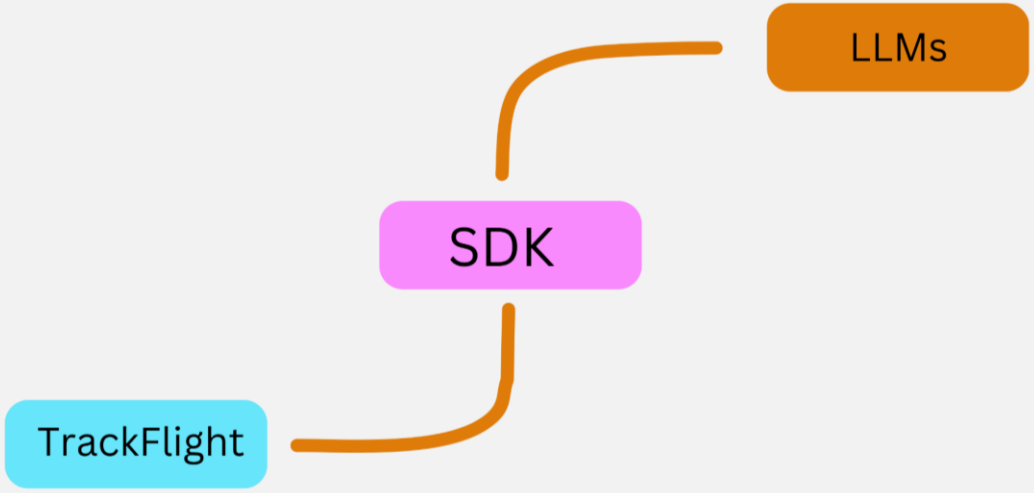
💭 Cognitive and Contextual AI Development



How do I use AI as a core of my app?

