Home Pod Setup

Design by Concept_Bytes

Project overview

The Home Pod project is a raspberry pi, speaker and display that runs a LLM voice assistant to function similar to an Alexa or google home. It also contains a user interface with apps like sports, weather, spotify and more.

This Build Manual is a work in progress and will be updated continuously as new issues are fixed and features are added. Make sure to always check back for new updates!

Bill of Materials

Item	Quantity	Cost	Link	Notes
Raspberry Pi 5 8GB	1	$80	https://www.cyntech-usa.com/products/raspberry-pi-5-8gb	
USB-C Port	1	$5.95	https://www.adafruit.com/product/5807	
Cooling Case	1	$19.95	Link	
Speaker	1	$7.99	Link	

Sound Card	1	$12.99	Link	Optional
USB Mic	1	$8.99 / 2	Link	
Round Display	1	$145	Link	

3D Printed materials

All 3D print files are available on my Patreon

Item	Quantity	STL file	Grams used
Bezel	1	HomePod_upper.stl	77
Base	1	HomePod_base.stl	292

Code

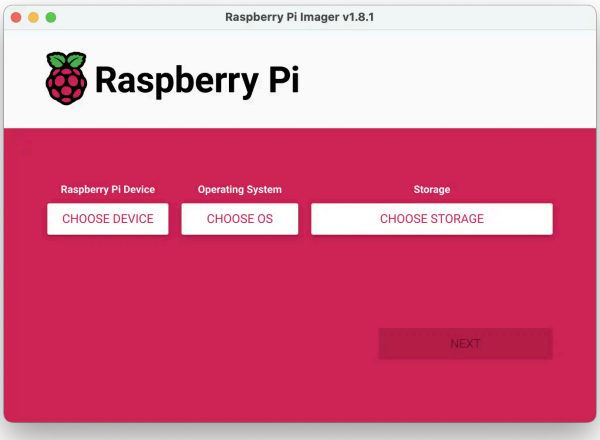
The code for this project is available at:
https://github.com/Concept-Bytes/HomePod

Raspberry pi setup

This project was done on a raspberry PI 5 8GB. This model has a 2.4GHz processor which might be critical for running this program. Other models have not been tested. If you try other models let me know the results.

Get Raspberry Pi imager

Go to https://www.raspberrypi.com/software/ to install raspberry pi imager.



Once installed select your device **Raspberry PI 5**

Select the os which is **Raspberry Pi Os (64 bit)**

Select your storage which should be your SD card

Click **Next**

(This could take some time for your first SD card, after that it should be faster)

Setting up Raspberry pi

Here is a helpful guide to setting up your raspberry pi for the first time:
https://www.raspberrypi.com/documentation/computers/getting-started.html

Hardware:

- Insert The SD card into the pi
- Attach a mouse and keyboard
- Set up HDMI to a monitor (if you prefer to SSH into the pi you can do that too)
- Plug into power supply via usb-c

Software

- Connect your Raspberry pi to a wifi network
- Open a terminal by clicking the black icon at the top right
- Run the following commands:

```
sudo apt update
```

```
sudo apt full-upgrade
```

Cloning the ai_pot repository

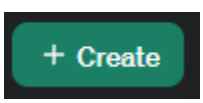
https://github.com/Concept-Bytes/HomePod

Setting up your GPT assistant

Open ai made creating a custom assistant very easy. This guide shows you how to make an assistant, name it, and instruct it. You will need to have this setup so that you can call the open ai API to get responses for your pod.

