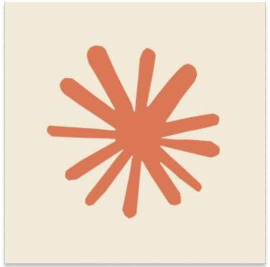
docker

Building AI workflows: from local experiments to serving users

Oleg Šelajev, Docker



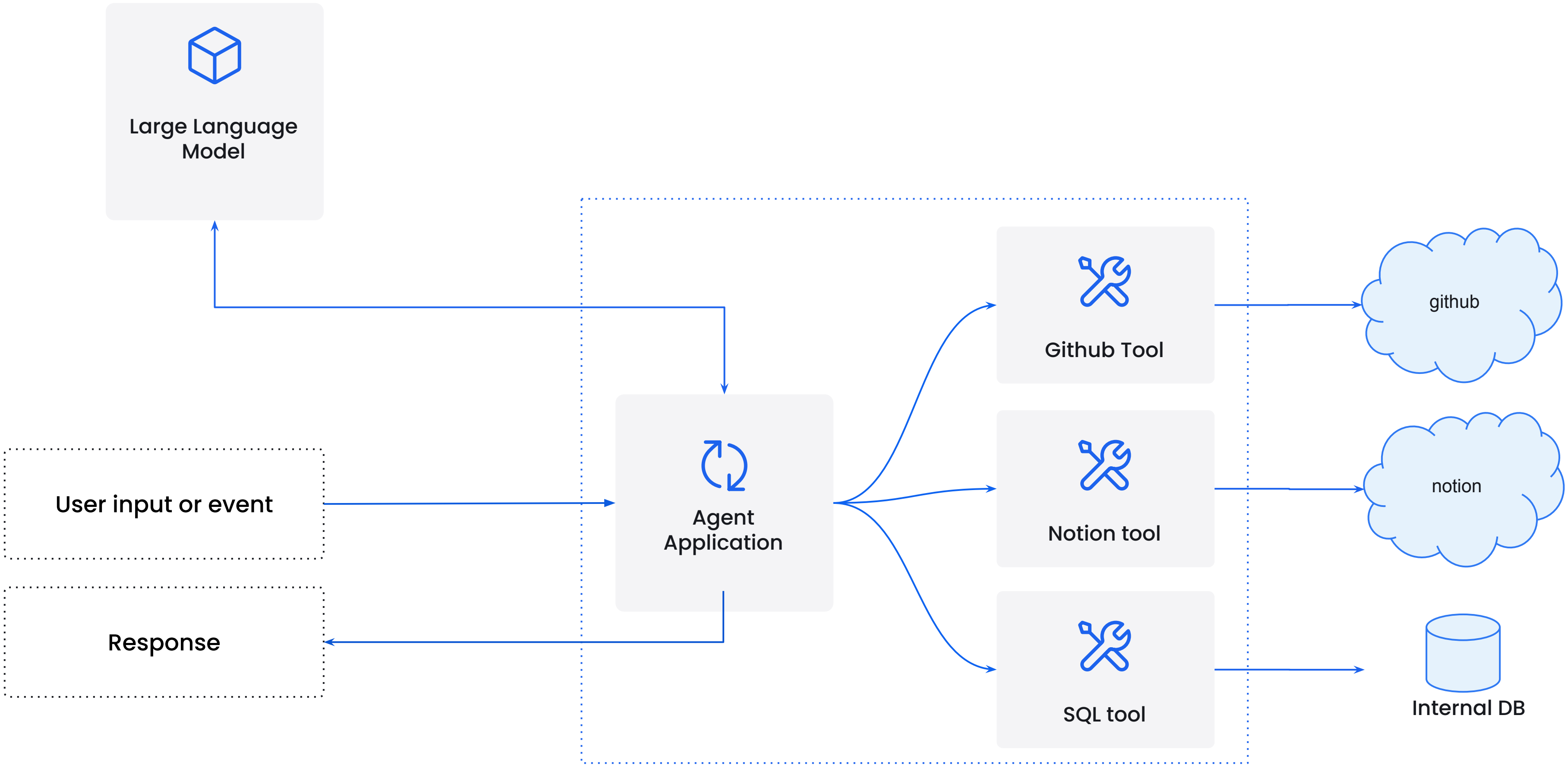
OI

Agentic applications need three things

Models

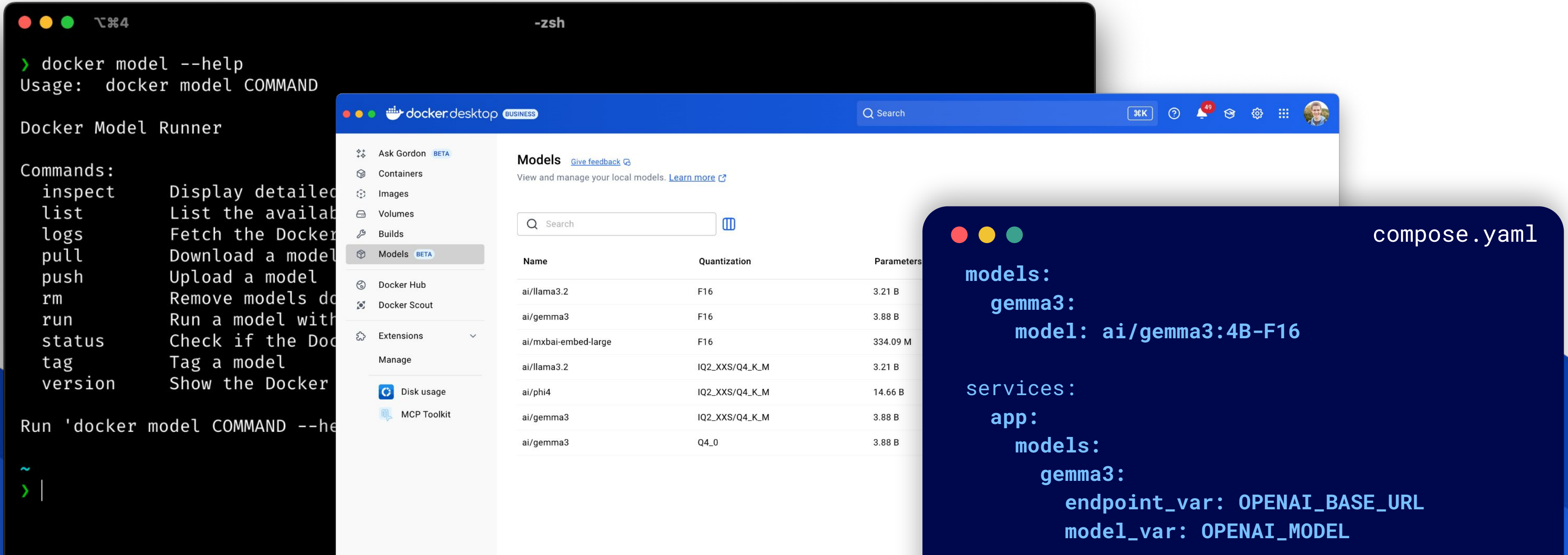
Tools

Code


Large Language Model
User input or event
Response
Agent Application
Github Tool
Notion tool
SQL tool
github
notion
Internal DB

The Docker Model Runner

Run models next to your other containerized services using the tools you're already using



```
❯ docker model --help
Usage:  docker model COMMAND

Docker Model Runner

Commands:
  inspect     Display detailed
  list        List the availab
  logs        Fetch the Docker
  pull        Download a model
  push        Upload a model
  rm          Remove models do
  run         Run a model with
  status      Check if the Doc
  tag         Tag a model
  version     Show the Docker

Run 'docker model COMMAND --he

~
❯ |
```

