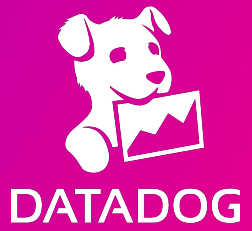
Building and Monitoring Generative AI Azure Logic Apps



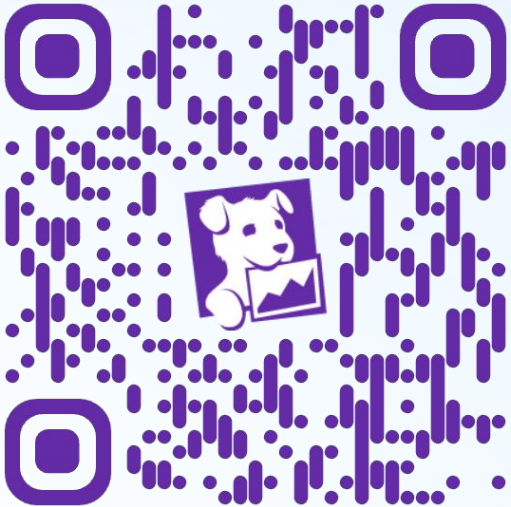
Jason Hand

Senior Developer Advocate

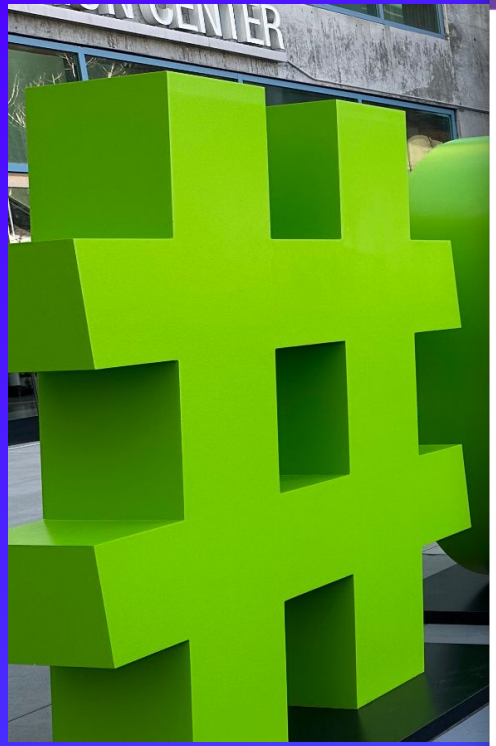
Datadog

DATADOG

Agenda



1. Building low-code applications using Azure Logic Apps
2. What can be collected from generative AI integrations
3. How to send logs and metrics to Datadog from Azure Logic Apps
4. How to add real user monitoring (RUM) to a web application
5. How to visualize Azure Logic App, OpenAI, and web app data in Datadog
6. Takeaways & Resources



Nvidia GTC Conf

notion.so/jasonhand/Nvidia-GTC-Conf-ca4a9c5952d84fc4bb...

New Chrome available

All Bookmarks

Conference Notes / Nvidia GTC Conf

Share

Nvidia GTC Conf

Sessions atte

17 sessions including th

- Nividia GTC 2024 -
- How to Apply Gener
- Multi-Modal Integrat
- LLMOps: The New F
- A Culture of Open a
- The Unsolved Chall
- Scaling Creative Ho
- Visualizing New Nar
- Energy-Efficient Ge
- Driving Accessibility
- Information Integrity
- How to Monitor You
- Learn How Educator
- "The Creative Sourc
- Modernizing Games
- State of PyTorch [S

How to Monitor Your Generat

notion.so/jasonhand/How-to-Monitor-Your-Generative-AI-Tec...

New Chrome available

All Bookmarks

Conference Notes / Nvidia GTC Conf / How to Monitor Your Generative AI ...

Share

How to Monitor Your Generative AI Tech Stack [EXPT63170]

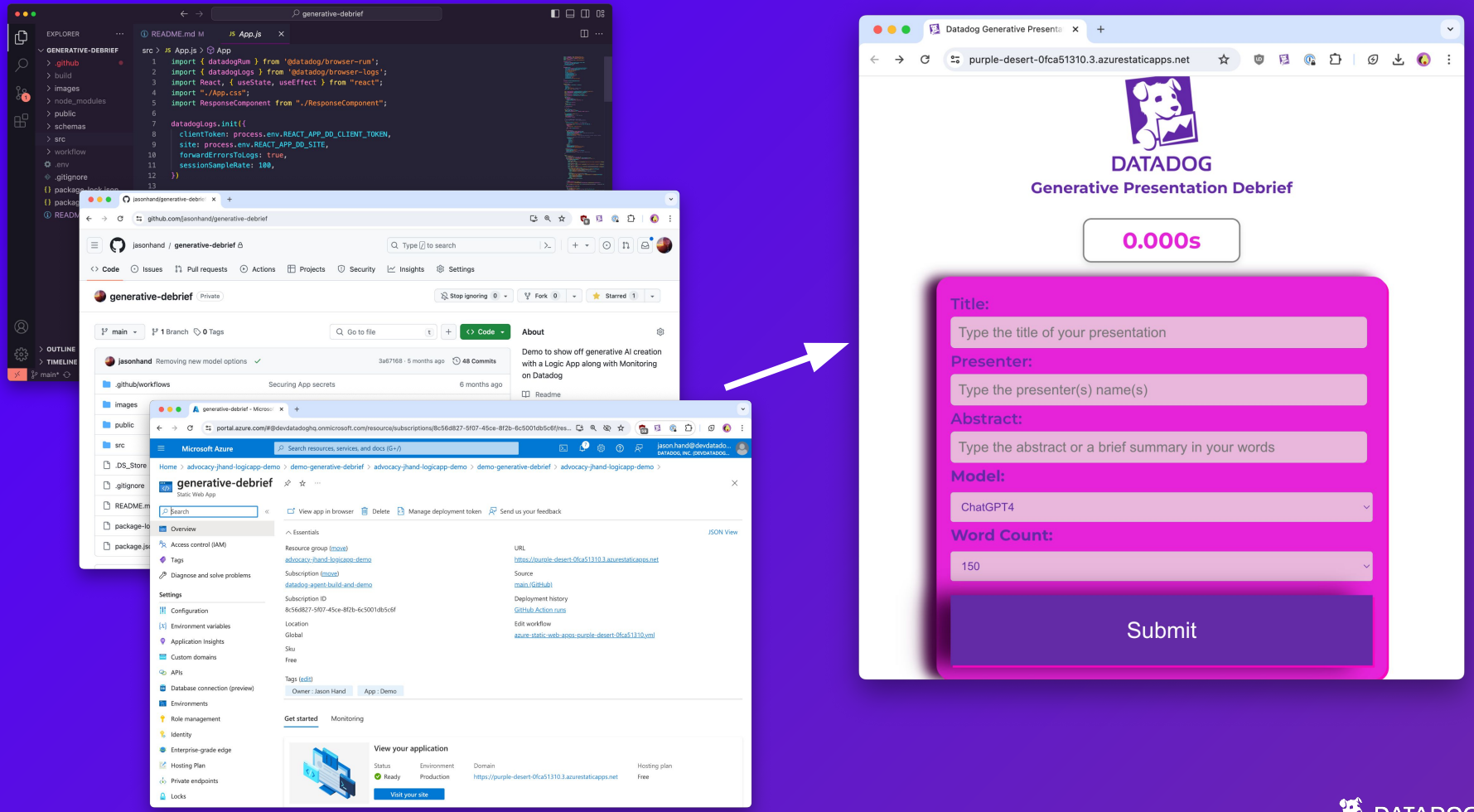
Datadog's generative AI integrations are designed to provide visibility into your applications and services that make use of large language models, and with these integrations you can effectively monitor the performance of your AI stack. Learn how these integrations work and how the various pieces fit together as a whole when deploying models or developing your own. By taking a layered approach, we'll delve into specific topics related to each layer, ensuring that the entire process is covered.

Ryan MacLean, Senior Tech Evangelist, Datadog

Industry: All Industries

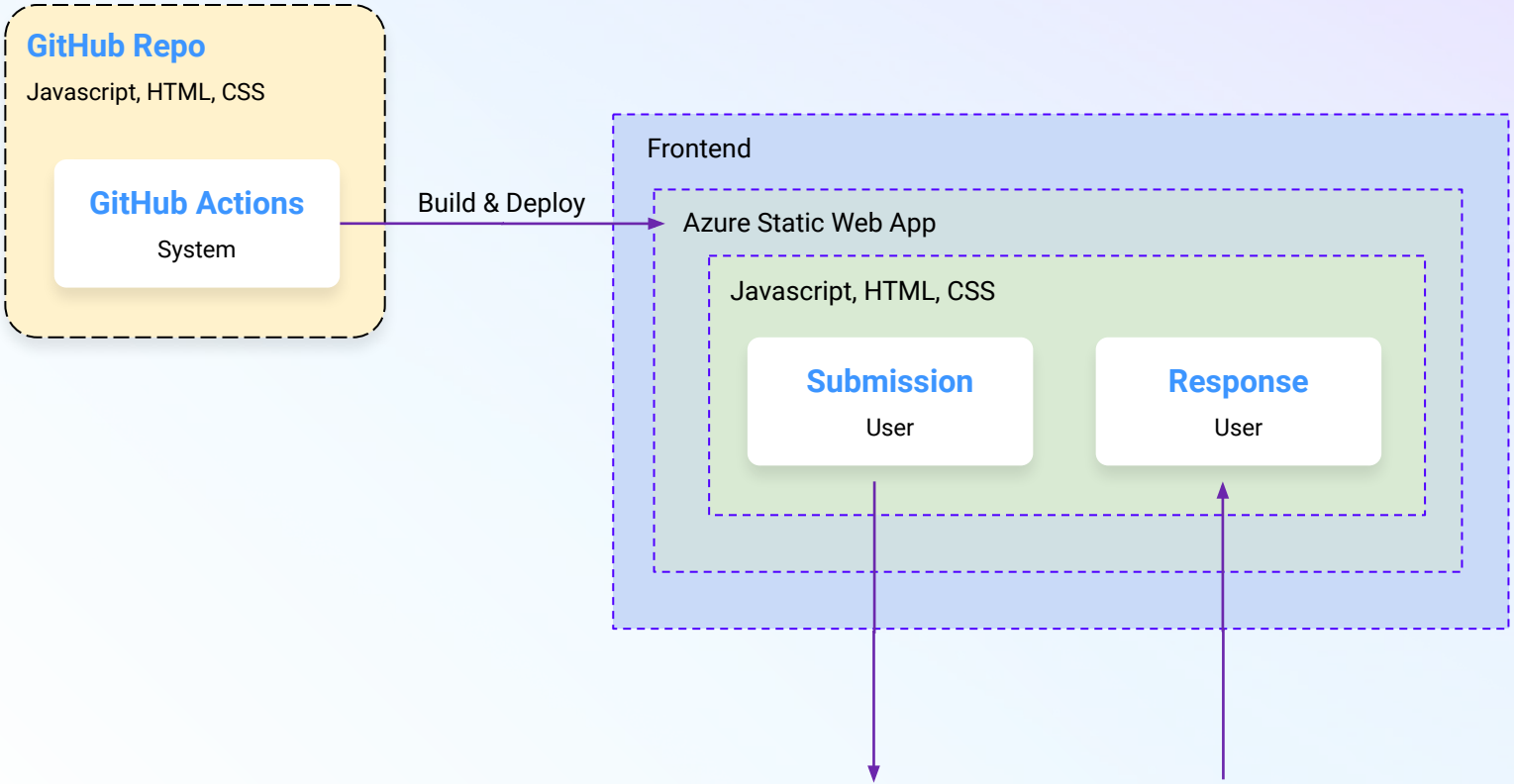
- **Wednesday, Mar 204:15 PM - 4:30 PM PDT** Expo Theater

The presentation focused on using Datadog's monitoring and analytics platform to enhance observability and security for AI and machine learning (ML) infrastructures, particularly Large Language Models (LLMs). Here's a summary of the key topics, takeaways, and notable quotes:

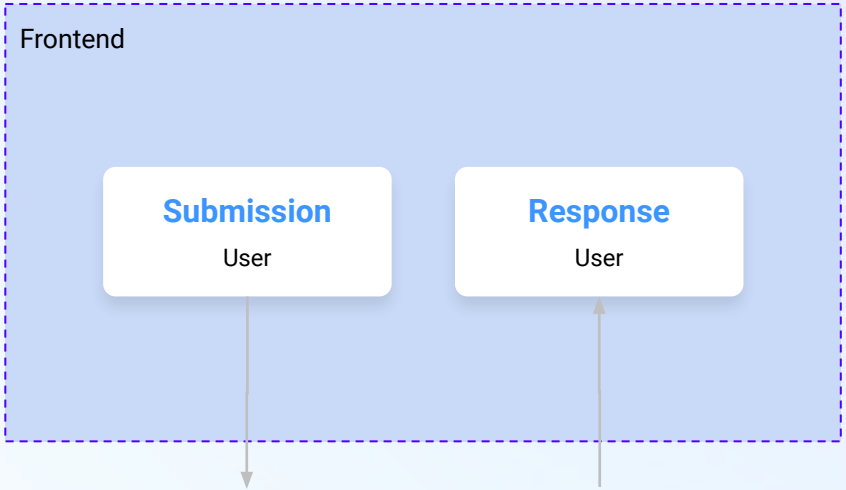

generative-debrief
README.md M
App.js
EXPLORER
GENERATIVE-DEBRIEF
.github
build
images
node_modules
public
schemas
src
workflow
.env
.gitignore
src > App.js > App
import { datadogRum } from '@datadog/browser-rum';
import { datadogLogs } from '@datadog/browser-logs';
import React, { useState, useEffect } from "react";
import "./App.css";
import ResponseComponent from "./ResponseComponent";
datadogLogs.init({
clientToken: process.env.REACT_APP_DD_CLIENT_TOKEN,
site: process.env.REACT_APP_DD_SITE,
forwardErrorsToLogs: true,
sessionSampleRate: 100,
})
github.com/jasonhand/generative-debrief
jasonhand / generative-debrief
Code
Issues
Pull requests
Actions
Projects
Security
Insights
Settings
generative-debrief
Private
main
1 Branch
0 Tags
Go to file
Code
About
Demo to show off generative AI creation with a Logic App along with Monitoring on Datadog
Readme
jasonhand Removing new model options
48 Commits
.github/workflows
Securing App secrets
6 months ago
images
public
src
.DS_Store
.gitignore
README.md
package-lock.json
package.json
portal.azure.com/#@devdatadoghq.onmicrosoft.com/resource/subscriptions/8c56d827-5f07-45ce-8f2b-6c5001db5c6f/res...
Microsoft Azure
Search resources, services, and docs (G+/)
Home > advocacy-jhand-logicapp-demo > demo-generative-debrief > advocacy-jhand-logicapp-demo > demo-generative-debrief > advocacy-jhand-logicapp-demo
generative-debrief
Static Web App
Search
Overview
Access control (IAM)
Tags
Diagnose and solve problems
Settings
Configuration
Environment variables
Application Insights
Custom domains
APIs
Database connection (preview)
Environments
Role management
Identity
Enterprise-grade edge
Hosting Plan
Private endpoints
Locks
View app in browser
Delete
Manage deployment token
Send us your feedback
Essentials
JSON View
Resource group (move)
advocacy-jhand-logicapp-demo
Subscription (move)
datadog-agent-build-and-demo
Subscription ID
8c56d827-5f07-45ce-8f2b-6c5001db5c6f
Location
Global
Sku
Free
URL
https://purple-desert-0fca51310.3.azurestaticapps.net
Source
main (GitHub)
Deployment history
GitHub Action runs
Edit workflow
azure-static-web-apps-purple-desert-0fca51310.yml
Tags (edit)
Owner : Jason Hand
App : Demo
Get started
Monitoring
View your application
Status
Ready
Environment
Production
Domain
https://purple-desert-0fca51310.3.azurestaticapps.net
Hosting plan
Free
Visit your site
Datadog Generative Presenta
purple-desert-0fca51310.3.azurestaticapps.net
DATADOG
Generative Presentation Debrief
0.000s
Title:
Type the title of your presentation
Presenter:
Type the presenter(s) name(s)
Abstract:
Type the abstract or a brief summary in your words
Model:
ChatGPT4
Word Count:
150
Submit