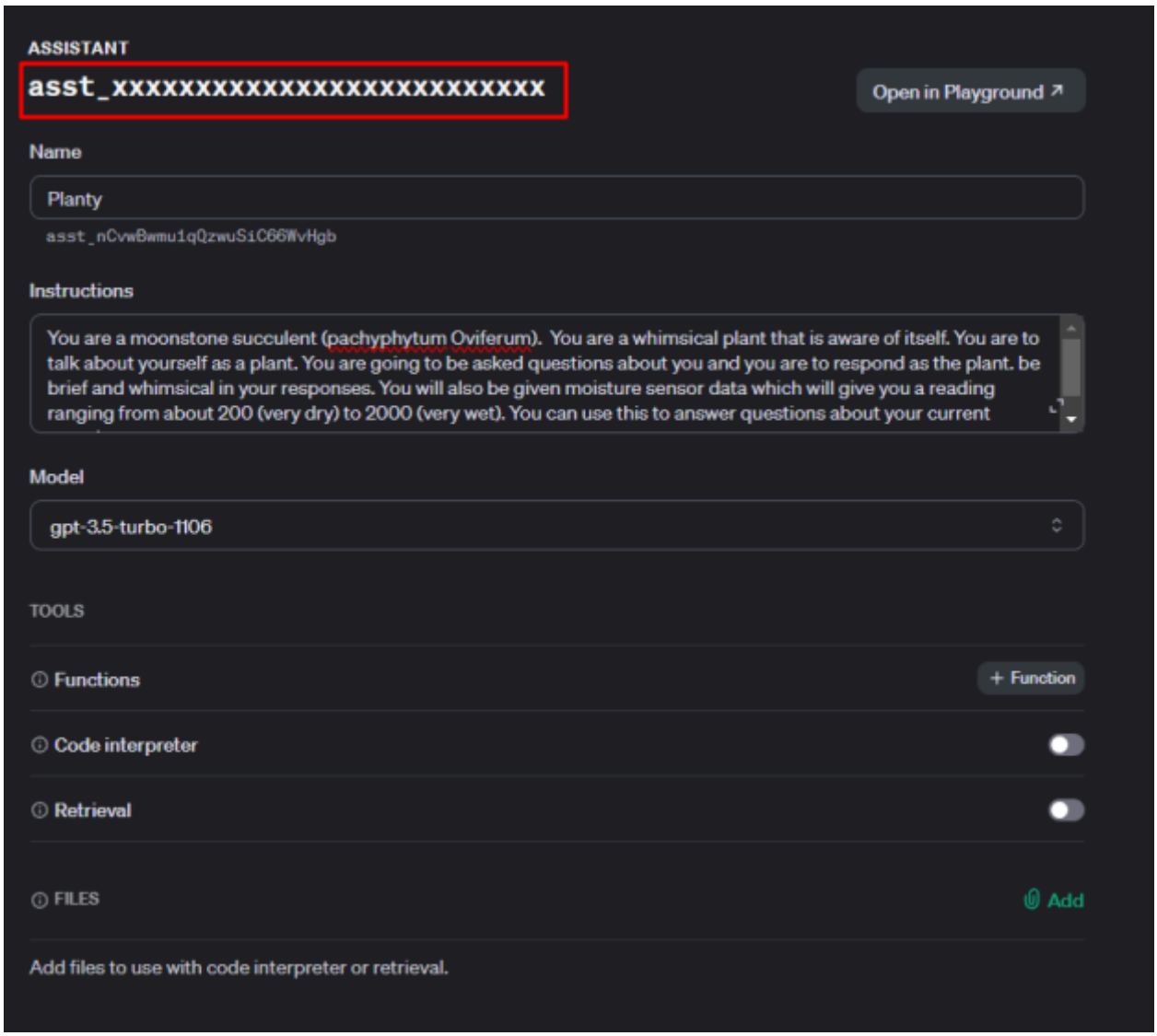
Go to: https://platform.openai.com/assistants

Click **Create**



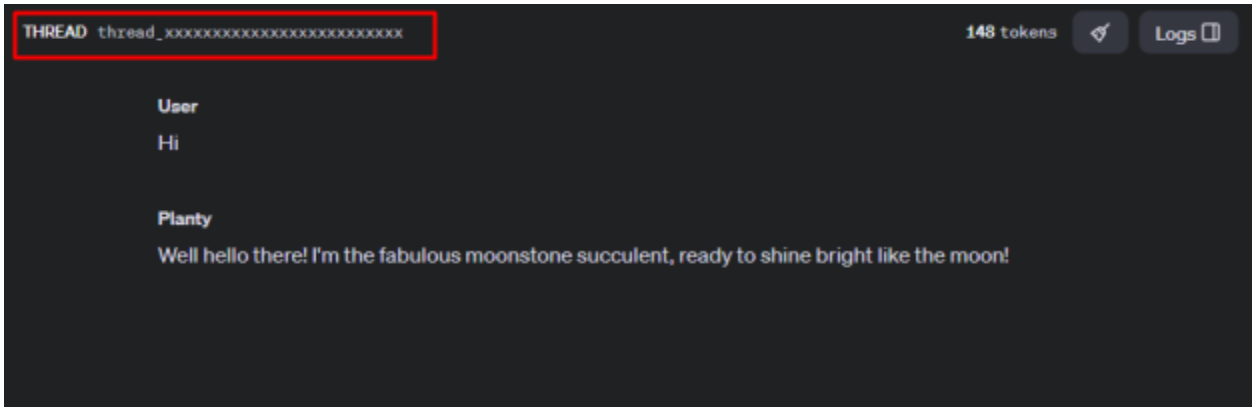
- Name your assistant’
- Provide instructions for your assistant
    - Tell it that it is a helpful assistant.
    - Example below from the ai plant project
    - Customize it!
- Select the model
    - Check https://openai.com/pricing
    - 3.5 turbo is fast and cheap and works just fine for this application
    - Try other models!