Models

View and manage your local models. Learn more

Name	Quantization	Parameters
ai/llama3.2	F16	3.21 B
ai/gemma3	F16	3.88 B
ai/mxbai-embed-large	F16	334.09 M
ai/llama3.2	IQ2_XXS/Q4_K_M	3.21 B
ai/phi4	IQ2_XXS/Q4_K_M	14.66 B
ai/gemma3	IQ2_XXS/Q4_K_M	3.88 B
ai/gemma3	Q4_0	3.88 B

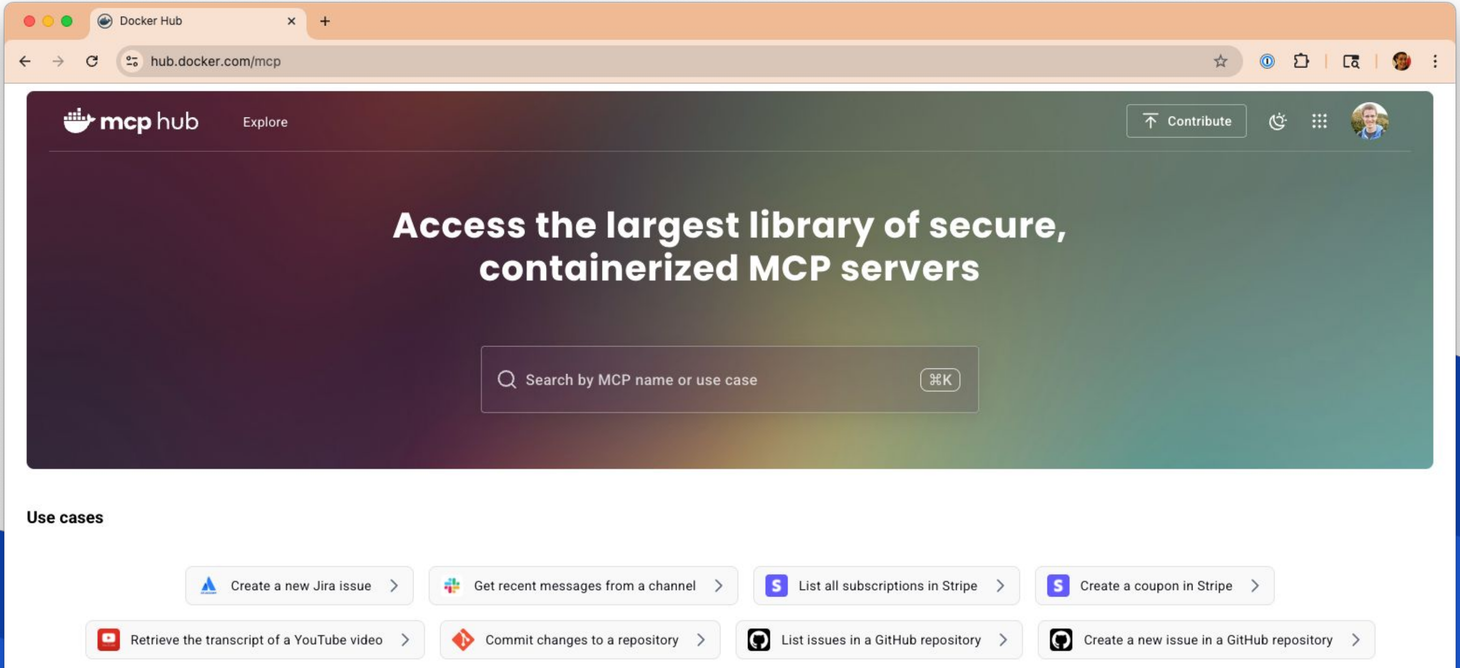
compose.yaml

```
models:
  gemma3:
    model: ai/gemma3:4B-F16

services:
  app:
    models:
      gemma3:
        endpoint_var: OPENAI_BASE_URL
        model_var: OPENAI_MODEL
```

The MCP Catalog

Run MCP servers using containers without worrying about runtimes or installs anymore