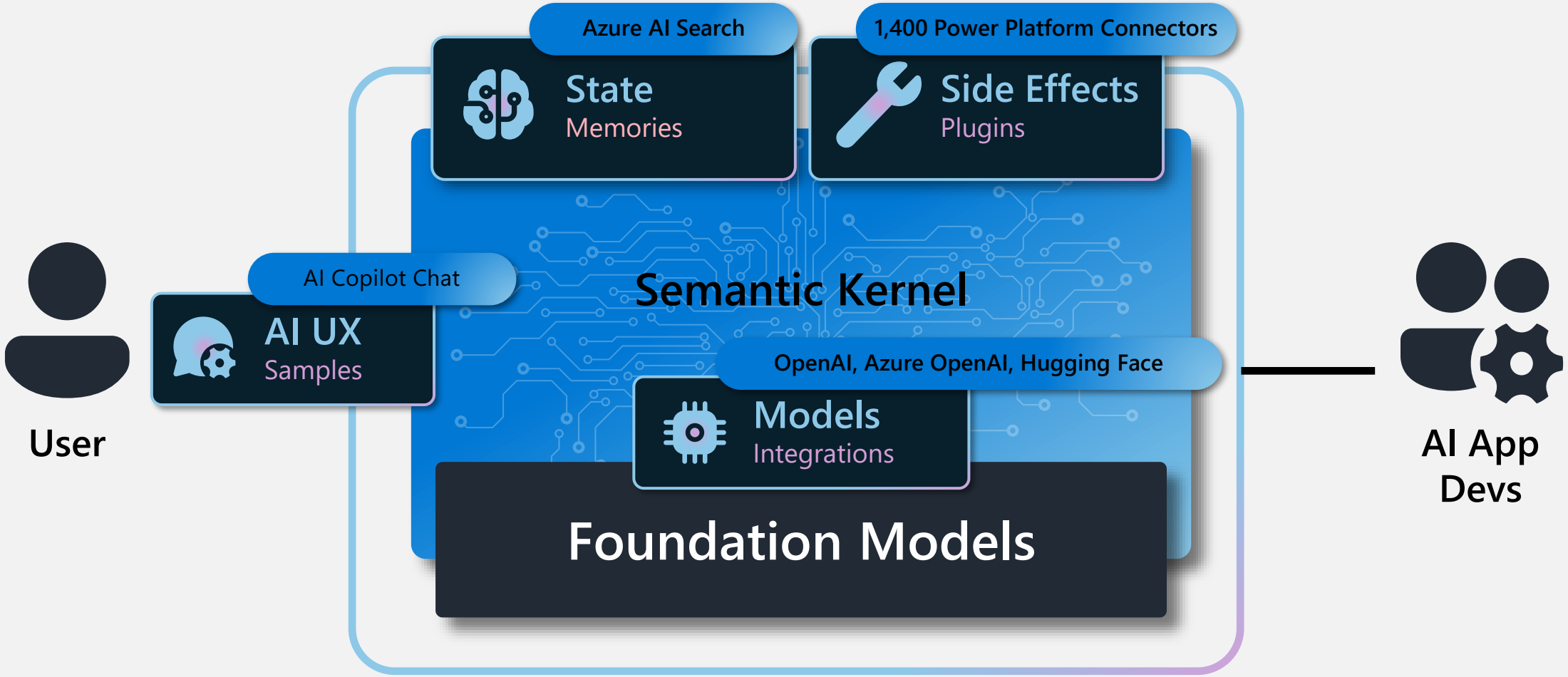


Own API Wrappers / Orchestration Layer

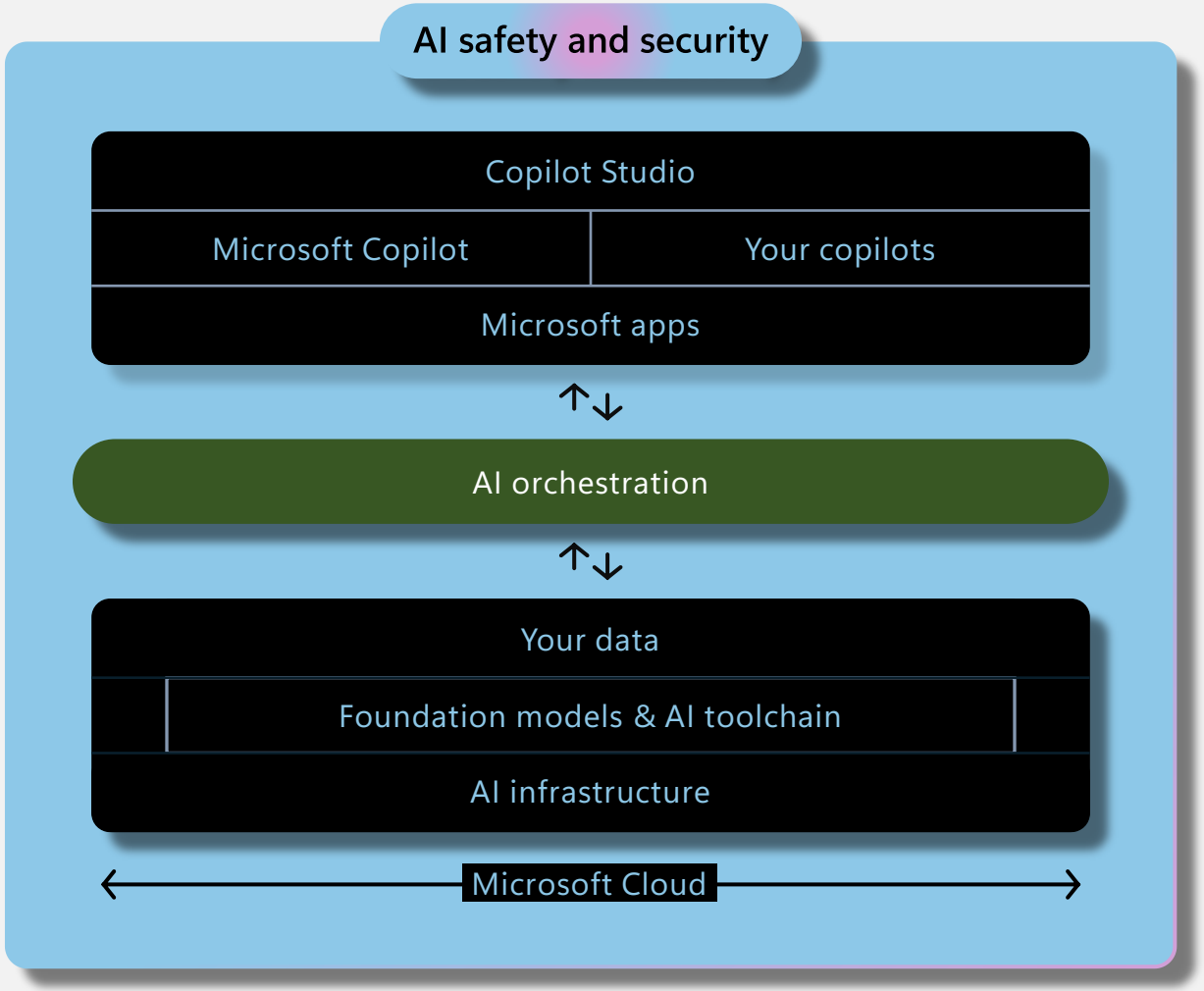


SDK

Semantic Kernel



Copilot Stack



AI Agents

- AI that can solve problems with you. More than just a boring chatbot.
  - Copilot: Special type of agent that works side-by-side with you
    - For example, writing an email or provide suggestions for what to write next
- Agents are comprised of 3 main building blocks
  - Retrieve the information - *Plugin*
  - Plan how to use that information - *Planner*
  - Respond back to user or perform an action - *Persona*

