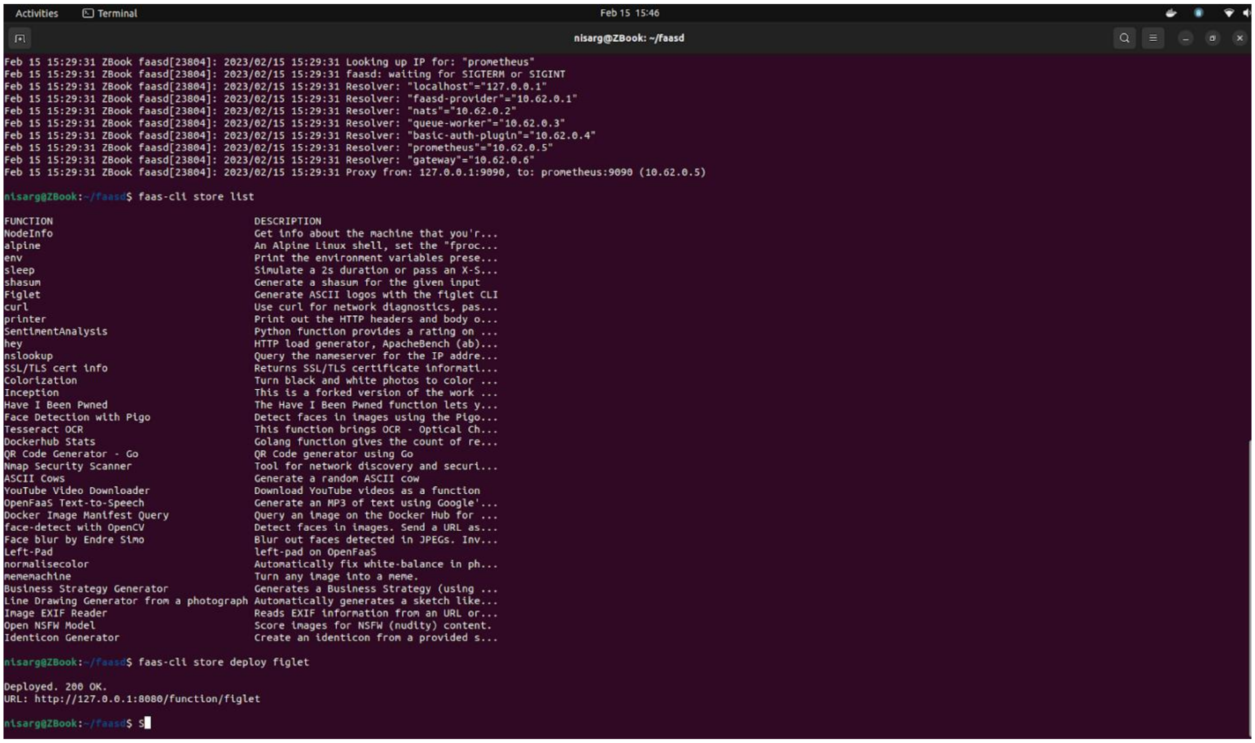
COEN 241: Homework 2

Nisarg Bhalavat W1649219

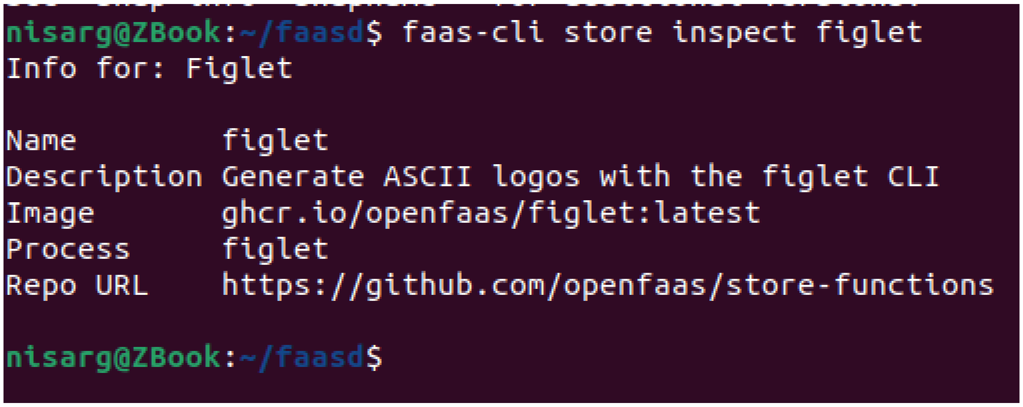
Figlet Deployment

We are deploying a function as a service (FaaS) called Figlet using faas-cli, the GitHub repository can be found on:

https://github.com/jmkhael/faas-figlet



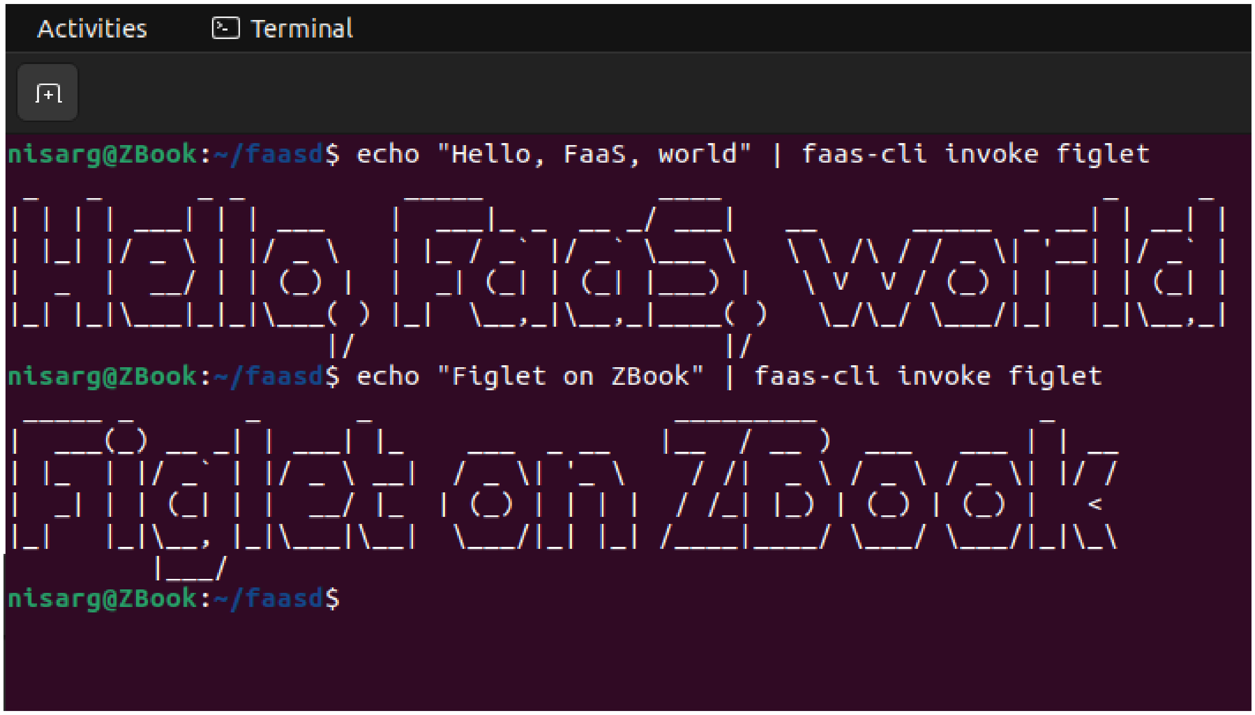
```
Feb 15 15:29:31 ZBook faasd[23804]: 2023/02/15 15:29:31 Looking up IP for: "prometheus"
Feb 15 15:29:31 ZBook faasd[23804]: 2023/02/15 15:29:31 faasd: waiting for SIGTERM or SIGINT
Feb 15 15:29:31 ZBook faasd[23804]: 2023/02/15 15:29:31 Resolver: "localhost"="127.0.0.1"
Feb 15 15:29:31 ZBook faasd[23804]: 2023/02/15 15:29:31 Resolver: "faasd-provider"="10.62.0.1"
Feb 15 15:29:31 ZBook faasd[23804]: 2023/02/15 15:29:31 Resolver: "nats"="10.62.0.2"
Feb 15 15:29:31 ZBook faasd[23804]: 2023/02/15 15:29:31 Resolver: "queue-worker"="10.62.0.3"
Feb 15 15:29:31 ZBook faasd[23804]: 2023/02/15 15:29:31 Resolver: "basic-auth-plugin"="10.62.0.4"
Feb 15 15:29:31 ZBook faasd[23804]: 2023/02/15 15:29:31 Resolver: "prometheus"="10.62.0.5"
Feb 15 15:29:31 ZBook faasd[23804]: 2023/02/15 15:29:31 Resolver: "gateway"="10.62.0.6"
Feb 15 15:29:31 ZBook faasd[23804]: 2023/02/15 15:29:31 Proxy from: 127.0.0.1:9090, to: prometheus:9090 (10.62.0.5)

nisarg@ZBook:~/faasd$ faas-cli store list

FUNCTION                                 DESCRIPTION
NodeInfo                                 Get info about the machine that you'r...
alpine                                   An Alpine Linux shell, set the "fproc...
env                                      Print the environment variables prese...
sleep                                    Simulate a 2s duration or pass an X-S...
shasum                                   Generate a shasum for the given input
Figlet                                   Generate ASCII logos with the figlet CLI
curl                                     Use curl for network diagnostics, pas...
printer                                  Print out the HTTP headers and body o...
SentimentAnalysis                        Python function provides a rating on ...
hey                                      HTTP load generator, ApacheBench (ab)...
nslookup                                 Query the nameserver for the IP addre...
SSL/TLS cert info                        Returns SSL/TLS certificate informati...
Colorization                             Turn black and white photos to color ...
Inception                                This is a forked version of the work ...
Have I Been Pwned                        The Have I Been Pwned function lets y...
Face Detection with Pigo                 Detect faces in images using the Pigo...
Tesseract OCR                            This function brings OCR - Optical Ch...
Dockerhub Stats                          Golang function gives the count of re...
QR Code Generator - Go                   QR Code generator using Go
Nmap Security Scanner                    Tool for network discovery and securi...
ASCII Cows                               Generate a random ASCII cow
YouTube Video Downloader                 Download YouTube videos as a function
OpenFaaS Text-to-Speech                  Generate an MP3 of text using Google'...
Docker Image Manifest Query              Query an image on the Docker Hub for ...
face-detect with OpenCV                  Detect faces in images. Send a URL as...
Face blur by Endre Simo                  Blur out faces detected in JPEGs. Inv...
Left-Pad                                 left-pad on OpenFaaS
normalisecolor                           Automatically fix white-balance in ph...
mememachine                              Turn any image into a meme.
Business Strategy Generator              Generates a Business Strategy (using ...
Line Drawing Generator from a photograph Automatically generates a sketch like...
Image EXIF Reader                        Reads EXIF information from an URL or...
Open NSFW Model                          Score images for NSFW (nudity) content.
Identicon Generator                      Create an identicon from a provided s...

nisarg@ZBook:~/faasd$ faas-cli store deploy figlet

Deployed. 200 OK.
URL: http://127.0.0.1:8080/function/figlet

nisarg@ZBook:~/faasd$ $
```



```
nisarg@ZBook:~/faasd$ faas-cli store inspect figlet
Info for: Figlet

Name        figlet
Description Generate ASCII logos with the figlet CLI
Image       ghcr.io/openfaas/figlet:latest
Process     figlet
Repo URL    https://github.com/openfaas/store-functions

nisarg@ZBook:~/faasd$
```

```
# Deploy figlet
$ faas-cli store deploy figlet
```