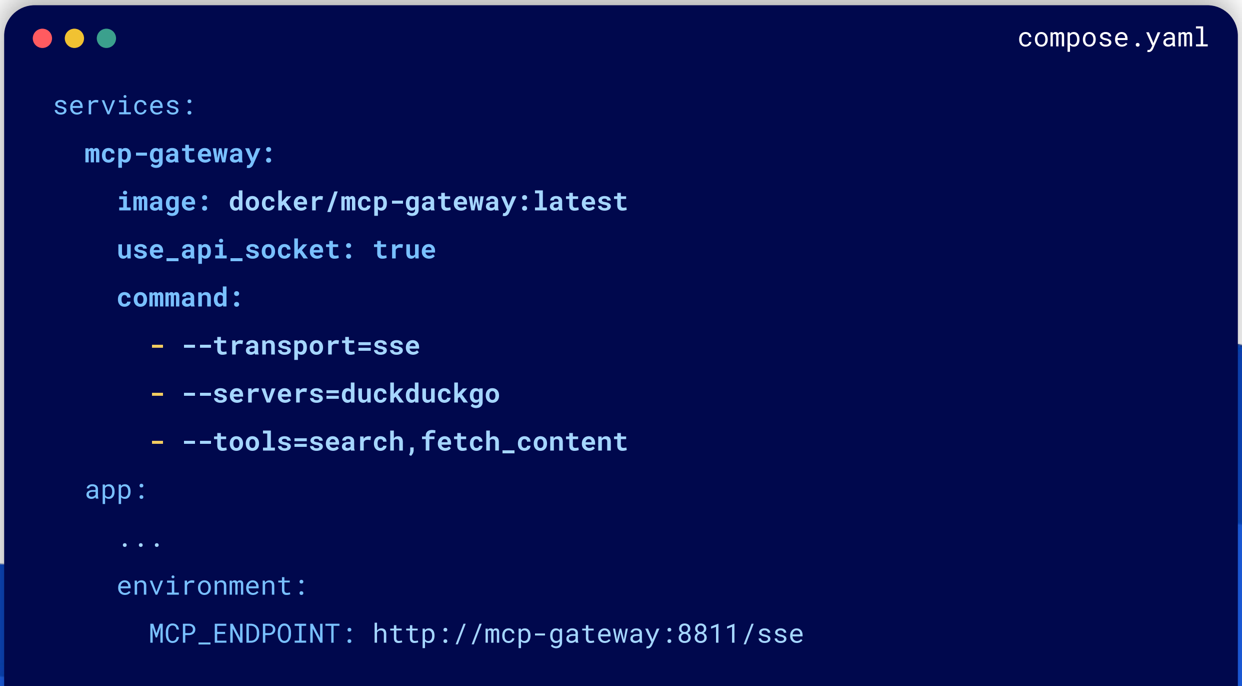
Access the largest library of secure, containerized MCP servers

Use cases

- Create a new Jira issue
- Get recent messages from a channel
- List all subscriptions in Stripe
- Create a coupon in Stripe
- Retrieve the transcript of a YouTube video
- Commit changes to a repository
- List issues in a GitHub repository
- Create a new issue in a GitHub repository