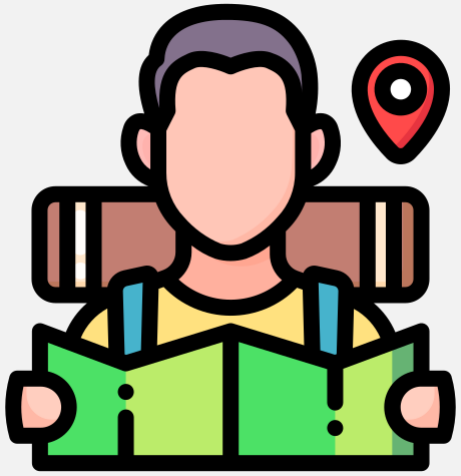
Plugins

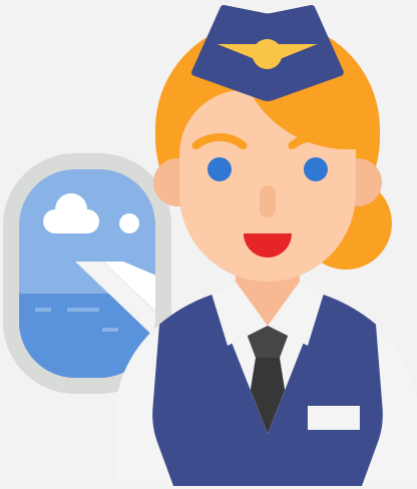
Planners

Personas

Personas

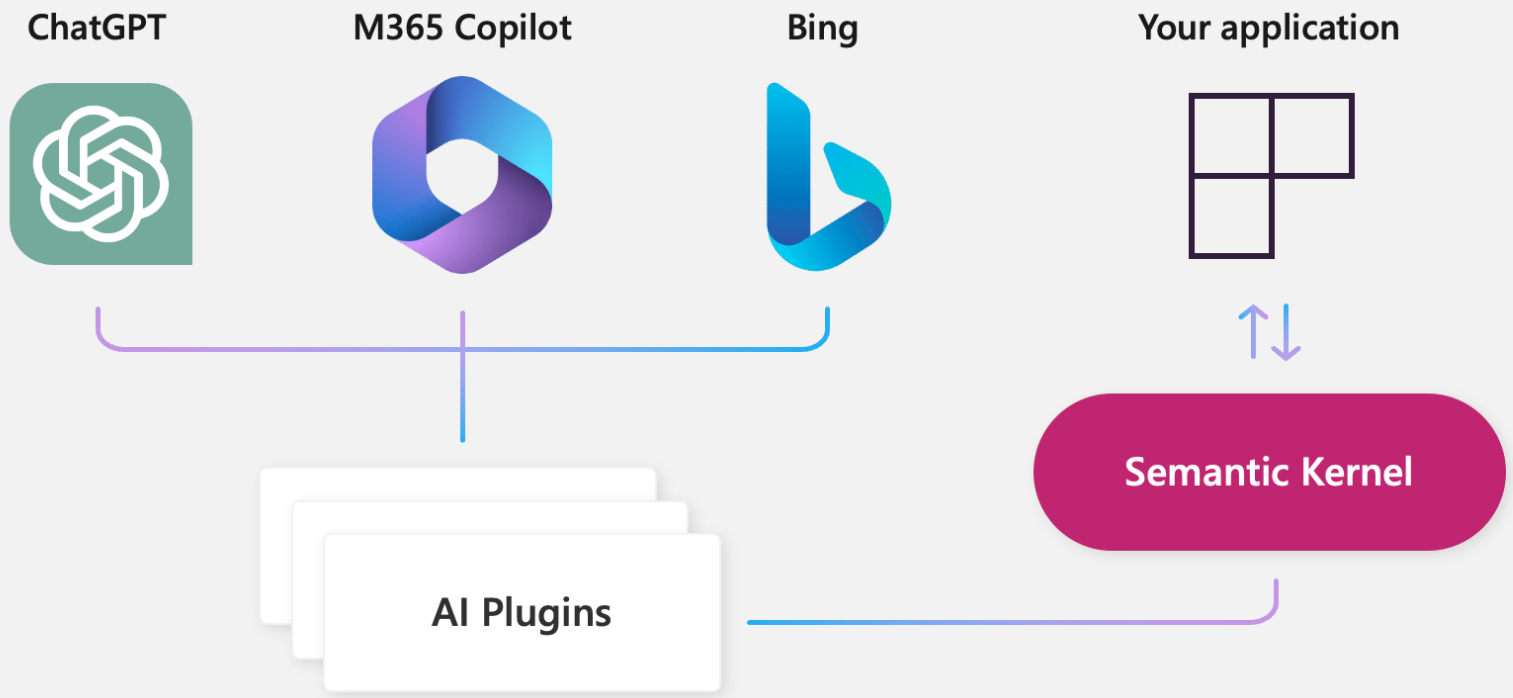
- Give your agent a personality by providing a sufficient metaprompt / instruction
  - *Is it friendly? Is it helpful? Is it sarcastic?*
- Helps us remain within our scope as well
  - For example, a simple flight tracker wouldn't give you the beach suggestions





