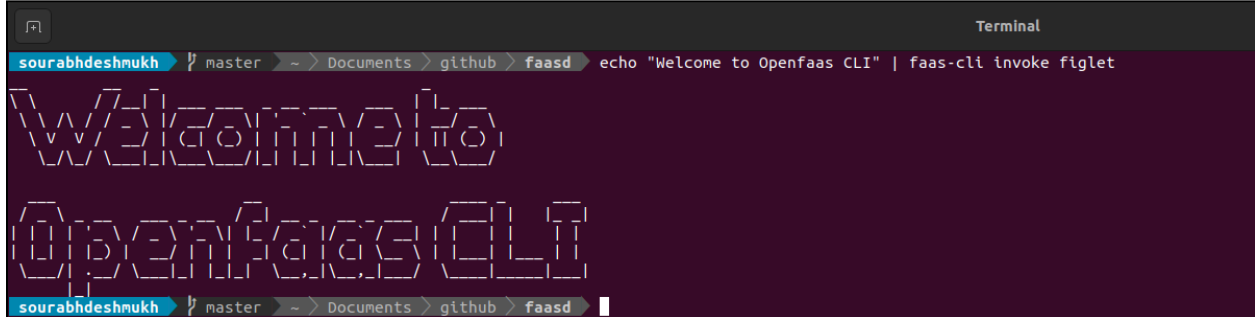


HOMEWORK 2

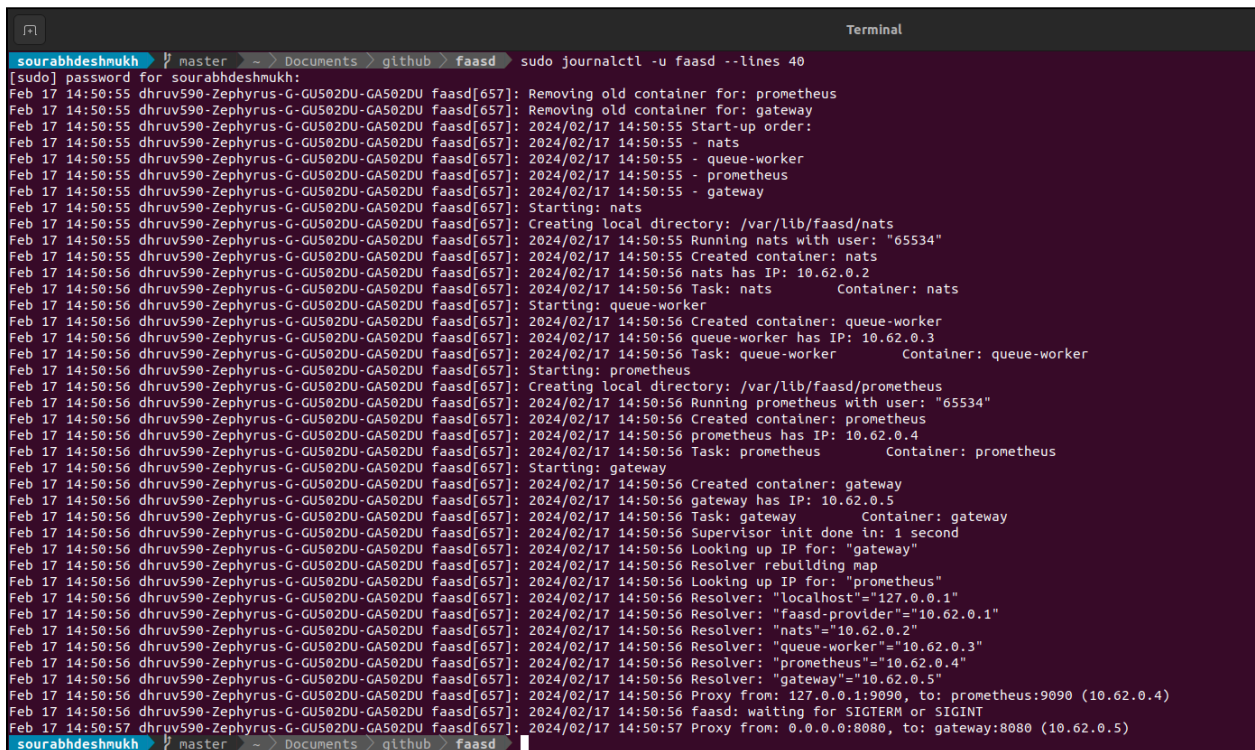
1 Provide a screenshot of invoking the figlet function.



```
sourabhdeshmukh master ~ Documents github faasd echo "Welcome to Openfaas CLI" | faas-cli invoke figlet
Welcome to
Openfaas CLI
sourabhdeshmukh master ~ Documents github faasd
```

2. Provide a screenshot of running the following command.

```
sourabhdeshmukh$ sudo journalctl -u faasd --lines 40
```



```
sourabhdeshmukh master ~ Documents github faasd sudo journalctl -u faasd --lines 40
[sudo] password for sourabhdeshmukh:
Feb 17 14:50:55 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: Removing old container for: prometheus
Feb 17 14:50:55 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: Removing old container for: gateway
Feb 17 14:50:55 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:55 Start-up order:
Feb 17 14:50:55 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:55 - nats
Feb 17 14:50:55 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:55 - queue-worker
Feb 17 14:50:55 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:55 - prometheus
Feb 17 14:50:55 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:55 - gateway
Feb 17 14:50:55 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: Starting: nats
Feb 17 14:50:55 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: Creating local directory: /var/lib/faasd/nats
Feb 17 14:50:55 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:55 Running nats with user: "65534"
Feb 17 14:50:55 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:55 Created container: nats
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 nats has IP: 10.62.0.2
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 Task: nats Container: nats
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: Starting: queue-worker
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 Created container: queue-worker
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 queue-worker has IP: 10.62.0.3
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 Task: queue-worker Container: queue-worker
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: Starting: prometheus
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: Creating local directory: /var/lib/faasd/prometheus
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 Running prometheus with user: "65534"
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 Created container: prometheus
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 prometheus has IP: 10.62.0.4
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 Task: prometheus Container: prometheus
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: Starting: gateway
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 Created container: gateway
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 gateway has IP: 10.62.0.5
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 Task: gateway Container: gateway
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 Supervisor init done in: 1 second
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 Looking up IP for: "gateway"
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 Resolver rebuilding map
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 Looking up IP for: "prometheus"
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 Resolver: "localhost"="127.0.0.1"
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 Resolver: "faasd-provider"="10.62.0.1"
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 Resolver: "nats"="10.62.0.2"
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 Resolver: "queue-worker"="10.62.0.3"
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 Resolver: "prometheus"="10.62.0.4"
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 Resolver: "gateway"="10.62.0.5"
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 Proxy from: prometheus:9090 (10.62.0.4)
Feb 17 14:50:56 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:56 faasd: waiting for SIGTERM or SIGINT
Feb 17 14:50:57 dhruv590-Zephyrus-G-GUS02DU-GAS02DU faasd[657]: 2024/02/17 14:50:57 Proxy from: 0.0.0.0:8080, to: gateway:8080 (10.62.0.5)
sourabhdeshmukh master ~ Documents github faasd
```

Complete slack-request/handler.py

<https://github.com/sourabhdeshmukh/Cloud-CSEN-241/blob/main/Assignments/HW2/functions/slack-request/handler.py>

```
import json

def handle(req):
    data = {
        "text": "Serverless Message",
        "attachments": [{
            "title": "The Awesome world of Cloud Computing! COEN 241",
            "fields": [{
                "title": "Amazing Level",
                "value": "100",
                "short": True
            }],
            "author_name": "Sourabh Deshmukh",
            "author_icon": "https://github.com/sourabhdeshmukh",
            "image_url":
"https://hub.docker.com/r/sourabhdeshmukh/slack-request"
        },
        {
            "title": "About COEN 241",
            "text": "COEN 241 is the most awesome class ever!."
        },
        {
            "fallback": "Would you recommend COEN 241 to your friends?",
            "title": "Would you recommend COEN 241 to your friends?",
            "callback_id": "response123",
            "color": "#3AA3E3",
            "attachment_type": "default",
            "actions": [
                {
                    "name": "recommend",
                    "text": "Of Course!",
                    "type": "button",
                    "value": "recommend"
                },
                {
                    "name": "definitely",
                    "text": "Most Definitely!",
```

```
        "type": "button",
        "value": "definitely"
    }
]
}]
}
return json.dumps(data)
```

