

COMSE6156 - Topics in SW Engineering:

Cloud Computing

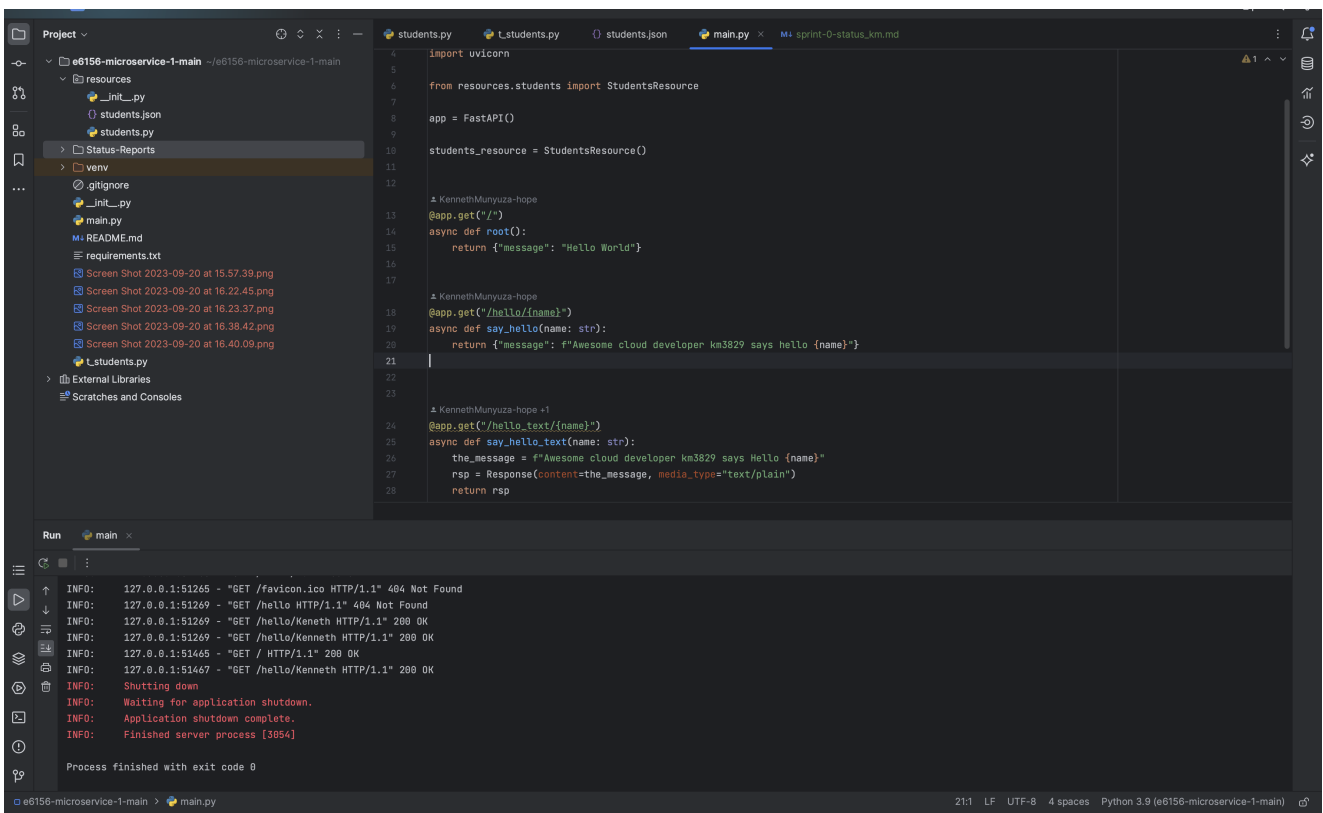
Sprint 0 Status Report KM

Overview

Sprint 0

Clone and Modify the Starter Project

- Place a screen capture of your directory below. Display the structure in a terminal window, PyCharm, file explorer or any other mechanism.