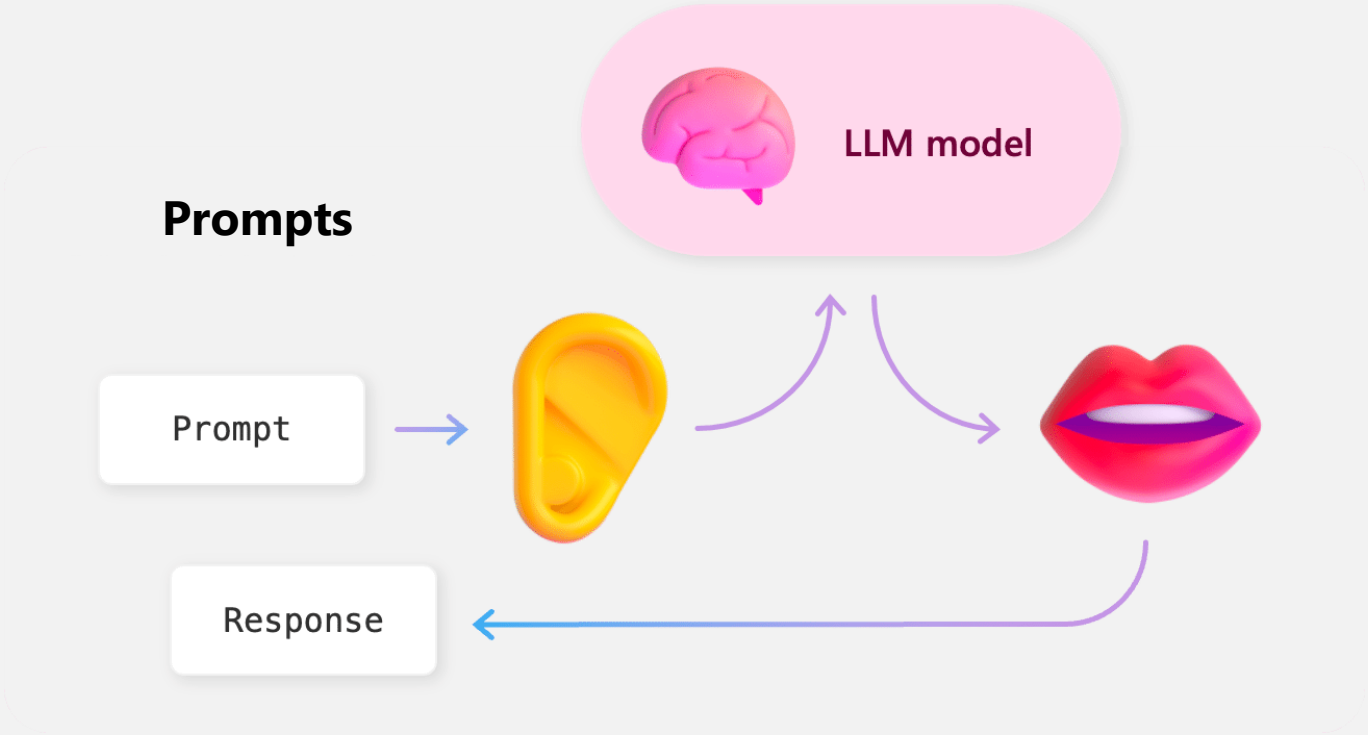
Plugins

- Group of functions that can be exposed to AI apps and services
  - Set of instructions or tasks that your app can perform
- Flight (Plugin) -> TrackFlight (Functions), BookFlight (Functions)
- Semantic description is utmost important while creating functions



