MIRACLE SOFTWARE SYSTEMS

Google Open Lab - Build an AI Chat Assistant using Gemini Flash

Open Lab | Digital Summit 2025

Pre-Requisites

- Node.js (v18 or above recommended) Download: https://nodejs.org
- Git Download: https://git-scm.com
- A code editor (VS Code recommended)

Steps to Get Started

STEP 1 | Clone the github repository

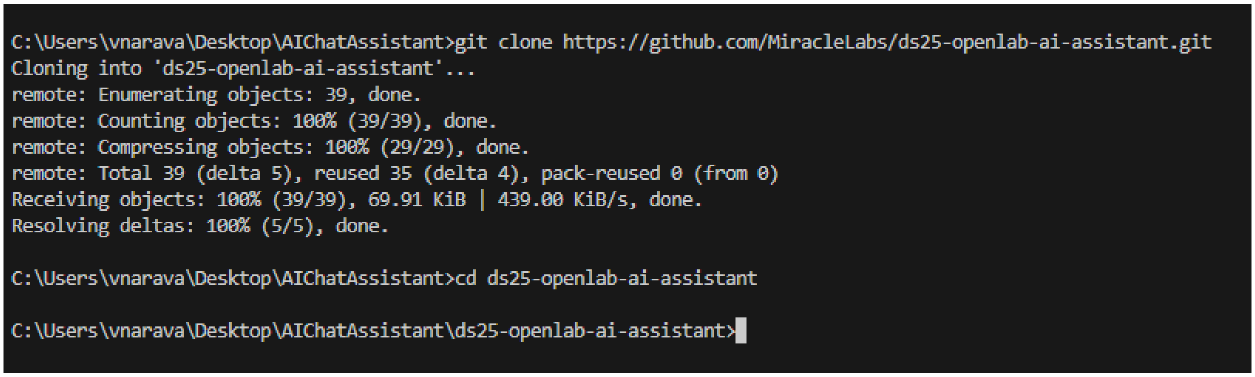
- Open terminal / command prompt
- Run the following command:

  git clone

- Move into the project folder:

  cd

The repository contains two folders: - server (Backend – Node.js) - client (Frontend – React + Tailwind)



```
C:\Users\vnarava\Desktop\AIChatAssistant>git clone https://github.com/MiracleLabs/ds25-openlab-ai-assistant.git
Cloning into 'ds25-openlab-ai-assistant'...
remote: Enumerating objects: 39, done.
remote: Counting objects: 100% (39/39), done.
remote: Compressing objects: 100% (29/29), done.
remote: Total 39 (delta 5), reused 35 (delta 4), pack-reused 0 (from 0)
Receiving objects: 100% (39/39), 69.91 KiB | 439.00 KiB/s, done.
Resolving deltas: 100% (5/5), done.

C:\Users\vnarava\Desktop\AIChatAssistant>cd ds25-openlab-ai-assistant

C:\Users\vnarava\Desktop\AIChatAssistant\ds25-openlab-ai-assistant>
```

STEP 2 | Set up and run server

- Navigate to server folder:

  cd server

- Install dependencies:

npm install

- Create .env file
- Inside the server folder
- Create a new file named: .env
- Add the following content to .env file:

  GOOGLE_GENERATIVE_AI_API_KEY=your_google_api_key_here

- Start the server:

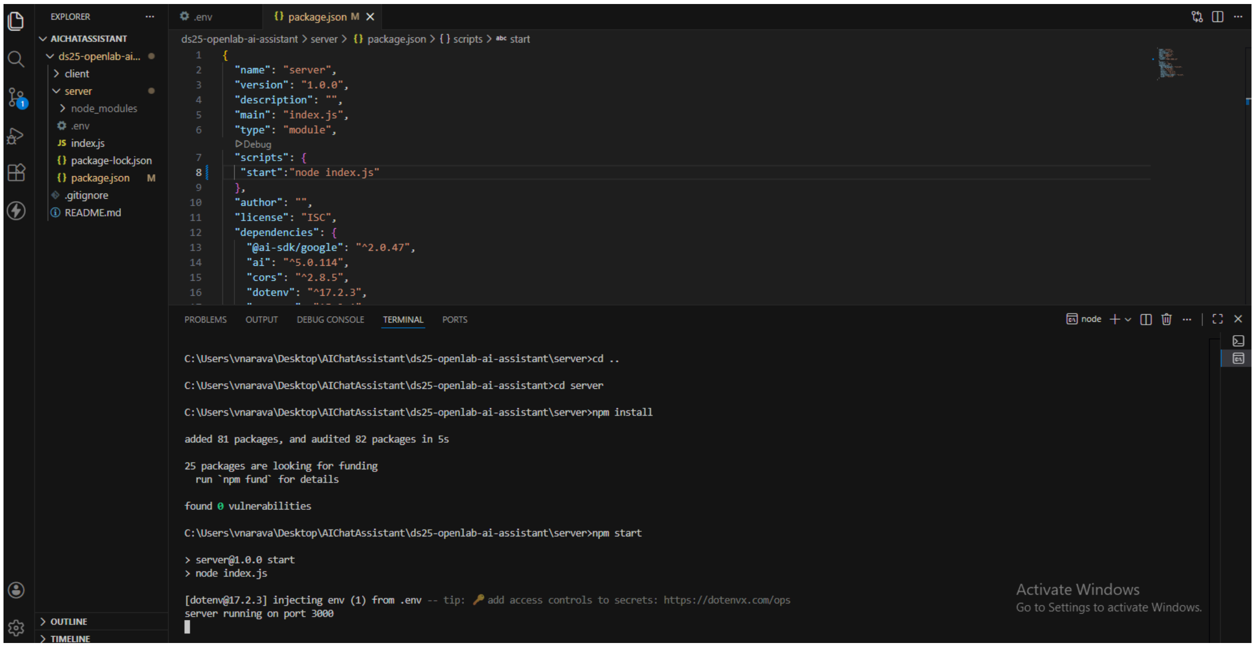
  npm start

- Add script in package.json under scripts

  “Start” : ”node index.js”

- Server should start on:

  http://localhost:3000



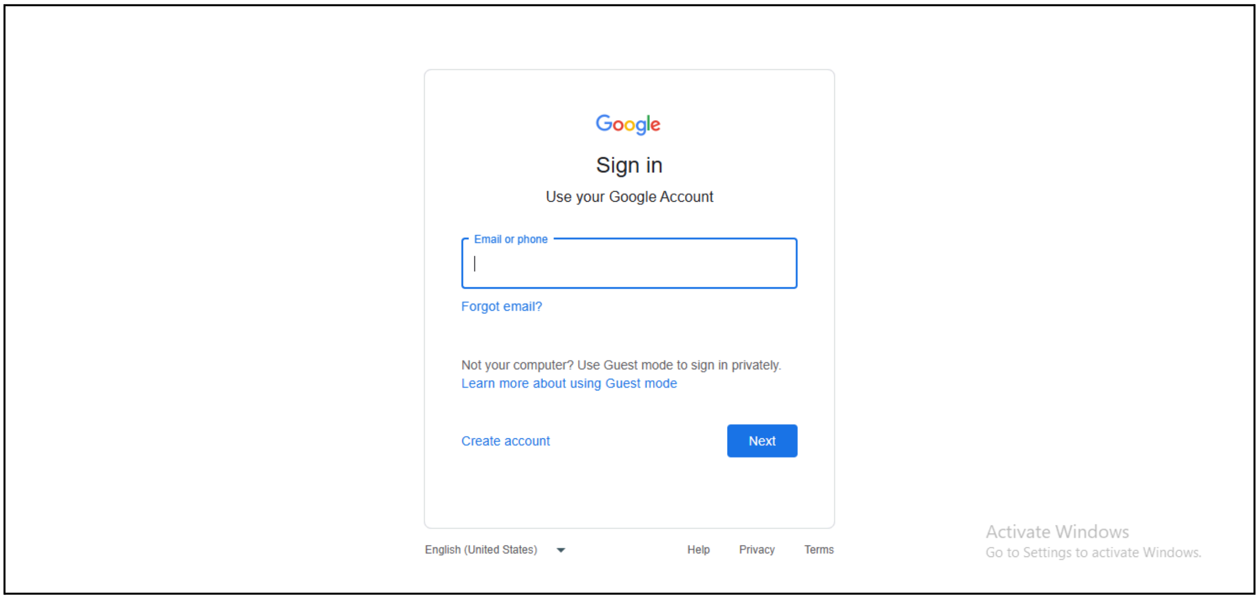
```
{
  "name": "server",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "type": "module",
  ▷Debug
  "scripts": {
   "start":"node index.js"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "@ai-sdk/google": "^2.0.47",
    "ai": "^5.0.114",
    "cors": "^2.8.5",
    "dotenv": "^17.2.3",
```

```
C:\Users\vnarava\Desktop\AIChatAssistant\ds25-openlab-ai-assistant\server>cd ..

C:\Users\vnarava\Desktop\AIChatAssistant\ds25-openlab-ai-assistant>cd server

C:\Users\vnarava\Desktop\AIChatAssistant\ds25-openlab-ai-assistant\server>npm install

added 81 packages, and audited 82 packages in 5s

25 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

C:\Users\vnarava\Desktop\AIChatAssistant\ds25-openlab-ai-assistant\server>npm start

> server@1.0.0 start
> node index.js

[dotenv@17.2.3] injecting env (1) from .env -- tip: 🔑 add access controls to secrets: https://dotenvx.com/ops
server running on port 3000
```