3. Complete slack-interactive/handler.py

<https://github.com/sourabhdeshmukh/Cloud-CSEN-241/blob/main/Assignments/HW2/functions/slack-interactive/handler.py>

```
import json
import urllib
def handle(req):
    urlstring = urllib.unquote(req).decode('utf8').strip('payload=')
    response = json.loads(urlstring)
    data = {
        "attachments": [
            {
                "replace_original": True,
                "response_type": "ephemeral",
                "fallback": "Required plain-text summary of the
attachment.",
                "color": "#36a64f",
                "pretext": "Ahh yeah! Great choice, COEN 241 is absolutely
amazing!",
                "author_name": "Sourabh Deshmukh",
                "author_link":
"https://github.com/sourabhdeshmukh/Cloud-CSEN-241",
                "author_icon": "https://github.com/sourabhdeshmukh",
                "title": "COEN 241",
                "title_link":
"https://www.scu.edu/engineering/academic-programs/departments-of-computer-
engineering/graduate/course-descriptions/",
                "text": "Head over to COEN 241",
                "image_url":
"https://www.scu.edu/media/offices/umc/scu-brand-guidelines/visual-identit
```

Sourabh Deshmukh
W1648445

```
y-amp-photography/visual-identity-toolkit/logos-amp-seals/Mission-Dont3.png",
    "thumb_url":
"https://www.scu.edu/engineering/academic-programs/department-of-computer-
engineering/graduate/course-descriptions/",
    "footer": "Slack Apps built on OpenFaas",
    "footer_icon":
"https://a.slack-edge.com/45901/marketing/img/_rebrand/meta/slack_hash_256
.png",
    "ts": 123456789
}
]
}
return json.dumps(data)
```

4. Provide a screenshot of your OpenFaaS gateway AFTER deploying figlet, slack-handler and slack-interactive functions

The screenshot shows the OpenFaaS Portal interface. On the left, a sidebar lists functions: 'figlet', 'slack-interactive' (selected), and 'slack-request'. The main panel displays details for the 'slack-interactive' function, including its status (Ready), replicas (1), invocation count (3), image (docker.io/sourabhdeshmukh/slack-interactive:latest), and URL (http://localhost:8080/function/slack-interactive). Below this, the 'Invoke function' section shows a request body of {"testing": "slack-interactive"} and a response body containing a JSON object with Slack message fields like footer, author_link, color, text, title, ts, author_name, title_link, and image_url.

Sourabh Deshmukh
W1648445

5. Provide a screenshot of invoking slack-request and slack-interactive functions

Slack request

The screenshot shows the OpenFaas web interface in a browser. On the left sidebar, the 'slack-request' function is selected. The main panel displays the function's status as 'Ready' with 1 replica and 1 invocation. The image is 'docker.io/sourabhdeshmukh/slack-request:latest' and the URL is 'http://localhost:8080/function/slack-request'. The function process is 'python index.py'. Below this, the 'Invoke function' section shows the 'Text' radio button selected. The request body is 'testing slack-request function'. The response status is 200 with a round-trip time of 0.105s. The response body is a JSON object:

```
{  "text": "Serverless Message",  "attachments": [    {      "fields": [        {          "short": true,          "value": "100",          "title": "Amazing Level"        }      ]    }  ]}
```

Slack-interactive

The screenshot shows the OpenFaas web interface in a browser. On the left sidebar, the 'slack-interactive' function is selected. The main panel displays the function's status as 'Ready' with 1 replica and 3 invocations. The image is 'docker.io/sourabhdeshmukh/slack-interactive:latest' and the URL is 'http://localhost:8080/function/slack-interactive'. The function process is 'python index.py'. Below this, the 'Invoke function' section shows the 'JSON' radio button selected. The request body is '{"testing": "slack-interactive"}'. The response status is 200 with a round-trip time of 0.117s. The response body is a JSON object:

```
{  "attachments": [    {      "footer": "Slack Apps built on OpenFaas",      "author_link": "https://github.com/sourabhdeshmukh/Cloud-CSEN-241",      "color": "#36a64f",      "text": "Head over to COEN 241",      "title": "COEN 241",      "ts": 123456789,      "author_name": "Sourabh Deshmukh",      "title_link": "https://www.scu.edu/engineering/academic-programs/department-of-computer-engineering/graduate/course-descriptions/"    }  ]}
```

6. Complete the chatbot with a yml file

```
import requests
import datetime
import random

# Chatbot Handler Function
def handle(req):
    processedText = req.lower().strip()
    preformat = list(processedText.split())

    if "name" in preformat:
        response = get_name()
    elif "time" in preformat:
        response = get_time()
    elif "figlet" in preformat:
        response = get_figlet(req)
    else:
        return "here is a sample questions you can ask.\n\n1. What is your name\n2. what is the current time\n3. figlet generate HI\n"
    return response

# Random name generator for chatbot
def get_name():
    names = ["Assistant", "Alexa", "Siri"]
    return "Hi, I am " + random.choice(names)

# Random format for current date and time generator.
def get_time():
    now = datetime.datetime.now()
    responses = [
        "Current time is {}".format(now.strftime('%I:%M %p')),
        "Current Date is {}".format(now.strftime('%A, %B %d, %Y')),
        "Current Date and Time is {}".format(now.strftime('%A, %B %d, %Y, %I:%M %p')),
    ]
    return random.choice(responses)
```

Sourabh Deshmukh
W1648445

```
# External figlet function calling function.
def get_figlet(req):
    text = req.split("generate", 1)[1].strip()
    url="http://10.62.0.1:8080/function/figlet"
    result = requests.post(url, data=text)
    return result.text
```

7. Provide a screenshot of invoking three different cases of the chatbot
- Chatbot help.

The screenshot shows the OpenFaaS web interface in a browser. The left sidebar contains a search bar and a list of functions: chatbot, figlet, slack-interactive, and slack-request. The main panel displays details for the 'chatbot' function, including its status (Ready), replicas (1), and invocation count (13). Below this, there is an 'Invoke function' section with a radio button selection for 'Text', 'JSON', or 'Download'. The 'Text' option is selected, and the request body is 'help'. The response status is 200, and the round-trip time is 0.935s. The response body contains a list of sample questions: '1. What is your name', '2. what is the current time', and '3. figlet generate HI'.

Status	Replicas	Invocation count
Ready	1	13

Image	URL
docker.io/sourabhdeshmukh/chatbot:latest	http://localhost:8080/function/chatbot

Function process
python3 index.py

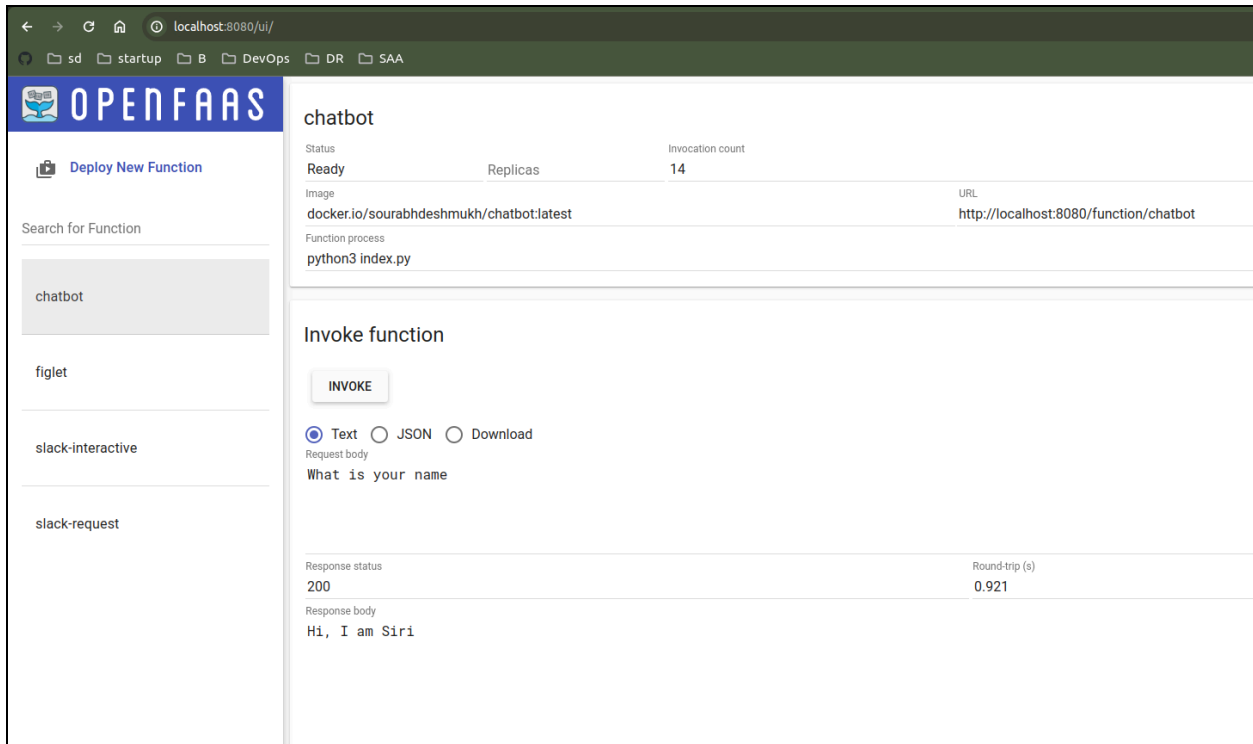
Response status	Round-trip (s)
200	0.935

Request body: help

Response body: here is a sample questions you can ask.