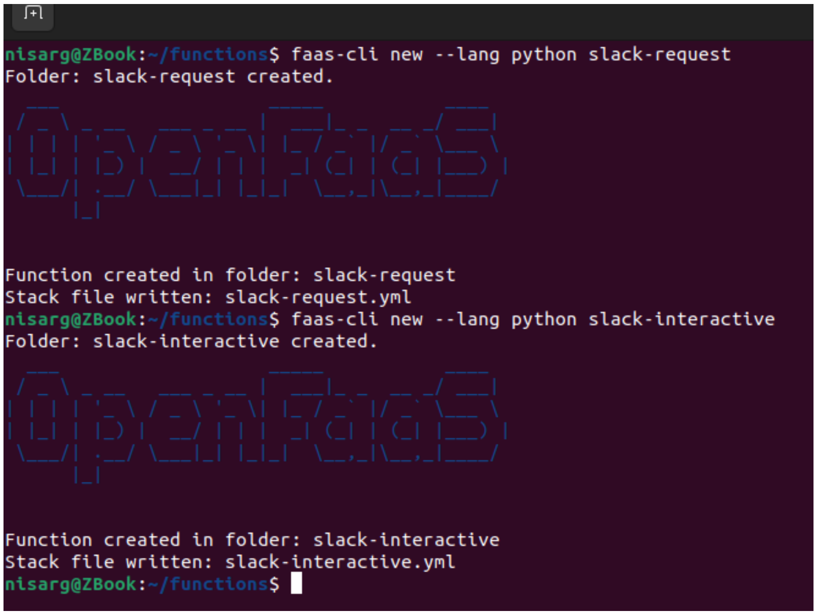
```
# Find the URLs for the function
$ faas-cli store inspect figlet

# Create some ASCII
$ echo "Hello, FaaS, world" | faas-cli invoke figlet
```



```
nisarg@ZBook:~/faasd$ echo "Hello, FaaS, world" | faas-cli invoke figlet
 _   _      _ _          _____           ____                       _     _ 
| | | | ___| | | ___    |  ___|_ _  __ _/ ___|  __      _____  _ __| | __| |
| |_| |/ _ \ | |/ _ \   | |_ / _` |/ _` \___ \  \ \ /\ / / _ \| '__| |/ _` |
|  _  |  __/ | | (_) |  |  _| (_| | (_| |___) |  \ V  V / (_) | |  | | (_| |
|_| |_|\___|_|_|\___( ) |_|  \__,_|\__,_|____( )  \_/\_/ \___/|_|  |_|\__,_|
                    |/                        |/
nisarg@ZBook:~/faasd$ echo "Figlet on ZBook" | faas-cli invoke figlet
 _____ _       _      _                   _________              _    
|  ___(_) __ _| | ___| |_    ___  _ __   |__  / __ )  ___   ___ | | __
| |_  | |/ _` | |/ _ \ __|  / _ \| '_ \    / /|  _ \ / _ \ / _ \| |/ /
|  _| | | (_| | |  __/ |_  | (_) | | | |  / /_| |_) | (_) | (_) |   < 
|_|   |_|\__, |_|\___|\__|  \___/|_| |_| /____|____/ \___/ \___/|_|\_\
         |___/
nisarg@ZBook:~/faasd$
```

Slack Functions

We will be creating our own slack functions which would be running on OpenFaaS similar to Figlet we ran above. However, the Figlet function is available on the OpenFaaS store, and our functions would be authored from scratch.



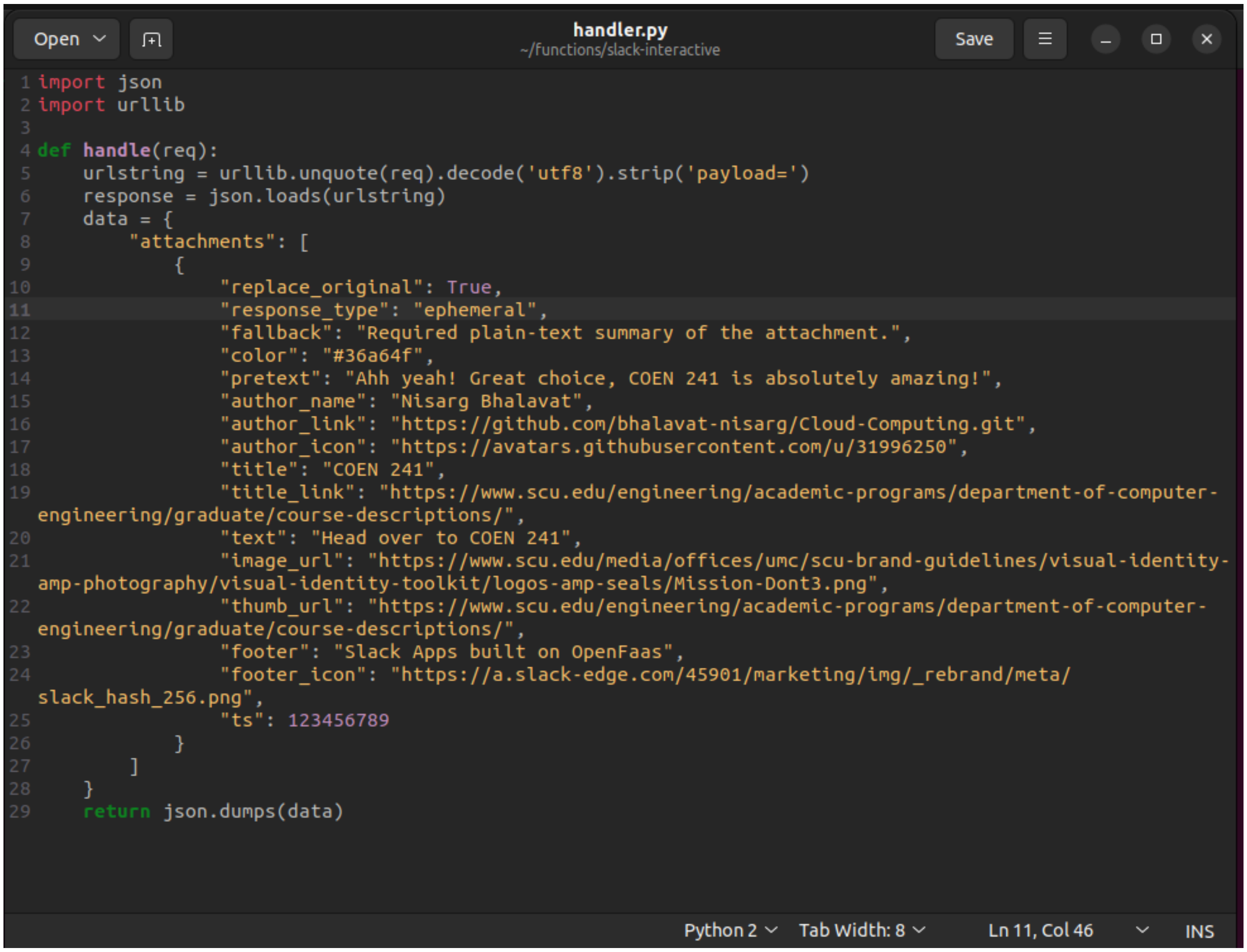
```
nisarg@ZBook:~/functions$ faas-cli new --lang python slack-request
Folder: slack-request created.
  ___                   _____           ____
 / _ \ _ __   ___ _ __ |  ___|_ _  __ _/ ___|
| | | | '_ \ / _ \ '_ \| |_ / _` |/ _` \___ \
| |_| | |_) |  __/ | | |  _| (_| | (_| |___) |
 \___/| .__/ \___|_| |_|_|  \__,_|\__,_|____/
      |_|


Function created in folder: slack-request
Stack file written: slack-request.yml
nisarg@ZBook:~/functions$ faas-cli new --lang python slack-interactive
Folder: slack-interactive created.
  ___                   _____           ____
 / _ \ _ __   ___ _ __ |  ___|_ _  __ _/ ___|
| | | | '_ \ / _ \ '_ \| |_ / _` |/ _` \___ \
| |_| | |_) |  __/ | | |  _| (_| | (_| |___) |
 \___/| .__/ \___|_| |_|_|  \__,_|\__,_|____/
      |_|


Function created in folder: slack-interactive
Stack file written: slack-interactive.yml
nisarg@ZBook:~/functions$
```

Slack Interactive

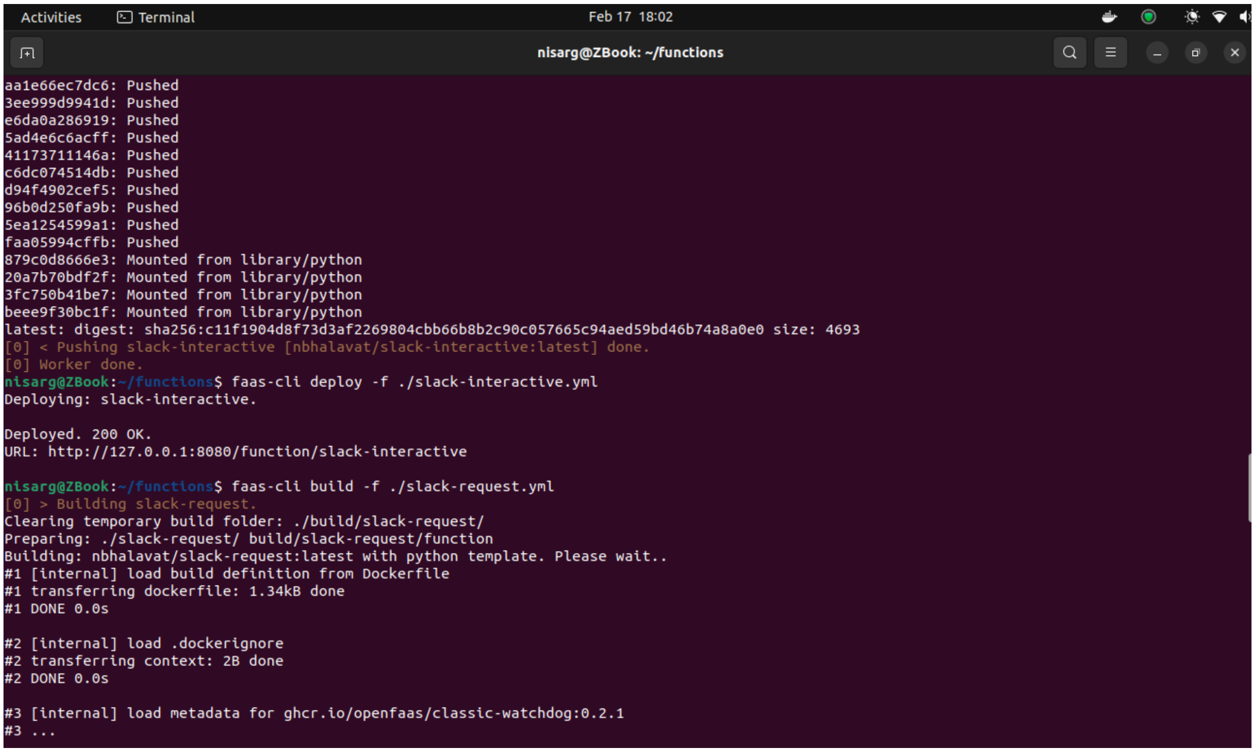
First, we are going to setup slack-interactive function. We can get the skeleton code by running the following command.

```
$ faas-cli new --lang python slack-interactive
```