The screenshot displays the PyCharm IDE interface. On the left, the Project Explorer shows the directory structure of the cloned project 'e6156-microservice-1-main'. The structure includes a 'resources' folder with 'init_.py', 'students.json', and 'students.py'; a 'Status-Reports' folder; a 'venv' folder; and a 'gitignore' file. The main editor window shows the 'main.py' file, which contains the following Python code:

```
1 import uvicorn
2
3 from resources.students import StudentsResource
4
5
6 app = FastAPI()
7
8 students_resource = StudentsResource()
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

The terminal window at the bottom shows the output of the application, including HTTP test results:

```
INFO: 127.0.0.1:51265 - "GET /favicon.ico HTTP/1.1" 404 Not Found
INFO: 127.0.0.1:51269 - "GET /hello HTTP/1.1" 404 Not Found
INFO: 127.0.0.1:51269 - "GET /hello/Kenneth HTTP/1.1" 200 OK
INFO: 127.0.0.1:51269 - "GET /hello/Kenneth HTTP/1.1" 200 OK
INFO: 127.0.0.1:51465 - "GET / HTTP/1.1" 200 OK
INFO: 127.0.0.1:51467 - "GET /hello/Kenneth HTTP/1.1" 200 OK
INFO: Shutting down
INFO: Waiting for application shutdown.
INFO: Application shutdown complete.
INFO: Finished server process [3054]