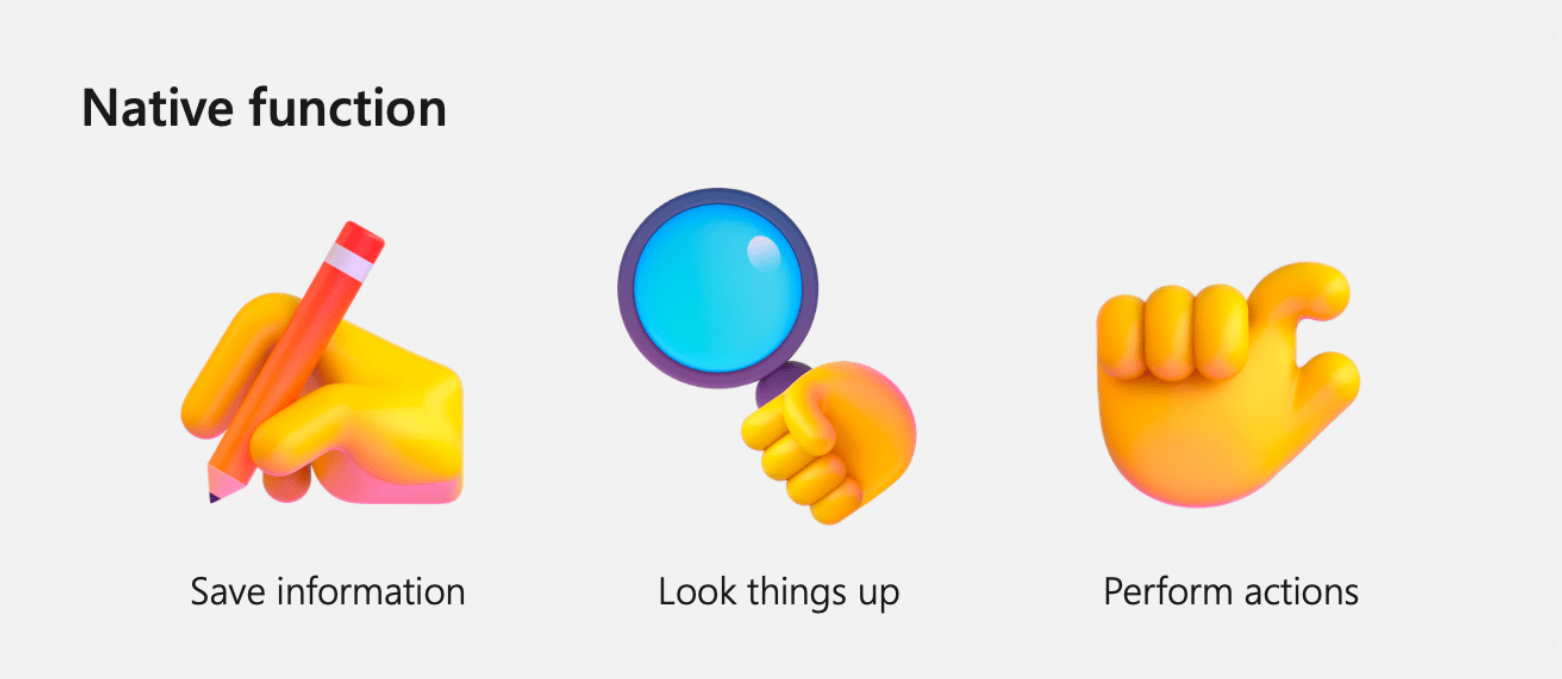
Prompts

- Individual steps that tells your app what to do and how to do it
- For example, if you're building a Weather Copilot
  - Get Current Weather from API
  - Take that current weather and format it into user-friendly message



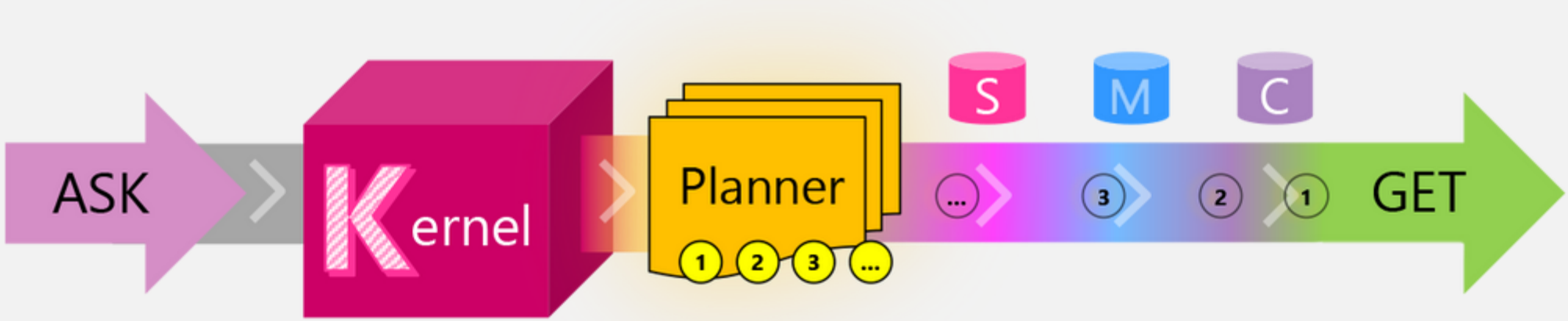
Native Function

- Tools that you use to when following a recipe. Such as a blender or a knife
- For example, if you’re building a Food Copilot
  - You may need to mix the ingredients together (combine data) hence you need a blender (Native Function)
  - Returns a cake batter (the result of the operation)