1. What is your name
2. what is the current time
3. figlet generate HI

- Generate 3 different names.



The screenshot shows the OpenFaaS UI at localhost:8080/ui/. The left sidebar contains a search bar and a list of functions: chatbot, figlet, slack-interactive, and slack-request. The main panel displays the details for the 'chatbot' function. It is in a 'Ready' status with 14 invocations. The image is 'docker.io/sourabhdeshmukh/chatbot:latest' and the function process is 'python3 index.py'. The URL is 'http://localhost:8080/function/chatbot'. Below this, the 'Invoke function' section shows a form with 'Text' selected as the response type. The request body is 'What is your name'. The response status is 200, the round-trip time is 0.921s, and the response body is 'Hi, I am Siri'.

Status	Replicas	Invocation count
Ready		14

Image	URL
docker.io/sourabhdeshmukh/chatbot:latest	http://localhost:8080/function/chatbot

Function process
python3 index.py

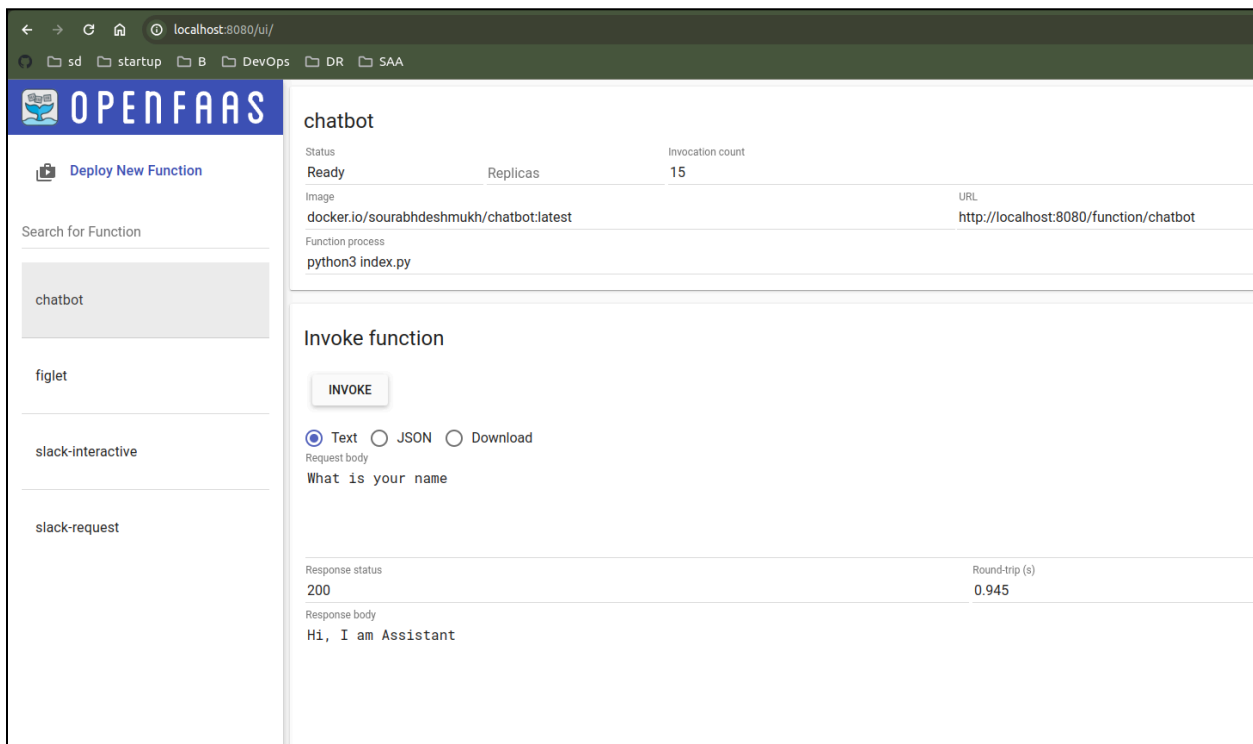
Invoke function

☒ Text ☐ JSON ☐ Download

Request body
What is your name

Response status	Round-trip (s)
200	0.921

Response body
Hi, I am Siri



This screenshot is similar to the first one, but the invocation count for the 'chatbot' function has increased to 15. The response body for the 'What is your name' request is now 'Hi, I am Assistant'.

Status	Replicas	Invocation count
Ready		15

Image	URL
docker.io/sourabhdeshmukh/chatbot:latest	http://localhost:8080/function/chatbot

Function process
python3 index.py

Invoke function

☒ Text ☐ JSON ☐ Download

Request body
What is your name

Response status	Round-trip (s)
200	0.945

Response body
Hi, I am Assistant

Sourabh Deshmukh
W1648445

The screenshot shows the OpenFaaS UI at localhost:8080/ui/. On the left, a sidebar lists functions: chatbot, figlet, slack-interactive, and slack-request. The main panel displays details for the 'chatbot' function:

Status	Replicas	Invocation count
Ready		16

Image: docker.io/sourabhdeshmukh/chatbot:latest | URL: <http://localhost:8080/function/chatbot>

Function process: python3 index.py

Invoke function

☒ Text ☐ JSON ☐ Download

Request body: What is your name

Response status	Round-trip (s)
200	0.939

Response body: Hi, I am Alexa

8. Generate 3 different formats for current time.

The screenshot shows the OpenFaaS UI at localhost:8080/ui/. The sidebar is the same. The main panel shows the 'chatbot' function details, with the invocation count now at 19.

Status	Replicas	Invocation count
Ready		19

Image: docker.io/sourabhdeshmukh/chatbot:latest | URL: <http://localhost:8080/function/chatbot>

Function process: python3 index.py

Invoke function

☒ Text ☐ JSON ☐ Download

Request body: what is the current time

Response status	Round-trip (s)
200	0.96

Response body: Current time is 11:57 PM

The screenshot shows the OpenFaaS UI at localhost:8080/ui/. The left sidebar lists functions: chatbot, figlet, slack-interactive, and slack-request. The main panel displays details for the 'chatbot' function:

Status	Replicas	Invocation count
Ready		20

Image: docker.io/sourabhdeshmukh/chatbot:latest
URL: http://localhost:8080/function/chatbot
Function process: python3 index.py

Invoke function

☒ Text ☐ JSON ☐ Download

Request body: what is the current time

Response status: 200
Round-trip (s): 0.926
Response body: Current Date is Saturday, February 24, 2024

The screenshot shows the OpenFaaS UI at localhost:8080/ui/. The left sidebar lists functions: chatbot, figlet, slack-interactive, and slack-request. The main panel displays details for the 'chatbot' function:

Status	Replicas	Invocation count
Ready		23

Image: docker.io/sourabhdeshmukh/chatbot:latest
URL: http://localhost:8080/function/chatbot
Function process: python3 index.py