Once, the files are updated, we need to deploy the functions.



```python
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": "Nisarg Bhalavat",
                "author_link": "https://github.com/bhalavat-nisarg/Cloud-Computing.git",
                "author_icon": "https://avatars.githubusercontent.com/u/31996250",
                "title": "COEN 241",
                "title_link": "https://www.scu.edu/engineering/academic-programs/department-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-identity-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)
```



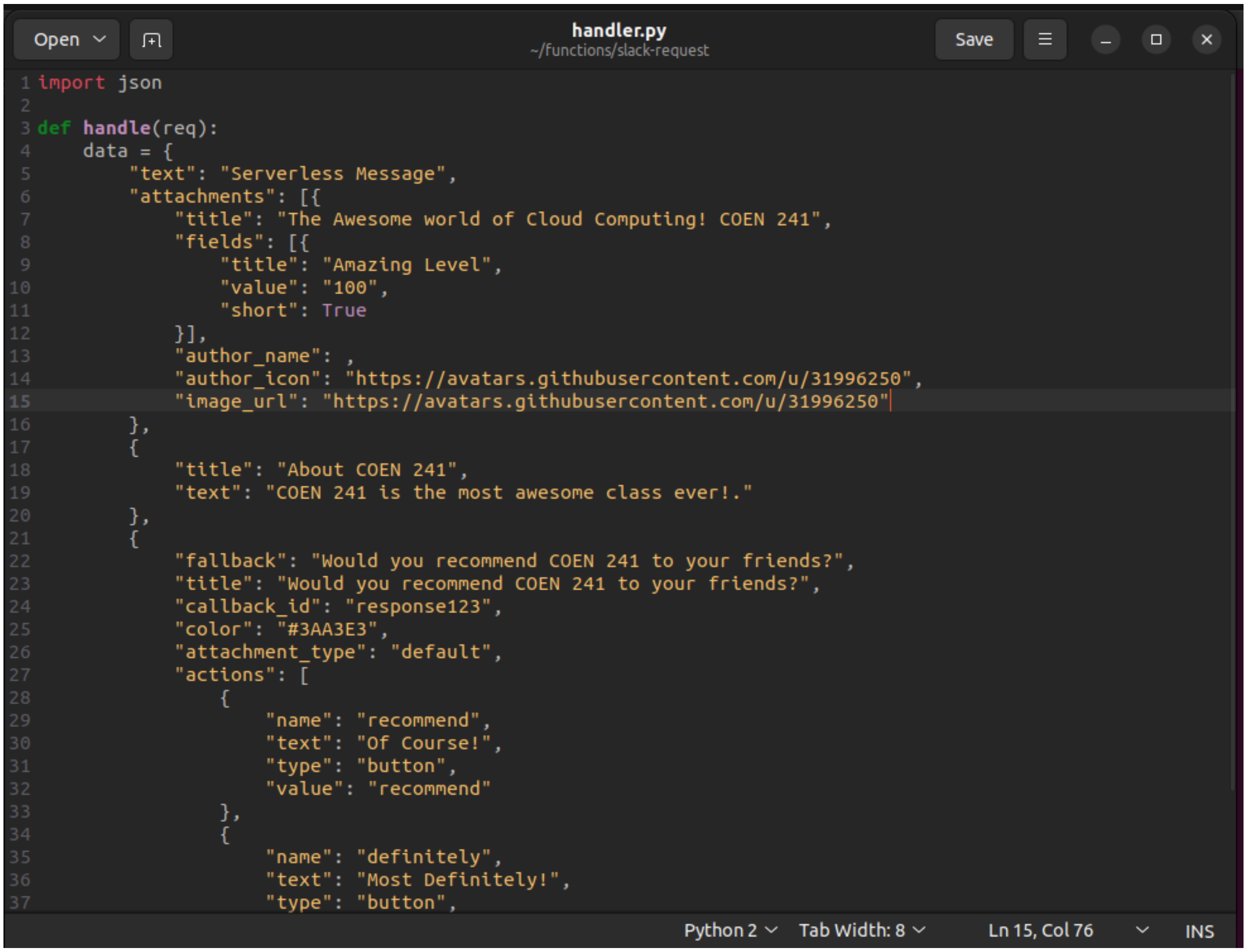
```
aa1e66ec7dc6: Pushed
3ee999d9941d: Pushed
e6da0a286919: Pushed
5ad4e6c6acff: Pushed
41173711146a: Pushed
c6dc074514db: Pushed
d94f4902cef5: Pushed
96b0d250fa9b: Pushed
5ea1254599a1: Pushed
faa05994cffb: Pushed
879c0d8666e3: Mounted from library/python
20a7b70bdf2f: Mounted from library/python
3fc750b41be7: Mounted from library/python
beee9f30bc1f: Mounted from library/python
latest: digest: sha256:c11f1904d8f73d3af2269804cbb66b8b2c90c057665c94aed59bd46b74a8a0e0 size: 4693
[0] < Pushing slack-interactive [nbhalavat/slack-interactive:latest] done.
[0] Worker done.
nisarg@ZBook:~/functions$ faas-cli deploy -f ./slack-interactive.yml
Deploying: slack-interactive.

Deployed. 200 OK.
URL: http://127.0.0.1:8080/function/slack-interactive

nisarg@ZBook:~/functions$ faas-cli build -f ./slack-request.yml
[0] > Building slack-request.
Clearing temporary build folder: ./build/slack-request/
Preparing: ./slack-request/ build/slack-request/function
Building: nbhalavat/slack-request:latest with python template. Please wait..
#1 [internal] load build definition from Dockerfile
#1 transferring dockerfile: 1.34kB done
#1 DONE 0.0s

#2 [internal] load .dockerignore
#2 transferring context: 2B done
#2 DONE 0.0s

#3 [internal] load metadata for ghcr.io/openfaas/classic-watchdog:0.2.1
#3 ...
```

Slack Request

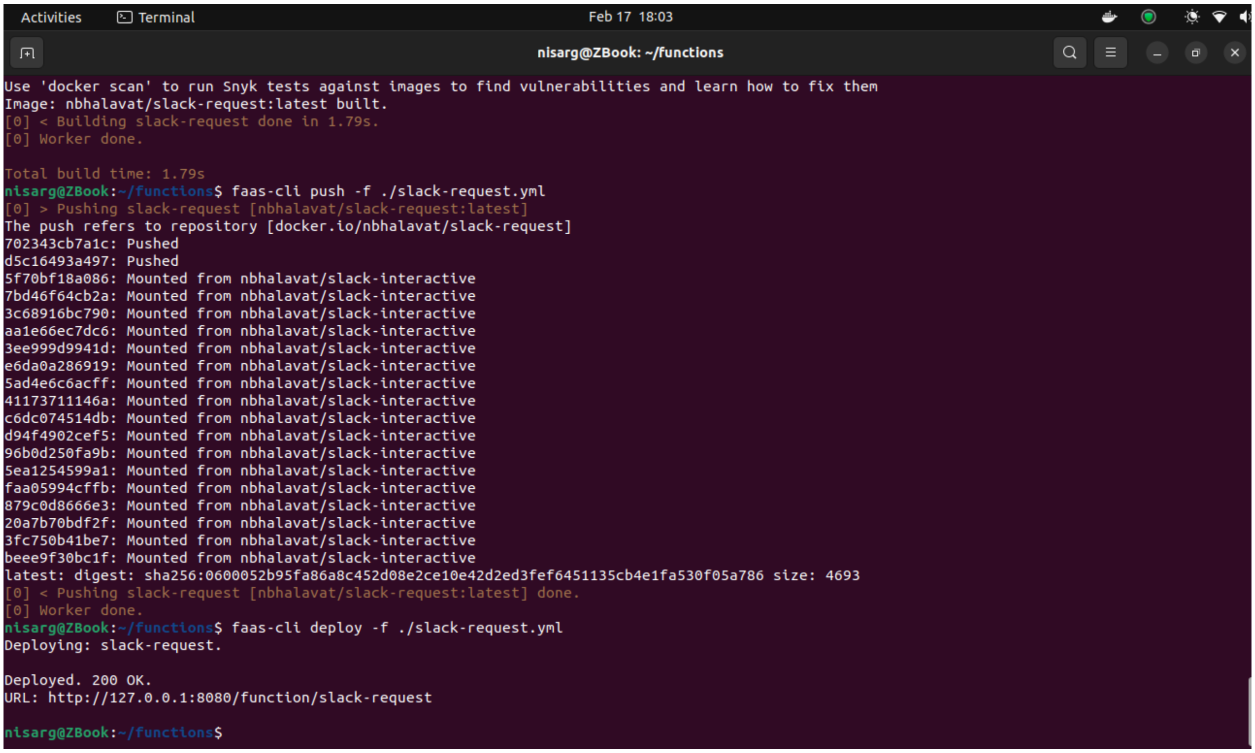
From the terminal, we need to create the skeleton code for slack-request and then update the handler files.

```
$ faas-cli new --lang python slack-request
```

Once, the files are updated, we need to deploy the functions.



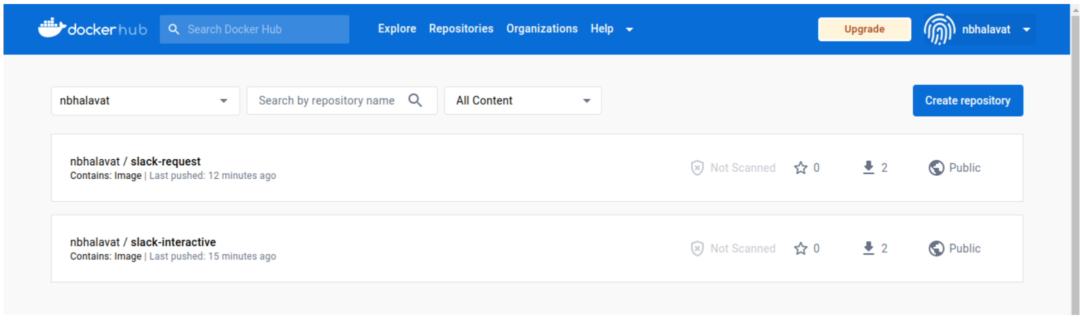
```python
import json

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": ,
            "author_icon": "https://avatars.githubusercontent.com/u/31996250",
            "image_url": "https://avatars.githubusercontent.com/u/31996250"
        },
        {
            "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",
```



```
Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
Image: nbhalavat/slack-request:latest built.
[0] < Building slack-request done in 1.79s.
[0] Worker done.

Total build time: 1.79s
nisarg@ZBook:~/functions$ faas-cli push -f ./slack-request.yml
[0] > Pushing slack-request [nbhalavat/slack-request:latest]
The push refers to repository [docker.io/nbhalavat/slack-request]
702343cb7a1c: Pushed
d5c16493a497: Pushed
5f70bf18a086: Mounted from nbhalavat/slack-interactive
7bd46f64cb2a: Mounted from nbhalavat/slack-interactive
3c68916bc790: Mounted from nbhalavat/slack-interactive
aa1e66ec7dc6: Mounted from nbhalavat/slack-interactive
3ee999d9941d: Mounted from nbhalavat/slack-interactive
e6da0a286919: Mounted from nbhalavat/slack-interactive
5ad4e6c6acff: Mounted from nbhalavat/slack-interactive
41173711146a: Mounted from nbhalavat/slack-interactive
c6dc074514db: Mounted from nbhalavat/slack-interactive
d94f4902cef5: Mounted from nbhalavat/slack-interactive
96b0d250fa9b: Mounted from nbhalavat/slack-interactive
5ea1254599a1: Mounted from nbhalavat/slack-interactive
faa05994cffb: Mounted from nbhalavat/slack-interactive
879c0d8666e3: Mounted from nbhalavat/slack-interactive
20a7b70bdf2f: Mounted from nbhalavat/slack-interactive
3fc750b41be7: Mounted from nbhalavat/slack-interactive
beee9f30bc1f: Mounted from nbhalavat/slack-interactive
latest: digest: sha256:0600052b95fa86a8c452d08e2ce10e42d2ed3fef6451135cb4e1fa530f05a786 size: 4693
[0] < Pushing slack-request [nbhalavat/slack-request:latest] done.
[0] Worker done.
nisarg@ZBook:~/functions$ faas-cli deploy -f ./slack-request.yml
Deploying: slack-request.

Deployed. 200 OK.
URL: http://127.0.0.1:8080/function/slack-request

nisarg@ZBook:~/functions$
```

