

DevOps Networks

1. Communication issues?
2. Create network

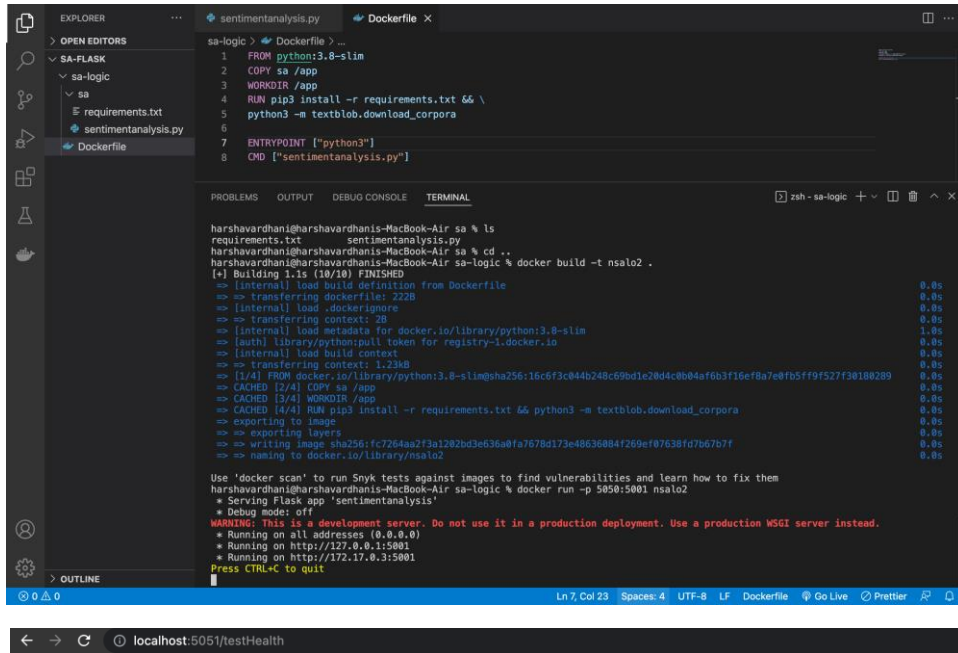
```
Last login: Thu Oct 13 17:09:10 on ttys000
harshavardhani@harshavardhanis-MacBook-Air ~ % docker network create --driver bridge sa_network
fee2717c9dfcf3df3aa1348616c0b0ca677b8f4d700aad2dcb4c1f6274ea9ddb
harshavardhani@harshavardhanis-MacBook-Air ~ % docker network inspect sa_network
[
  {
    "Name": "sa_network",
    "Id": "fee2717c9dfcf3df3aa1348616c0b0ca677b8f4d700aad2dcb4c1f6274ea9ddb",
    "Created": "2022-10-17T01:33:03.48488809Z",
    "Scope": "local",
    "Driver": "bridge",
    "EnableIPv6": false,
    "IPAM": {
      "Driver": "default",
      "Options": {},
      "Config": [
        {
          "Subnet": "172.18.0.0/16",
          "Gateway": "172.18.0.1"
        }
      ]
    },
    "Internal": false,
    "Attachable": false,
    "Ingress": false,
    "ConfigFrom": {
      "Network": ""
    },
    "ConfigOnly": false,
    "Containers": {},
    "Options": {},
    "Labels": {}
  }
]
harshavardhani@harshavardhanis-MacBook-Air ~ %
```