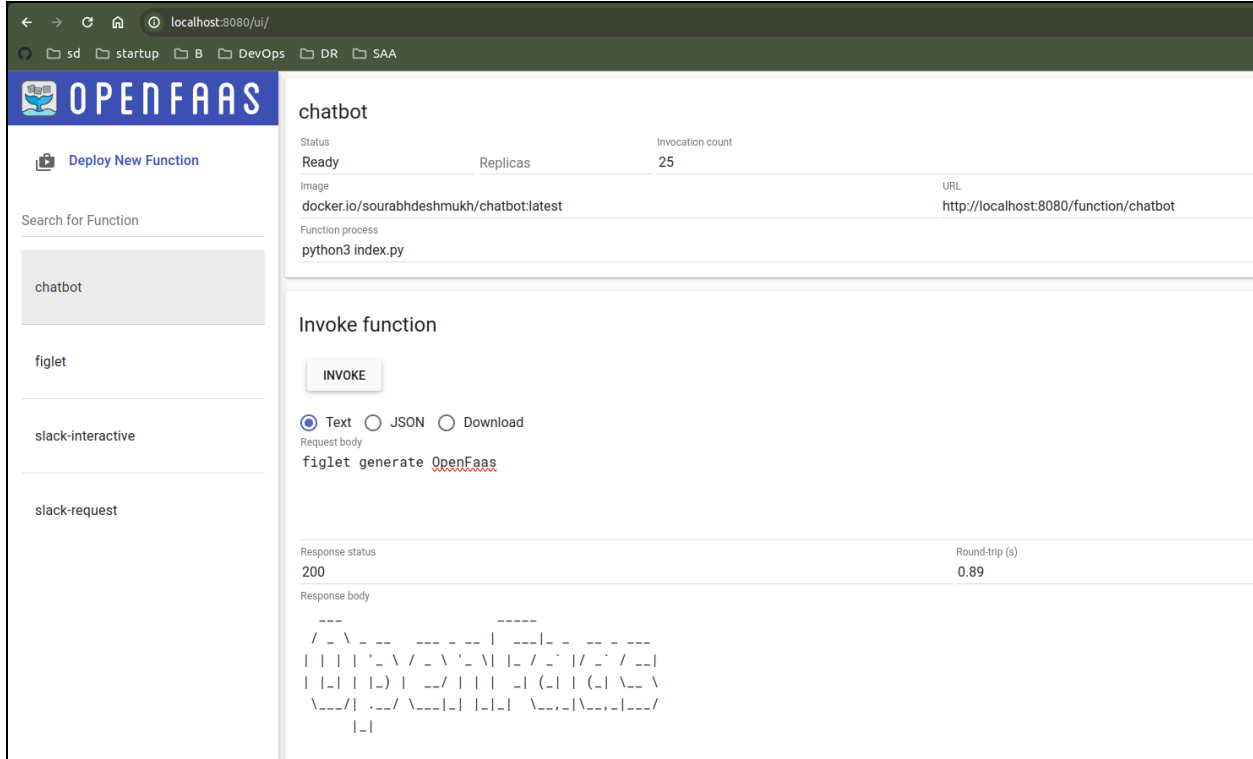
Invoke function

☒ Text ☐ JSON ☐ Download

Request body: what is the current time

Response status: 200
Round-trip (s): 0.917
Response body: Current Date and Time is Saturday, February 24, 2024, 11:58 PM

- Invoking figlet function

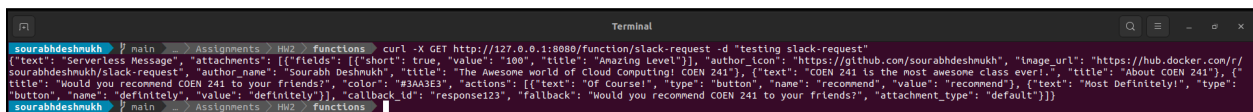


The screenshot shows the OpenFaaS web interface in a browser. The left sidebar has a search bar and a list of functions: chatbot, figlet, slack-interactive, and slack-request. The main area displays details for the 'chatbot' function, including its status (Ready), replicas (25), invocation count (25), image (docker.io/sourabhdeshmukh/chatbot:latest), and function process (python3 index.py). Below this is the 'Invoke function' section with an 'INVOKE' button and radio buttons for Text (selected), JSON, and Download. The request body is 'figlet generate OpenFaaS'. The response status is 200, the round-trip time is 0.89s, and the response body is a ASCII art representation of the text 'figlet generate OpenFaaS'.

9 What is the command to invoke the slack-request function (2 pts)?

- Via Curl

```
sourabhdeshmukh$ curl -X GET http://127.0.0.1:8080/function/slack-request -d "testing slack-request"
```



The screenshot shows a terminal window with the command `curl -X GET http://127.0.0.1:8080/function/slack-request -d "testing slack-request"` being executed. The output is a JSON response from the function.

- Via faas-cli

```
sourabhdeshmukh$ echo "invoke slack-request" | faas-cli invoke slack-request
```

```

sourabhdeshmukh@main:~/Assignments/HH2/functions$ echo "invoke slack-request" | faas-cli invoke slack-request
{"text": "Serverless Message", "attachments": [{"fields": [{"short": true, "value": "100", "title": "Amazing Level"}], "author_icon": "https://github.com/sourabhdeshmukh", "image_url": "https://hub.docker.com/r/sourabhdeshmukh/slack-request", "author_name": "Sourabh Deshmukh", "title": "The Awesome world of Cloud Computing! COEN 241"}, {"text": "COEN 241 is the most awesome class ever!.", "title": "About COEN 241"}, {"button": {"name": "definitely", "value": "definitely"}}, {"callback_id": "response123", "fallback": "Would you recommend COEN 241 to your friends?", "attachment_type": "default"}]}

```

10 What is the output you see when you invoke the slack-request function?

- When we provide the text based or json based argument then we get the json data object defined in the handle function of slack-request handler.py
- OUTPUT

```

Response body
{
  "text": "Serverless Message",
  "attachments": [
    {
      "fields": [
        {
          "short": true,
          "value": "100",
          "title": "Amazing Level"
        }
      ],
      "author_icon": "https://github.com/sourabhdeshmukh",
      "image_url": "https://hub.docker.com/r/sourabhdeshmukh/slack-request",
      "author_name": "Sourabh Deshmukh",
      "title": "The Awesome world of Cloud Computing! COEN 241"
    },
    {
      "text": "COEN 241 is the most awesome class ever!.",
      "title": "About COEN 241"
    },
    {
      "title": "Would you recommend COEN 241 to your friends?",
      "color": "#3AA3E3",
      "actions": [
        {
          "text": "Of Course!",
          "type": "button",
          "name": "recommend",
          "value": "recommend"
        },
        {
          "text": "Most Definitely!",
          "type": "button",
          "name": "definitely",
          "value": "definitely"
        }
      ],
      "callback_id": "response123",
      "fallback": "Would you recommend COEN 241 to your friends?",
      "attachment_type": "default"
    }
  ]
}

```

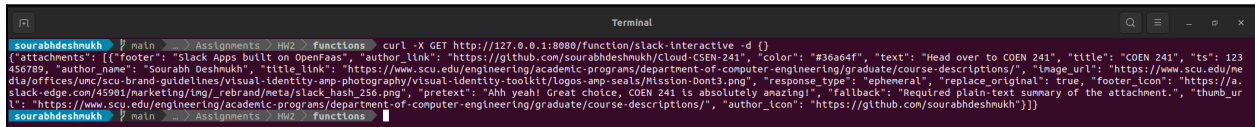
11. What is the command to invoke the slack-interactive function?

- Via curl

```
sourabhdeshmukh$ curl -X GET
```

Sourabh Deshmukh
W1648445

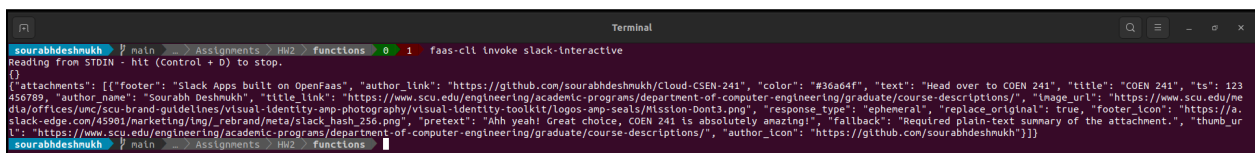
```
http://127.0.0.1:8080/function/slack-interactive -d {}
```



```
Terminal
sourabhdeshmukh ~ main > Assignments > HW2 > Functions > curl -X GET http://127.0.0.1:8080/function/slack-interactive -d {}
{"attachments": [{"footer": "Slack Apps built on OpenFaas", "author_link": "https://github.com/sourabhdeshmukh/Cloud-CSEN-241", "color": "#36a64f", "text": "Head over to COEN 241", "title": "COEN 241", "ts": 123456789, "author_name": "Sourabh Deshmukh", "title_link": "https://www.scu.edu/engineering/academic-programs/department-of-computer-engineering/graduate/course-descriptions/", "image_url": "https://www.scu.edu/media/offices/umc/scu-brand-guidelines/visual-identity-amp-photography/visual-identity-toolkit/logos-amp-seals/Mission-Dont3.png", "response_type": "ephemeral", "replace_original": true, "footer_icon": "https://a.slack-edge.com/45901/marketing/img/_rebrand/meta/slack_hash_256.png", "pretext": "Ahh yeah! Great choice, COEN 241 is absolutely amazing!", "fallback": "Required plain-text summary of the attachment.", "thumb_url": "https://www.scu.edu/engineering/academic-programs/department-of-computer-engineering/graduate/course-descriptions/", "author_icon": "https://github.com/sourabhdeshmukh"}]}
```

