



Groq Connected System for [Appian](#)
V1.0.0

Appian Corporation

Version 1.0.0

Table of Contents

Overview	3
Features	3
Connected System Configuration	4
Steps to obtain Groq API Key	5
Integration Configuration	6

Overview

The Groq Connected System plugin seamlessly integrates Groq's cutting-edge AI acceleration technology with the Appian platform, empowering users to harness advanced AI capabilities within their workflows.

Utilizing API authentication mechanisms, the plugin facilitates secure access to Groq's powerful AI acceleration services, ensuring data integrity and privacy.

By bridging Appian with Groq's AI acceleration technology, this plugin streamlines text processing tasks, offering users the ability to enhance productivity and automate text-based operations effortlessly. Simply input text prompts, and the Groq Connected System plugin swiftly generates intelligent text completions and insights, altering the way users interact with and derive value from their textual data.

Features

- Make API calls to Groq's Chat Completion endpoint to generate text completions using any of the following models: LLaMA3 8b, LLaMA3 70b, LLaMA2 70b, Mixtral 8x7b and Gemma 7b
- Configure other optional inputs such as temperature, topP, max tokens, temperature and stop sequence.
- Seamlessly integrate completions generation into Appian workflows with easy setup of connected system

Connected System Configuration

Connected System Properties



Groq Connected System

Connected System that makes calls to Groq API
Version: 1

Name *

GSA CS Groq

Description

Groq Connected System

Groq Connected System Configuration

Groq API Key ?*

.....