DATADOG

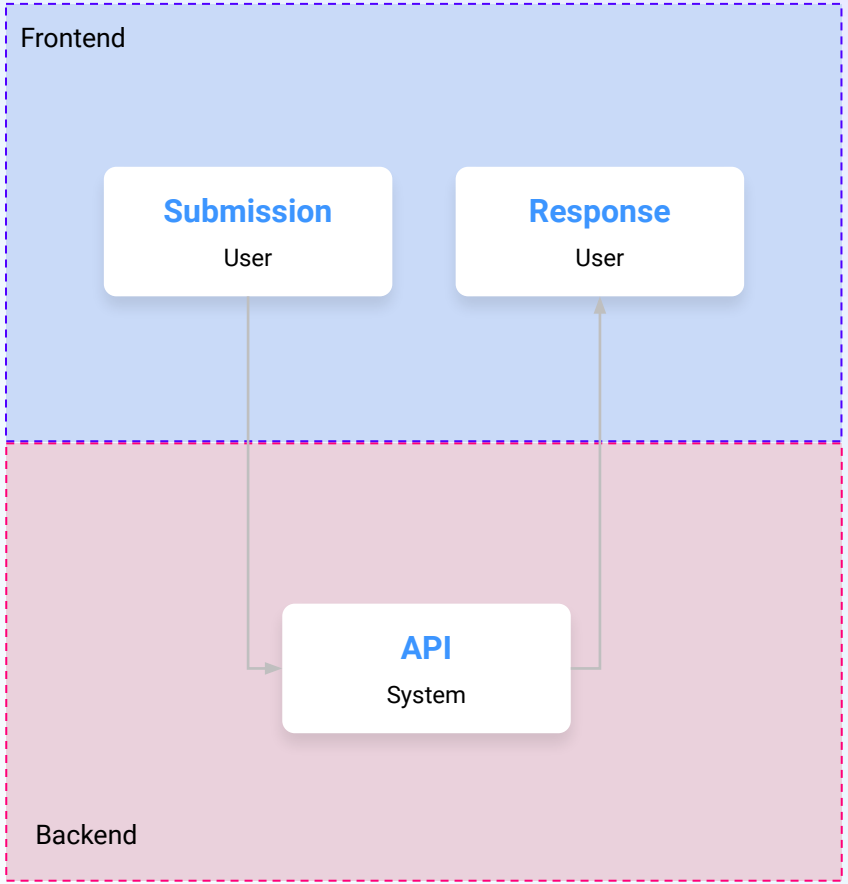
Generative Debrief

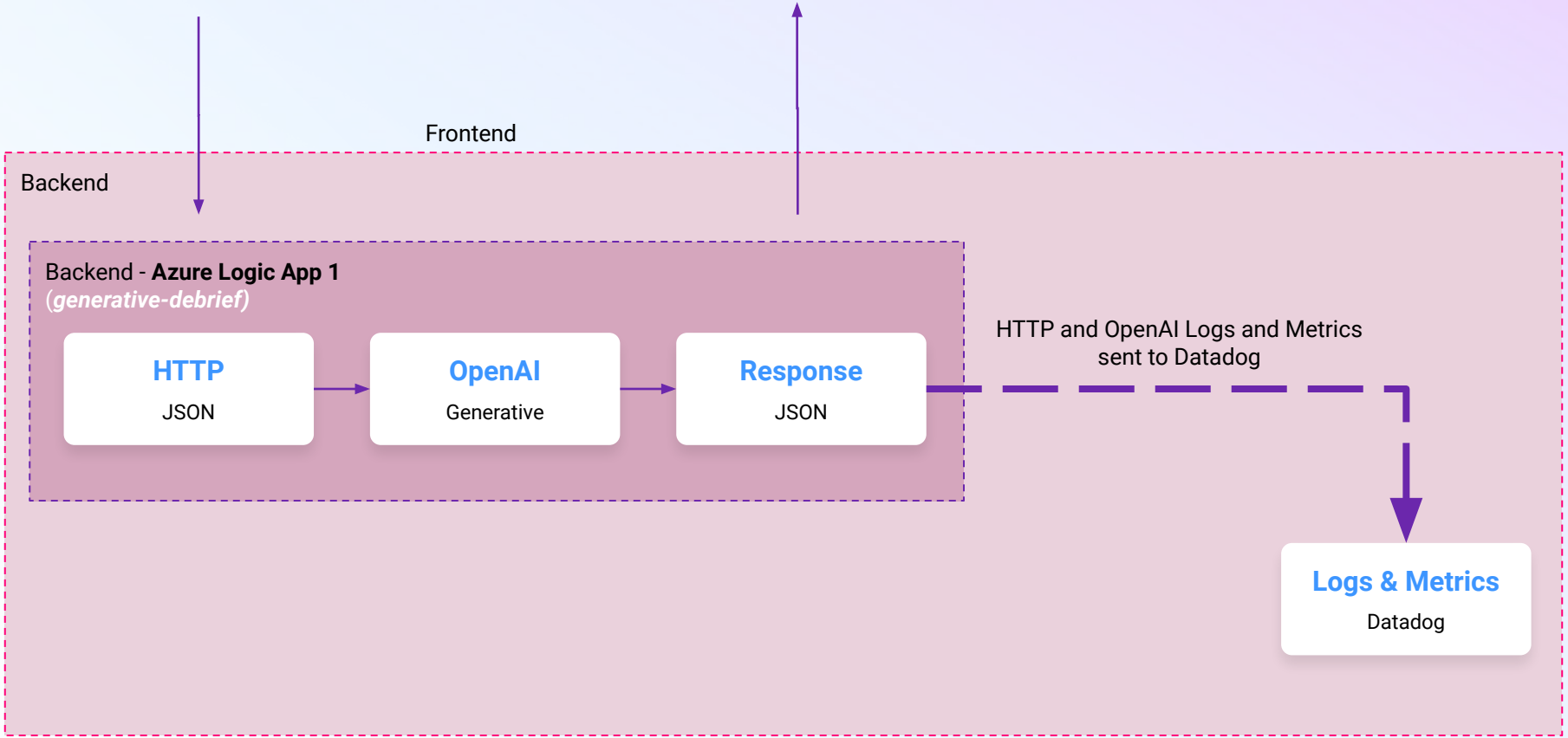


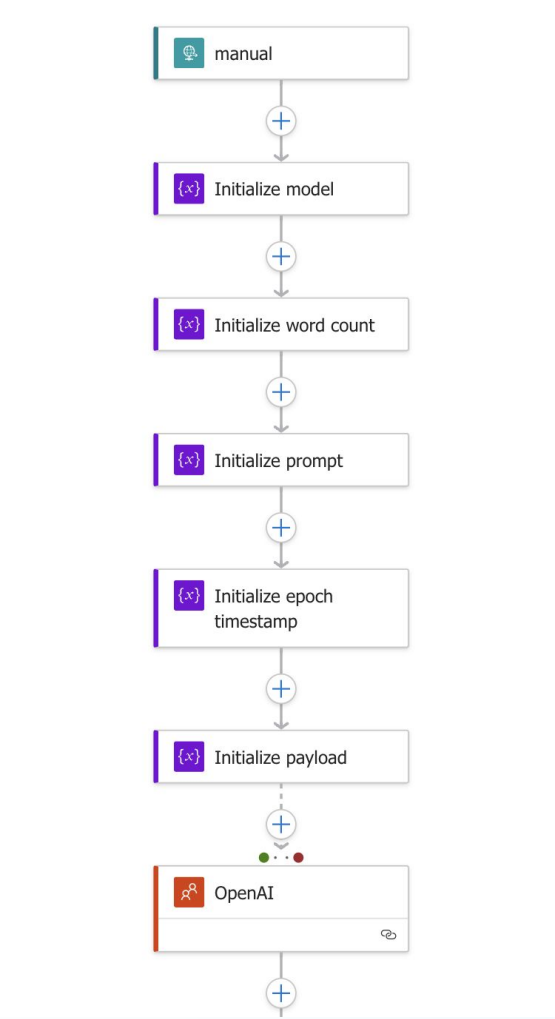
Simplified

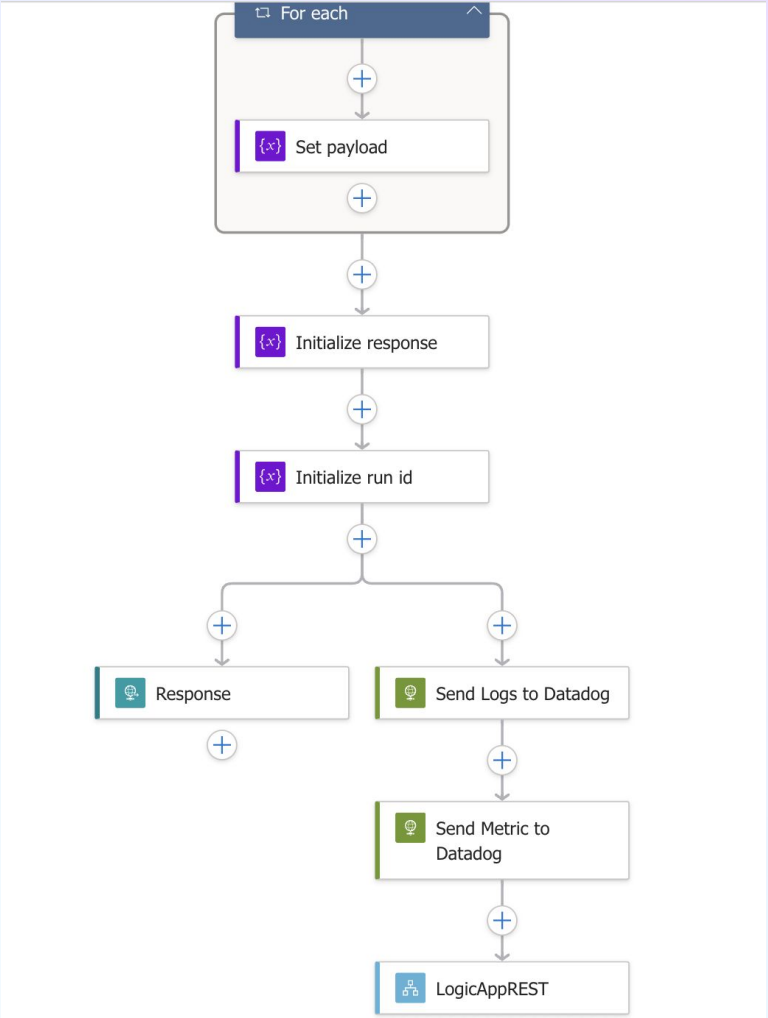


Generative Debrief

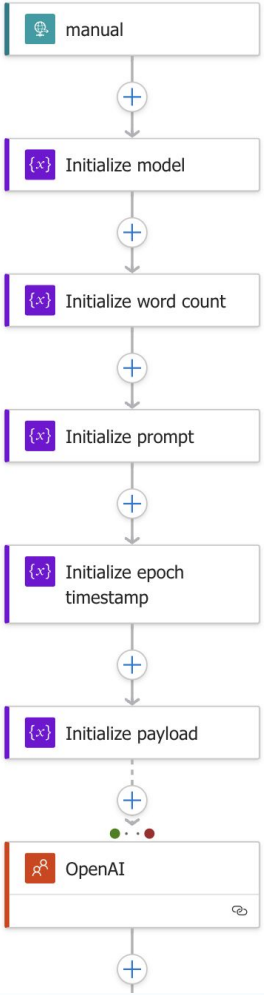



Frontend
Backend
Backend - **Azure Logic App 1**
(generative-debrief)
HTTP
JSON
OpenAI
Generative
Response
JSON
HTTP and OpenAI Logs and Metrics
sent to Datadog
Logs & Metrics
Datadog


manual
Initialize model
Initialize word count
Initialize prompt
Initialize epoch timestamp
Initialize payload
OpenAI


For each
Set payload
Initialize response
Initialize run id
Response
Send Logs to Datadog
Send Metric to Datadog
LogicAppREST

DATADOG



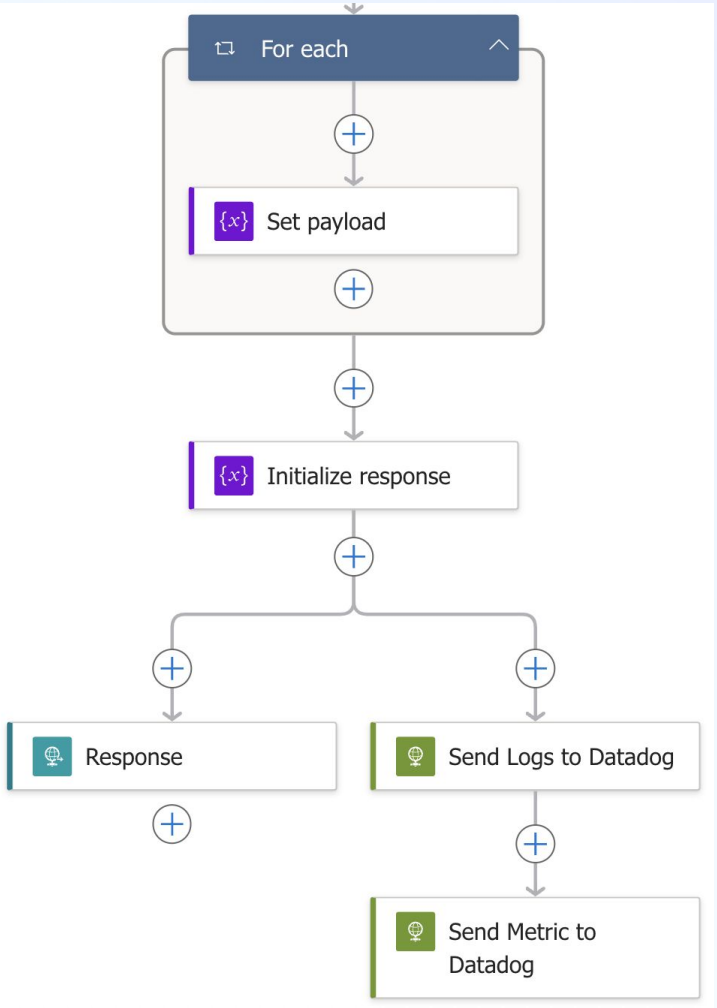
HTTP Request

Create variables

- Model
- Word count
- Prompt
- Timestamp
- Payload

Call OpenAI

DATADOG



Put openai response in variable

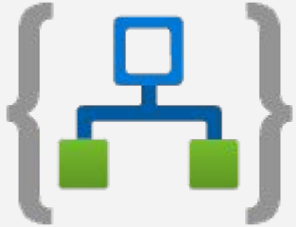
Create response variable

Respond to User

Send Logs & Metrics to Datadog

DATADOG

Live Demo
(Low / No Code + Generative AI + Datadog)

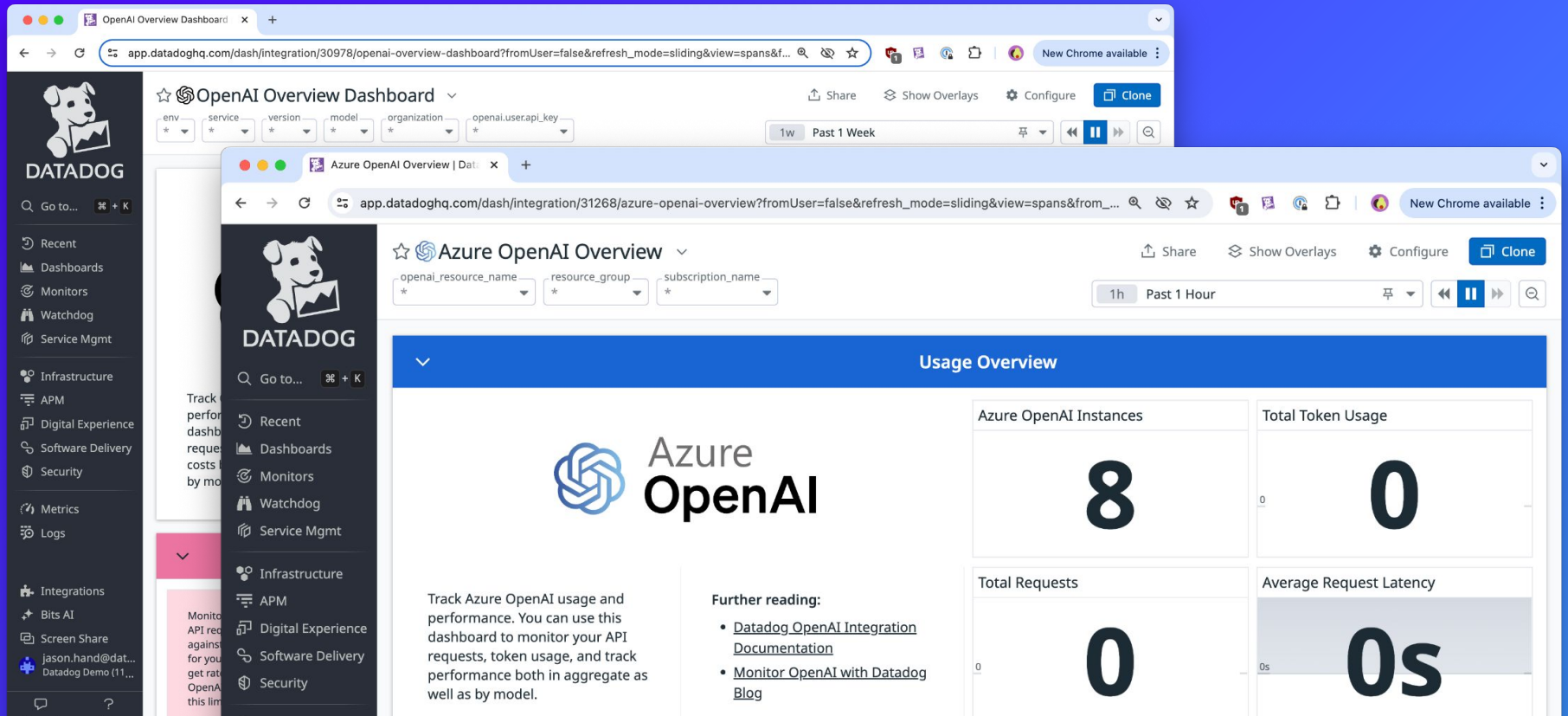


Azure Logic Apps

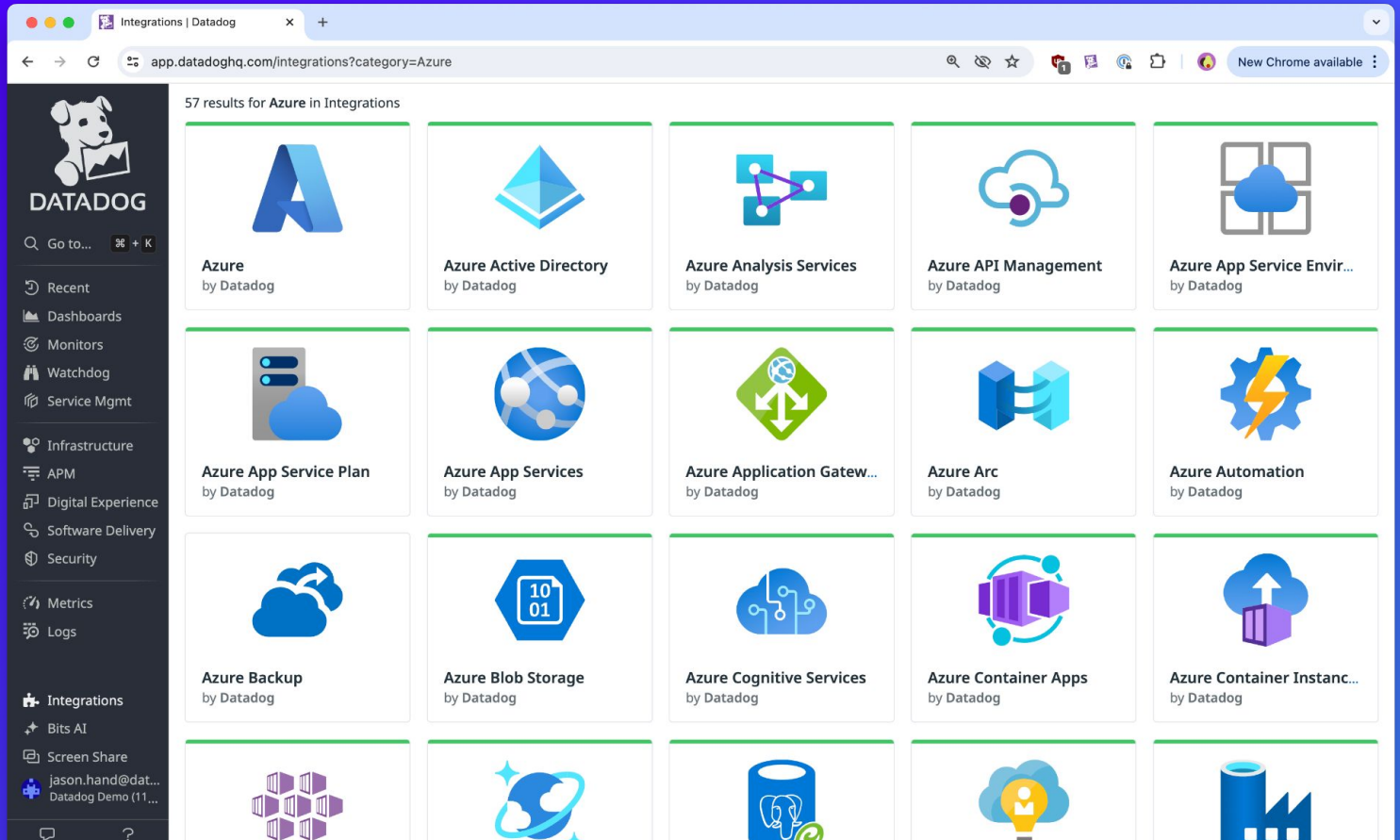
Let’s look at the Logic App inside Azure