3. Connect a container to a network

```
harshavardhani@harshavardhanis-MacBook-Air ~ % docker run --network=sa_network -itd --name=sa-fe-container sa-fe-image
Unable to find image 'sa-fe-image:latest' locally
docker: Error response from daemon: pull access denied for sa-fe-image, repository does not exist or may require 'docker login': denied: requested access to the resource is denied.
See 'docker run --help'.
harshavardhani@harshavardhanis-MacBook-Air ~ % docker container ls
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS           NAMES
46ba8f2ac787   harsharachagri02/sentiment-analysis-frontend  "/docker-entrypoint..."  39 minutes ago  Up 39 minutes  0.0.0.0:8091->80/tcp  elegant_joliot
harshavardhani@harshavardhanis-MacBook-Air ~ % docker run --network=sa_network -itd --name=elegant_joliot sa-fe-image
Unable to find image 'sa-fe-image:latest' locally
docker: Error response from daemon: pull access denied for sa-fe-image, repository does not exist or may require 'docker login': denied: requested access to the resource is denied.
See 'docker run --help'.
harshavardhani@harshavardhanis-MacBook-Air ~ % docker run --network=sa_network -itd --name=harsharachagri02/sentiment-analysis-frontend sa-fe-image
Unable to find image 'sa-fe-image:latest' locally
docker: Error response from daemon: pull access denied for sa-fe-image, repository does not exist or may require 'docker login': denied: requested access to the resource is denied.
See 'docker run --help'.
harshavardhani@harshavardhanis-MacBook-Air ~ % docker run --network=sa_network -itd --name=elegant_joliot2 sa-fe-image
Unable to find image 'sa-fe-image:latest' locally
docker: Error response from daemon: pull access denied for sa-fe-image, repository does not exist or may require 'docker login': denied: requested access to the resource is denied.
See 'docker run --help'.
harshavardhani@harshavardhanis-MacBook-Air ~ % docker run --network=sa_network -itd --name=elegant_joliot harsharachagri02/sentiment-analysis-frontend
docker: Error response from daemon: Conflict. The container name "/elegant_joliot" is already in use by container "46ba8f2ac78728a63388c9af552bab68fc9c5b341d21a58d4b7d2ce799fede". You have to remove (or rename) that container to be able to reuse that name.
See 'docker run --help'.
harshavardhani@harshavardhanis-MacBook-Air ~ % docker run --network=sa_network -itd --name=elegant_joliot2 harsharachagri02/sentiment-analysis-frontend
8c547c9266c89d74ab76d4d46b834558a715e37c2a9d6d8e88bf524ebd7e8a0
harshavardhani@harshavardhanis-MacBook-Air ~ % docker container ls
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS           NAMES
8c547c9266c8   harsharachagri02/sentiment-analysis-frontend  "/docker-entrypoint..."  About a minute ago  Up About a minute  80/tcp           elegant_joliot2
46ba8f2ac787   harsharachagri02/sentiment-analysis-frontend  "/docker-entrypoint..."  48 minutes ago    Up 48 minutes    0.0.0.0:8091->80/tcp  elegant_joliot
harshavardhani@harshavardhanis-MacBook-Air ~ %
```

4. SA-LOGIC:

BUILD & RUN



The screenshot shows the VS Code interface with the Explorer sidebar on the left, the Dockerfile editor in the center, and the Terminal at the bottom. The Dockerfile contains the following content:

```
sa-logic > Dockerfile > ...
1 FROM python:3.8-slim
2 COPY sa /app
3 WORKDIR /app
4 RUN pip3 install -r requirements.txt && \
5 python3 -m textblob.download_corpora
6
7 ENTRYPOINT ["python3"]
8 CMD ["sentimentanalysis.py"]
```

The Terminal shows the output of the build and run commands:

```
harshavardhan@harshavardhanis-MacBook-Air sa % ls
requirements.txt  sentimentanalysis.py
harshavardhan@harshavardhanis-MacBook-Air sa % cd ..
harshavardhan@harshavardhanis-MacBook-Air sa-logic % docker build -t nsalo2 .
[*] Building 1.1s (10/10) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 222B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/python:3.8-slim
=> [internal] load metadata for registry-1.docker.io
=> [internal] load build context
=> => transferring context: 1.23kB
=> [1/4] FROM docker.io/library/python:3.8-slimgha256:16c6f3c044b248c69bd1e2044c0b04af6b3f16ef8a7e0fb5f9f527f30100209
=> CACHED [2/4] COPY sa /app
=> CACHED [3/4] WORKDIR /app
=> CACHED [4/4] RUN pip3 install -r requirements.txt && python3 -m textblob.download_corpora
=> exporting to image
=> writing image sha256:fc7264aa273a1282bd3e636a0fa7678d173e40630804f209ef07638fd7067b7f
=> naming to docker.io/library/nsalo2

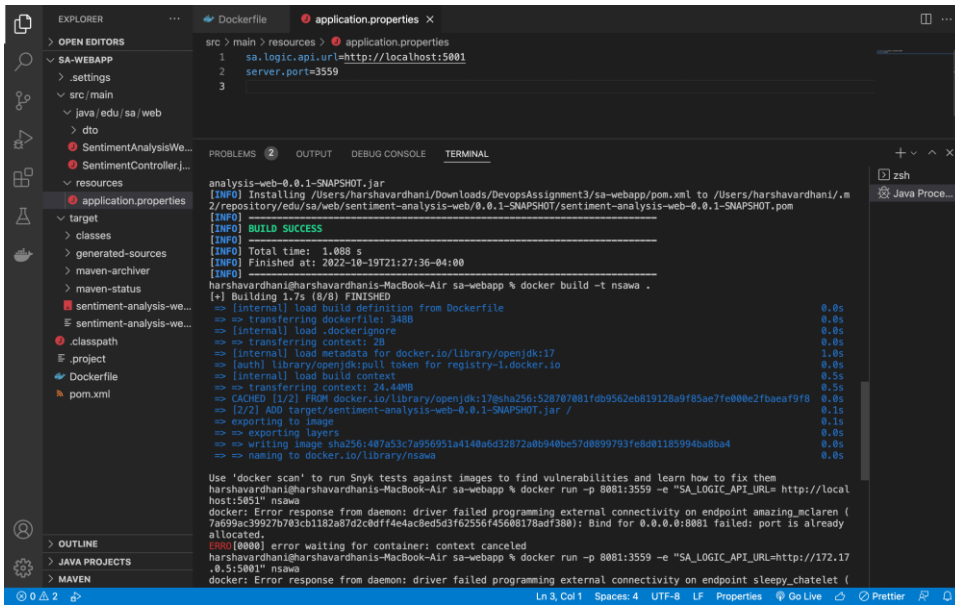
Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
harshavardhan@harshavardhanis-MacBook-Air sa-logic % docker run -p 5050:5001 nsalo2
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5001
* Running on http://172.17.0.3:5001
Press CTRL+C to quit
```

