SHELLY

1.0.1

User Manual

GREENCODERS www.greencoders.net

How to Create an OpenAI API Account and Assistant for Shelly

Here are the steps to get started with your OpenAI API Account and creating your first Assistant:

1. Navigate to Projects: https://platform.openai.com/settings/organization/projects
2. Create a new project (name it as you wish).
3. Navigate to the Assistants option and click “Create” to set up a new AI Assistant.
4. Enter a “Name” and paste Shelly's System instructions included in the downloaded ZIP file.

   You can also find the System instructions inside the app by navigating to: Settings → AI Training → Load Training. Please ensure the “Response format” is set to “text”.

Supported models for Shelly 1.0.1:

- gpt-4o-mini
- gpt-4o
- gpt-4.1
- o3-mini

We recommend using “gpt-4o-mini” for general tasks and low token costs.

> ***Important**: the selected Assistant model inside the OpenAI platform cannot be changed from inside Shelly Settings. The “AI Model” selection inside Shelly only applies to AI Agents.*
> *For more information, please refer to the Technical Documentation.*

5. Save your new AI Assistant.
6. Copy the unique AI Assistant ID.

Generate a New OpenAI API Key

1. Navigate to the API keys dashboard: https://platform.openai.com/api-keys
2. Click “Create new secret key”.
3. Type a relevant name and select your previously created project.
4. Click Create secret key.

> **Important:** Copy the secret key to a file and save it to a secure location. Once you close the window, the API key will be hidden and cannot be retrieved again for security reasons. If you forget the key, you will need to generate a new one.

That’s it. You should now have an Assistant ID and an OpenAI API key. **Paste both into your Shelly App and restart the app.**

Shelly – User Manual

Introduction

Shelly is a Windows-based application designed to run on the .NET framework (tested with versions 7.0 and 8.0). It leverages the OpenAI API and requires a valid paid OpenAI account to operate.

Shelly does not collect any user data in any form.

Powered by OpenAI models, Shelly can engage in conversational chat, generate and execute PowerShell scripts, and run custom .NET functions directly on the user's machine.

Getting Started

To begin using Shelly, simply click on the application interface. This action opens the Chat Window, the main interaction area.

Chat Window Overview

- **Prompt Field:** Enter your request or question to the AI.
- **Response Field:** Displays the AI-generated response.
- **Send Request:** Submits your prompt to the AI.
- **New Conversation:** Starts a new session by clearing previous history.
- **Use Voice:** Activates speech-to-text (STT) input. A 3-second pause is required to indicate the end of dictation.
- **Cancel Task:** Interrupts and cancels any ongoing AI task.
- **Settings:** Opens the application settings window.
- **Close App:** Exits and shuts down Shelly.

Copilot Integration

You can optionally use Microsoft Copilot for real-time voice or text input. Once your conversation is complete, you can pass it to Shelly for further processing or execution.

Note: For best results, sign in to Copilot using your Microsoft account.

Console Mode

Shelly features a command-based console for advanced interactions. Type 'help' to view all available commands.

Custom Functions

Shelly supports a range of built-in Custom Functions that allow the AI to interact with your system, files, and browser.

To maximize accuracy, it’s recommended to use the provided prompt templates for each function.

Function List and Recommended Prompts

📝 WriteInsideFileOrWindow

- **Purpose**: Inserts AI-generated content directly into an open file or window.
- **How it works**: After sending your prompt, click inside the target window where the text should appear.
- **Prompt Example**:
  "Write inside my opened file: [your request]"

🖥️ CheckMyScreenAndAnswer

- **Purpose**: Allows the AI to respond based on what's visible on your screen.
- **How it works**: Shelly captures a screenshot of your main display and replies based on the content.
- **Prompt Example**:
  "Check my screen and tell me what's wrong."

🚀 StartOrRunApplicationByName

- **Purpose**: Launches an application from your Start Menu.
- **Prompt Example**:
  "Run Notepad"

ReadFileAndAnswer

- **Purpose**: Reads a file in multiple parts and replies based on its contents.
- **Prompt Example**:
  "Based on this file: [path], please summarize the content."

GenerateLargeFileWithTextOrCode

- **Purpose**: Generates large documents or code files.
- **Prompt Example**:
  "Generate a 4-page Word document about WW2 and save it to [path]."

UpdateFileByChunks

- **Purpose**: Updates an existing file with new content or corrections.
- **Prompt Example**:
  "Update this file [path] by fixing the grammar."

Note: Ideal for code fixes, content updates, or expansions.

GenerateImages

- **Purpose**: Creates AI-generated images.
- **Prompt Example**:
  "Generate a cartoon-style image of a robot and save it to [folder path]."

ChangeOrSetVolume

- **Purpose**: Adjusts system volume.
- **Prompt Example**:
  "Set volume to 50."

SendMediaKey

- **Purpose**: Sends play/pause/mute commands to your system's media player or browser.
- **Prompt Example**:
  "Pause the music."

WebSearchAndRespondBasedOnPageContent

- **Purpose**: Performs an online search and responds based on the first result.
- **How it works**: Uses a built-in browser and Google to search and read content.
- **Prompt Examples**:
  "Search Amazon.com for HP printers and list their prices."
  "Find the latest AI news and summarize the top 5 stories."

TakePrintScreenOrScreenShot

- **Purpose**: Captures a screenshot and saves it to a folder.
- **Prompt Example**:
  "Take a screenshot and save it to [folder path]."

ImageAnswer

- **Purpose**: Analyzes a specified image.
- **Prompt Example**:
  "Analyze this image: [path] and describe its contents."

ReadCopilotConversation

- **Purpose**: Allows Shelly to respond based on a Copilot conversation.
- **Prompt Example**:
  "Based on my Copilot chat, can you write a summary?"

🔍 **SearchForTextInsideFiles**

- **Purpose**: Searches for text inside files in a specified folder.
- **Prompt Example**:
  "Search all files in [folder path] for the word 'license'."

Note: Some antivirus software may block this function.

📄 **ReadWebPageAndRespondBasedOnPageContent**

- **Purpose**: Reads a specific URL and replies based on its contents.
- **Prompt Example**:
  "Check this webpage: [https://...] and tell me what it says."

📜 **GenerateBatchAndPs1File**

- **Purpose**: Creates and saves a batch or PowerShell script file.
- **Prompt Example**:
  "Generate a batch file that backs up my Documents folder."

Best practices and warnings

1. Always create a copy of existing files that are about to be manipulated by Shelly or OpenAI.
2. Some PowerShell scripts may cause file deletion, corruption, or damage to your computer. Use clear instructions and remind the AI to generate 100% safe scripts.
3. Shelly Beta 1.0.1 is based on AI Agents. AI Agents do not store in memory any generated PowerShell scripts or content from Custom Functions. Only final responses and previous user prompts are stored. ******This feature is available in the Shelly PRO version.*****

End of document