DATADOG

What can be collected from generative AI integrations

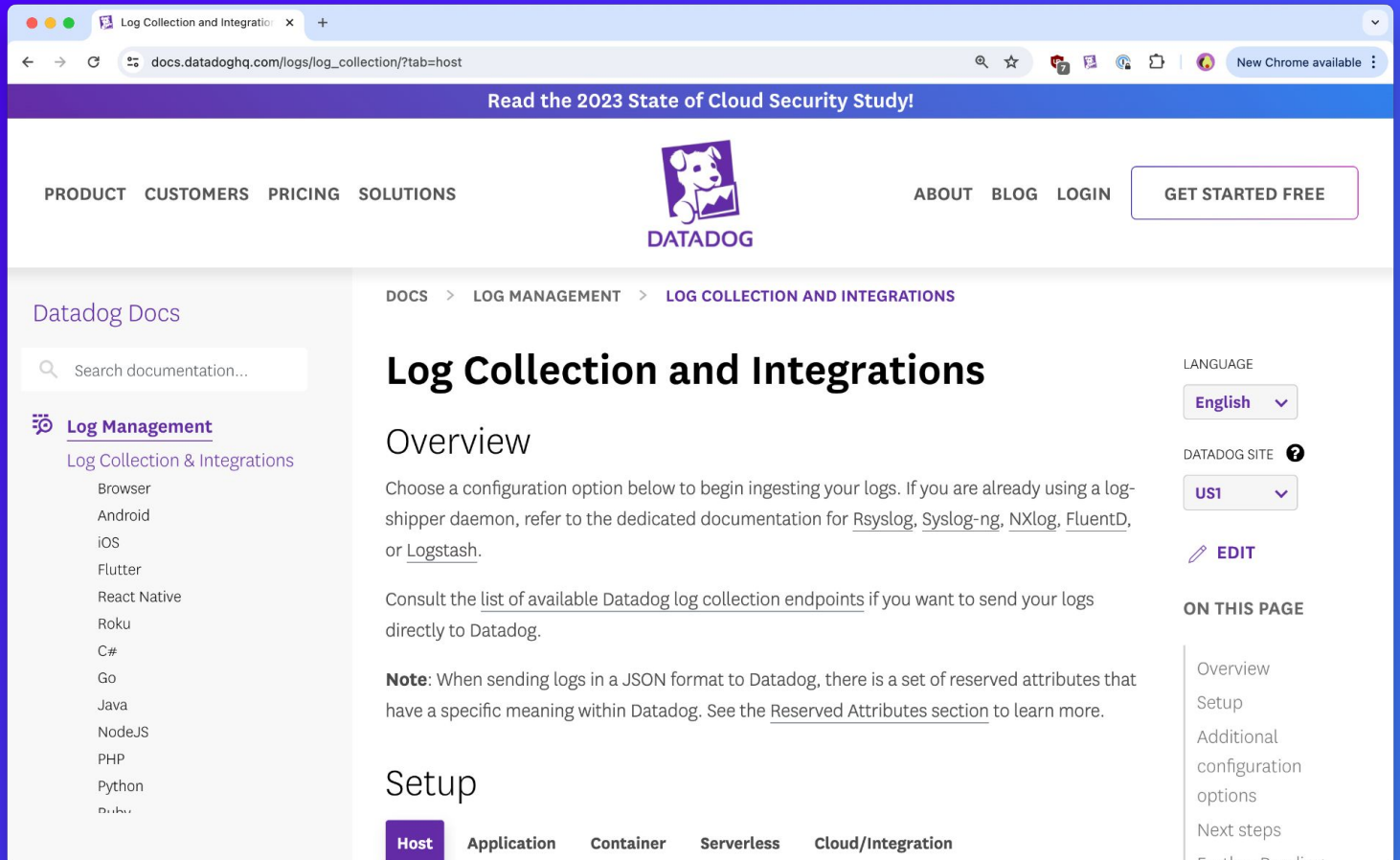


Azure Monitoring & Integrations

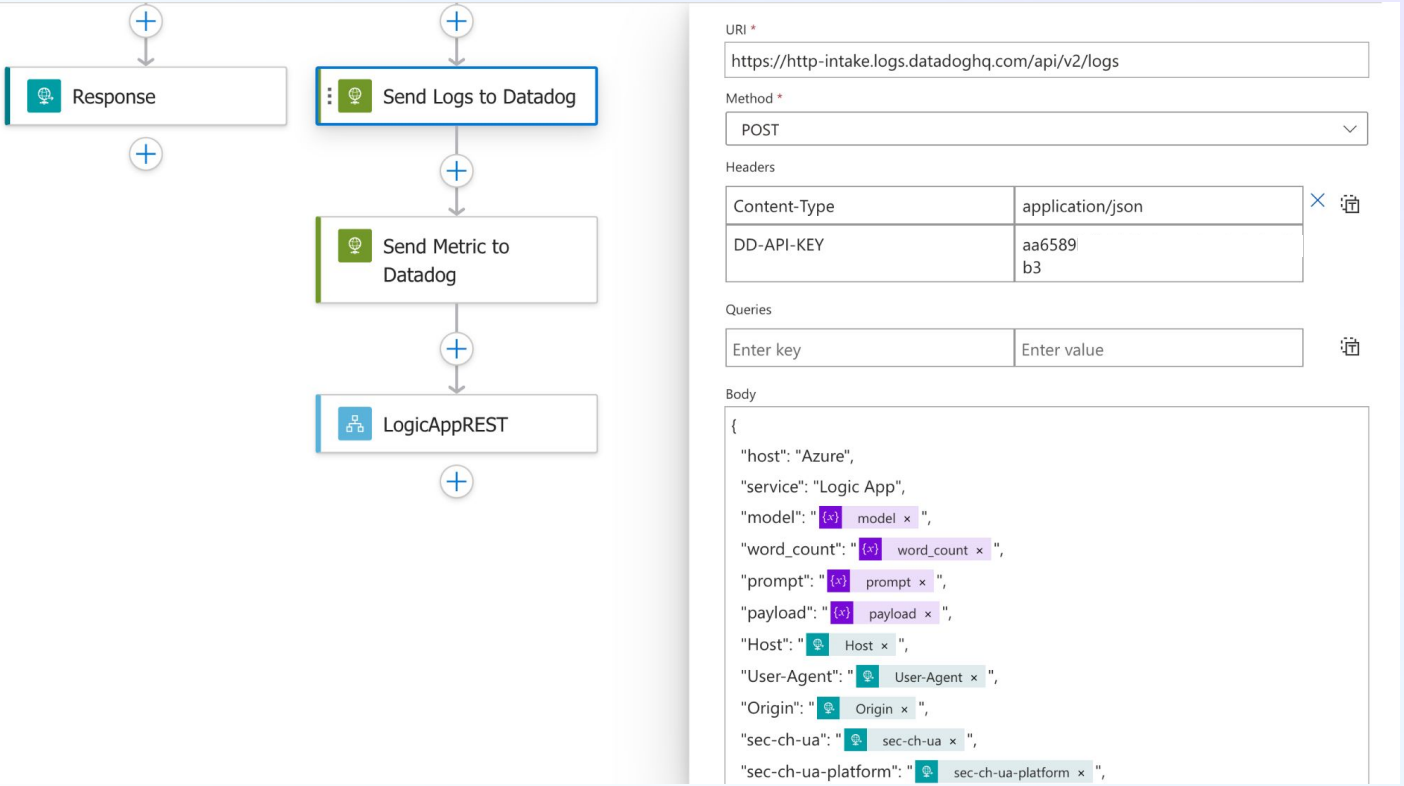


DATADOG

How to send logs to Datadog from Azure Logic Apps



DATADOG



Action:
Send Logs to Datadog

NOTE:
You will need a Datadog API Key.

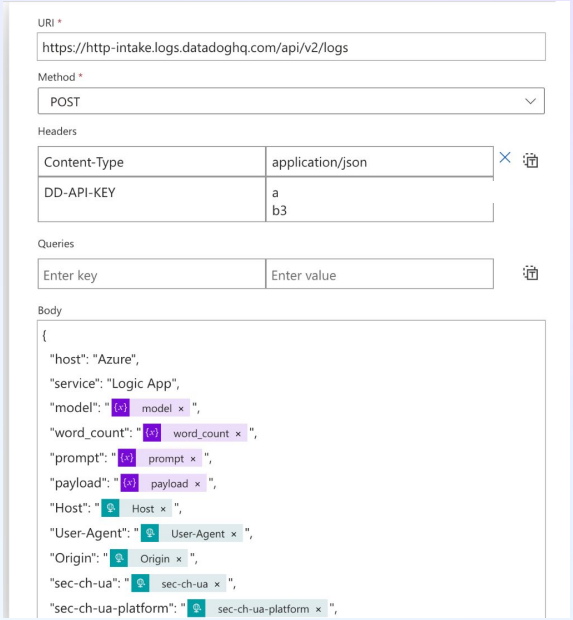
DATADOG

Action:
Send Logs to Datadog

Logs API Endpoint URL:

https://http-intake.logs.datadoghq.com/api/v2/logs

JSON sent to Datadog
Use the Body below

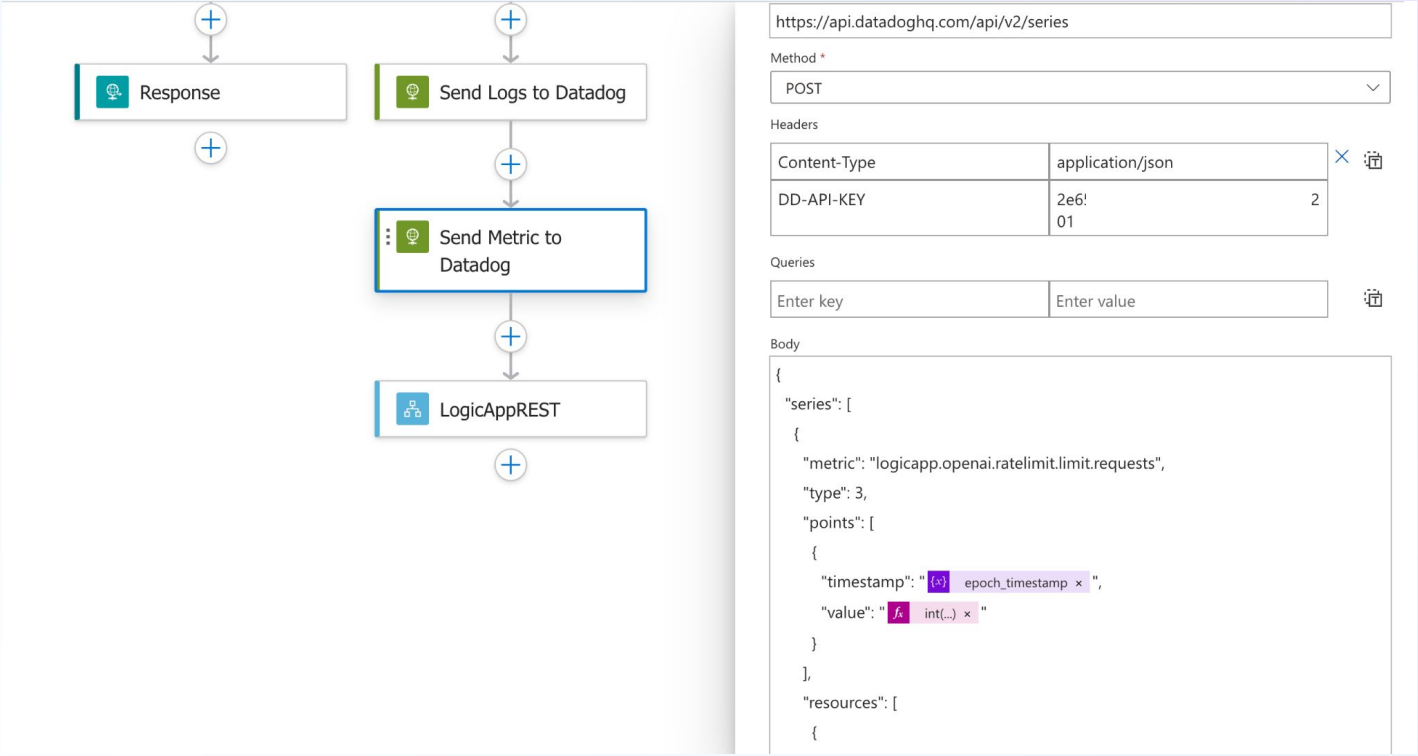


Action:
Send Logs to Datadog

```
{
  "service": "Logic App",
  "model": "@{variables('model')}",
  "word_count": "@{variables('word_count')}",
  "prompt": "@{variables('prompt')}",
  "payload": "@{variables('payload')}",
  "Host": "@{triggerOutputs()?['headers']?['Host']}",
  "User-Agent": "@{triggerOutputs()?['headers']?['User-Agent']}",
  "Origin": "@{triggerOutputs()?['headers']?['Origin']}",
  "sec-ch-ua": "@{triggerOutputs()?['headers']?['sec-ch-ua']}",
  "sec-ch-ua-platform": "@{triggerOutputs()?['headers']?['sec-ch-ua-platform']}",
  "CLIENT-IP": "@{triggerOutputs()?['headers']?['CLIENT-IP']}",
  "DISGUISED-HOST": "@{triggerOutputs()?['headers']?['DISGUISED-HOST']}",
  "Content-Length": "@{triggerOutputs()?['headers']?['Content-Length']}"
}
```