The bottom status bar shows the browser address bar with the URL `localhost:5051/testHealth`.

Hello from python sentiment analysis flask app!

5. SA-WEBAPP

-- BUILD



The screenshot shows the VS Code interface with the Explorer on the left showing the project structure. The main editor displays the `Dockerfile` and `application.properties` files. The `application.properties` file contains:

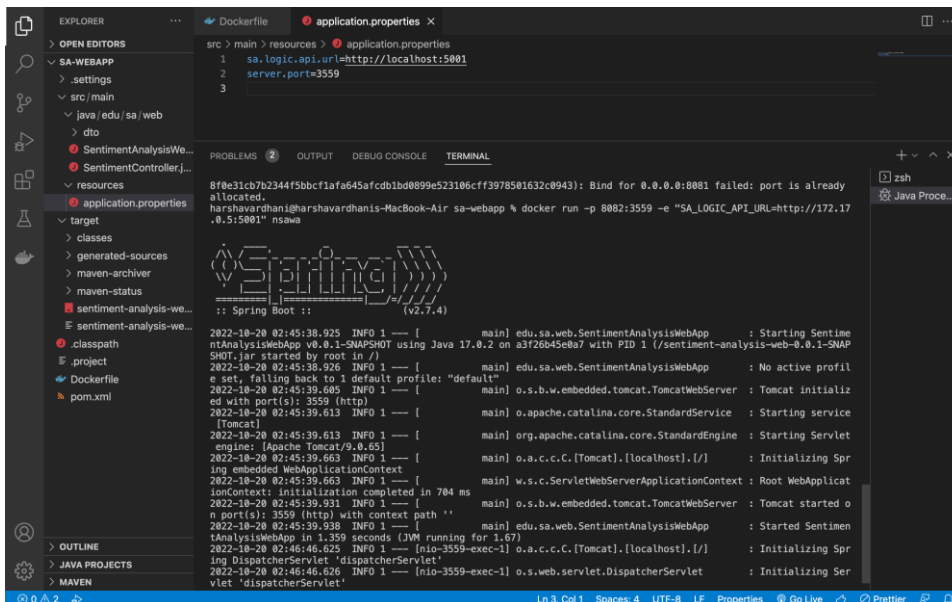
```
src > main > resources > application.properties
1 sa.logic.api.url=http://localhost:5001
2 server.port=3559
3
```

