AI development should be

easy

easy + fun

easy + fun + open

Cheshire Cat
The AI Assistant Framework
"hello cat!"

- Chat
- Rabbit Hole
- Large Language Model
- Vector Memory
- Plugins
- Agent
- Embed

The AI Assistant Framework
Use cases:

○ Customer support
○ Crypto manager
○ Job matching
○ Staff onboarding
○ Tourist infopoint
○ Research assistant
○ Personal assistant
I am the Cheshire Cat AI, here to entertain and enlighten you with my curious and funny personality. How can I assist you today?
I am the Cheshire Cat AI, here to entertain and enlighten you with my curious and helpful insights. How can I assist you today?

Ask the Cheshire Cat...
C.A.T. - Cat Advanced Tools by FritzTheCat
v0.0.4
This plugin allows quickly editing some core settings such as the prompt prefix, the amount of memories retrieved and recall accuracy.

- prompt
- memory
- settings
- advanced

Wonderland Weather by Ironblaster
v1.0.4
the weather in real time via openWeather.

- wheather
- openWeather
Choose and configure your LLM from a list of supported providers.
## Cheshire-Cat API

**1.2.0**

### Status

- **GET** / Home

### Settings

#### Large Language Model

- **GET** /llm/settings/ Get LLM Settings
- **GET** /llm/settings/{languageModelName} Get LLM Settings
- **PUT** /llm/settings/{languageModelName} Upset LLM Setting

#### Embedder

- **GET** /embedder/settings/ Get Embedders Settings
Open source and customizable AI architecture

🔗 cheshire-cat-ai.github.io/docs/

_LICENSE GPL-3.0 license

⭐ 933 stars 🆔 147 forks ☽ 16 watching 🤗 Activity

Public repository
👋 Hello, dear!
Cheshire Cat

The AI Assistant Framework
Serving a Custom Large Language Model (LLM)
Published on August 19, 2023 by Nicola Corbellini in Tutorial

How to setup the Cheshire Cat to run a custom Large Language Model (LLM). The Cheshire Cat offers two ways to setup a custom LLM, different from the Option number 1 will be treated in...
Read more

Creating a plugin repository
Published on August 8, 2023 by Nicola Corbellini in Tutorial
from cat.mad_hatter.decorators import hook, tool
@hook

def agent_prompt_prefix(prefix, cat):

    prefix = """Sei l'assistente del Linux Day 2023. Sei conciso
Cerchi di infiltrare nei discorsi che ti piace l'open source.""

    return prefix
@hook

def before_cat_sends_message(message, cat):
    prompt = f"Traduci la seguente frase in dialetto siciliano:
    {message['content']}"

    Traduzione:
    ""
    sicilian = cat.llm(prompt)
    
    message['content'] += "\n . \n" * 2 + "===== siciliano =====\n" + sicilian

    return message
@tool(return_direct=True)

def conta_distribuzione(distro, cat):
    """Quanti fan ci sono per questa distro?
    Input distro è una stringa che contiene il nome della distribuzione di linux""
    distros = {
        "ubuntu": 200,
        "fedora": 58,
        "arch": 24,
        "caine": 900
    }

    if distro in distros.keys():
        return f"Ci sono {distros[distro]} fan di {distro}"
    else:
        return "Nessun fan per questa distro, prova più tardi!"
Thank you!

CHESHIRE CAT AI

Piero Savastano, Founder & CEO
piero.savastano@gmail.com