- Via faas-cli

```
sourabhdeshmukh$ faas-cli invoke slack-interactive
```

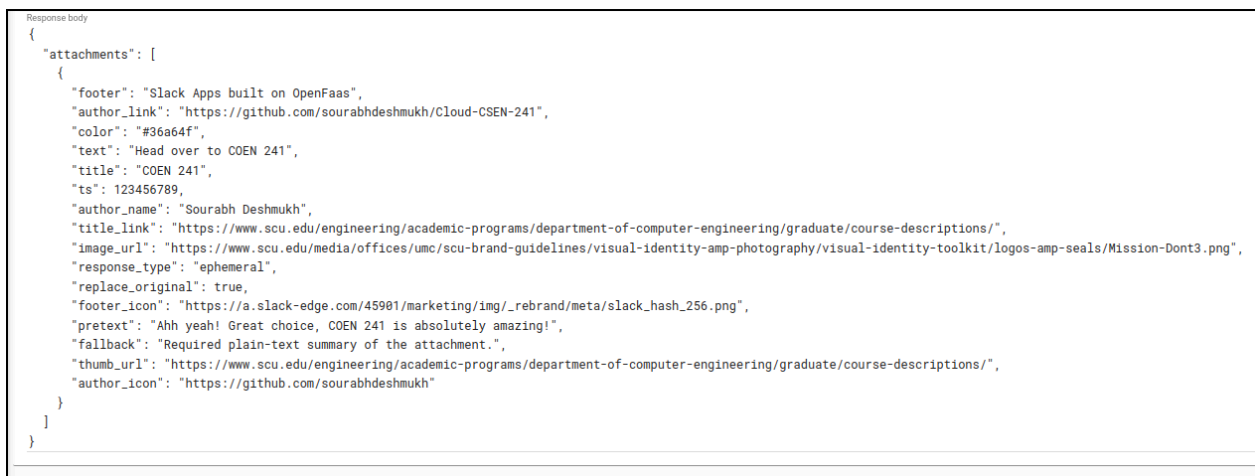


```
Terminal
sourabhdeshmukh ~ main > Assignments > HW2 > Functions > 0 1 faas-cli invoke slack-interactive
Reading from STDIN - hit (Control + D) to stop.
{"attachments": [{"footer": "Slack Apps built on OpenFaas", "author_link": "https://github.com/sourabhdeshmukh/Cloud-CSEN-241", "color": "#36a64f", "text": "Head over to COEN 241", "title": "COEN 241", "ts": 123456789, "author_name": "Sourabh Deshmukh", "title_link": "https://www.scu.edu/engineering/academic-programs/department-of-computer-engineering/graduate/course-descriptions/", "image_url": "https://www.scu.edu/media/offices/umc/scu-brand-guidelines/visual-identity-amp-photography/visual-identity-toolkit/logos-amp-seals/Mission-Dont3.png", "response_type": "ephemeral", "replace_original": true, "footer_icon": "https://a.slack-edge.com/45901/marketing/img/_rebrand/meta/slack_hash_256.png", "pretext": "Ahh yeah! Great choice, COEN 241 is absolutely amazing!", "fallback": "Required plain-text summary of the attachment.", "thumb_url": "https://www.scu.edu/engineering/academic-programs/department-of-computer-engineering/graduate/course-descriptions/", "author_icon": "https://github.com/sourabhdeshmukh"}]}
```

12. What is the output you see when you invoke the slack-interactive function?

- When we provide the text based or json based argument then we get the json data object defined in the handle function of slack-request handler.py

- OUTPUT



```
Response body
{
  "attachments": [
    {
      "footer": "Slack Apps built on OpenFaas",
      "author_link": "https://github.com/sourabhdeshmukh/Cloud-CSEN-241",
      "color": "#36a64f",
      "text": "Head over to COEN 241",
      "title": "COEN 241",
      "ts": 123456789,
      "author_name": "Sourabh Deshmukh",
      "title_link": "https://www.scu.edu/engineering/academic-programs/department-of-computer-engineering/graduate/course-descriptions/",
      "image_url": "https://www.scu.edu/media/offices/umc/scu-brand-guidelines/visual-identity-amp-photography/visual-identity-toolkit/logos-amp-seals/Mission-Dont3.png",
      "response_type": "ephemeral",
      "replace_original": true,
      "footer_icon": "https://a.slack-edge.com/45901/marketing/img/_rebrand/meta/slack_hash_256.png",
      "pretext": "Ahh yeah! Great choice, COEN 241 is absolutely amazing!",
      "fallback": "Required plain-text summary of the attachment.",
      "thumb_url": "https://www.scu.edu/engineering/academic-programs/department-of-computer-engineering/graduate/course-descriptions/",
      "author_icon": "https://github.com/sourabhdeshmukh"
    }
  ]
}
```

12 . How would you pass different arguments to the functions?

- There are two types of arguments we can pass to the function
 - Text
 - Json
- We can pass the arguments while invoking the openfaas function
 - We can invoke the function on the web UI of the openfaas.

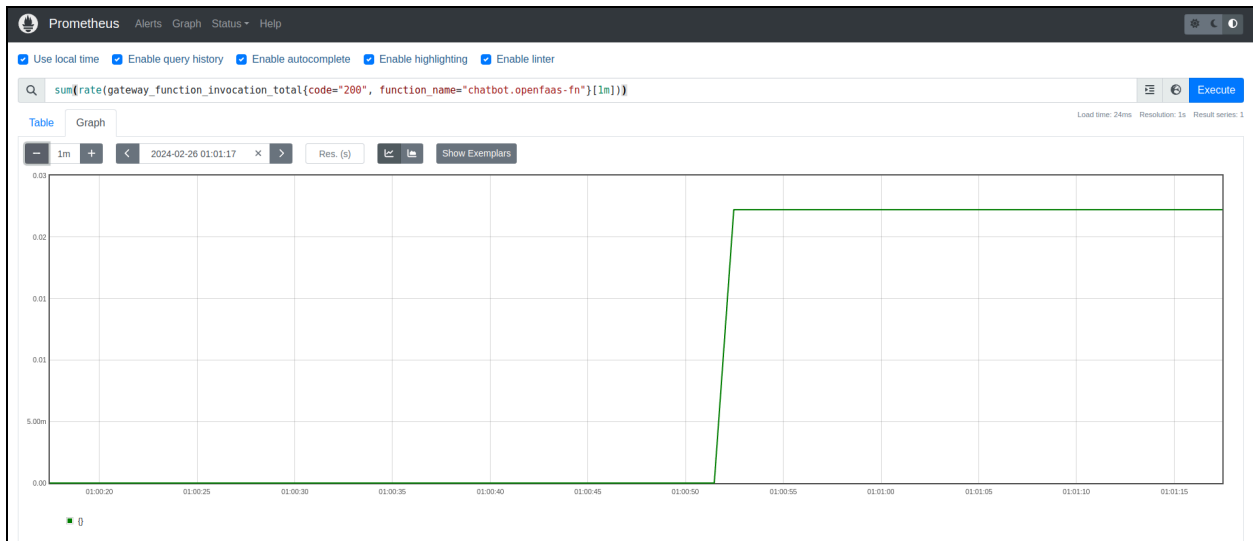
- We can invoke the function using the curl command and using the -d option we can provide the json or text argument.
- Another way is by using the faas-cli. We can invoke the function using `echo "arguments" | faas-cli invoke <function-name>`.
- We can provide the content type inside the yaml file either we can provide the application/json or text/plain content type.
- We can also alter the arguments to the function and depending upon the request we can perform the changes inside the handler function to process the text based or the json based request.

14. How would you change the slack-interactive function to react to different inputs?

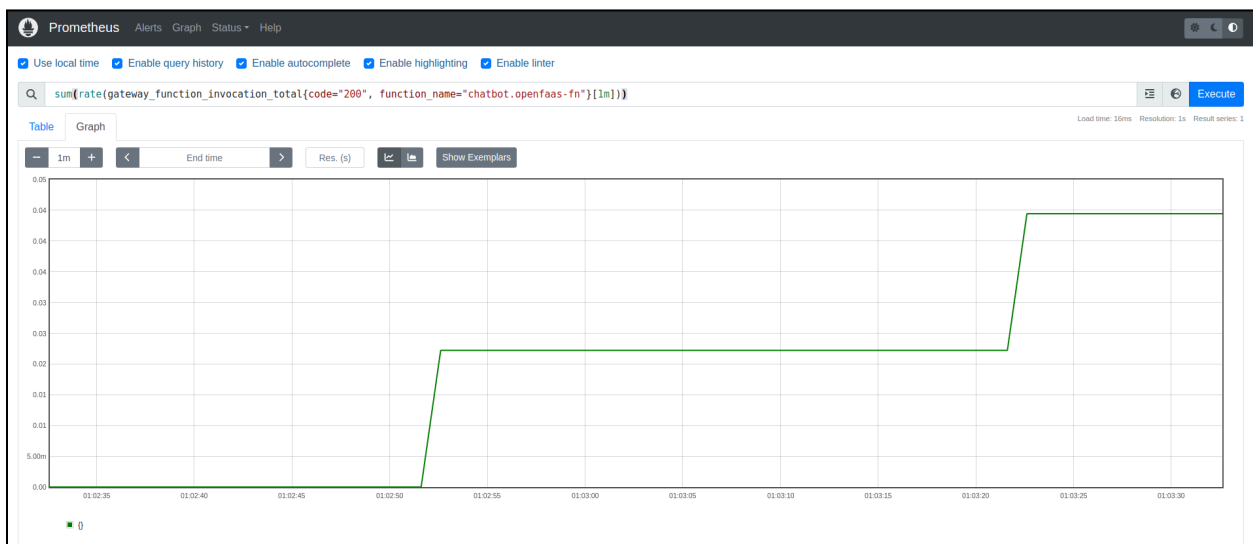
- In order to provide the additional support to function to react to different inputs we can leverage the response object, which serves as a Python dictionary containing the user's input. The function already processes the request as JSON and stores it within the response object, so direct access to the user input becomes feasible.
- The important steps that needs to be performed in the function are:
 1. Accepting the request in either text or JSON format within the handler.py file.
 2. Utilizing the unquote function from the urllib package to decode the request string, typically employing UTF-8 encoding.
 3. Making use of the json.load() method to convert the valid JSON request into a Python dictionary, which is then stored in the response object.
 4. At present the response object remains unused within the function.
 5. To activate the function to respond to varied inputs, extracting the user input data from the response object becomes very important, given its structure as a Python dictionary. While returning json.dumps(data), the function becomes capable of reacting to different inputs based on the data present in the response object.