STEP 3 | Get google AI Studio API KEY

1. Access Google AI Studio

- Click on this link to login to Google AI Studio
- Log in with your Google Account credentials (If prompted) by entering your email, password and click continue.

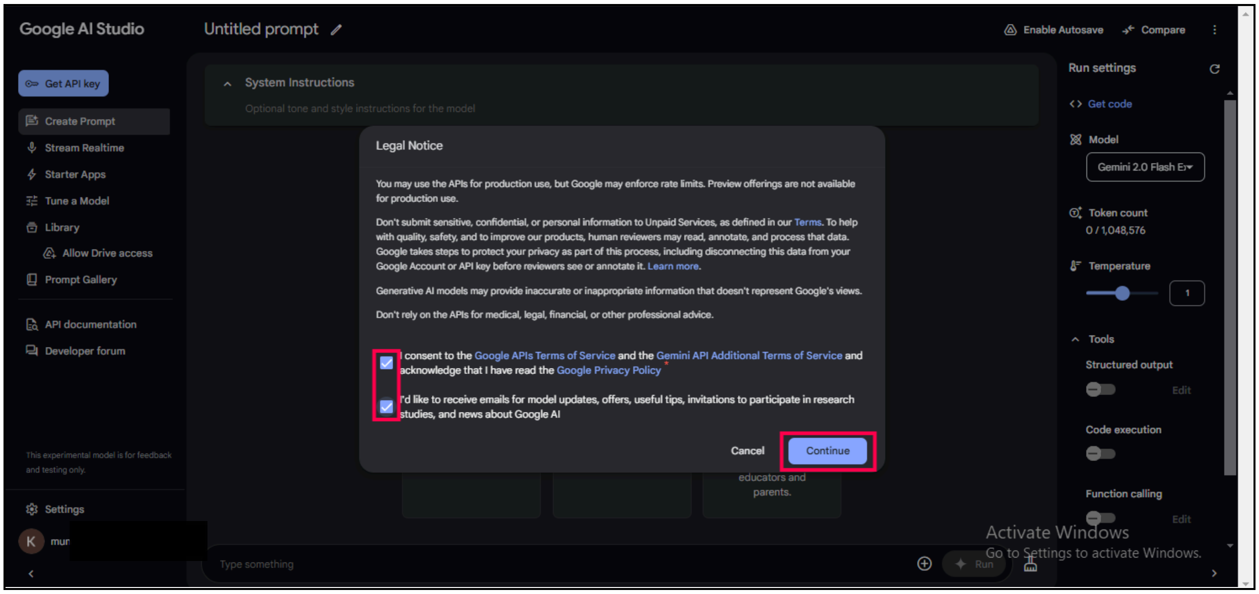


Sign in

Use your Google Account

Not your computer? Use Guest mode to sign in privately.

- After logging into your **Google Account**, you'll be redirected to the AI Studio Dashboard, where you need to accept the **Terms of Service** and click "**Continue**" to proceed.



[redacted]

Legal Notice

You may use the APIs for production use, but Google may enforce rate limits. Preview offerings are not available for production use.

Don't submit sensitive, confidential, or personal information to Unpaid Services, as defined in our Terms. To help with quality, safety, and to improve our products, human reviewers may read, annotate, and process that data. Google takes steps to protect your privacy as part of this process, including disconnecting this data from your Google Account or API key before reviewers see or annotate it. Learn more.