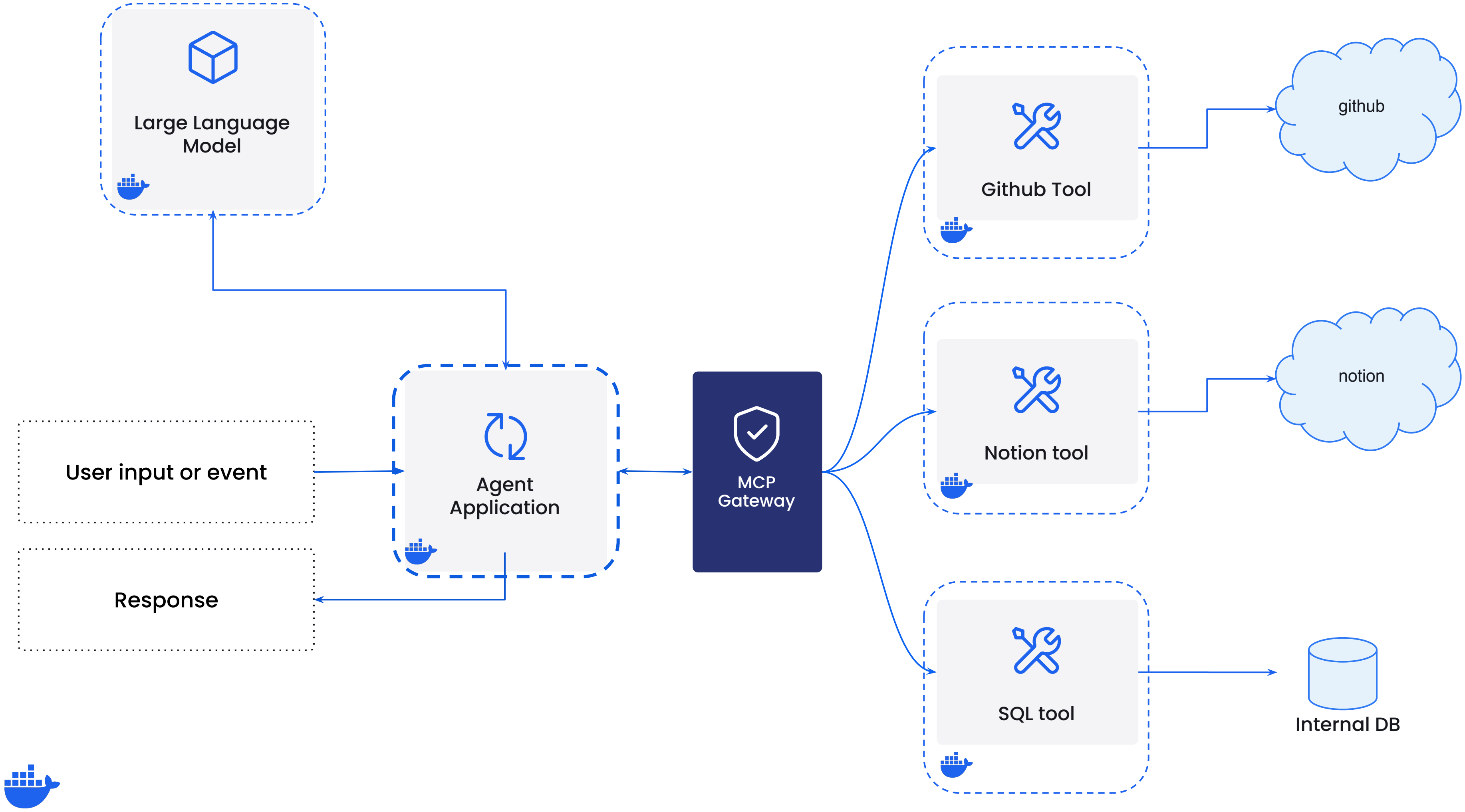
The MCP Gateway

Run containerized MCP servers safely and securely directly in your application stack