Please provide API Key obtained from Groq

Connection successful

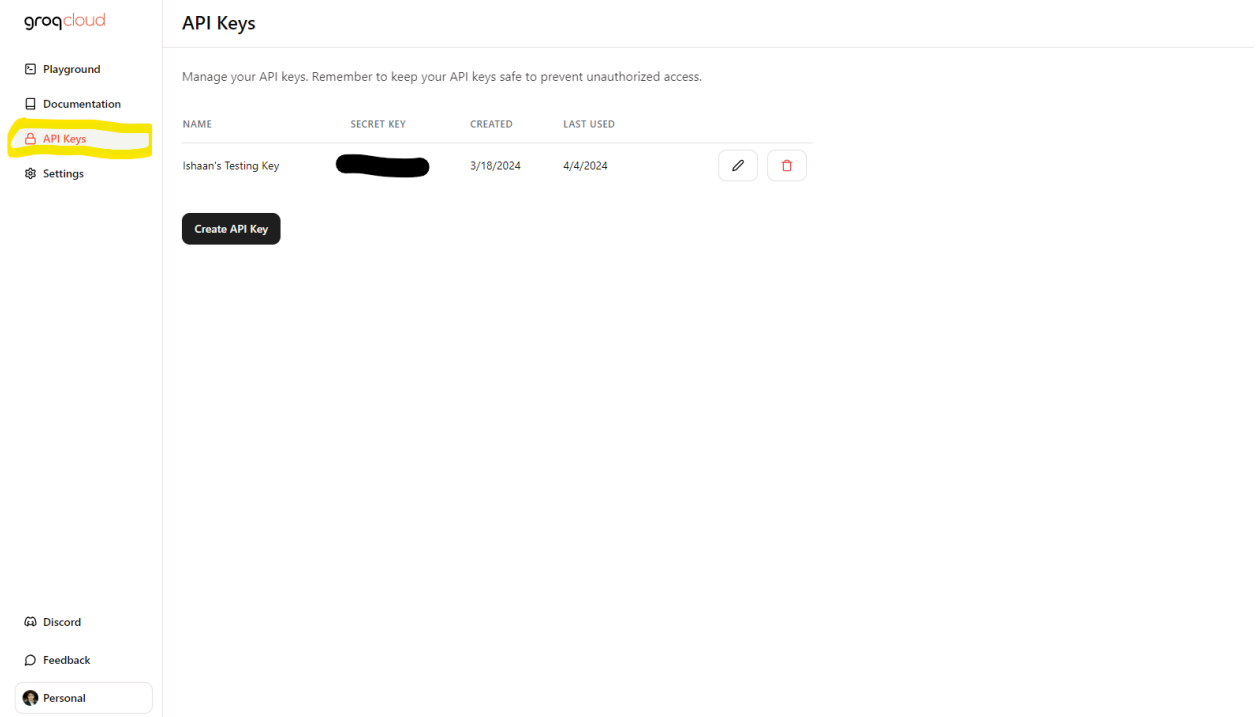
TEST CONNECTION

CANCEL

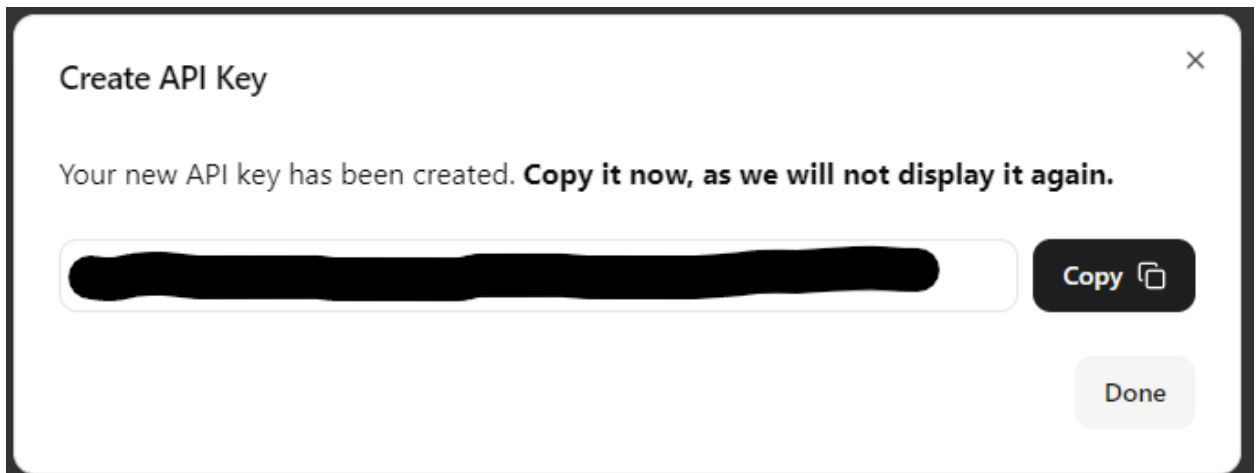
USE IN NEW INTEGRATION **SAVE**

Steps to obtain Groq API Key

1. **View API Keys:** Once you have logged into [Groq Cloud](#) through your Appian email, click on **API Keys** to view all the API keys associated with your account.



2. **Use the API Key:** You can use the existing API key or Generate a new key using the **Create API Key** button. Remember to copy your API Key and store it in a secured location as it will not be displayed again.



Integration Configuration

Inputs:

Content (Text) - Required - Text input as the starting point for generating the response.

Model (Text Dropdown) - Required - Variant of the Groq model to generate with. Valid values: LLaMA3-8b, LLaMA3-70b, LLaMA2-70b, Mixtral-8x7b, Gemma-7b-it.

Temperature (Number(Decimal)) - Optional - Controls randomness of responses. A lower temperature leads to more predictable outputs while a higher temperature results in more varies and sometimes more creative outputs.

Max Tokens (Number(Integer)) - Optional - The maximum number of tokens that the model can process in a single response. This limits ensures computational efficiency and resource management.

Top P (Number(Decimal)) - Optional - A method of text generation where a model will only consider the most probable next tokens that make up the probability p. 0.5 means half of all likelihood-weighted options are considered.

Stop Sequence (Text) - Optional - A stop sequence is a predefined or user-specified text string that signals an AI to stop generating content, ensuring its responses remain focused and concise.

Rule Input Name	Expression	Value
model (Text)	1 null	null (Text)
content (Text)	1 "what are the main advantages of low latency LLMs?"	"What are the main advantages of low latency LLMs?"
temperature (Number (Decimal))	1 null	null (Number (Decimal))
maxTokens (Number (Integer))	1 null	null (Number (integer))
topP (Number (Decimal))	1 null	null (Number (Decimal))
stopSequence (Text)	1 null	null (Text)

```
id "chatcmpl-8d7f1688-f422-4d91-bb27-2f455d22369e" (Text)
  choices List of Dictionary - 1 item
  Dictionary
    finish_reason "stop" (Text)
    index 0 (Number (Decimal))
    message Dictionary
      role "assistant" (Text)
      content "There are several advantages of using low-latency large language models (LLMs), including: 1. Faster response time: LLMs can process and generate text in near real-time, al"
      logprobs null (Null)
      system_fingerprint "fp_90d05a079a" (Text)
      object "chat.completion" (Text)
    error null (Null)
  authType Diagnostic
```

Input:

```
JavaScript
{
  top_p: 1,
  stop: "random",
  max_tokens: 120,
  temperature: 0.7,
  messages: [
    {
      role: "user",
      content: "What are the 2 advantages of low latency
LLMs in short sentences?",
    },
  ],
  model: "llama2-70b-4096",
}
```

Output:

```
JavaScript
{
  id: "chatcmpl-1c34c029-d196-4f92-a5af-9ee3e30686c6",
  object: "chat.completion",
  created: 1712242000,
  model: "llama2-70b-4096",
  choices: [
    {
      index: 0,
      message: {
        role: "assistant",
        content: "Lower latency LLMs can provide faster
processing of information and faster generation of responses,
which can be beneficial in time-sensitive applications such as
chatbots, voice assistants, and real-time translation. Also, low
```

latency LLMs can improve the responsiveness and interactive experience of applications, making them feel more natural and conversational.",

```
    },
    logprobs: null,
    finish_reason: "stop",
  },
],
usage: {
  prompt_tokens: 44,
  prompt_time: 0.015,
  completion_tokens: 75,
  completion_time: 0.254,
  total_tokens: 119,
  total_time: 0.269,
},
system_fingerprint: "fp_90d05a079a",
x_groq: {
  id: "req_01htmryegnfwprrp6m10vn5xwz",
},
}
```