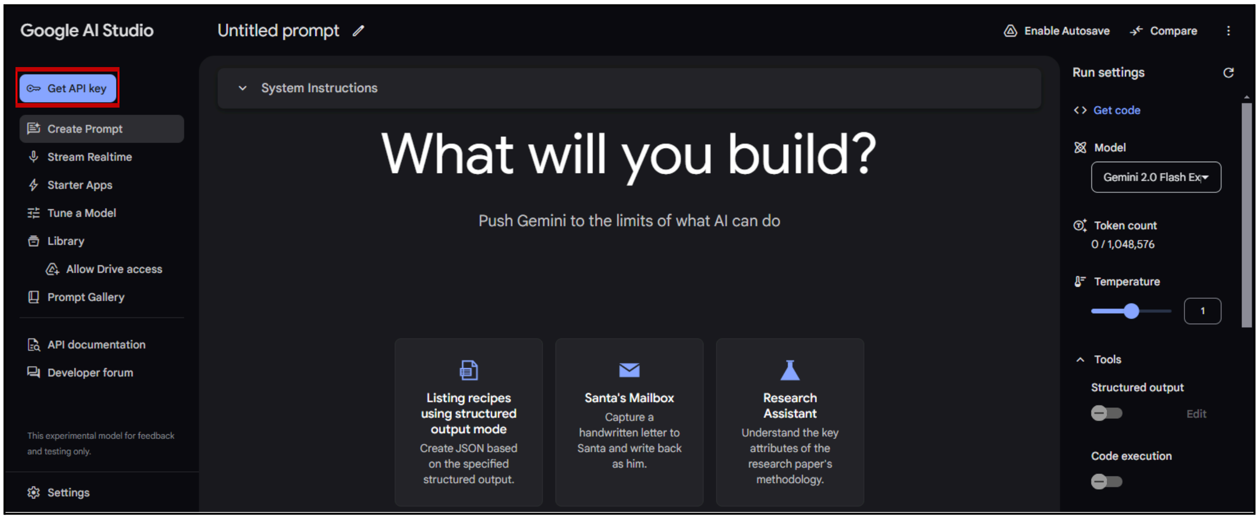
Generative AI models may provide inaccurate or inappropriate information that doesn't represent Google's views.

Don't rely on the APIs for medical, legal, financial, or other professional advice.

☑ I consent to the Google APIs Terms of Service and the Gemini API Additional Terms of Service and acknowledge that I have read the Google Privacy Policy *

☑ I'd like to receive emails for model updates, offers, useful tips, invitations to participate in research studies, and news about Google AI

You will then be redirected to the Dashboard.



What will you build?

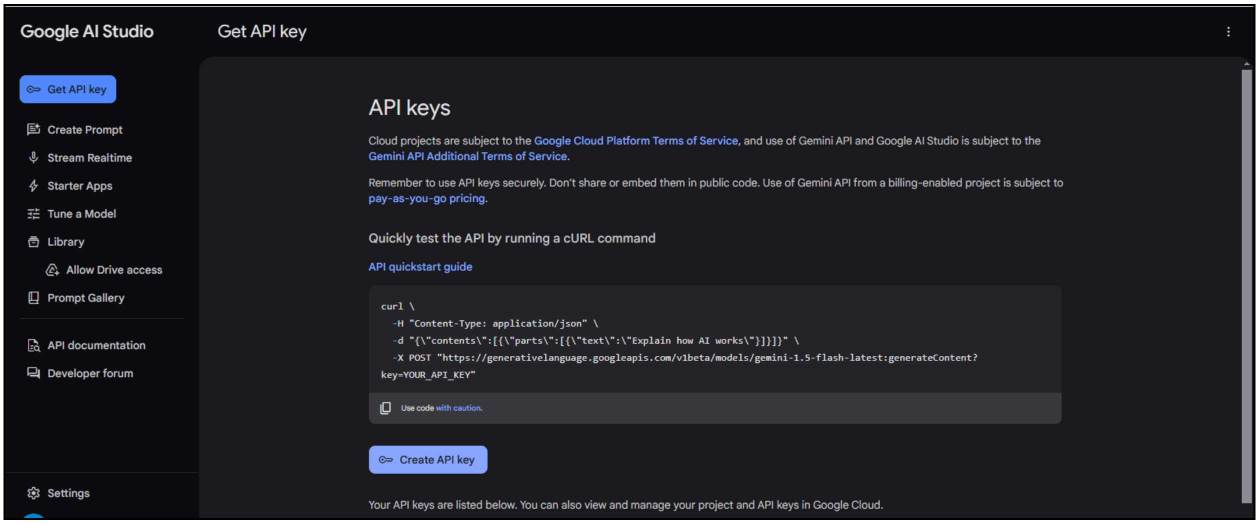
Push Gemini to the limits of what AI can do

Listing recipes using structured output mode
Create JSON based on the specified structured output.