The following commands were used to deploy the functions:

```
$ faas-cli build -f ./slack-interactive.yml
$ faas-cli push -f ./slack-interactive.yml
$ faas-cli deploy -f ./slack-interactive.yml

$ faas-cli build -f ./slack-request.yml
$ faas-cli push -f ./slack-request.yml
$ faas-cli deploy -f ./slack-request.yml
```



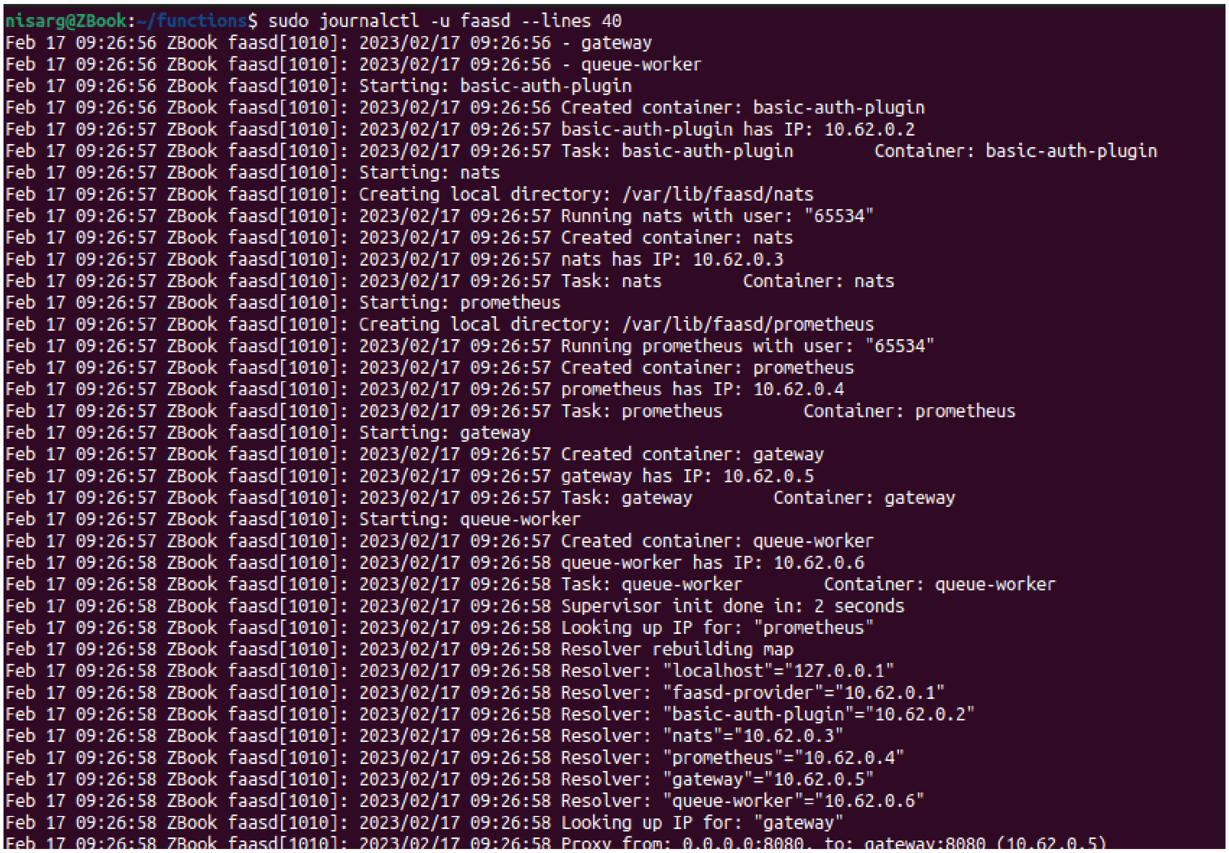
nbhalavat / slack-request
Contains: Image | Last pushed: 12 minutes ago
Not Scanned ☆ 0 ⤓ 2 Public

nbhalavat / slack-interactive
Contains: Image | Last pushed: 15 minutes ago
Not Scanned ☆ 0 ⤓ 2 Public

Journal CTL command

Now, we are running journal CTL command to query and display the logs from journald and sysemd's logging service.

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

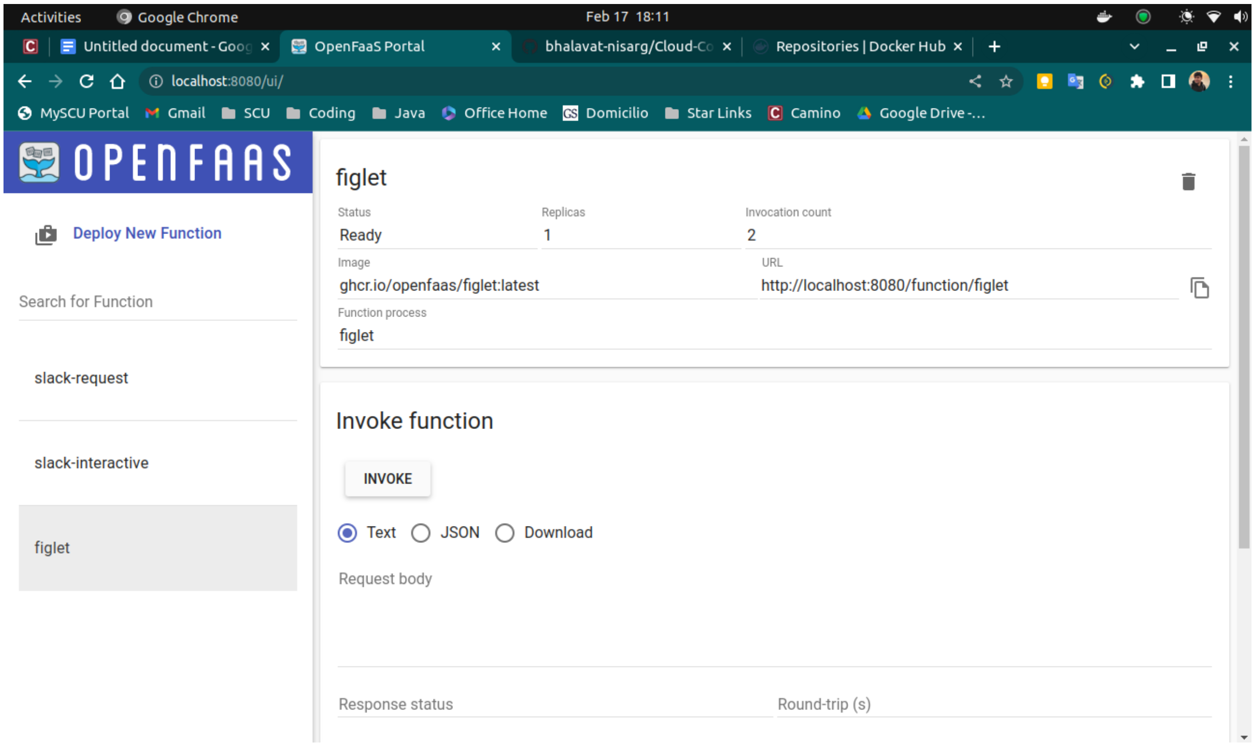


```
nisarg@ZBook:~/functions$ sudo journalctl -u faasd --lines 40
Feb 17 09:26:56 ZBook faasd[1010]: 2023/02/17 09:26:56 - gateway
Feb 17 09:26:56 ZBook faasd[1010]: 2023/02/17 09:26:56 - queue-worker
Feb 17 09:26:56 ZBook faasd[1010]: Starting: basic-auth-plugin
Feb 17 09:26:56 ZBook faasd[1010]: 2023/02/17 09:26:56 Created container: basic-auth-plugin
Feb 17 09:26:57 ZBook faasd[1010]: 2023/02/17 09:26:57 basic-auth-plugin has IP: 10.62.0.2
Feb 17 09:26:57 ZBook faasd[1010]: 2023/02/17 09:26:57 Task: basic-auth-plugin        Container: basic-auth-plugin
Feb 17 09:26:57 ZBook faasd[1010]: Starting: nats
Feb 17 09:26:57 ZBook faasd[1010]: Creating local directory: /var/lib/faasd/nats
Feb 17 09:26:57 ZBook faasd[1010]: 2023/02/17 09:26:57 Running nats with user: "65534"
Feb 17 09:26:57 ZBook faasd[1010]: 2023/02/17 09:26:57 Created container: nats
Feb 17 09:26:57 ZBook faasd[1010]: 2023/02/17 09:26:57 nats has IP: 10.62.0.3
Feb 17 09:26:57 ZBook faasd[1010]: 2023/02/17 09:26:57 Task: nats        Container: nats
Feb 17 09:26:57 ZBook faasd[1010]: Starting: prometheus
Feb 17 09:26:57 ZBook faasd[1010]: Creating local directory: /var/lib/faasd/prometheus
Feb 17 09:26:57 ZBook faasd[1010]: 2023/02/17 09:26:57 Running prometheus with user: "65534"
Feb 17 09:26:57 ZBook faasd[1010]: 2023/02/17 09:26:57 Created container: prometheus
Feb 17 09:26:57 ZBook faasd[1010]: 2023/02/17 09:26:57 prometheus has IP: 10.62.0.4
Feb 17 09:26:57 ZBook faasd[1010]: 2023/02/17 09:26:57 Task: prometheus        Container: prometheus
Feb 17 09:26:57 ZBook faasd[1010]: Starting: gateway
Feb 17 09:26:57 ZBook faasd[1010]: 2023/02/17 09:26:57 Created container: gateway
Feb 17 09:26:57 ZBook faasd[1010]: 2023/02/17 09:26:57 gateway has IP: 10.62.0.5
Feb 17 09:26:57 ZBook faasd[1010]: 2023/02/17 09:26:57 Task: gateway        Container: gateway
Feb 17 09:26:57 ZBook faasd[1010]: Starting: queue-worker
Feb 17 09:26:57 ZBook faasd[1010]: 2023/02/17 09:26:57 Created container: queue-worker
Feb 17 09:26:58 ZBook faasd[1010]: 2023/02/17 09:26:58 queue-worker has IP: 10.62.0.6
Feb 17 09:26:58 ZBook faasd[1010]: 2023/02/17 09:26:58 Task: queue-worker        Container: queue-worker
Feb 17 09:26:58 ZBook faasd[1010]: 2023/02/17 09:26:58 Supervisor init done in: 2 seconds
Feb 17 09:26:58 ZBook faasd[1010]: 2023/02/17 09:26:58 Looking up IP for: "prometheus"
Feb 17 09:26:58 ZBook faasd[1010]: 2023/02/17 09:26:58 Resolver rebuilding map
Feb 17 09:26:58 ZBook faasd[1010]: 2023/02/17 09:26:58 Resolver: "localhost"="127.0.0.1"
Feb 17 09:26:58 ZBook faasd[1010]: 2023/02/17 09:26:58 Resolver: "faasd-provider"="10.62.0.1"
Feb 17 09:26:58 ZBook faasd[1010]: 2023/02/17 09:26:58 Resolver: "basic-auth-plugin"="10.62.0.2"
Feb 17 09:26:58 ZBook faasd[1010]: 2023/02/17 09:26:58 Resolver: "nats"="10.62.0.3"
Feb 17 09:26:58 ZBook faasd[1010]: 2023/02/17 09:26:58 Resolver: "prometheus"="10.62.0.4"
Feb 17 09:26:58 ZBook faasd[1010]: 2023/02/17 09:26:58 Resolver: "gateway"="10.62.0.5"
Feb 17 09:26:58 ZBook faasd[1010]: 2023/02/17 09:26:58 Resolver: "queue-worker"="10.62.0.6"
Feb 17 09:26:58 ZBook faasd[1010]: 2023/02/17 09:26:58 Looking up IP for: "gateway"
Feb 17 09:26:58 ZBook faasd[1010]: 2023/02/17 09:26:58 Proxy from: 0.0.0.0:8080, to: gateway:8080 (10.62.0.5)
```