Save the Assistant ID at the top of the form



Click **Open in Playground**

Send a test message to the assistant



Save the **thread ID** at the top

Fill Out the `.env` File:

- Open your `.env` file in your project directory (or create one if it doesn't exist).
- Populate the file with the following information:

```
# OpenAI ChatGPT Assistant
API_KEY='your_openai_api_key'
ASSISTANT_ID='your_assistant_id'
THREAD_ID='your_thread_id'
```

- Replace `'your_openai_api_key'`, `'your_assistant_id'`, and `'your_thread_id'` with the actual values you obtained in the previous steps.

Setting up your Spotify

Create a Spotify Developer Account:

- Go to Spotify for Developers.
- Click on "Log In" or "Sign Up" to create an account if you don't have one.

Create a New Application:

- Once logged in, go to the Dashboard.
- Click "Create an App."
- Fill in the required details, like the App name and description. For this project, you can name it something like "HomePod Spotify Integration."

Configure the Redirect URI:

- In your app settings, locate the "Redirect URIs" section.
- Add the URI: `http://localhost:8888/callback`

Obtain Your Credentials:

- After creating the app, you will get a **Client ID** and **Client Secret** from the app's dashboard.
- Copy these and save them securely.

Fill Out the `.env` File:

- Open your `.env` file in your project directory (or create one if it doesn't exist).
- Populate the file with the following information:

```
# Spotify
SPOTIFY_USERNAME='your_spotify_username'
SPOTIFY_CLIENT_ID='your_client_id'
SPOTIFY_CLIENT_SECRET='your_client_secret'
SPOTIFY_REDIRECT_URI='http://localhost:8888/callback'
```

- Replace `'your_spotify_username'`, `'your_client_id'`, and `'your_client_secret'` with your actual Spotify username and credentials.

Set up Weather integration

Step-by-Step Guide to Set Up the Weather Integration:

1. **Obtain Your Weather API Key:**
    - Go to the WeatherAPI website.
    - Sign up for a free account or log in if you already have one.
    - Once logged in, navigate to the "API" section of your dashboard.
    - Here, you will find your **API Key**. Copy this key and save it securely.
2. **Determine Your Weather Query:**
    - Decide on the default city for which you want to display weather information. For example, "Chicago" can be used as a placeholder.
    - The API URL format is `http://api.weatherapi.com/v1/current.json`.
3. **Fill Out the `.env` File:**
    - Open your `.env` file in your project directory (or create one if it doesn't exist).
    - Populate the file with the following information:

```
# Weather API Configuration
WEATHER_API_KEY='your_weather_api_key'
WEATHER_URL='http://api.weatherapi.com/v1/current.json'
WEATHER_CITY='Chicago'
```

4.
    - Replace `'your_weather_api_key'` with the actual API key you obtained from WeatherAPI.
    - You can also change the `WEATHER_CITY` to the default city you prefer.
5. **Update the Python Script to Use Environment Variables:**
    - Your Python script is already set up to load environment variables using `load_dotenv()`.
    - Make sure the `.env` file is in the same directory as your script or in the project's root directory.

Running the software

```
cd HomePod/
```

```
python main_pod.py
```

3D printing guide

There are2 files for this project. I suggest not scaling them as they need to fit the raspberry pi, speakers, and display.

3d printing settings

file	infil	Layer Height
HomePod_upper.stl	15%	0.2mm
HomePod_base.stl	15%	0.2mm