Santa's Mailbox
Capture a handwritten letter to Santa and write back as him.

Research Assistant
Understand the key attributes of the research paper's methodology.

Click on '**Get API Key**' in the top left corner to proceed to the screen shown below.



API keys

Cloud projects are subject to the Google Cloud Platform Terms of Service, and use of Gemini API and Google AI Studio is subject to the Gemini API Additional Terms of Service.

Remember to use API keys securely. Don't share or embed them in public code. Use of Gemini API from a billing-enabled project is subject to pay-as-you-go pricing.

Quickly test the API by running a cURL command

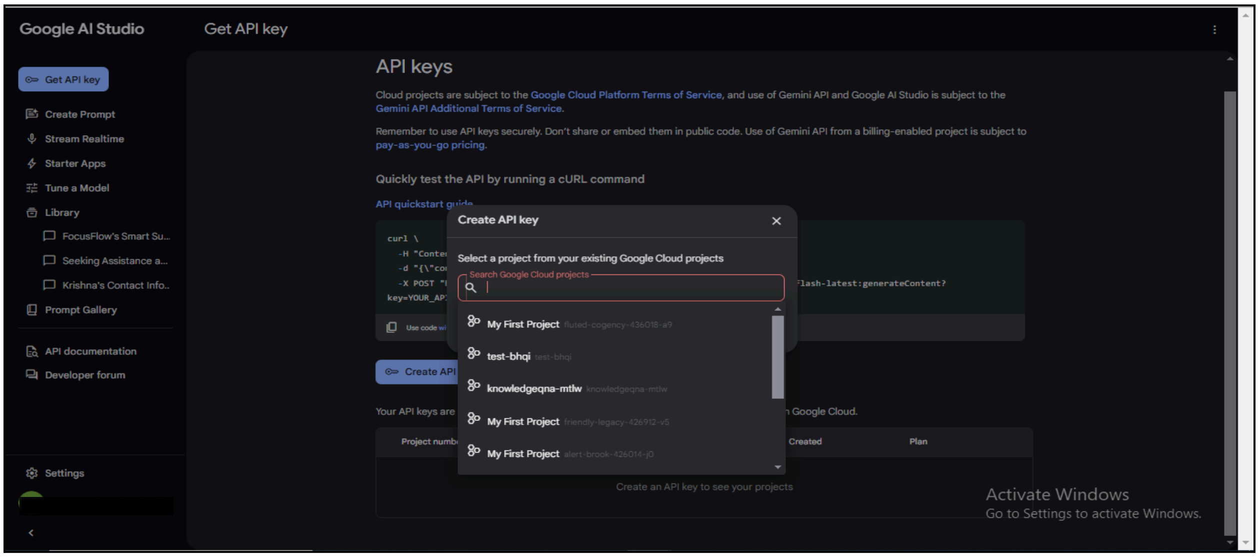
API quickstart guide

```
curl \
  -H "Content-Type: application/json" \
  -d "{\"contents\":[{\"parts\":[{\"text\":\"Explain how AI works\"}]}]}" \
  -X POST "https://generativelanguage.googleapis.com/v1beta/models/gemini-1.5-flash-latest:generateContent?
key=YOUR_API_KEY"
```

Use code with caution.

Your API keys are listed below. You can also view and manage your project and API keys in Google Cloud.

Click on '**Create API Key**' and choose the default or any available project as shown below.



[redacted]

API keys

Cloud projects are subject to the Google Cloud Platform Terms of Service, and use of Gemini API and Google AI Studio is subject to the Gemini API Additional Terms of Service.

Remember to use API keys securely. Don't share or embed them in public code. Use of Gemini API from a billing-enabled project is subject to pay-as-you-go pricing.