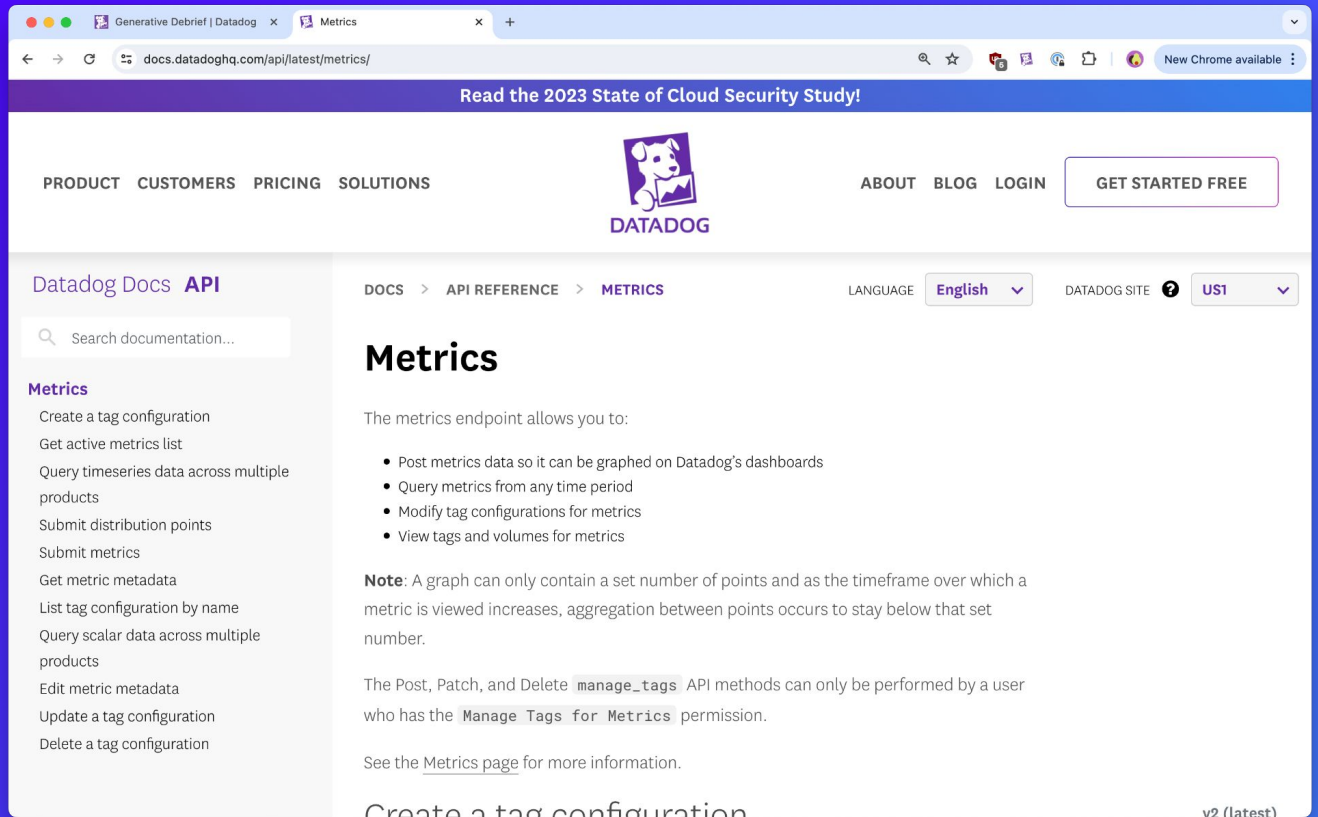
DATADOG

Action: Send Metrics to Datadog



DATADOG

How to send metrics to Datadog from Azure Logic Apps



DATADOG

Action:
Send Metrics
to Datadog

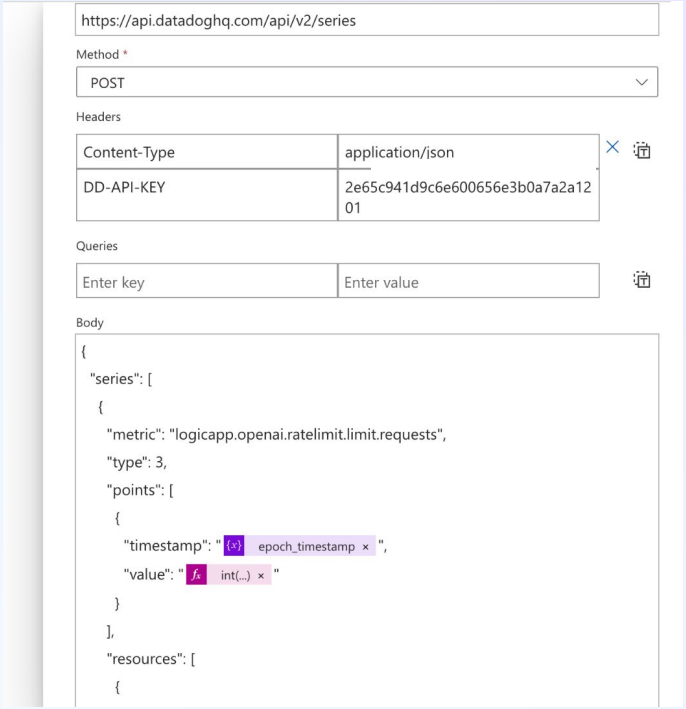
Metrics API Endpoint URL:

https://api.datadoghq.com/api/v2/series

```
{
  "series": [
    {
      "metric": "logicapp.openai.ratelimit.limit.requests",
      "type": 3,
      "points": [
        {
          "timestamp": "@{variables('epoch_timestamp')}",
          "value": "@{int(outputs('OpenAI')['headers']['x-ratelimit-limit-requests'])}"
        }
      ],
      "resources": [
        {
          "name": "generativedebrief",
          "type": "logicapp"
        }
      ],
      "tags": [
        "@{outputs('OpenAI')['headers']['openai-model']}",
        "openai-call"
      ]
    },
    {
      "metric": "logicapp.openai.ratelimit.limit.tokens",
      "type": 3,
      "points": [
        {
          "timestamp": "@{variables('epoch_timestamp')}",
          "value": "@{int(outputs('OpenAI')['headers']['x-ratelimit-limit-tokens'])}"
        }
      ],
      "resources": [
        {
          "name": "generativedebrief",
          "type": "logicapp"
        }
      ],
      "tags": [
        "@{outputs('OpenAI')['headers']['openai-model']}",
        "openai-call"
      ]
    },
```

Part 1

Action:
Send Metrics to Datadog

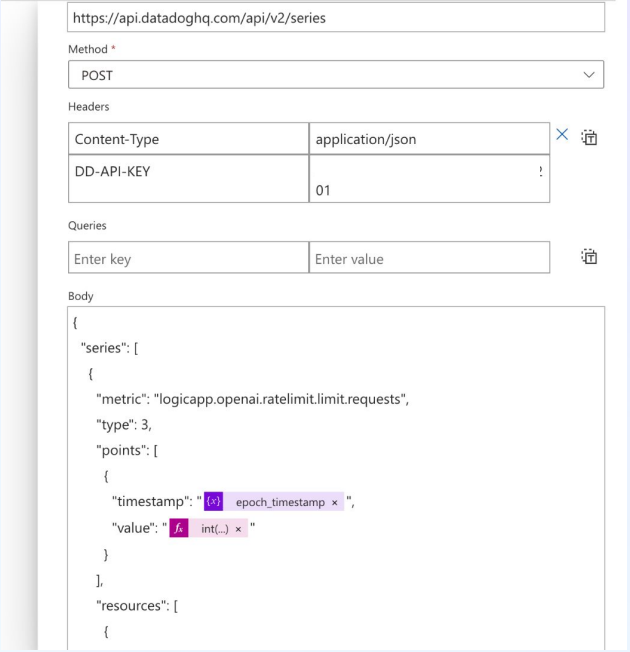


DATADOG

```
{
      "metric": "logicapp.openai.ratelimit.remaining.requests",
      "type": 3,
      "points": [
        {
          "timestamp": "@{variables('epoch_timestamp')}",
          "value": "@{int(outputs('OpenAI')['headers']['x-ratelimit-remaining-requests'])}"
        }
      ],
      "resources": [
        {
          "name": "generativedebrief",
          "type": "logicapp"
        }
      ],
      "tags": [
        "@{outputs('OpenAI')['headers']['openai-model']}",
        "openai-call"
      ]
    },
    {
      "metric": "logicapp.openai.ratelimit.remaining.tokens",
      "type": 3,
      "points": [
        {
          "timestamp": "@{variables('epoch_timestamp')}",
          "value": "@{int(outputs('OpenAI')['headers']['x-ratelimit-remaining-tokens'])}"
        }
      ],
      "resources": [
        {
          "name": "generativedebrief",
          "type": "logicapp"
        }
      ],
      "tags": [
        "@{outputs('OpenAI')['headers']['openai-model']}",
        "openai-call"
      ]
    }
  ]
}
```