The terminal output shows the Docker build process:

```
analysis-web-0.0.1-SNAPSHOT.jar
[INFO] Installing /Users/harshavardhani/Downloads/DevopsAssignment3/sa-webapp/pom.xml to /Users/harshavardhani/.m2/repository/edu/sa/web/sentiment-analysis-web/0.0.1-SNAPSHOT/sentiment-analysis-web-0.0.1-SNAPSHOT.pom
[INFO] BUILD SUCCESS
[INFO] Total time: 1.088 s
[INFO] Finished at: 2022-10-19T21:27:36-04:00
[INFO]
harshavardhani@harshavardhani-MacBook-Air sa-webapp % docker build -t nsawa .
[+] Building 1.7s (8/8) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring Dockerfile: 348B 0.0s
=> [internal] load .dockerignore 0.0s
=> => transferring context: 2B 0.0s
=> [internal] load metadata for docker.io/library/openjdk:17 1.0s
=> [auth] library/openjdk:pull token for registry-1.docker.io 0.0s
=> [internal] load build context 0.5s
=> => transferring context: 24.40kB 0.5s
=> CACHED [1/2] FROM docker.io/library/openjdk:17@sha256:526707081f0b9562eb81912ba9f85ac7fe00e2fbaea19f8 0.0s
=> [2/2] ADD target/sentiment-analysis-web-0.0.1-SNAPSHOT.jar / 0.1s
=> => exporting layers 0.1s
=> => exporting image sha256:407a53c7a956951a4140a6d32b72a0b940be57d0899793febd01185994ba8ba4 0.0s
=> => naming to docker.io/library/nsawa 0.0s

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
harshavardhani@harshavardhani-MacBook-Air sa-webapp % docker run -p 8081:3559 -e "SA_LOGIC_API_URL=http://localhost:5001" nsawa
docker: Error response from daemon: driver failed programming external connectivity on endpoint amazing_mclaren (7a699ac39927b783cb1182a87d2c8dff4e4ac8ed5d3f62556f45608178adf380): Bind for 0.0.0.0:8081 failed: port is already allocated.
[8081] error waiting for container: context canceled
harshavardhani@harshavardhani-MacBook-Air sa-webapp % docker run -p 8081:3559 -e "SA_LOGIC_API_URL=http://172.17.0.5:5001" nsawa
docker: Error response from daemon: driver failed programming external connectivity on endpoint sleepy_chatelet (
```

-- RUN



The screenshot shows the VS Code interface with the terminal output of a successful Docker run. The terminal output displays the Spring Boot application startup logs:

```
8f0e31cb7b2344f5bbc1fa645afcd1bd0889e523106cff3978501632c0943): Bind for 0.0.0.0:8081 failed: port is already allocated.
harshavardhani@harshavardhani-MacBook-Air sa-webapp % docker run -p 8082:3559 -e "SA_LOGIC_API_URL=http://172.17.0.5:5001" nsawa
:: Spring Boot :: (v2.7.4)

2022-10-20 02:45:38.925 INFO 1 --- [main] edu.sa.web.SentimentAnalysisWebApp : Starting Sentimen
tAnalysisWebApp v0.0.1-SNAPSHOT using Java 17.0.2 on a3f28045e0a7 with PID 1 (/sentiment-analysis-web-0.0.1-SNAP
SHOT.jar started by root in /)
2022-10-20 02:45:38.926 INFO 1 --- [main] edu.sa.web.SentimentAnalysisWebApp : No active profil
e set, falling back to 1 default profile: "default"
2022-10-20 02:45:39.605 INFO 1 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initializ
ed with port(s): 3559 (http)
2022-10-20 02:45:39.613 INFO 1 --- [main] o.apache.catalina.core.StandardService : Starting service
[Tomcat]
2022-10-20 02:45:39.613 INFO 1 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet
engine: [Apache Tomcat/9.0.65]
2022-10-20 02:45:39.663 INFO 1 --- [main] o.a.c.c.C.[Tomcat].[localhost]./ : Initializing Spr
ing embedded WebApplicationContext
2022-10-20 02:45:39.663 INFO 1 --- [main] w.s.c.ServletWebServerApplicationContext : Root WebApplicat
ionContexts initialization completed in 784 ms
2022-10-20 02:45:39.931 INFO 1 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started o
n port(s): 3559 (http) with context path ''
2022-10-20 02:45:39.938 INFO 1 --- [main] edu.sa.web.SentimentAnalysisWebApp : Started Sentimen
tAnalysisWebApp in 1.359 seconds (JVM running for 1.67)
2022-10-20 02:46:46.625 INFO 1 --- [nio-3559-exec-1] o.a.c.c.C.[Tomcat].[localhost]./ : Initializing Spr
ing DispatcherServlet 'dispatcherServlet'
2022-10-20 02:46:46.626 INFO 1 --- [nio-3559-exec-1] o.s.web.servlet.DispatcherServlet : Initializing Ser
vlet 'dispatcherServlet'
```