Quickly test the API by running a cURL command

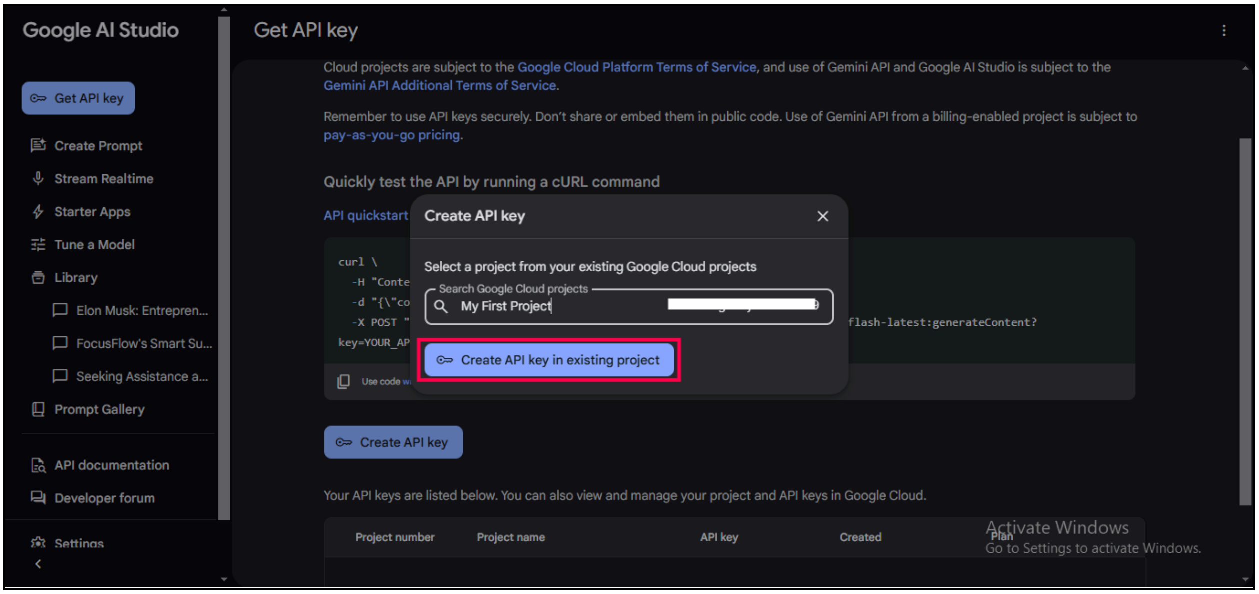
API quickstart guide

Select a project from your existing Google Cloud projects

- My First Project fluted-cogency-436018-a9
- test-bhqi test-bhqi
- knowledgeqna-mtlw knowledgeqna-mtlw
- My First Project friendly-legacy-426912-v5
- My First Project alert-brook-426014-j0

Create an API key to see your projects

Click on '**Create API Key**' in an existing project to generate a key.



Get API key

Cloud projects are subject to the Google Cloud Platform Terms of Service, and use of Gemini API and Google AI Studio is subject to the Gemini API Additional Terms of Service.

Remember to use API keys securely. Don't share or embed them in public code. Use of Gemini API from a billing-enabled project is subject to pay-as-you-go pricing.

Quickly test the API by running a cURL command

API quickstart

```
curl \
  -H "Conte
  -d "{\"co
  -X POST "                                        flash-latest:generateContent?
key=YOUR_AP
```

Create API key

Select a project from your existing Google Cloud projects

Search Google Cloud projects: My First Project [redacted]

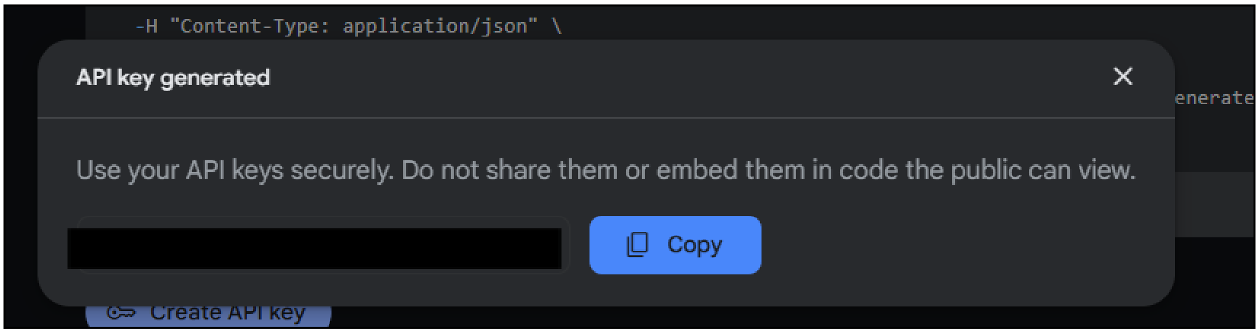
Create API key in existing project

Create API key

Your API keys are listed below. You can also view and manage your project and API keys in Google Cloud.