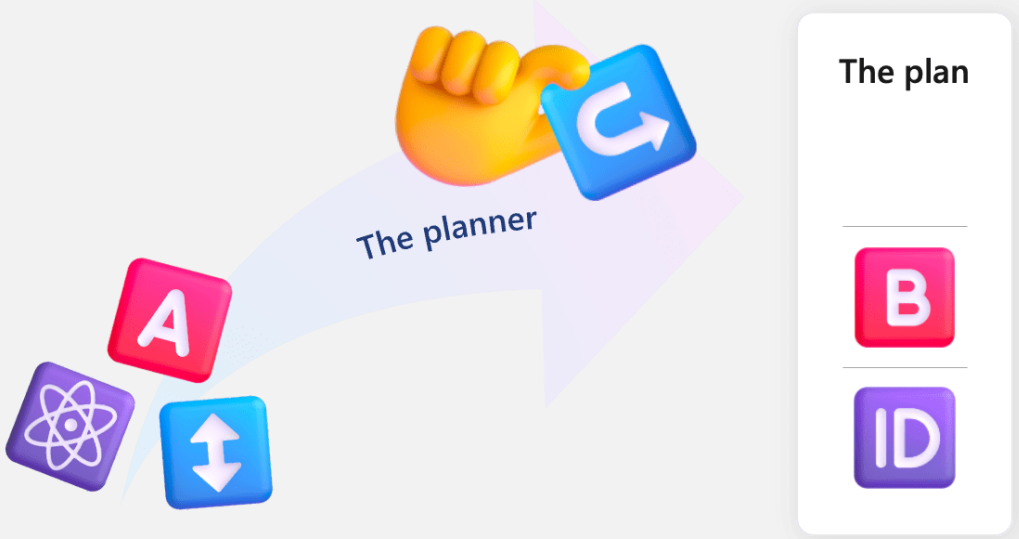
Planners

- Planners are a set of tools that allow developers to automatically orchestrate Prompts and Native functions
- Use AI to mix-and-match the plugins registered in the kernel to complete a goal
- It saves time and cost by optimising the number of calls to LLMs