OpenFaaS UI

We can access the OpenFaaS UI page by going onto following url:

http://localhost:8080/ui/

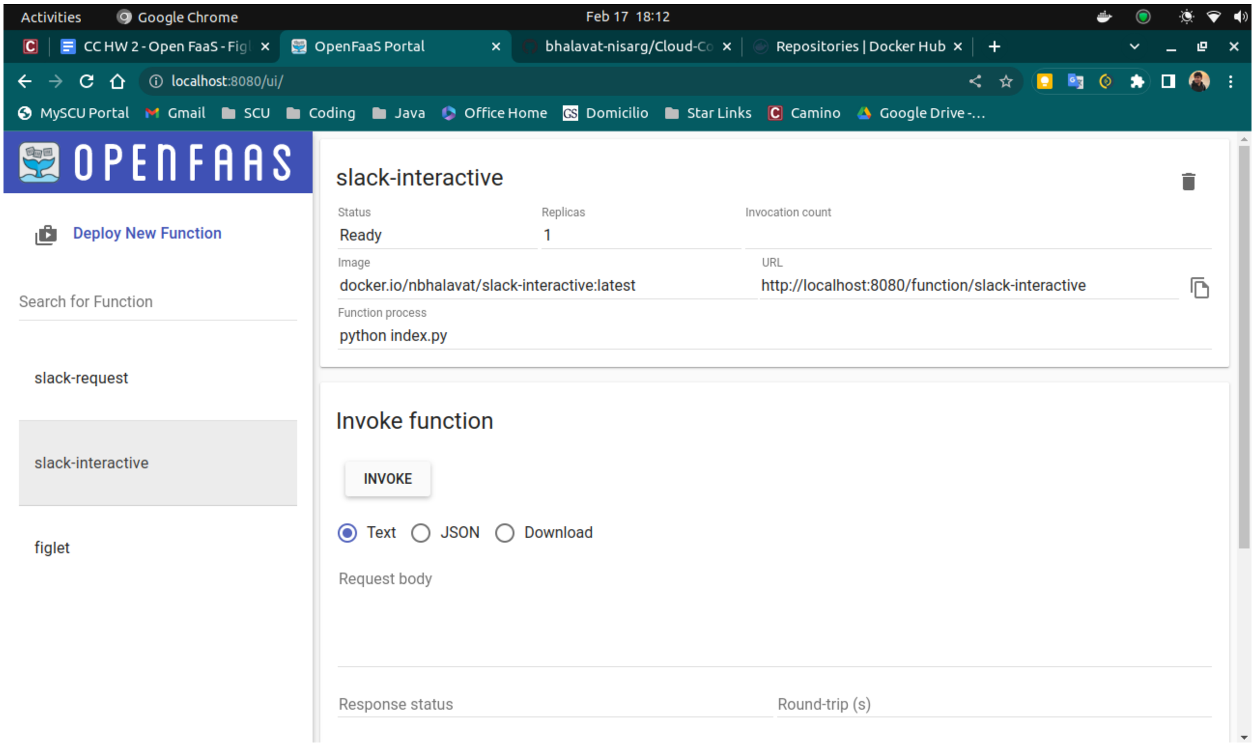


figlet

Status: Ready
Replicas: 1
Invocation count: 2
Image: ghcr.io/openfaas/figlet:latest
URL: http://localhost:8080/function/figlet
Function process: figlet

Invoke function

● Text ○ JSON ○ Download

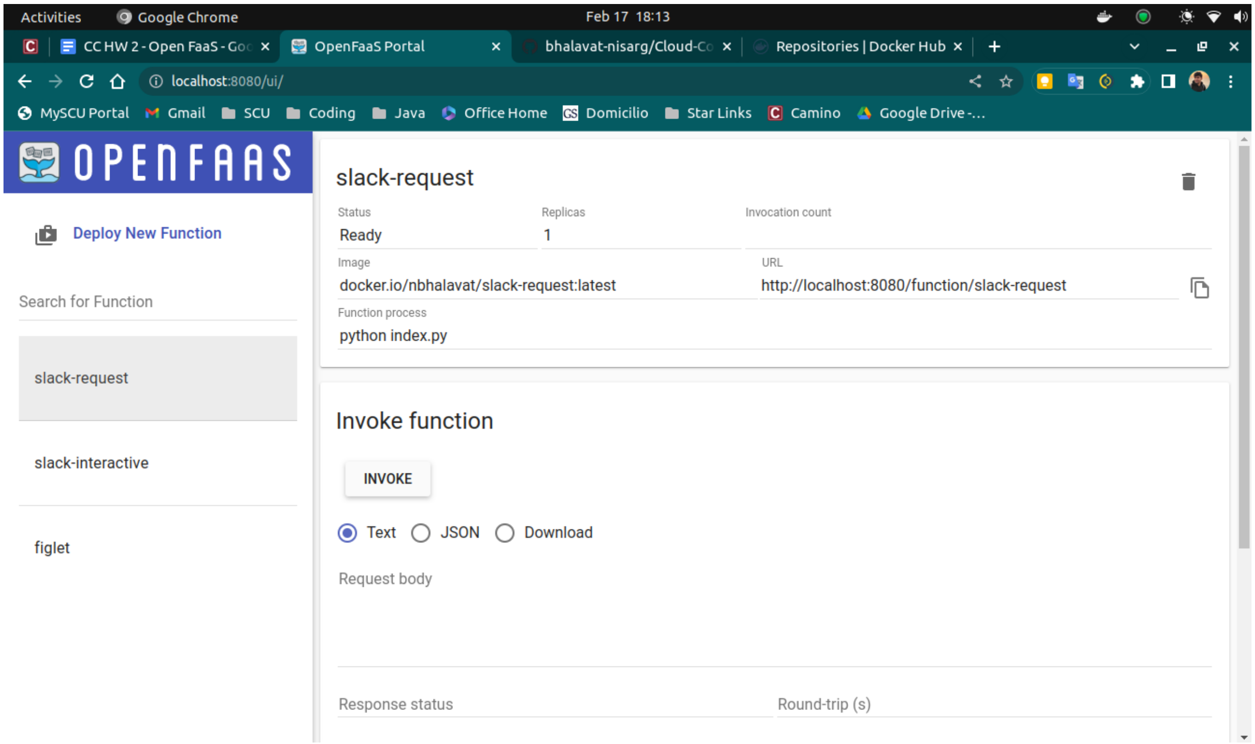


slack-interactive

Status: Ready
Replicas: 1
Invocation count:
Image: docker.io/nbhalavat/slack-interactive:latest
URL: http://localhost:8080/function/slack-interactive
Function process: python index.py

Invoke function

● Text ○ JSON ○ Download



slack-request

Status: Ready
Replicas: 1
Invocation count:
Image: docker.io/nbhalavat/slack-request:latest
URL: http://localhost:8080/function/slack-request
Function process: python index.py

Invoke function

● Text ○ JSON ○ Download

Request body

Response status

Round-trip (s)

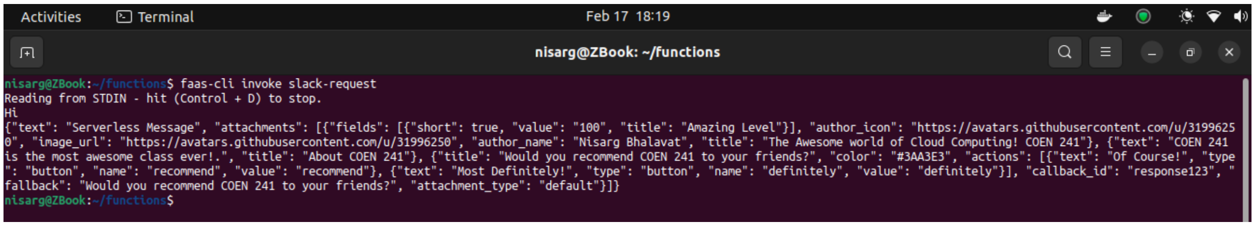
Invoking Functions

Now, we will invoke the functions from the CLI and the UI.