localhost:8082/testHealth

hello world!

```
--BUILD
```

The screenshot shows a VS Code editor with a Dockerfile and its build output. The Dockerfile defines a Node.js application with a health check. The build output shows the successful creation of a Docker image.

```

10 super(props); this.state = {
11   sentence: '',
12   polarity: undefined;
13 };
14 analyzeSentence() {
15   fetch('http://localhost:8082/sentence', {
16     method: 'POST',
17     headers: {
18       'Content-Type': 'application/json'
19     },
20     body: JSON.stringify({sentence: this.textField.getValue()})
21   })
22   .then(response => response.json())
23   .then(data => this.setState(data));
24 }
25
26 // EnterPress = e => {
27
28
29 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```

http://bit.ly/2vY8Kkr

harshavardhani@harshavardhani-MacBook-Air sa-frontend % docker build -t nsafe .
[*] Building 0.2s (7/7) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 170B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/nginx:latest
=> [internal] load build context
=> transferring context: 1.89MB
=> [1/2] FROM docker.io/library/nginx
=> CACHED [2/2] COPY build /usr/share/nginx/html
=> exporting to image
=> exporting layers
=> writing image sha256:4edbdaf89991ec879f7b440726115a7611cdce637db4883b6f947cf949
=> naming to docker.io/library/nsafe

```

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
harshavardhani@harshavardhani-MacBook-Air sa-frontend % docker run -p 8086:80 nsafe
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf

```
--RUN
```

The screenshot shows the VS Code interface with the Dockerfile for 'sa-frontend' open in the editor. The Dockerfile contains instructions to build an image from a local source, set the entrypoint to a custom script, and expose port 8082. The terminal shows the output of the 'docker build' command, which successfully built the image 'sa-frontend' and pushed it to Docker Hub.

```

Dockerfile
10 super(props); this.state = {
11   sentences: [],
12   polarity: undefined;
13 };
14 analyzeSentence() {
15   fetch('http://localhost:8082/sentiment', {
16     method: 'POST',
17     headers: {
18       'Content-Type': 'application/json'
19     },
20     body: JSON.stringify({sentence: this.textField.getValue()})
21   })
22   .then(response => response.json())
23   .then(data => this.setState(data));
24 }
25 }
26 export default class App extends React.Component {
27   render() {
28     return (
29       <div>
30         <h1>Sentiment Analysis</h1>
31         <div>
32           <input type="text" value={this.state.sentence} />
33           <button onClick={this.analyzeSentence}>Analyze</button>
34         </div>
35         <div>
36           {this.state.sentences.map(sentence => (
37             <div>
38               <div>{sentence}</div>
39               <div>{this.state.polarity}</div>
40             </div>
41           ))}
42         </div>
43       </div>
44     );
45   }
46 }
47 export default App;
48 
```

```

sa-frontend > src > App.js > App > analyzeSentence
10 super(props); this.state = {
11   sentences: [],
12   polarity: undefined;
13 };
14 analyzeSentence() {
15   fetch('http://localhost:8082/sentiment', {
16     method: 'POST',
17     headers: {
18       'Content-Type': 'application/json'
19     },
20     body: JSON.stringify({sentence: this.textField.getValue()})
21   })
22   .then(response => response.json())
23   .then(data => this.setState(data));
24 }
25 }
26 export default class App extends React.Component {
27   render() {
28     return (
29       <div>
30         <h1>Sentiment Analysis</h1>
31         <div>
32           <input type="text" value={this.state.sentence} />
33           <button onClick={this.analyzeSentence}>Analyze</button>
34         </div>
35         <div>
36           {this.state.sentences.map(sentence => (
37             <div>
38               <div>{sentence}</div>
39               <div>{this.state.polarity}</div>
40             </div>
41           ))}
42         </div>
43       </div>
44     );
45   }
46 }
47 export default App;
48 
```