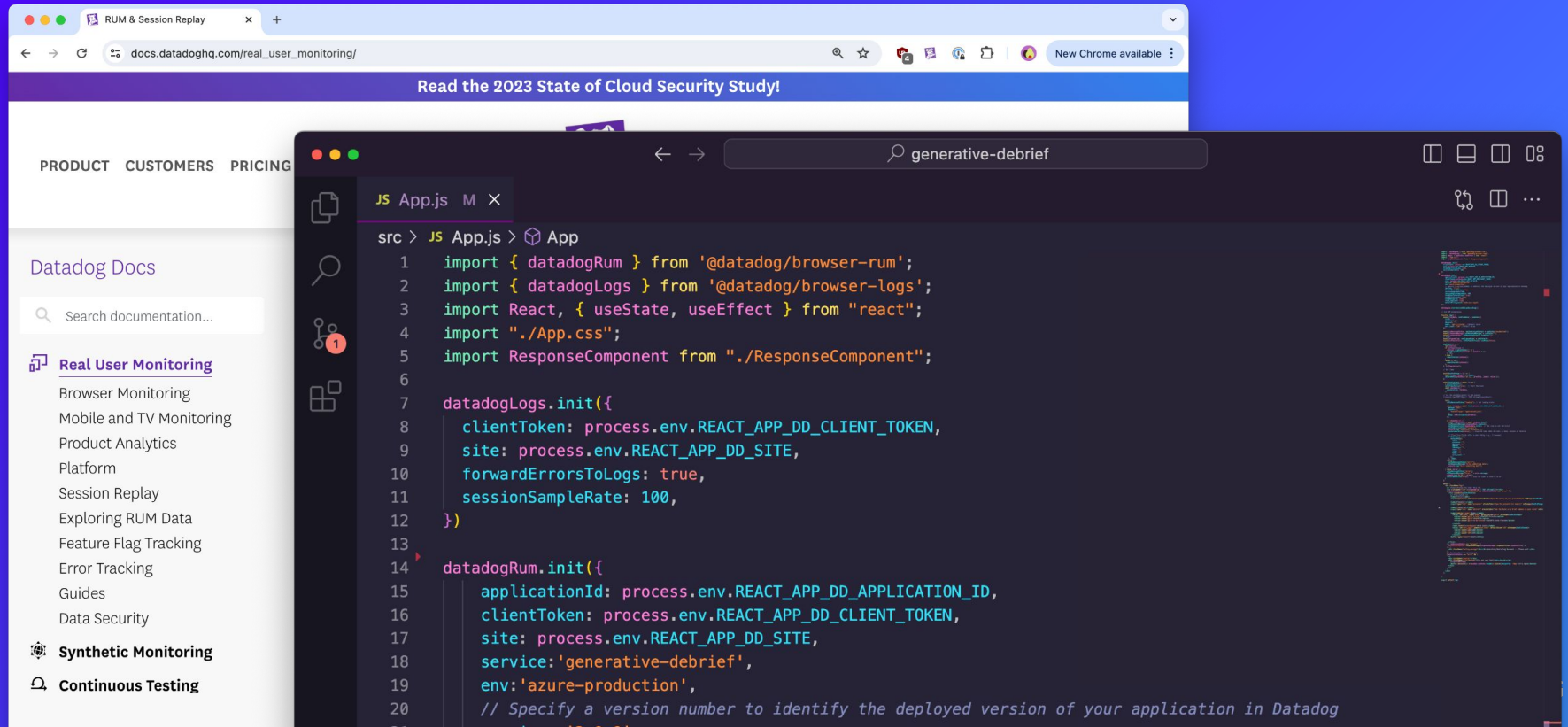
Action: Send Metrics to Datadog

Part 2

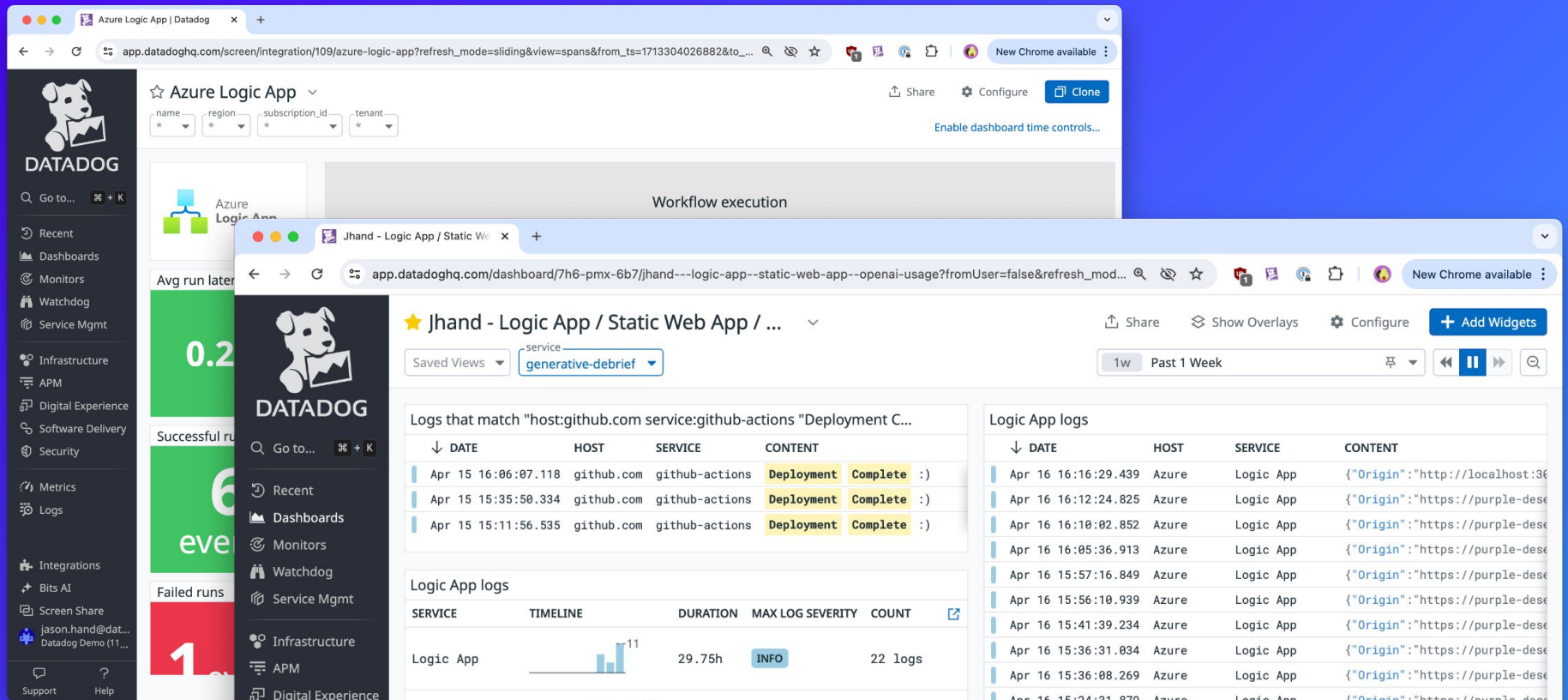


DATADOG

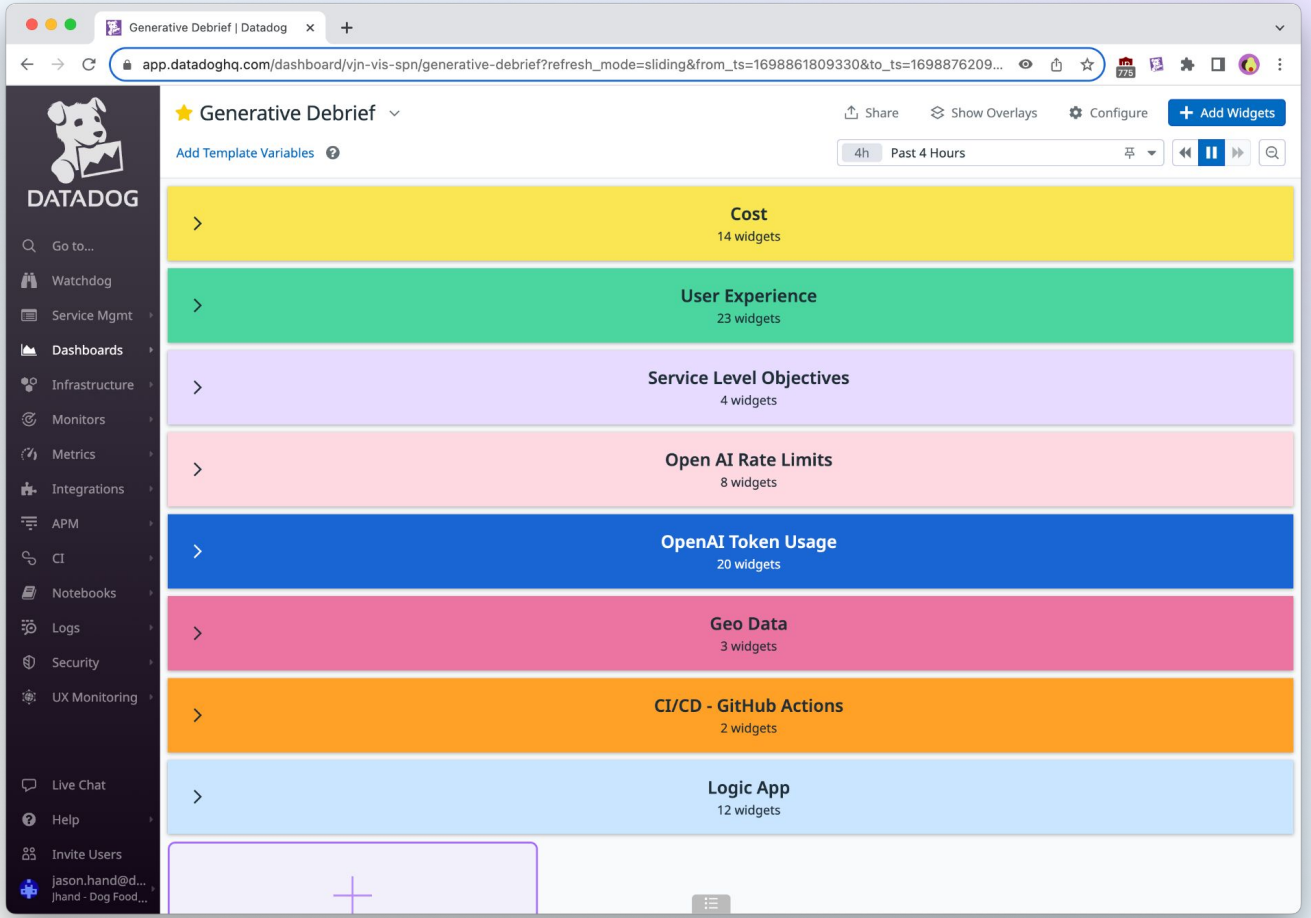
How to add real user monitoring (RUM) to a web application



How to visualize Azure Logic App, OpenAI, and web app data in Datadog



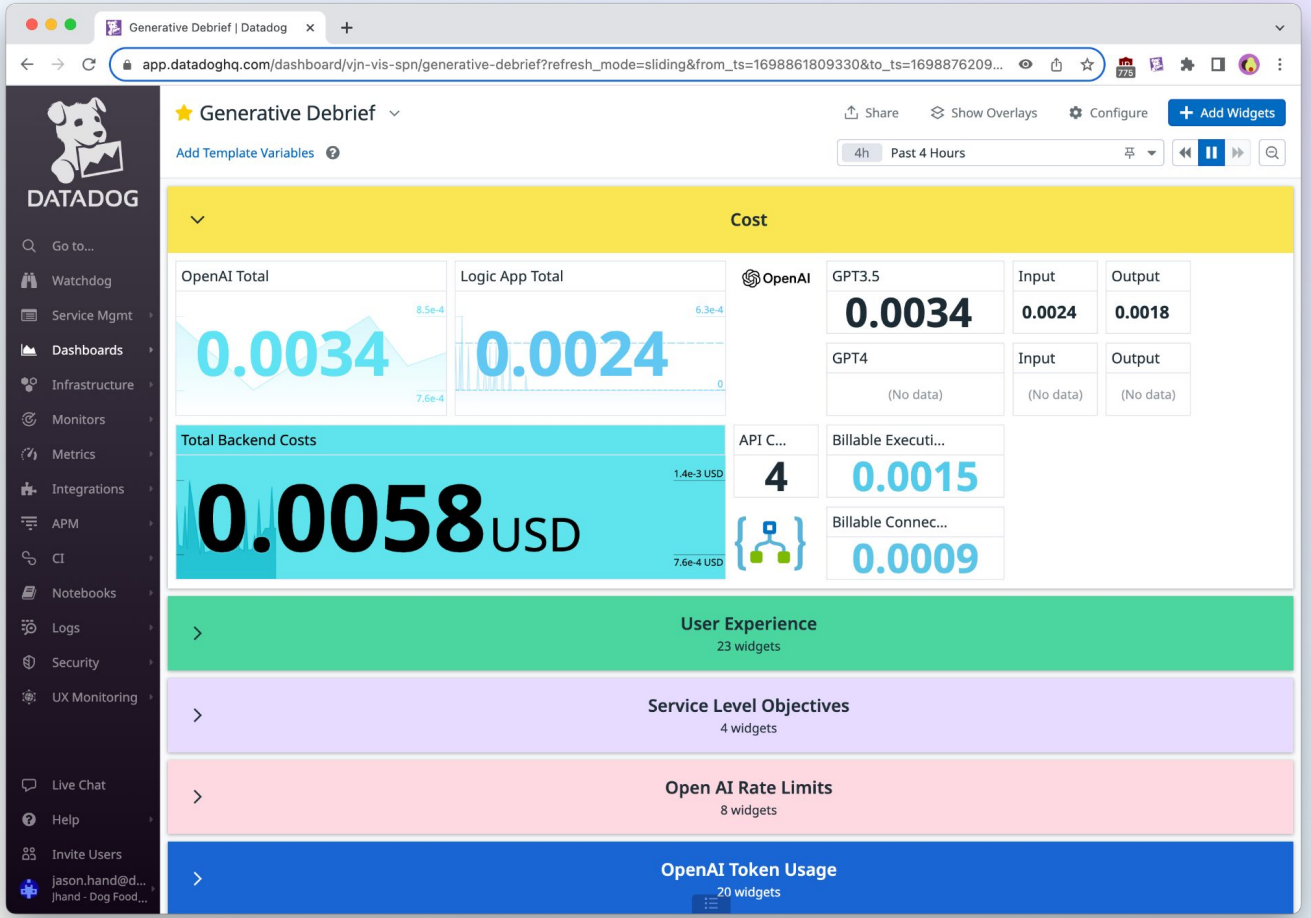
Datadog Dashboards



Dashboard Groups

- Cost
- User Experience
- Service Level Objectives
- OpenAI Rate Limits
- OpenAI Token Usage
- Geo Data
- CI/CD - GitHub Actions
- Logic App

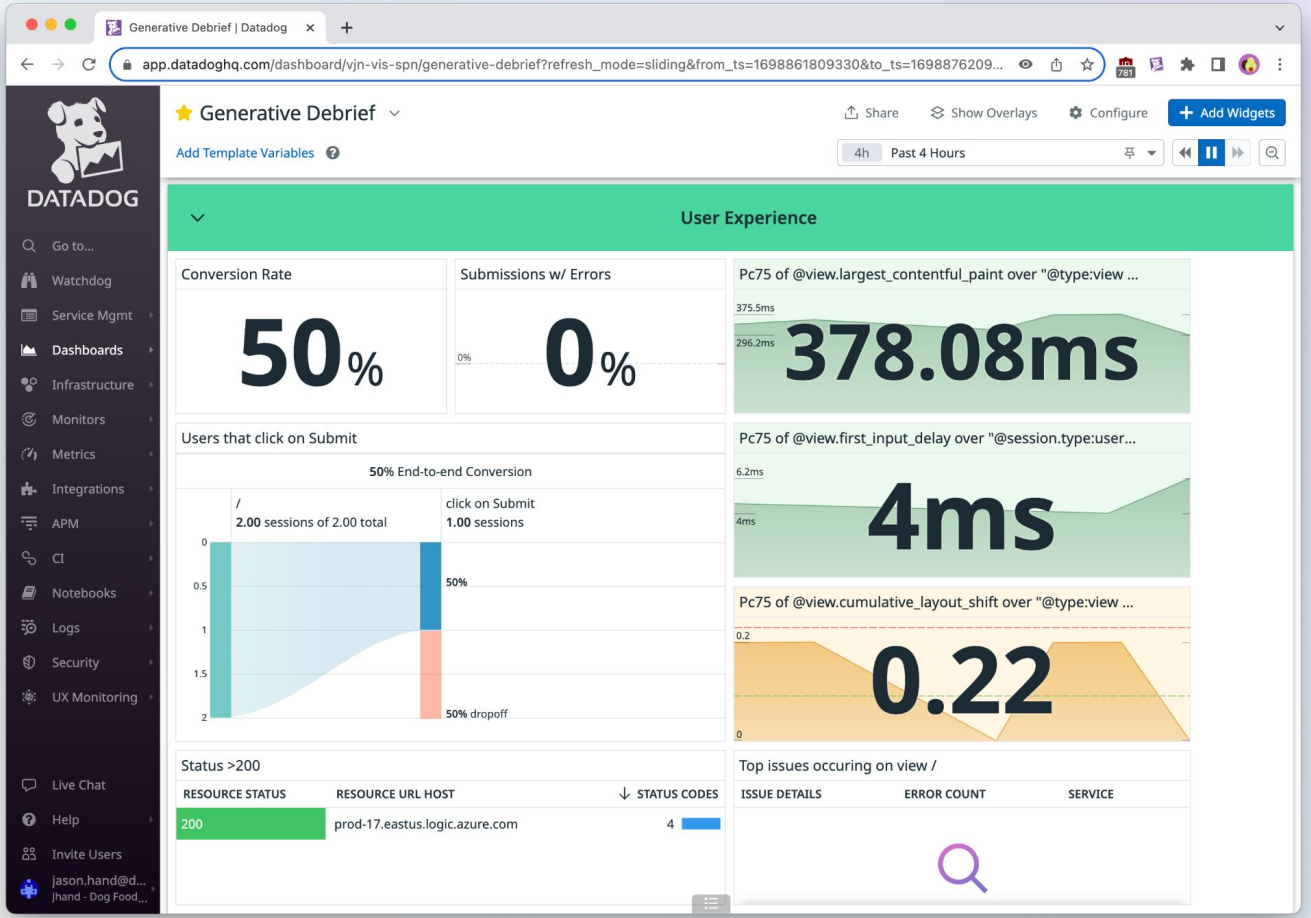
DATADOG



Operational Costs

- Total Backend Costs
- OpenAI Total Cost
- Logic App Total Cost
- Breakdown of model costs
- Breakdown of billable Logic App executions