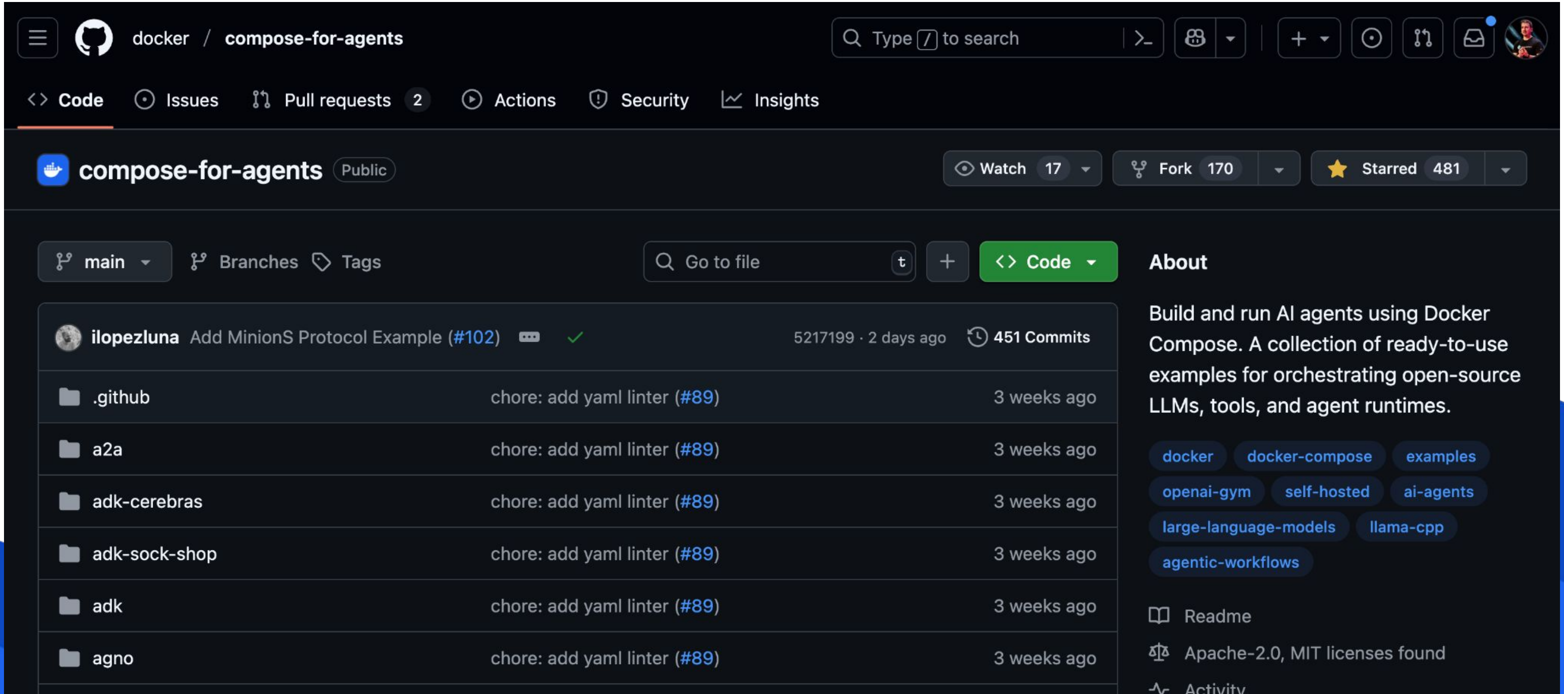
compose.yaml

```yaml
services:
  mcp-gateway:
    image: docker/mcp-gateway:latest
    use_api_socket: true
    command:
      - --transport=sse
      - --servers=duckduckgo
      - --tools=search,fetch_content
  app:
    ...
    environment:
      MCP_ENDPOINT: http://mcp-gateway:8811/sse
```


Large Language Model
User input or event
Response
Agent Application
MCP Gateway
Github Tool
Notion tool
SQL tool
github
notion
Internal DB

Compose for agents

Build and run AI agents using Docker Compose



docker / compose-for-agents

compose-for-agents Public

ilopezluna Add MinionS Protocol Example (#102) 5217199 · 2 days ago 451 Commits

.github	chore: add yaml linter (#89)	3 weeks ago
a2a	chore: add yaml linter (#89)	3 weeks ago
adk-cerebras	chore: add yaml linter (#89)	3 weeks ago
adk-sock-shop	chore: add yaml linter (#89)	3 weeks ago
adk	chore: add yaml linter (#89)	3 weeks ago
agno	chore: add yaml linter (#89)	3 weeks ago

About

Build and run AI agents using Docker Compose. A collection of ready-to-use examples for orchestrating open-source LLMs, tools, and agent runtimes.

docker docker-compose examples openai-gym self-hosted ai-agents large-language-models llama-cpp agentic-workflows

Readme

Apache-2.0, MIT licenses found

Activity

Cloud Run and Docker Compose

Deploy your compose.yaml directly to Cloud Run

cloud.google.com/blog/products/serverless/cloud-run-and-docker-collaboration