15. How long does it take for the chat response to come back?

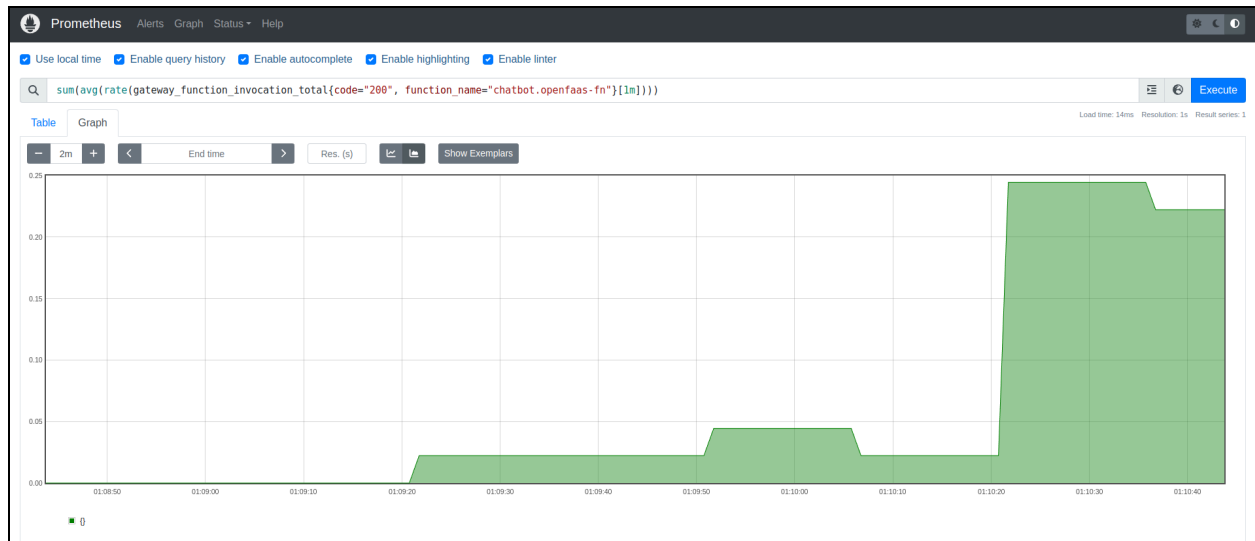
- o For the first request that does not call figlet
-> **0.2657 sec**



- For the second request that does not call figlet
-> **0.0069 sec**



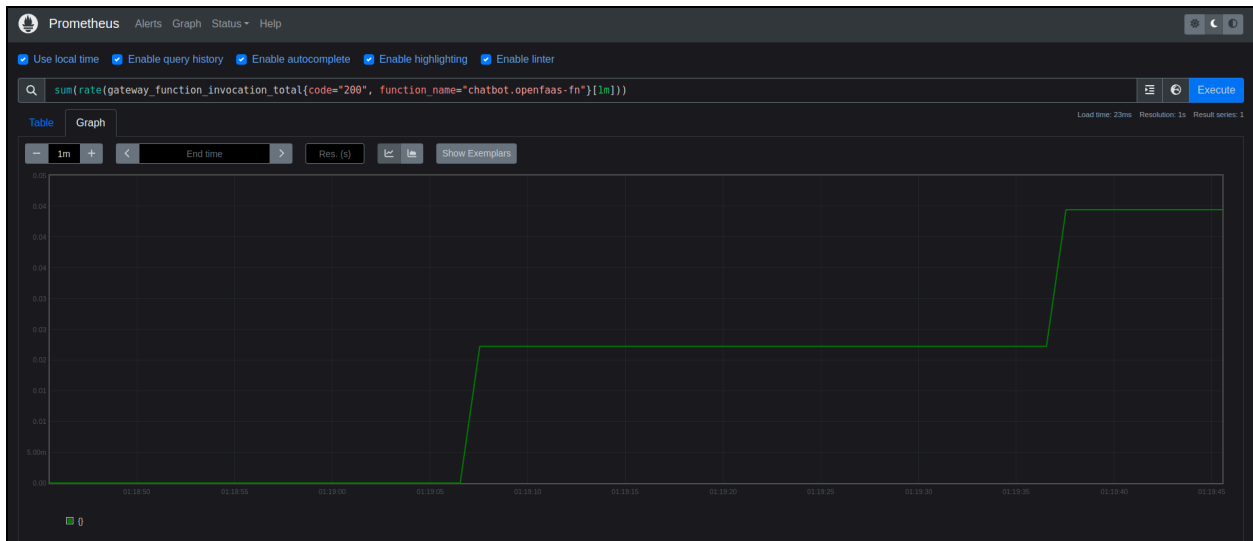
- Average over 10 requests that do not call figlet
-> **0.6732 sec**



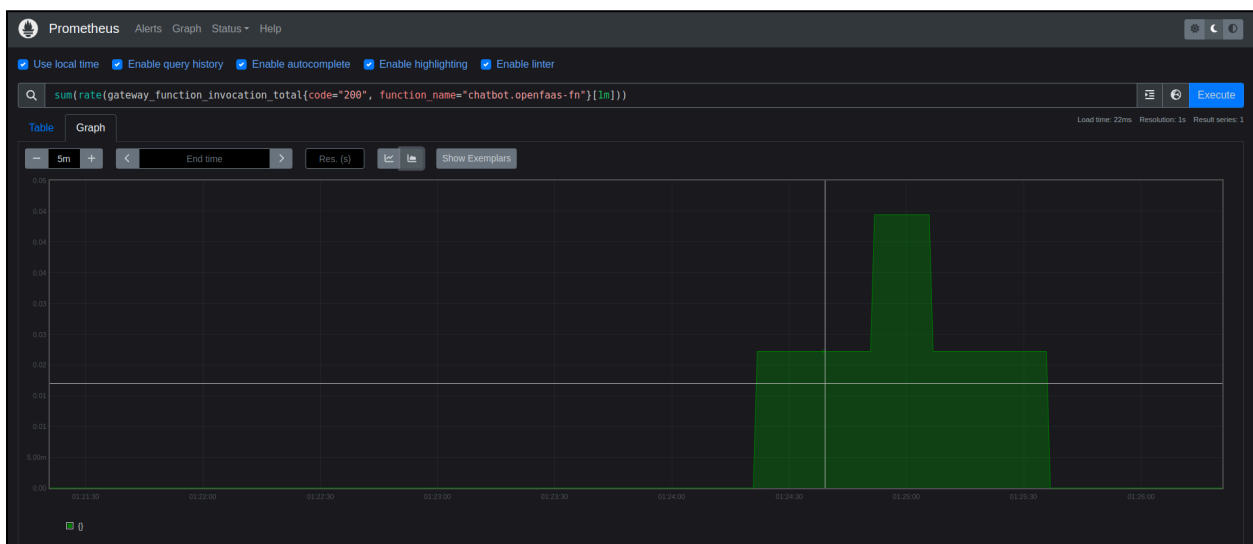
- For the first request that calls figlet
-> **1.0195 sec**



- For the second request that calls figlet
-> **0.8624 sec**



- For the second request that calls figlet that follows the first request that does not call figlet
-> **0.8988 sec**



- Average over 10 requests that do call figlet
-> **0.8074 sec**



Overall metrics result for above questions.

```
Terminal
sourabhdeshmukh$ python metrics.py
Response time for the first request using no figlet call: 0.2657 seconds
Response time for the second request using no figlet call: 0.0069 seconds
Average response time over 10 requests using no figlet call: 0.6732 seconds
Response time for the first request using with figlet call: 1.0195 seconds
Response time for the second request using with figlet call: 0.8624 seconds
Response time for the second request using with figlet and after without figlet call: 0.8988 seconds
Average response time over 10 requests using figlet call: 0.8074 seconds
sourabhdeshmukh$
```

16. Now try sending a series of requests to the chatbot in parallel. At what queries per second does OpenFaaS add a new instance of the function?