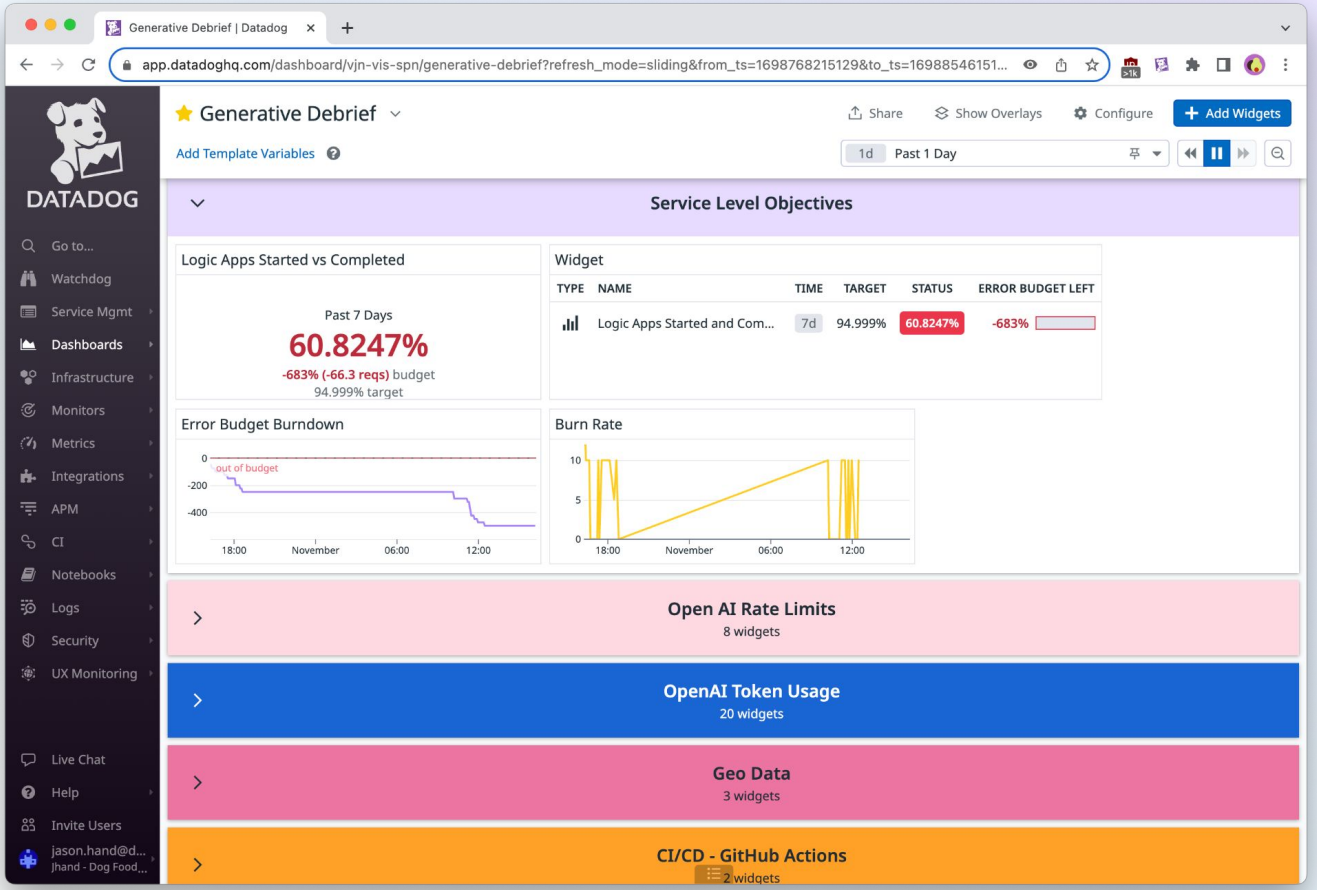
DATADOG



User Experience

- Conversion Rate
- Submission Errors
- “Submit” count
- Pc75 - content pain
- Pc75 - input delay
- Pc75 - cumulative layout shift
- API call responses
- Errors