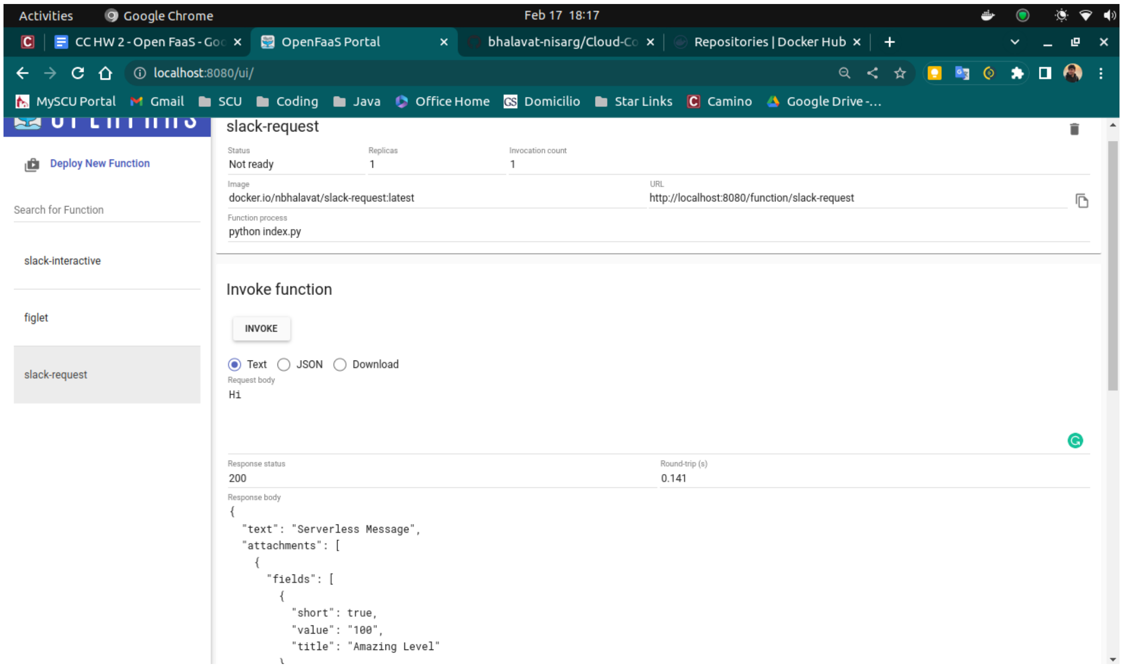
Slack Request

In the terminal, we can run the following command:

```
$ faas-cli invoke slack-request
```



```
nisarg@ZBook:~/functions$ faas-cli invoke slack-request
Reading from STDIN - hit (Control + D) to stop.
Hi
{"text": "Serverless Message", "attachments": [{"fields": [{"short": true, "value": "100", "title": "Amazing Level"}], "author_icon": "https://avatars.githubusercontent.com/u/3199625
0", "image_url": "https://avatars.githubusercontent.com/u/31996250", "author_name": "Nisarg Bhalavat", "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"}]}
nisarg@ZBook:~/functions$
```



slack-request

Status: Not ready
Replicas: 1
Invocation count: 1
Image: docker.io/nbhalavat/slack-request:latest
URL: http://localhost:8080/function/slack-request
Function process: python index.py

Invoke function

INVOKE

● Text ○ JSON ○ Download

Request body

```
Hi
```

Response status: 200
Round-trip (s): 0.141

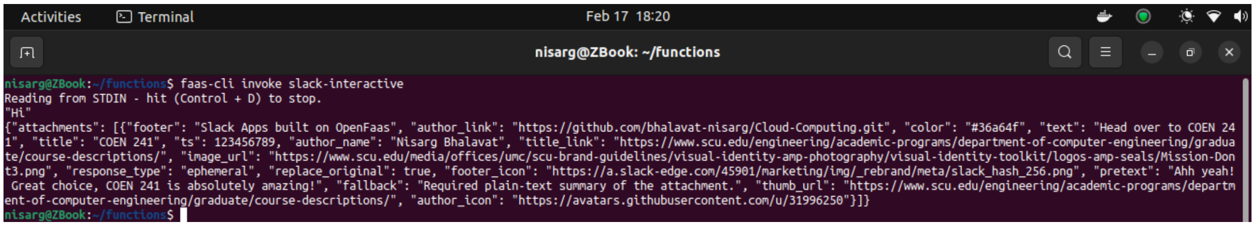
Response body

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

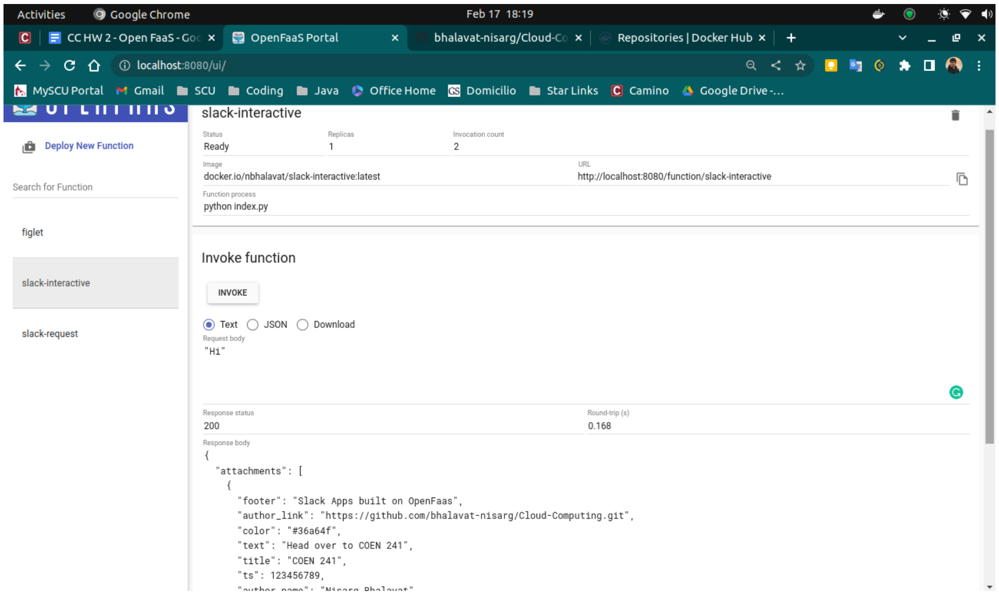
Slack Interactive

In the terminal, we can run the following command:

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



```
nisarg@ZBook:~/functions$ faas-cli invoke slack-interactive
Reading from STDIN - hit (Control + D) to stop.
"Hi"
{"attachments": [{"footer": "Slack Apps built on OpenFaas", "author_link": "https://github.com/bhalavat-nisarg/Cloud-Computing.git", "color": "#36a64f", "text": "Head over to COEN 24
1", "title": "COEN 241", "ts": 123456789, "author_name": "Nisarg Bhalavat", "title_link": "https://www.scu.edu/engineering/academic-programs/department-of-computer-engineering/gradua
te/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-Don
t3.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/departm
ent-of-computer-engineering/graduate/course-descriptions/", "author_icon": "https://avatars.githubusercontent.com/u/31996250"}]}
nisarg@ZBook:~/functions$
```



slack-interactive

Status: Ready
Replicas: 1
Invocation count: 2
Image: docker.io/nbhalavat/slack-interactive:latest
URL: http://localhost:8080/function/slack-interactive
Function process: python index.py

Invoke function

INVOKE

● Text ○ JSON ○ Download

Request body

```
"Hi"
```

Response status: 200
Round-trip (s): 0.168

Response body

```
{
  "attachments": [
    {
      "footer": "Slack Apps built on OpenFaas",
      "author_link": "https://github.com/bhalavat-nisarg/Cloud-Computing.git",
      "color": "#36a64f",
      "text": "Head over to COEN 241",
      "title": "COEN 241",
      "ts": 123456789,
      "author_name": "Nisarg Bhalavat"
```

Questions

Q1.

What is the command to invoke the slack-request function?

a. Via curl

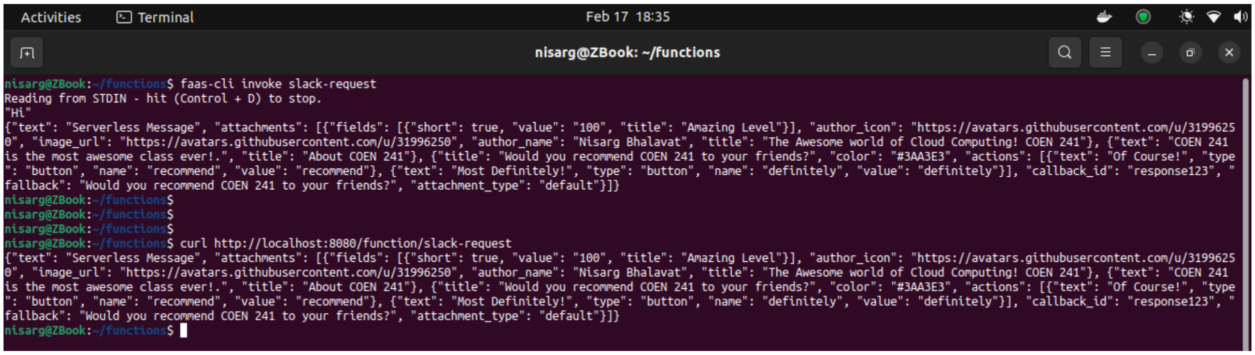
```
$ curl http://localhost:8080/function/slack-request
```

b. Via faas-cli

```
$ faas-cli invoke slack-request
```

Q2.

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



```
nisarg@ZBook:~/functions$ faas-cli invoke slack-request
Reading from STDIN - hit (Control + D) to stop.
"Hi"
{"text": "Serverless Message", "attachments": [{"fields": [{"short": true, "value": "100", "title": "Amazing Level"}], "author_icon": "https://avatars.githubusercontent.com/u/3199625
0", "image_url": "https://avatars.githubusercontent.com/u/31996250", "author_name": "Nisarg Bhalavat", "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"}]}
nisarg@ZBook:~/functions$
nisarg@ZBook:~/functions$
nisarg@ZBook:~/functions$
nisarg@ZBook:~/functions$ curl http://localhost:8080/function/slack-request
{"text": "Serverless Message", "attachments": [{"fields": [{"short": true, "value": "100", "title": "Amazing Level"}], "author_icon": "https://avatars.githubusercontent.com/u/3199625
0", "image_url": "https://avatars.githubusercontent.com/u/31996250", "author_name": "Nisarg Bhalavat", "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"}]}
nisarg@ZBook:~/functions$
```

Q3.

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

a. Via curl

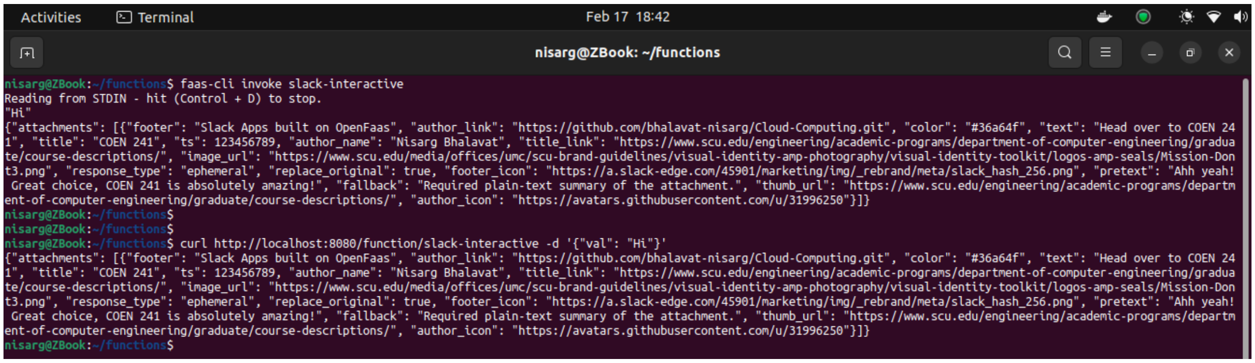
```
$ curl http://localhost:8080/ slack-interactive -d '{"key": "Hi"}'
```

b. Via faas-cli

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

Q4.