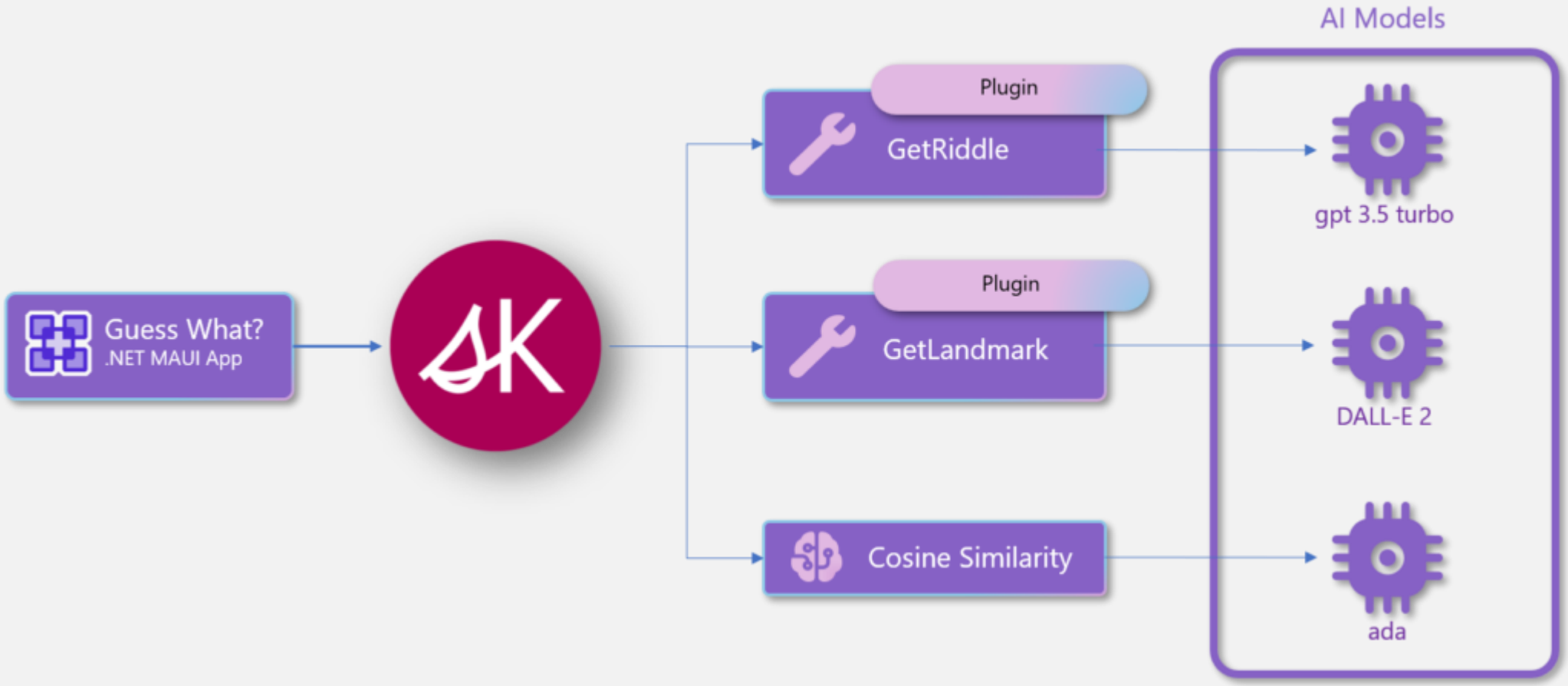
Gen-4 and Gen-5 Planners

- **Handlebar Planners**
  - Planner that uses Handlebars syntax to generate a plan
  - Good support of loops and conditional statements
- **Function calling stepwise planners**
  - Use ReAct methodology
    - Ability to make a function call, reason over it and then make another function call
  - More robust than a vanilla function call by OpenAI



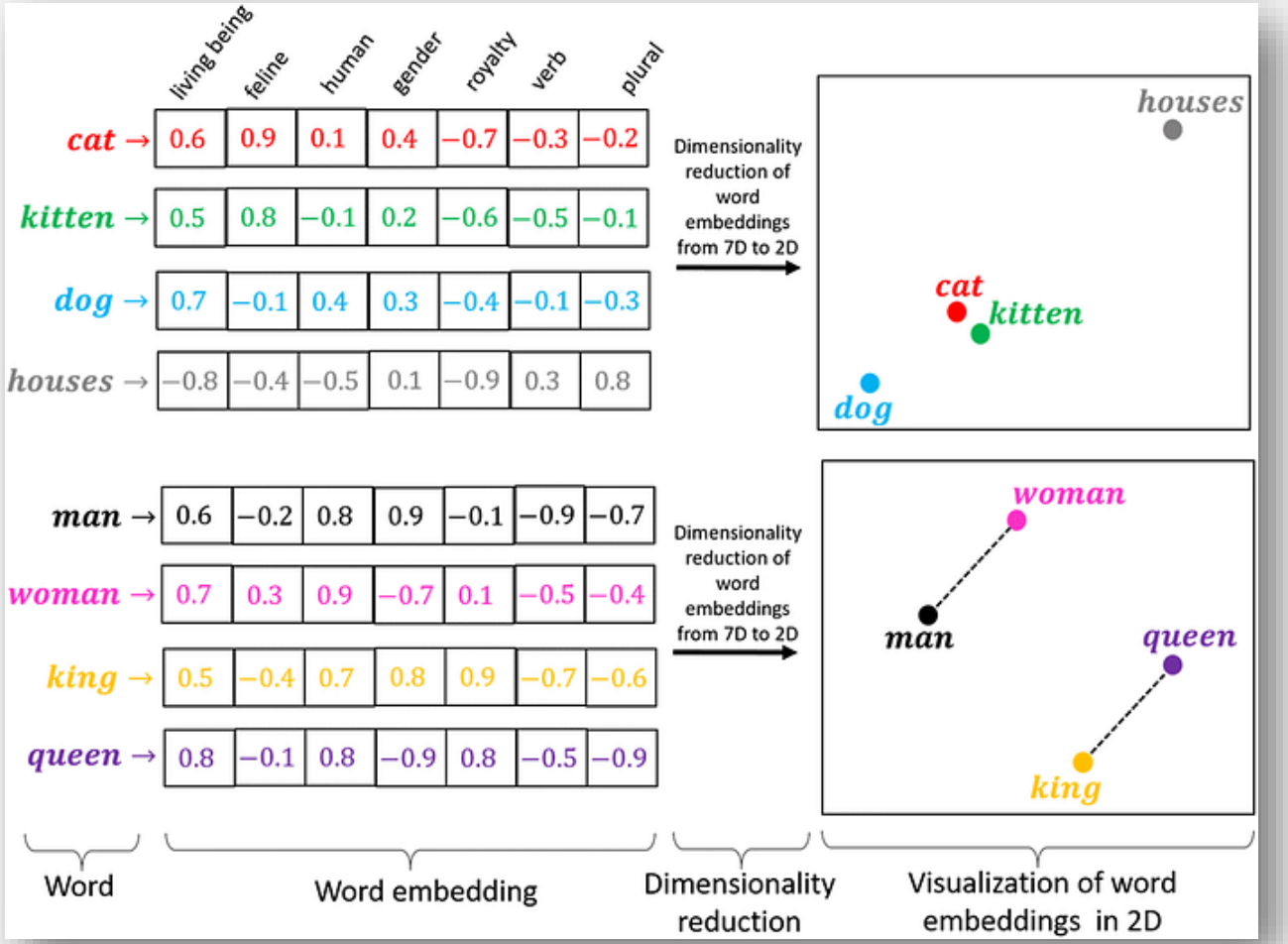
Introducing GuessWhat?