->

I have used the hey load testing tool to send requests to my function. Below is the command I used to send 10 requests per second for 60 seconds, with a payload of data=time to the /function/chatbot endpoint on localhost port 8080:

```
sourabhdeshmukh$ hey -z 60s -q 20 -c 10 -m POST -d=time'
http://127.0.0.1:8080/function/chatbot
```

The alert expression in Prometheus Alertmanager is configured to trigger an alert when the function invocation count is greater than 1 over a 5 second period. During my test, the invocation count exceeded 1, which triggered the alert to fire and caused OpenFaaS to scale the function by adding another instance. At that point, the queries per second (QPS) was around 1.5425 as the new instance was launching.

Please find below screenshots obtained during testing

OpenFaaS Portal Prometheus Time Series Prometheus Time Series Grafana - OpenFaaS

127.0.0.1:9090/alerts?search=

sd startup B DevOps DR SAA

Prometheus Alerts Graph Status Help

☒ Inactive (0) ☒ Pending (0) ☒ Firing (1) ☒ Show annotations

/etc/prometheus/alert.rules.yml > openfaas firing (1)

APIHighInvocationRate (1 active)

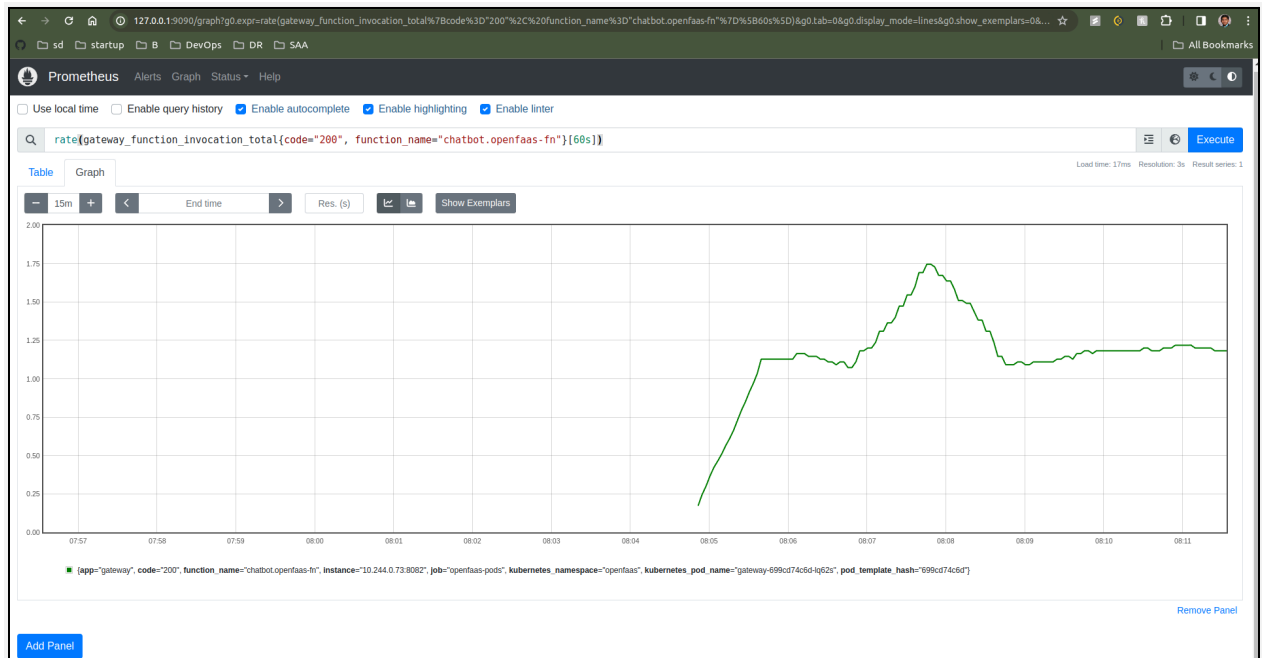
name: APIHighInvocationRate
expr: sum by (function_name) (rate(gateway_function_invocation_total{code="200"}[18s])) > 1
for: 5s
labels:
 service: gateway
 severity: major
annotations:
 description: High invocation total on "{{labels.function_name}}"
 summary: High invocation total on "{{labels.function_name}}"

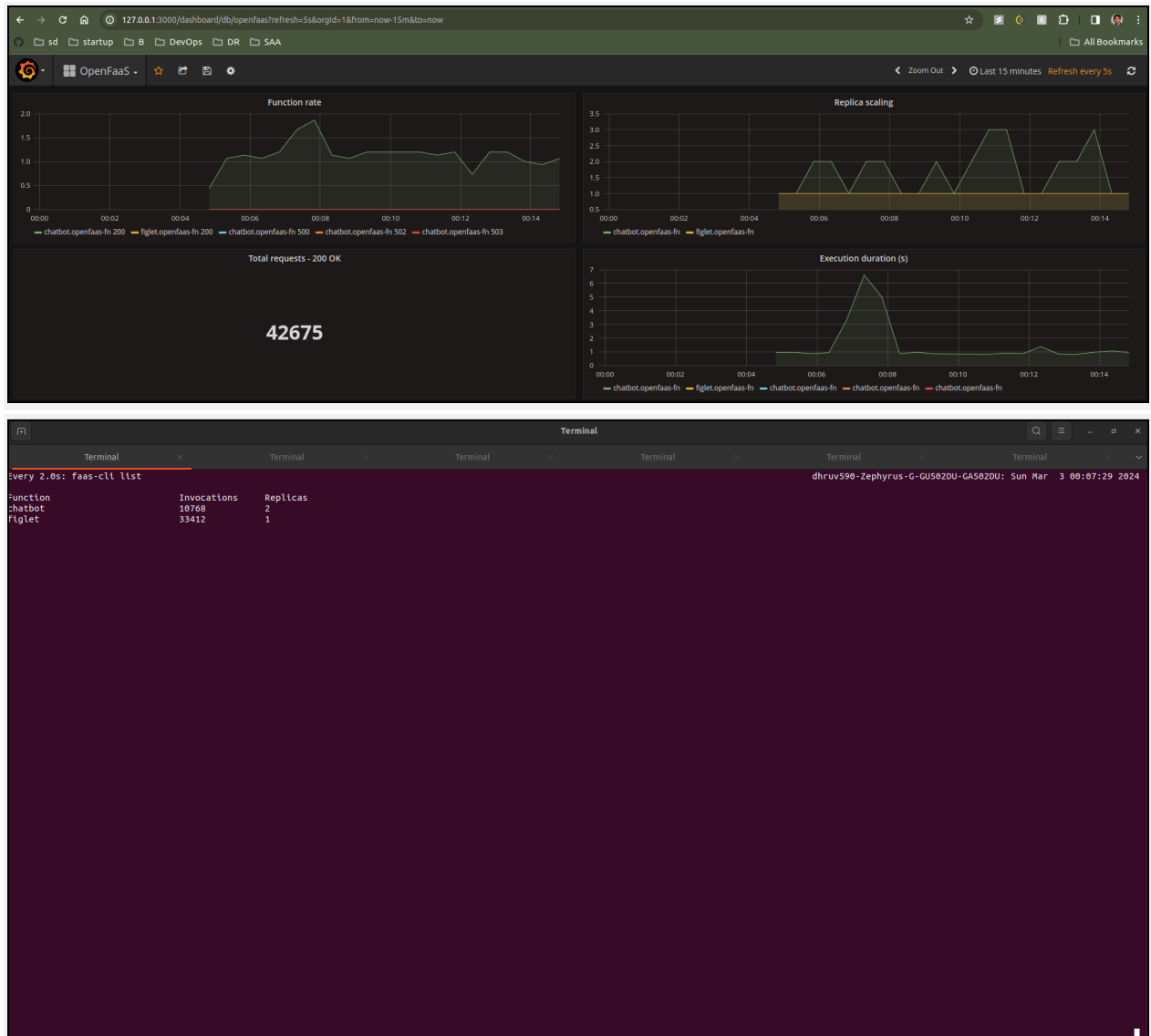
Labels	State	Active Since	Value
alertname:APIHighInvocationRate function_name:chatbot.openfaas-fn service:gateway severity:major	FIRING	2024-03-03T08:06:57.891727015Z	1.4000000000000001