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



```
nisarg@ZBook:~/functions$ faas-cli invoke slack-interactive
Reading from STDIN - hit (Control + D) to stop.
"Hi"
{"attachments": [{"footer": "Slack Apps built on OpenFaas", "author_link": "https://github.com/bhalavat-nisarg/Cloud-Computing.git", "color": "#36a64f", "text": "Head over to COEN 24
1", "title": "COEN 241", "ts": 123456789, "author_name": "Nisarg Bhalavat", "title_link": "https://www.scu.edu/engineering/academic-programs/department-of-computer-engineering/gradua
te/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-Don
t3.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/departm
ent-of-computer-engineering/graduate/course-descriptions/", "author_icon": "https://avatars.githubusercontent.com/u/31996250"}]}
nisarg@ZBook:~/functions$
nisarg@ZBook:~/functions$
nisarg@ZBook:~/functions$ curl http://localhost:8080/function/slack-interactive -d '{"val": "Hi"}'
{"attachments": [{"footer": "Slack Apps built on OpenFaas", "author_link": "https://github.com/bhalavat-nisarg/Cloud-Computing.git", "color": "#36a64f", "text": "Head over to COEN 24
1", "title": "COEN 241", "ts": 123456789, "author_name": "Nisarg Bhalavat", "title_link": "https://www.scu.edu/engineering/academic-programs/department-of-computer-engineering/gradua
te/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-Don
t3.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/departm
ent-of-computer-engineering/graduate/course-descriptions/", "author_icon": "https://avatars.githubusercontent.com/u/31996250"}]}
nisarg@ZBook:~/functions$
```

Q5.

How would you pass different arguments to the functions?

We have various ways to pass different arguments to the function, i.e., if we are using curl, then we can provide data using the “-d” flag, use “-X” to provide a different REST method or use “-H” to provide a custom header to the function. In some cases like for Figlet, we are using a pipe operator to pass the argument to the function with the echo command. Additionally, if we have access to the web UI portal for OpenFaaS, then we can pass arguments directly into the UI.

Q6.

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

In our handler.py file, we are taking the request payload, which is transformed to JSON to Object if the string is valid. We can use that Object to manipulate the response, i.e., either send back the response based on the input parameter received or pass it as part of the response body. The highlighted line in the following screenshot shows one of the methods to change the response based on different input parameters.

Extra Credit

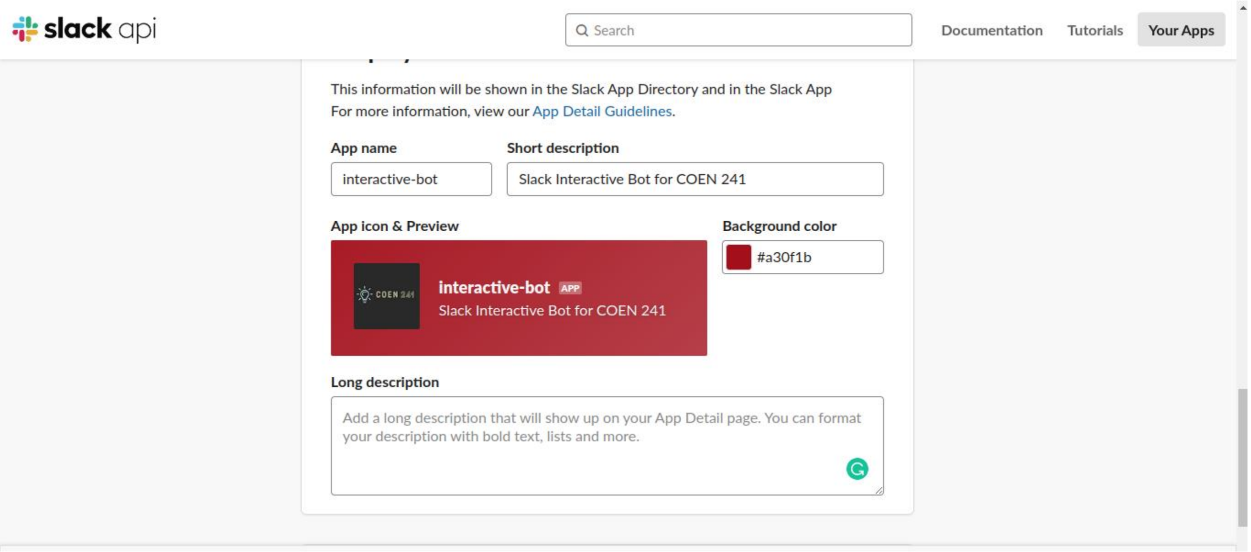
The slack interactive bot was successfully tested on slack by adding it as a function running from my localhost tunneled using ngrok for exposing my localhost to the internet.

The app can be found on following URL:

https://api.slack.com/apps/A04QANJ6NAY/app-home?

The channel where the app was tested can be joined using following link (please use @scu.edu email):

https://join.slack.com/t/coen-241-cc/shared_invite/zt-1pk92dkgi-R48pfnFGYXcq39H1ONotnw



This information will be shown in the Slack App Directory and in the Slack App
For more information, view our App Detail Guidelines.

App name: interactive-bot

Short description: Slack Interactive Bot for COEN 241

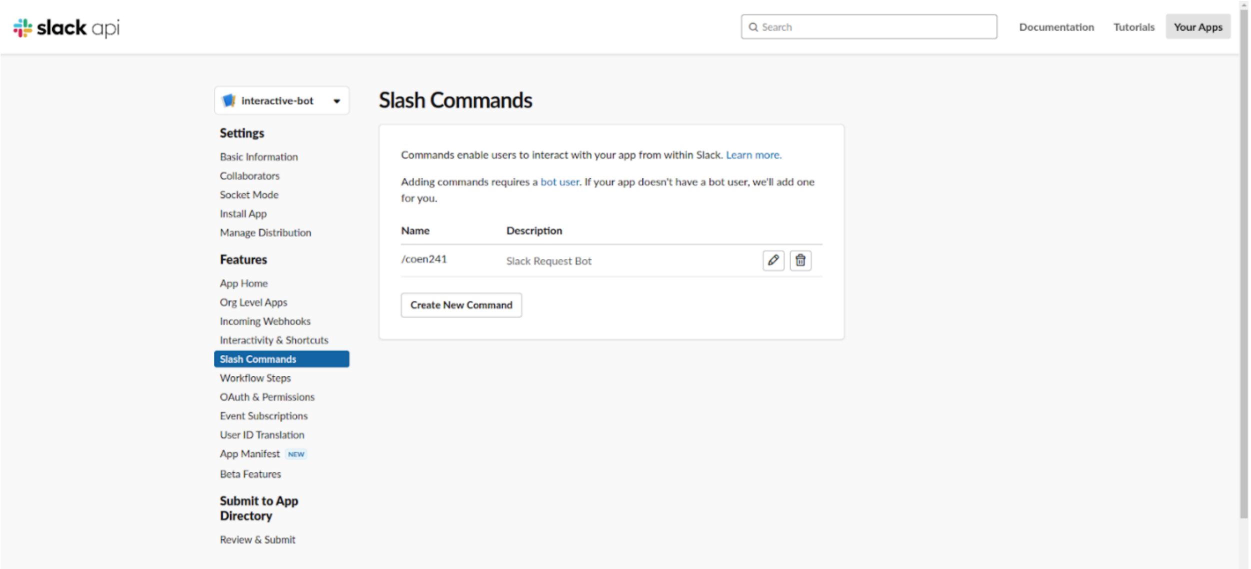
App icon & Preview

interactive-bot APP
Slack Interactive Bot for COEN 241

Background color: #a30f1b

Long description

Add a long description that will show up on your App Detail page. You can format your description with bold text, lists and more.

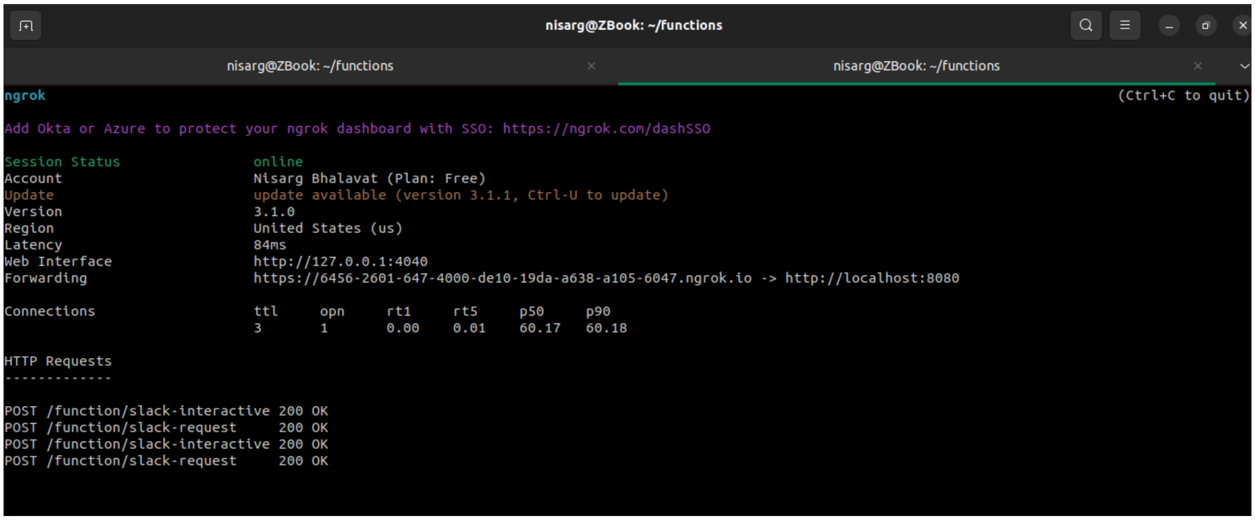


Slash Commands

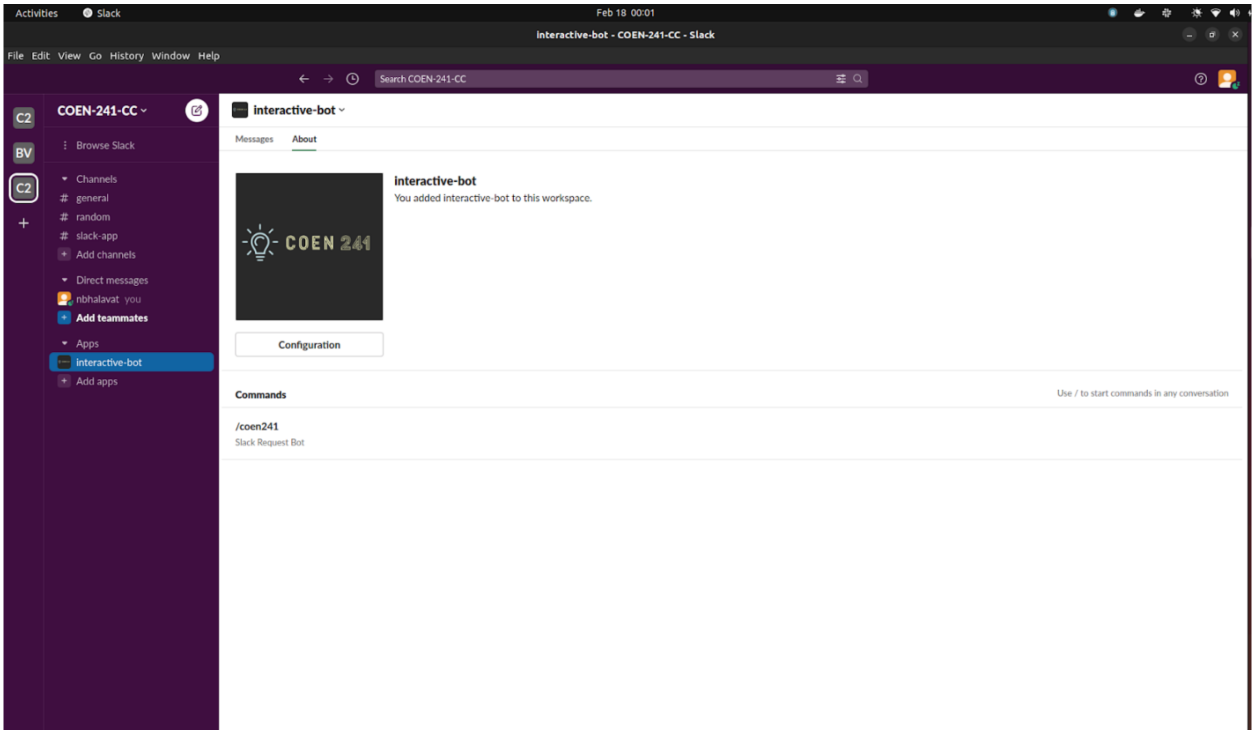
Commands enable users to interact with your app from within Slack. Learn more.