Project number	Project name	API key	Created	Plan

Now you will see the API Key like the one in the below image.



```
-H "Content-Type: application/json" \
```

API key generated

Use your API keys securely. Do not share them or embed them in code the public can view.

[redacted]

- Paste the API key into the .env file created earlier:

  GOOGLE_GENERATIVE_AI_API_KEY=PASTE_KEY_HERESTEP

STEP 4 | Set up and Run Client (Frontend)

- Open a new terminal window
- Navigate to client folder:

  cd client

- Install dependencies:

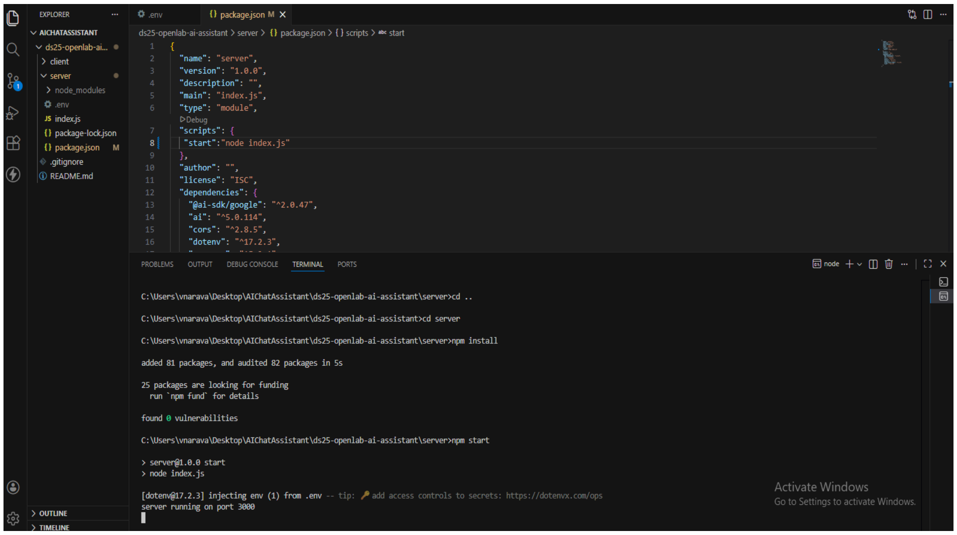
npm install

- Start the React application:

  npm run dev

- Open browser and go to:

  http://localhost:5173

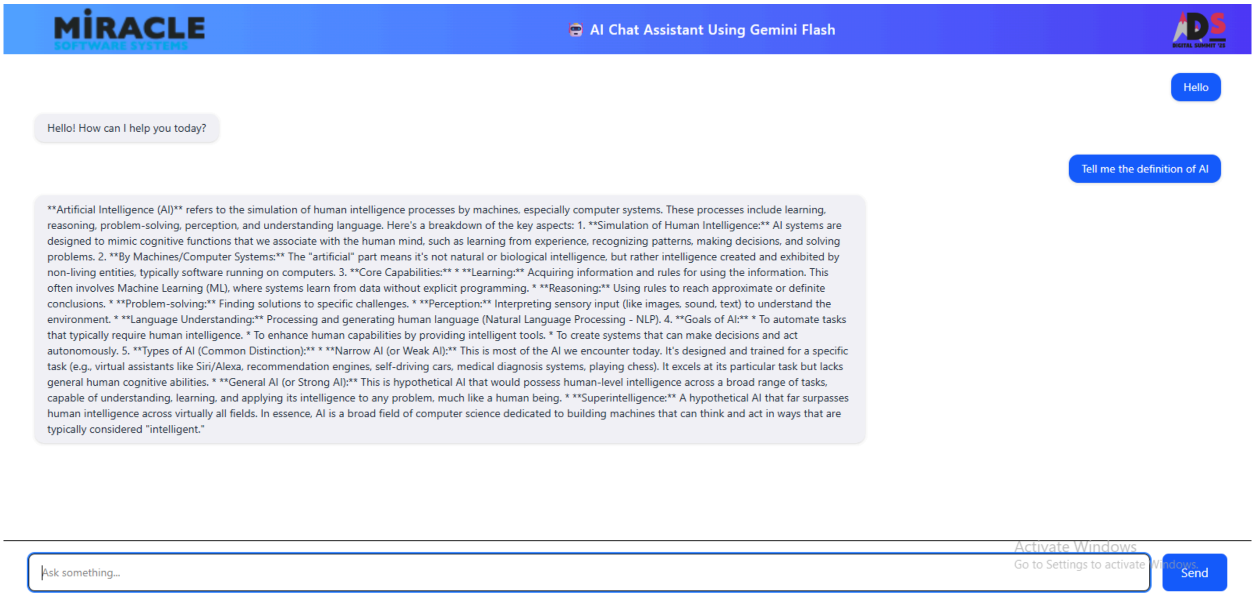


```
{
  "name": "server",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "type": "module",
  "scripts": {
   "start":"node index.js"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "@ai-sdk/google": "^2.0.47",
    "ai": "^5.0.114",
    "cors": "^2.8.5",
    "dotenv": "^17.2.3",
```

```
C:\Users\vnarava\Desktop\AIChatAssistant\ds25-openlab-ai-assistant\server>cd ..

C:\Users\vnarava\Desktop\AIChatAssistant\ds25-openlab-ai-assistant>cd server

C:\Users\vnarava\Desktop\AIChatAssistant\ds25-openlab-ai-assistant\server>npm install

added 81 packages, and audited 82 packages in 5s

25 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

C:\Users\vnarava\Desktop\AIChatAssistant\ds25-openlab-ai-assistant\server>npm start

> server@1.0.0 start
> node index.js

[dotenv@17.2.3] injecting env (1) from .env -- tip: 🔑 add access controls to secrets: https://dotenvx.com/ops
server running on port 3000
```

STEP 5 | Verify Application

- The header with logos should be visible
- Chat input box should appear at the bottom
- Type a message and press Send
- AI response should be displayed


AI Chat Assistant Using Gemini Flash
Ask something...
Send
Activate Windows
Go to Settings to activate Windows.

Hello

Hello! How can I help you today?

Tell me the definition of AI

Artificial Intelligence (AI) refers to the simulation of human intelligence processes by machines, especially computer systems. These processes include learning, reasoning, problem-solving, perception, and understanding language. Here's a breakdown of the key aspects: 1. **Simulation of Human Intelligence:** AI systems are designed to mimic cognitive functions that we associate with the human mind, such as learning from experience, recognizing patterns, making decisions, and solving problems. 2. **By Machines/Computer Systems:** The "artificial" part means it's not natural or biological intelligence, but rather intelligence created and exhibited by non-living entities, typically software running on computers. 3. **Core Capabilities:** * **Learning:** Acquiring information and rules for using the information. This often involves Machine Learning (ML), where systems learn from data without explicit programming. * **Reasoning:** Using rules to reach approximate or definite conclusions. * **Problem-solving:** Finding solutions to specific challenges. * **Perception:** Interpreting sensory input (like images, sound, text) to understand the environment. * **Language Understanding:** Processing and generating human language (Natural Language Processing - NLP). 4. **Goals of AI:** * To automate tasks that typically require human intelligence. * To enhance human capabilities by providing intelligent tools. * To create systems that can make decisions and act autonomously. 5. **Types of AI (Common Distinction):** * **Narrow AI (or Weak AI):** This is most of the AI we encounter today. It's designed and trained for a specific task (e.g., virtual assistants like Siri/Alexa, recommendation engines, self-driving cars, medical diagnosis systems, playing chess). It excels at its particular task but lacks general human cognitive abilities. * **General AI (or Strong AI):** This is hypothetical AI that would possess human-level intelligence across a broad range of tasks, capable of understanding, learning, and applying its intelligence to any problem, much like a human being. * **Superintelligence:** A hypothetical AI that far surpasses human intelligence across virtually all fields. In essence, AI is a broad field of computer science dedicated to building machines that can think and act in ways that are typically considered "intelligent."

COMMON ISSUES & FIXES

- “API key not found”
    - Check .env file exists inside server folder
    - Restart server after updating .env
- Server not responding
    - Make sure server is running on port 3000
- Client cannot connect to server
    - Verify API URL in frontend code
    - Check CORS configuration