- A cross platform app built using .NET MAUI, powered by Semantic Kernel



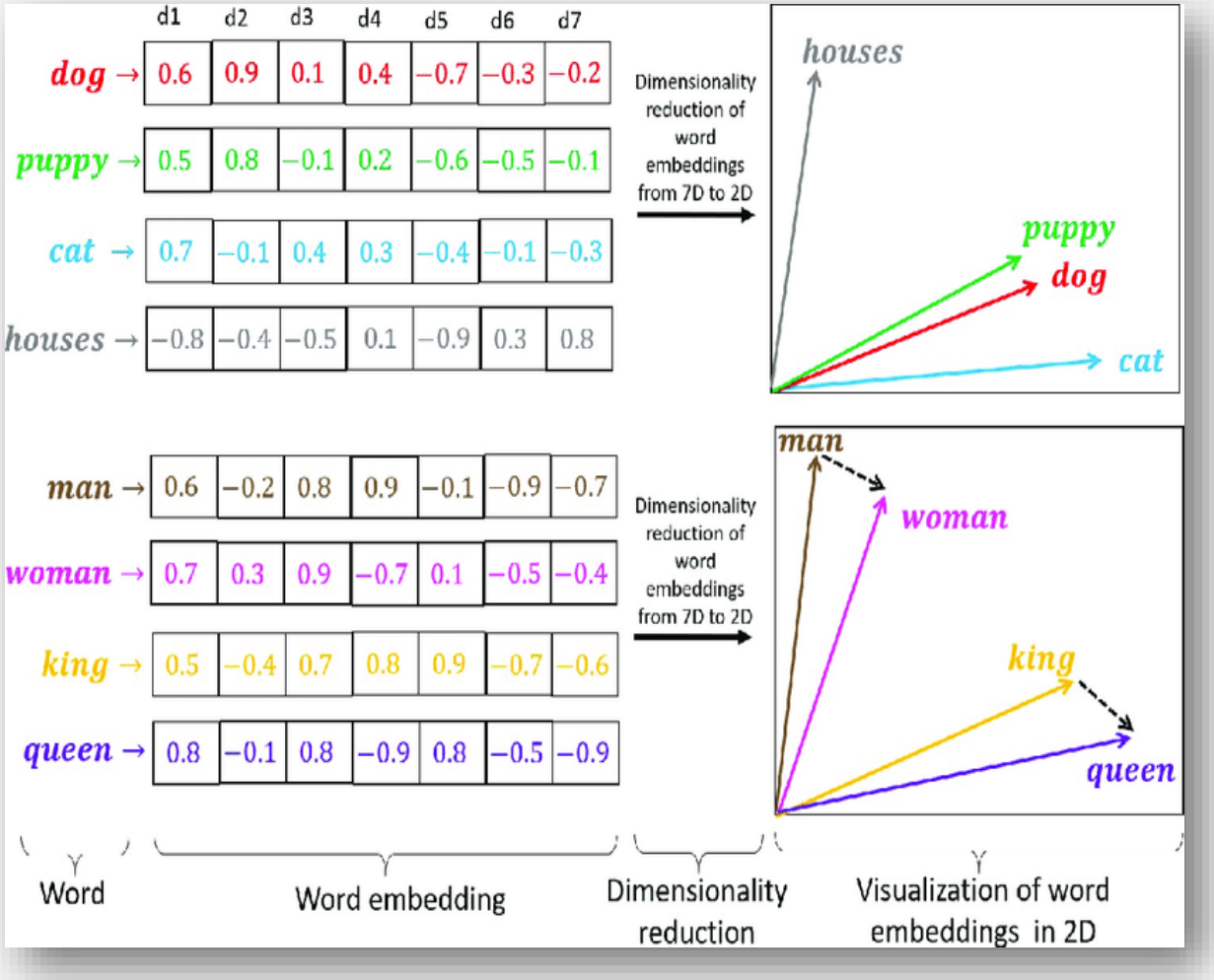
Embeddings

- Computers only understand numbers
- Words are stored in high-dimensional vectors
- For example, “cat” and “kitten” may be closer in the vector space



Cosine Similarity

- Comparing two things but in a different way
- Cosine similarity is calculated by taking the cosine of the angle between two vectors
- If they're in the same direction then it's 1 otherwise -1. If it is in 90° then it's 0



Questions

arafat@outlook.com
linkedin.com/in/arafattehsin
x.com/arafattehsin

arafattehsin.com/blog