Adding commands requires a bot user. If your app doesn't have a bot user, we'll add one for you.

Name	Description
/coen241	Slack Request Bot


nisarg@ZBook: ~/functions
nisarg@ZBook: ~/functions
nisarg@ZBook: ~/functions

```
ngrok                                                                                       (Ctrl+C to quit)

Add Okta or Azure to protect your ngrok dashboard with SSO: https://ngrok.com/dashSSO

Session Status                online
Account                       Nisarg Bhalavat (Plan: Free)
Update                        update available (version 3.1.1, Ctrl-U to update)
Version                       3.1.0
Region                        United States (us)
Latency                       84ms
Web Interface                 http://127.0.0.1:4040
Forwarding                    https://6456-2601-647-4000-de10-19da-a638-a105-6047.ngrok.io -> http://localhost:8080

Connections                   ttl     opn     rt1     rt5     p50     p90
                              3       1       0.00    0.01    60.17   60.18

HTTP Requests
-------------

POST /function/slack-interactive 200 OK
POST /function/slack-request     200 OK
POST /function/slack-interactive 200 OK
POST /function/slack-request     200 OK
```



interactive-bot

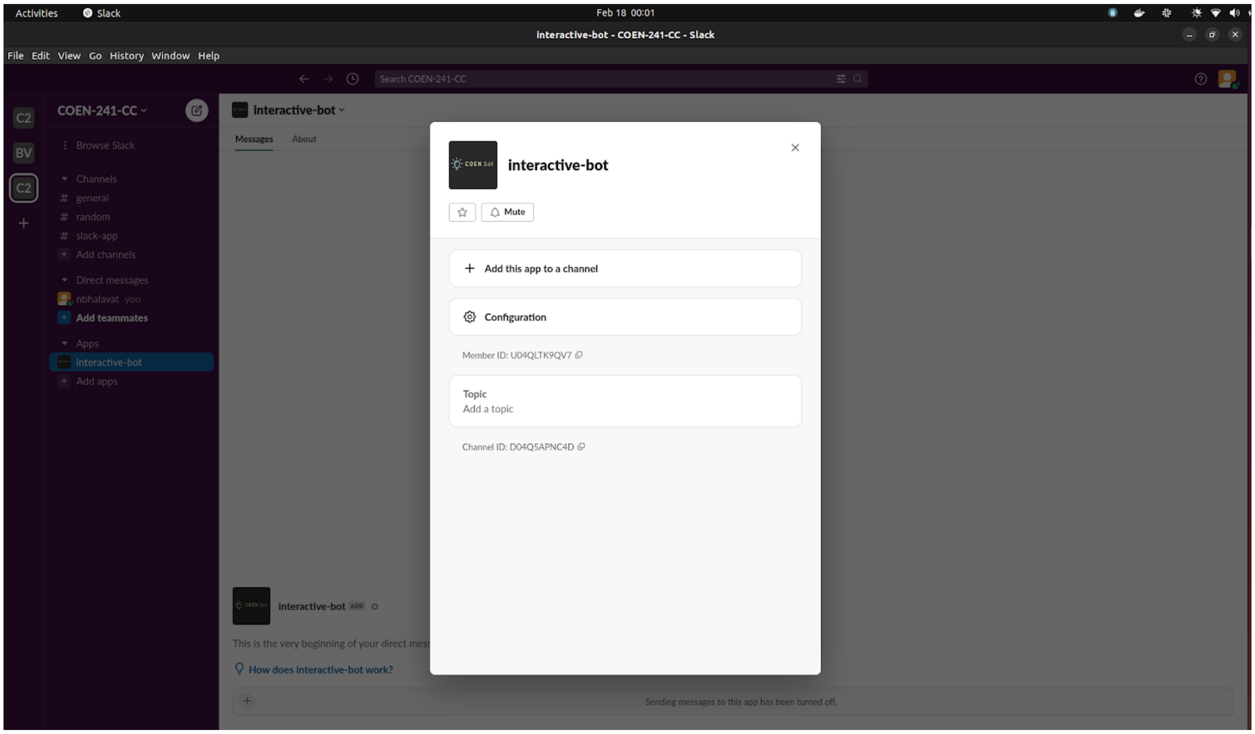
You added interactive-bot to this workspace.

Commands

Use / to start commands in any conversation

/coen241

Slack Request Bot



This is the very beginning of your direct mes

Sending messages to this app has been turned off.

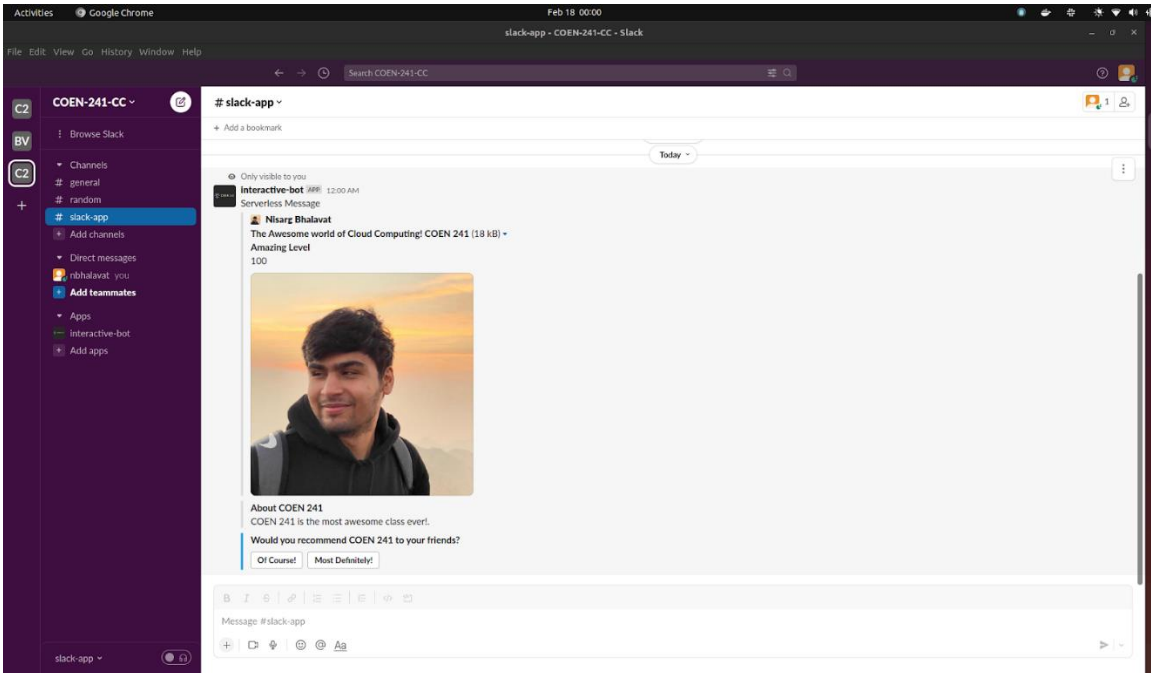
interactive-bot

Member ID: U04QLTK9QV7

Topic

Add a topic

Channel ID: D04Q5APNC4D



Only visible to you

interactive-bot APP 12:00 AM

Serverless Message

Nisarg Bhalavat

The Awesome world of Cloud Computing! COEN 241 (18 kB)

Amazing Level

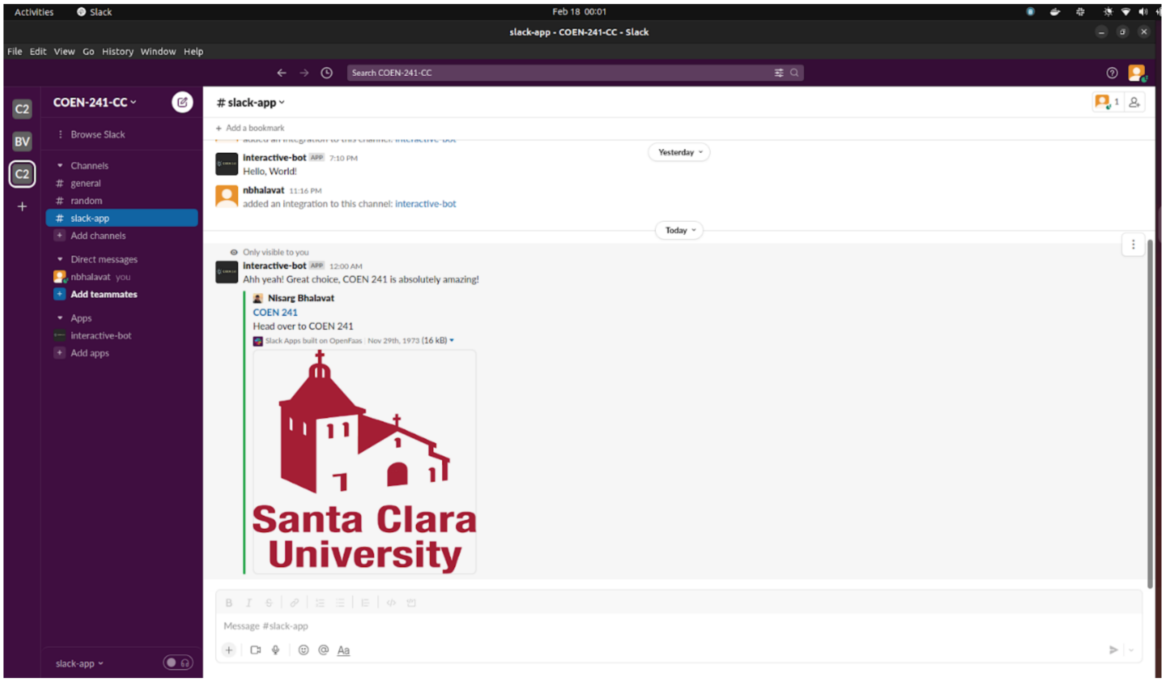
100

About COEN 241

COEN 241 is the most awesome class ever!.

Would you recommend COEN 241 to your friends?

Of Course! | Most Definitely!



interactive-bot APP 7:10 PM

Hello, World!

nbhalavat 11:16 PM

added an integration to this channel: interactive-bot

Only visible to you

interactive-bot APP 12:00 AM

Ahh yeah! Great choice, COEN 241 is absolutely amazing!

Nisarg Bhalavat

COEN 241

Head over to COEN 241

Slack Apps built on OpenFaas | Nov 29th, 1973 (16 kB)

Git Repository

Account Name	bhalavat-nisarg
Repository Name	Cloud-Computing
Folder Containing HW	HW2
Link to Repo	https://github.com/bhalavat-nisarg/Cloud-Computing