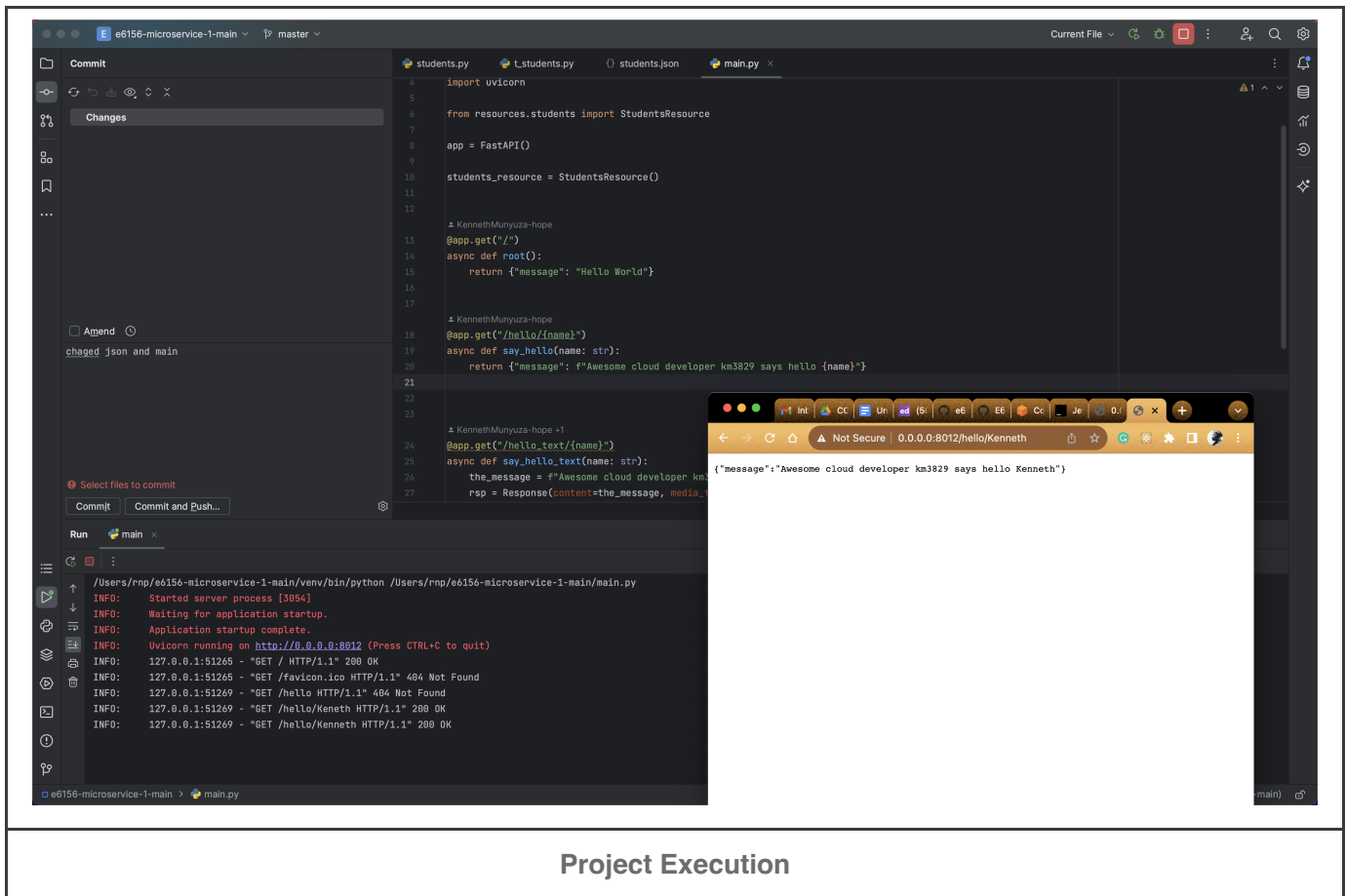
Process finished with exit code 0
```

Below the screenshot, the text 'Cloned Project' is displayed.

Cloned Project

Execute the Starter Project

You can choose any formats for the screen capture as long as it shows your project executing. My simple example shows and execution window and browser side-by-side.



Create Your Own GitHub Project

Make a copy of the project and create a project in GitHub. Show the project.

Kennethm-spec / e6156-microservice-1

Code

Issues

Pull requests

Actions

Projects

Security

Insights

Settings

Unwatch 1

Fork 0

Star 0

master

1 branch

0 tags

Go to file

Add file

Code

Kennethm-spec added gitignore for venv

978cec4 last week

4 commits

.idea

new repo

last week

resources

required changes completed: (uni, file path)

last week

.gitignore

added gitignore for venv

last week

README.md

new repo

last week

__init__.py

new repo

last week

main.py

required changes completed: (uni, file path)

last week

requirements.txt

new repo

last week

t_students.py

new repo

last week

README.md

E6156 - Topics in SW Engineering: Cloud Computing

First Example Microservice

Author: Donald F. Ferguson

Local Project Execution

Setup

- The first step for local project execution is to create a [virtual environment](#). This is a best practice for developing, testing and running multiple Python applications on a single system. There are many [explanations](#) for the benefits of virtual environments.

About

No description, website, or topics provided.

Readme

Activity

0 stars

1 watching

0 forks

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

Languages

Python 100.0%

Suggested Workflows

Based on your tech stack

Actions Importer

Set up

Automatically convert C/CD files to YAML for GitHub Actions.

Project on GitHub

Deploy and Test Project on AWS

Show the information about your VM through the console.

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#instanceDetails:instanceId=i-04964ad74b9b03f16

Services

Search

[Option+S]

New EC2 Experience

EC2 Dashboard

EC2 Global View

Events

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity Reservations

Images

AMIs

AMI Catalog

Elastic Block Store

Volumes

Snapshots

Lifecycle Manager

Network & Security

Security Groups

Elastic IPs

Placement Groups

Key Pairs

Network Interfaces

Load Balancing

Load Balancers

CloudShell

Feedback

EC2 > Instances > i-04964ad74b9b03f16

Instance summary for i-04964ad74b9b03f16 (microservice)

Updated less than a minute ago

Connect

Instance state

Actions

Instance ID

i-04964ad74b9b03f16 (microservice)

IPv6 address

-

Hostname type

IP name: ip-172-31-47-65.ec2.internal

Answer private resource DNS name

IPv4 (A)

Auto-assigned IP address

54.83.135.77 [Public IP]

IAM Role

-

IMDSv2

Required

Public IPv4 address

54.83.135.77 [open address]

Instance state

Running

Private IP DNS name (IPv4 only)

ip-172-31-47-65.ec2.internal

Instance type

t2.micro

VPC ID

vpc-0f13838fd2407a2a1

Subnet ID

subnet-0a32ef80a518adf21

Private IPv4 addresses

172.31.47.65

Public IPv4 DNS

ec2-54-83-135-77.compute-1.amazonaws.com [open address]

Elastic IP addresses

-

AWS Compute Optimizer finding

Opt-in to AWS Compute Optimizer for recommendations. | Learn more

Auto Scaling Group name

-

Details

Security

Networking

Storage

Status checks

Monitoring

Tags

Instance details

Platform

Amazon Linux (Inferred)

Platform details

Linux/UNIX

Stop protection

Disabled

Instance auto-recovery

Default

AMI ID

ami-04cb4ca688797756f

AMI name

al2023-ami-2023.1.20230912.0-kernel-6.1-x86_64

Launch time

Wed Sep 20 2023 16:31:54 GMT-0400 (Eastern Daylight Time) (8 minutes)

Lifecycle

normal

Monitoring

disabled

Termination protection

Disabled

AMI location

amazon/al2023-ami-2023.1.20230912.0-kernel-6.1-x86_64

Stop-hibernate behavior

Disabled

AMI launch index

Key pair assigned at launch

Stop-termination reason

© 2023, Amazon Web Services, Inc. or its affiliates.

Privacy

Terms

Cookie preferences

AWS Console

Show the application running in the EC2 terminal and browser.

The screenshot displays the AWS Management Console interface. On the left, the 'Services' menu is visible, and the 'Instances' page is selected. An EC2 instance named 'i-04964ad74b9b03f16 (microservice)' is highlighted. The 'Monitor' tab is active, showing the instance's status as 'Running'. Below the status, the public and private IP addresses are listed: Public IPs: 54.83.135.77, Private IPs: 172.31.47.65.

On the right, a browser window is open, showing the application running on the EC2 instance. The browser address bar displays 'Not Secure | ec2-54-83-135-77.compute-1.amazonaws.com:8012'. The page content shows a JSON message: `{\"message\": \"Awesome cloud developer km3829 says hello Kenneth\"}`.

The terminal window on the left shows the following commands and output:

```
Downloading typing_extensions-4.7.1-py3-none-any.whl (33 kB)
Collecting uvicorn==0.23.2
  Downloading uvicorn-0.23.2-py3-none-any.whl (59 kB)
    59 kB 8.5 MB/s
Collecting exceptiongroup
  Downloading exceptiongroup-1.1.3-py3-none-any.whl (14 kB)
Installing collected packages: typing-extensions, sniffio, idna, exceptiongroup, pydantic-core,
anyio, annotated-types, starlette, pydantic, h11, click, uvicorn, fastapi
ERROR: pip's dependency resolver does not currently take into account all the packages that are
installed. This behaviour is the source of the following dependency conflicts.
requests 2.25.1 requires idna<3,>=2.5, but you have idna 3.4 which is incompatible.
Successfully installed annotated-types-0.5.0 anyio-3.7.1 click-8.1.7 exceptiongroup-1.1.3 fasta
pi-0.103.1 h11-0.14.0 idna-3.4 pydantic-2.3.0 pydantic-core-2.6.3 sniffio-1.3.0 starlette-0.27.
0 typing-extensions-4.7.1 uvicorn-0.23.2
[ec2-user@ip-172-31-47-65 e6156-microservice-1]$ python --version
Python 3.9.16
[ec2-user@ip-172-31-47-65 e6156-microservice-1]$ python main.py
INFO: Started server process [25472]
INFO: Waiting for application startup.
INFO: Application startup complete.
INFO: Uvicorn running on http://0.0.0.0:8012 (Press CTRL+C to quit)
^CINFO: Shutting down
INFO: Waiting for application shutdown.
INFO: Application shutdown complete.
INFO: Finished server process [25472]
[ec2-user@ip-172-31-47-65 e6156-microservice-1]$ python main.py &
[1] 25473
[ec2-user@ip-172-31-47-65 e6156-microservice-1]$ INFO: Started server process [25473]
INFO: Waiting for application startup.
INFO: Application startup complete.
INFO: Uvicorn running on http://0.0.0.0:8012 (Press CTRL+C to quit)

[ec2-user@ip-172-31-47-65 e6156-microservice-1]$ curl 0.0.0.0:8012
INFO: 127.0.0.1:138900 - "GET / HTTP/1.1" 200 OK
{"message": "Hello World"}[ec2-user@ip-172-31-47-65 e6156-microservice-1]$ curl 0.0.0.0:8012/hel
lo/Kenneth
INFO: 127.0.0.1:152650 - "GET /hello/Kenneth HTTP/1.1" 200 OK
{"message": "Awesome cloud developer km3829 says hello Kenneth"}[ec2-user@ip-172-31-47-65 e6156-
microservice-1]$ INFO: 209.2.224.52:51444 - "GET / HTTP/1.1" 200 OK
INFO: 209.2.224.52:51444 - "GET /favicon.ico HTTP/1.1" 404 Not Found
INFO: 209.2.224.52:51448 - "GET /hello/Kenneth HTTP/1.1" 200 OK
INFO: 23.228.131.196:45536 - "GET /hello/Kenneth HTTP/1.1" 200 OK
```