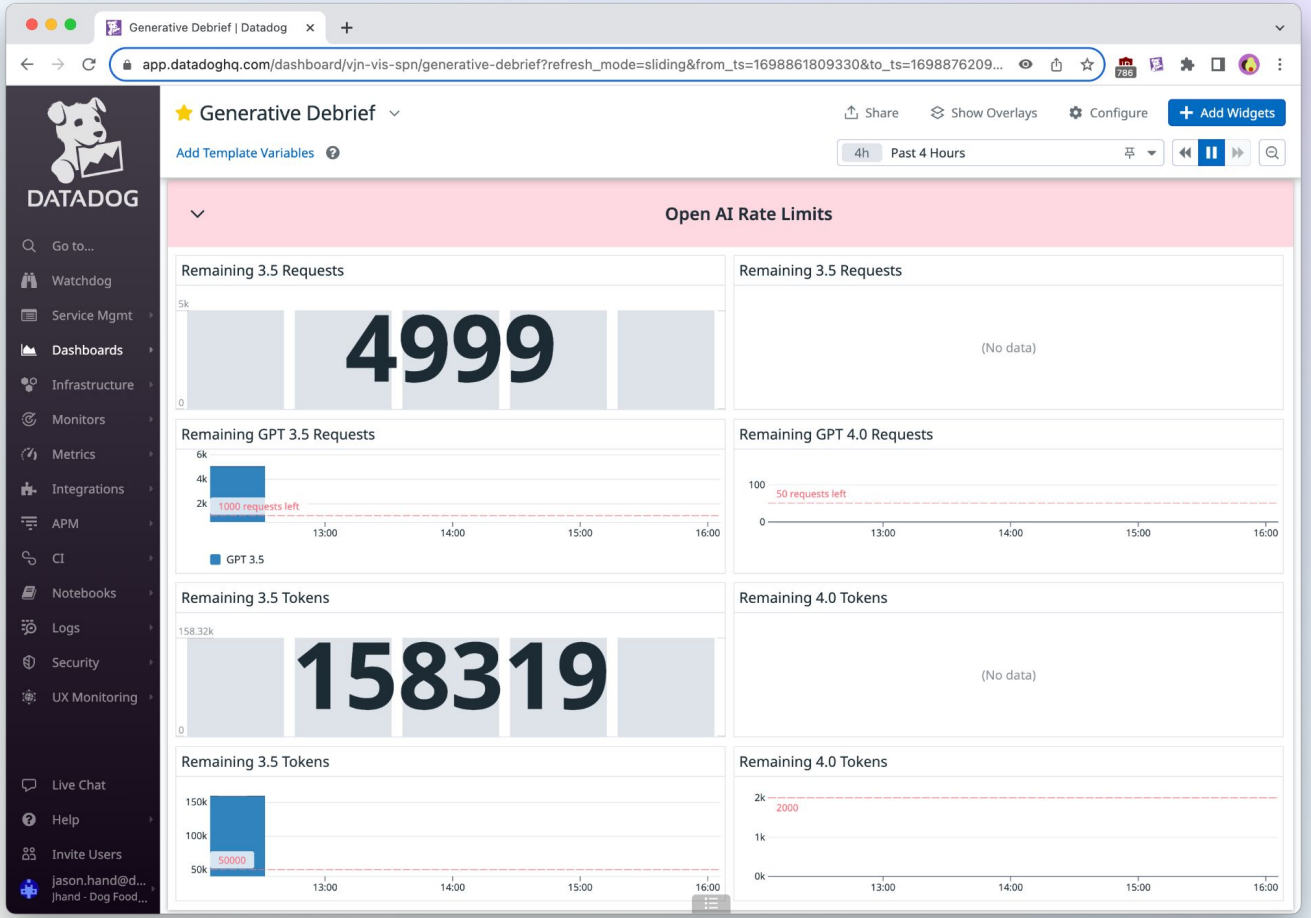
DATADOG



Service Level Objectives

- Error Budget Burndown
- Burn Rate

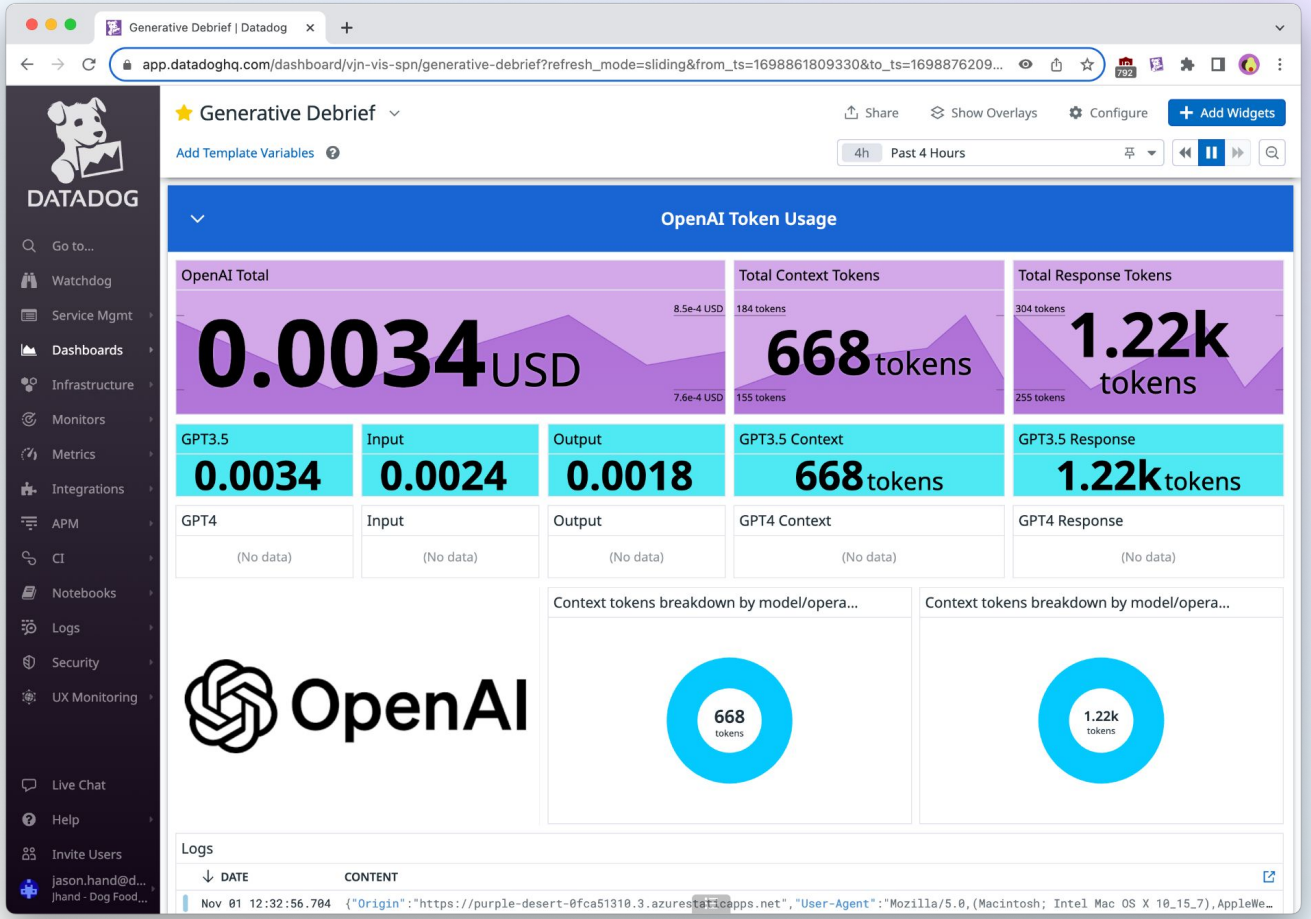
DATADOG



OpenAI Rate Limits

- Remaining Tokens
- Remaining Requests

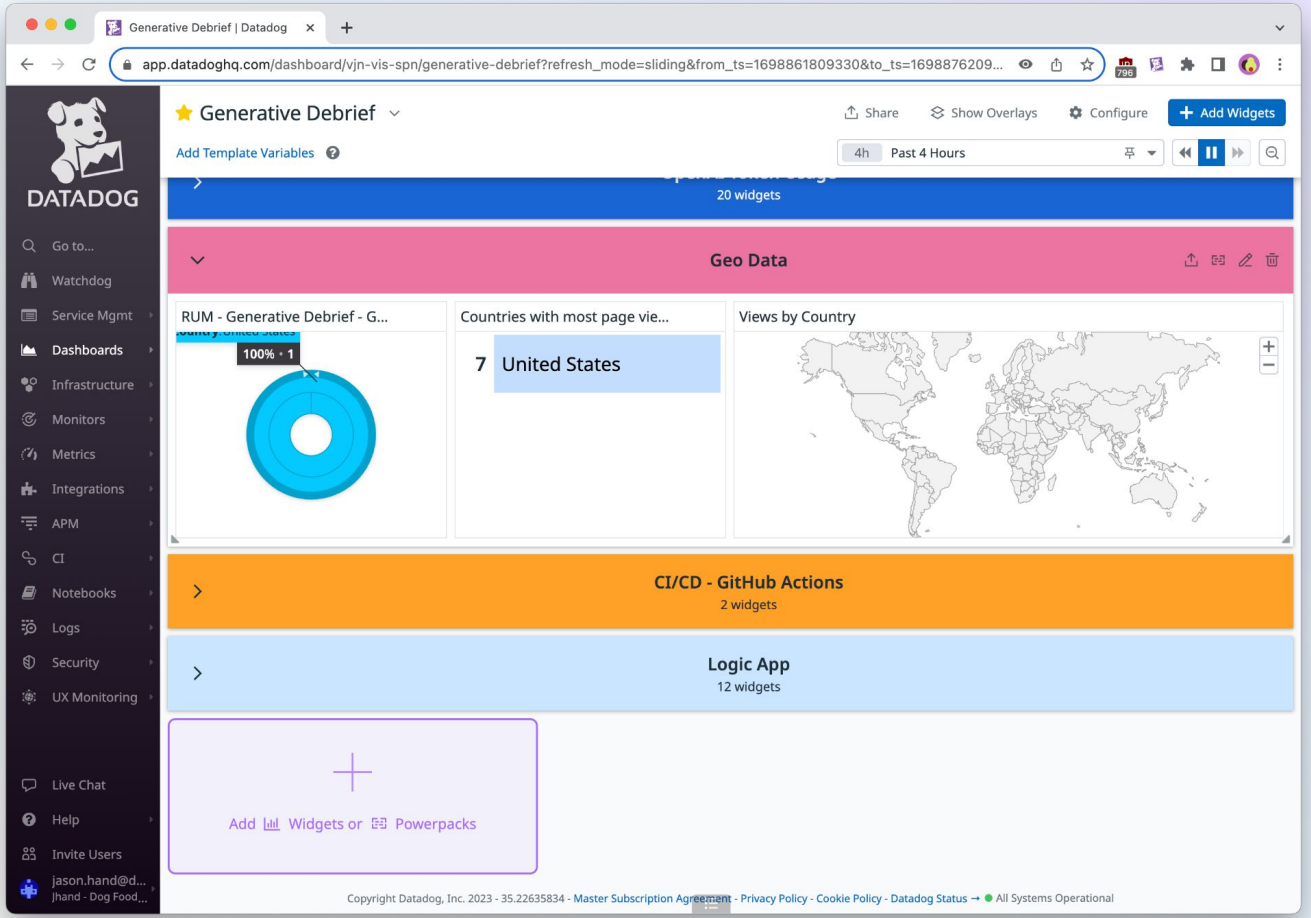
DATADOG



OpenAI token Usage

- Total cost
- Total context tokens
- Total response tokens
- Breakdown of model usage

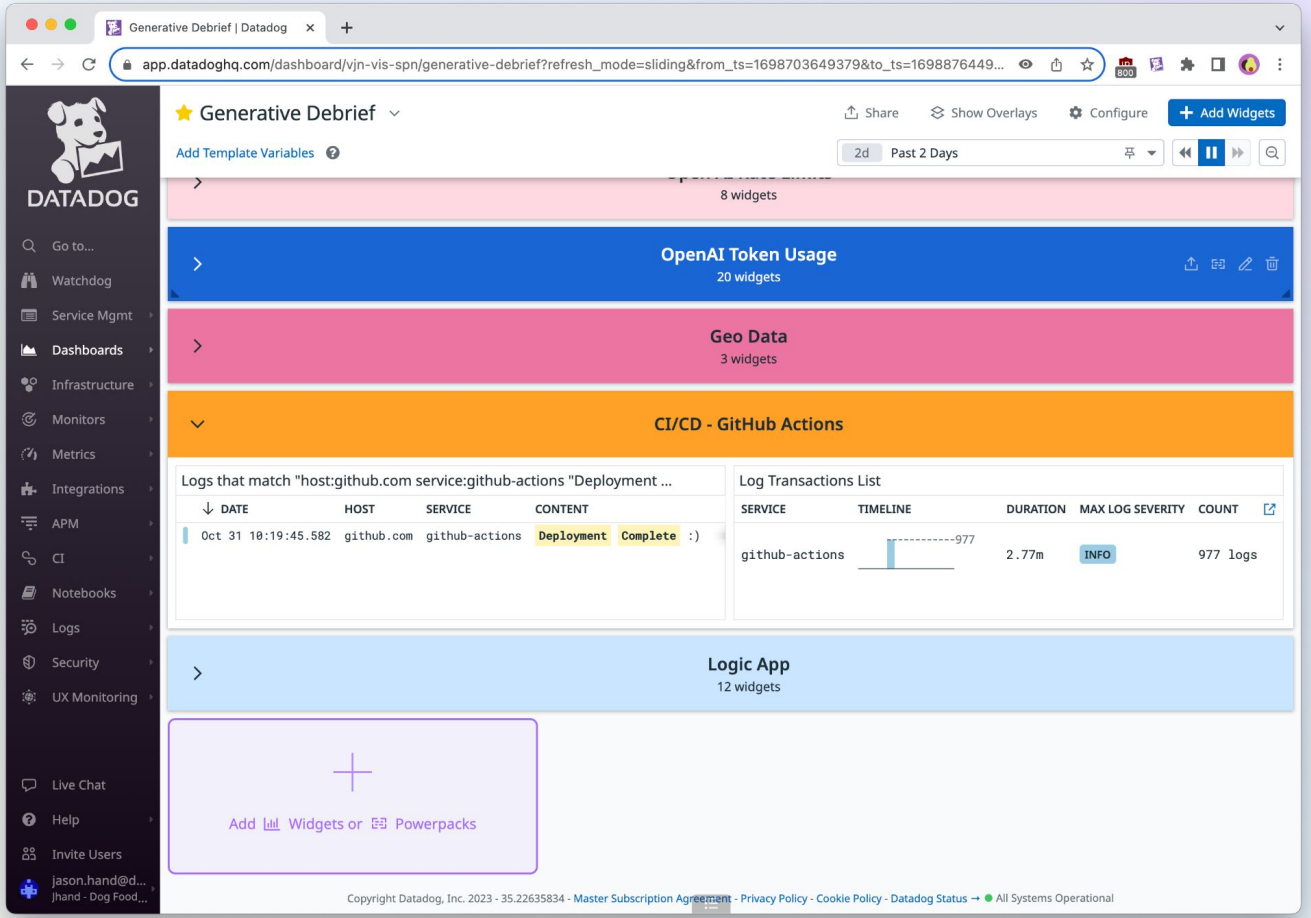
DATADOG



Geo data

- RUM - page views
- RUM - location

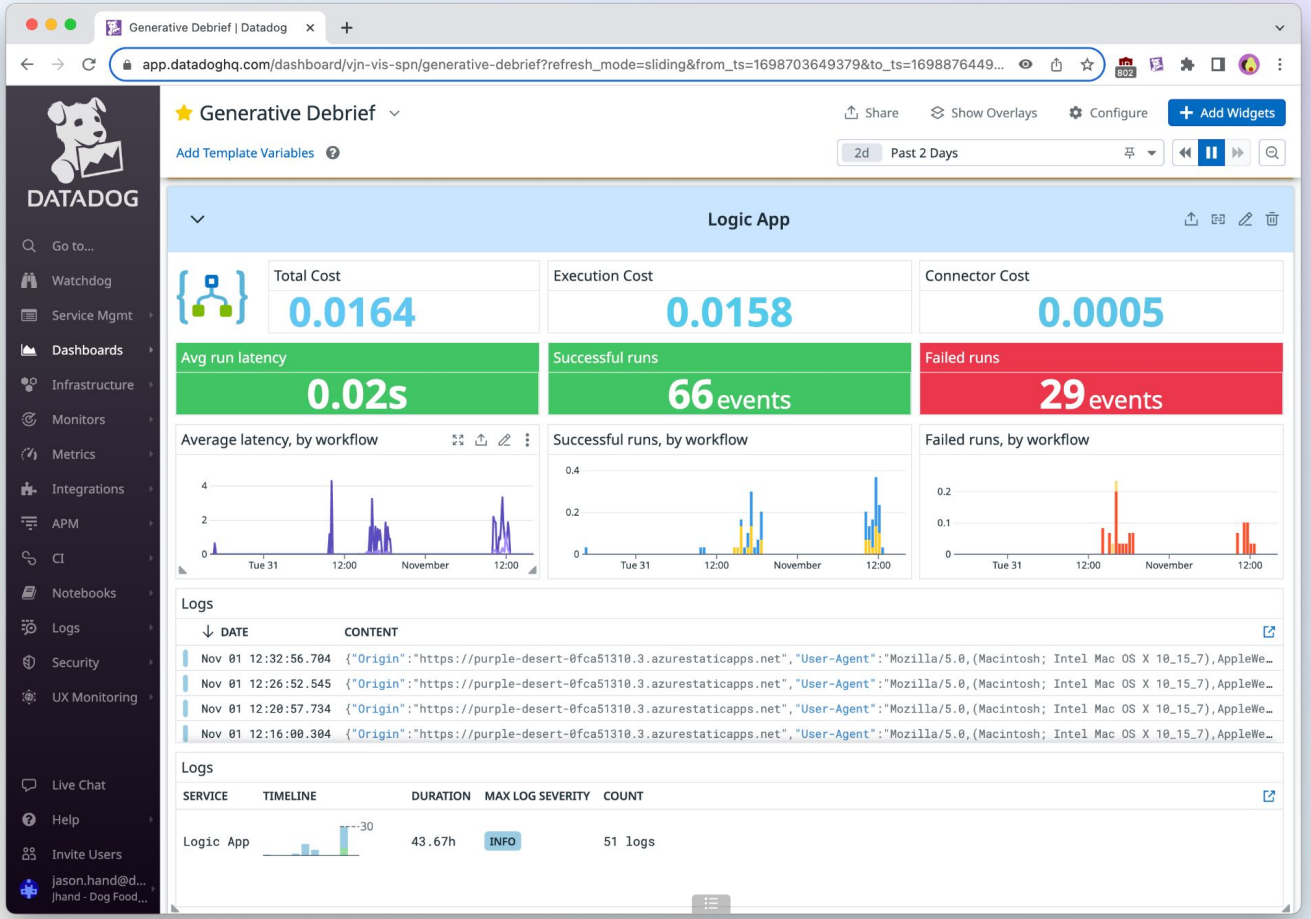
DATADOG



CI/CD - GitHub Actions

- Logs - “Deployment Complete”

DATADOG



Logic App

- Total cost
- Latency
- Successful runs
- Failed runs
- Logs

DATADOG

Takeaways

- Integrations for Low Code tools (like Azure Logic Apps)
- Integrations for Generative tools (such as OpenAI)
- Augment with Logs & Metrics
- Measure customer experience (RUM)
- Visualize, monitor, and alert on what is important
- Continue to learn and experiment

Thank You

DATADOG

Resources

- OpenAI Integration
- OpenAI Integration Blog
- OpenAI Monitoring
- Azure Integration
- Logic App Integration Docs
- Logic App Integration Blog
- RUM Documentation
- Azure OpenAI Integration Docs
- CI Pipelines
- Monitor your GitHub Actions workflows with Datadog CI Visibility
- GitHub Integration