Annotations

description
High invocation total on "chatbot.openfaas-fn"

summary
High invocation total on "chatbot.openfaas-fn"





Sourabh Deshmukh
W1648445

Extra Credit

Link to the application

<https://api.slack.com/apps/A06LA0P690E>

Shareable URL for the application

https://slack.com/oauth/v2/authorize?client_id=6709310927441.6690023213014&scope=chat:write.commands.incoming-webhook.mpim:write.im:write&user_scope=

Link to join OpenFaas Chatbot Workspace

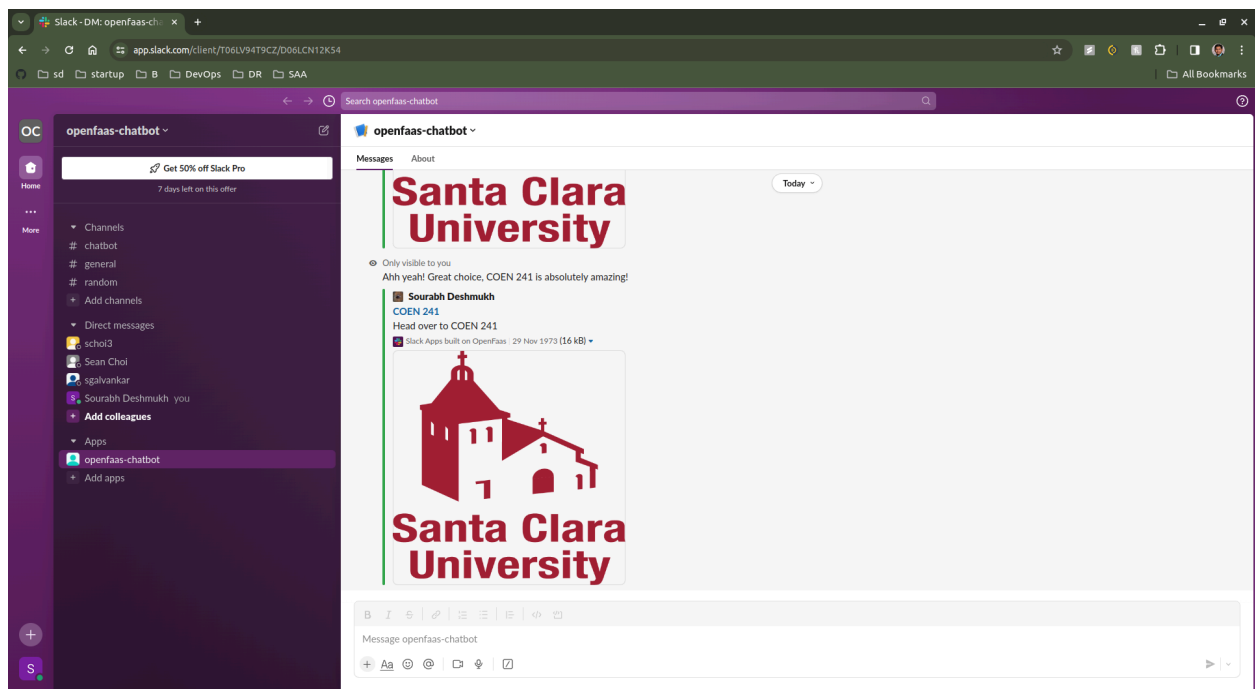
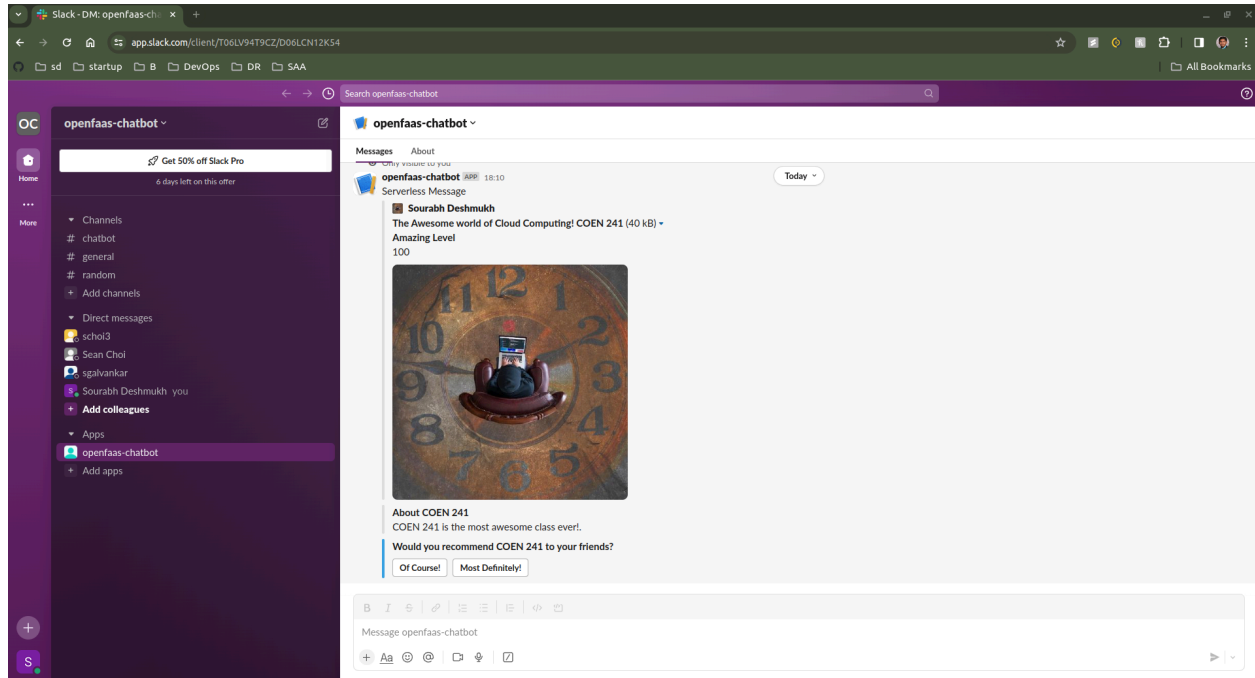
https://join.slack.com/t/openfaas-chatbotgroup/shared_invite/zt-2dm7qis1d-XxjsrZBXDWhgcm~9O2_GAA

URL for Slash Command (/coen241)

<https://charming-formally-grouper.ngrok-free.app/function/slack-request>

Slack Application Bot

Sourabh Deshmukh
W1648445



Sourabh Deshmukh
W1648445

The screenshot shows the OpenFaas Portal interface. On the left, there's a sidebar with a search bar and a list of functions: chatbot, slack-request (selected), figlet, and slack-interactive. The main panel displays details for the 'slack-request' function. It shows a status of 'Ready', 8 replicas, and an invocation count of 8. The image is 'docker.io/sourabhdeshmukh/slack-request:latest' and the URL is 'https://charming-formally-grouper.ngrok-free.app/function/slack-request'. The function process is 'python index.py'. Below this, there's an 'Invoke function' section with a button labeled 'INVOKE'. The request body is set to 'Text' and contains 'test function'. The response status is 200, the round trip time is 0.091s, and the response body is a JSON object:

```
{  "text": "Serverless Message",  "attachments": [    {      "fields": [        {          "short": true,          "value": "100",          "title": "Amazing Level"        }      ]    }  ]}
```

 At the bottom, there's a footer: "author: Sourabh Deshmukh - https://github.com/sourabhdeshmukh".

The screenshot shows the OpenFaas Portal interface for the 'slack-interactive' function. The sidebar shows 'slack-interactive' selected. The main panel displays details for this function. It shows a status of 'Ready', 8 replicas, and an invocation count of 8. The image is 'docker.io/sourabhdeshmukh/slack-interactive:latest' and the URL is 'https://charming-formally-grouper.ngrok-free.app/function/slack-interactive'. The function process is 'python index.py'. Below this, there's an 'Invoke function' section with a button labeled 'INVOKE'. The request body is set to 'JSON' and contains '{"test function"}'. The response status is 200, the round trip time is 0.163s, and the response body is a JSON object:

```
{  "attachments": [    {      "footer": "Slack Apps built on OpenFaas",      "author_link": "https://github.com/sourabhdeshmukh/Cloud-CSEN-241",      "color": "#36a64f",      "text": "Head over to COEN 241",      "title": "COEN 241",      "ts": 123456789,      "author_name": "Sourabh Deshmukh",      "title_link": "https://www.scu.edu/engineering/academic-programs/department-of-computer-engineering/graduate/course-descriptions/",      "image_url": "https://www.scu.edu/media/offices/umc/scu-brand-guidelines/visual-identity-app-photography/visual-identity-toolkit/1000-app-seals/Mission..."    }  ]}
```