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
==> Writing image sha256:2e4a0d4ddcf183adad42cbb081de5f0e4f492350a47540b263ed511c8513 0.0s
==> Naming to docker.io/library/nsafe 0.0s
Use the 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
harshavardhan@harshavardhan-MacBook-Air sa-frontend % docker run -p 8080:80 nsafe
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
2022/10/20 02:54:36 [notice] 1#1: Using the 'epoll' event method
2022/10/20 02:54:36 [notice] 1#1: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
2022/10/20 02:54:36 [notice] 1#1: Starting the nginx event engine
2022/10/20 02:54:36 [notice] 1#1: /etc/nginx/conf.d/default.conf
2022/10/20 02:54:36 [notice] 1#1: Configuration complete; ready for start up
2022/10/20 02:54:36 [notice] 1#1: using the 'epoll' event method
2022/10/20 02:54:36 [notice] 1#1: nginx/1.23.1
2022/10/20 02:54:36 [notice] 1#1: built by gcc 10.2.1 20211010 (Debian 10.2.1-6)
2022/10/20 02:54:36 [notice] 1#1: OS: Linux 5.10.124-linuxkit
2022/10/20 02:54:36 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2022/10/20 02:54:36 [notice] 1#1: start worker processes
2022/10/20 02:54:36 [notice] 1#1: start worker process 31
2022/10/20 02:54:36 [notice] 1#1: start worker process 32
2022/10/20 02:54:36 [notice] 1#1: start worker process 33
2022/10/20 02:54:36 [notice] 1#1: start worker process 34
2022/10/20 02:54:36 [notice] 1#1: start worker process 35
2022/10/20 02:54:36 [notice] 1#1: start worker process 36
2022/10/20 02:54:36 [notice] 1#1: start worker process 37
2022/10/20 02:54:36 [notice] 1#1: start worker process 38
2022/10/20 02:54:36 [notice] 1#1: start worker process 39
2022/10/20 02:54:36 [notice] 1#1: start worker process 40
2022/10/20 02:54:36 [notice] 1#1: start worker process 41
2022/10/20 02:54:36 [notice] 1#1: start worker process 42
2022/10/20 02:54:36 [notice] 1#1: start worker process 43
2022/10/20 02:54:36 [notice] 1#1: start worker process 44
2022/10/20 02:54:36 [notice] 1#1: start worker process 45
2022/10/20 02:54:36 [notice] 1#1: start worker process 46
2022/10/20 02:54:36 [notice] 1#1: start worker process 47
2022/10/20 02:54:36 [notice] 1#1: start worker process 48
2022/10/20 02:54:36 [notice] 1#1: start worker process 49
2022/10/20 02:54:36 [notice] 1#1: start worker process 50
2022/10/20 02:54:36 [notice] 1#1: start worker process 51
2022/10/20 02:54:36 [notice] 1#1: start worker process 52
2022/10/20 02:54:36 [notice] 1#1: start worker process 53
2022/10/20 02:54:36 [notice] 1#1: start worker process 54
2022/10/20 02:54:36 [notice] 1#1: start worker process 55
2022/10/20 02:54:36 [notice] 1#1: start worker process 56
2022/10/20 02:54:36 [notice] 1#1: start worker process 57
2022/10/20 02:54:36 [notice] 1#1: start worker process 58
2022/10/20 02:54:36 [notice] 1#1: start worker process 59
2022/10/20 02:54:36 [notice] 1#1: start worker process 60
2022/10/20 02:54:36 [notice] 1#1: start worker process 61
2022/10/20 02:54:36 [notice] 1#1: start worker process 62
2022/10/20 02:54:36 [notice] 1#1: start worker process 63
2022/10/20 02:54:36 [notice] 1#1: start worker process 64
2022/10/20 02:54:36 [notice] 1#1: start worker process 65
2022/10/20 02:54:36 [notice] 1#1: start worker process 66
2022/10/20 02:54:36 [notice] 1#1: start worker process 67
2022/10/20 02:54:36 [notice] 1#1: start worker process 68
2022/10/20 02:54:36 [notice] 1#1: start worker process 69
2022/10/20 02:54:36 [notice] 1#1: start worker process 70
2022/10/20 02:54:36 [notice] 1#1: start worker process 71
2022/10/20 02:54:36 [notice] 1#1: start worker process 72
2022/10/20 02:54:36 [notice] 1#1: start worker process 73
2022/10/20 02:54:36 [notice] 1#1: start worker process 74
2022/10/20 02:54:36 [notice] 1#1: start worker process 75
2022/10/20 02:54:36 [notice] 1#1: start worker process 76
2022/10/20 02:54:36 [notice] 1#1: start worker process 77
2022/10/20 02:54:36 [notice] 1#1: start worker process 78
2022/10/20 02:54:36 [notice] 1#1: start worker process 79
2022/10/20 02:54:36 [notice] 1#1: start worker process 80
2022/10/20 02:54:36 [notice] 1#1: start worker process 81
2022/10/20 02:54:36 [notice] 1#1: start worker process 82
2022/10/20 02:54:36 [notice] 1#1: start worker process 83
2022/10/20 02:54:36 [notice] 1#1: start worker process 84
2022/10/20 02:54:36 [notice] 1#1: start worker process 85
2022/10/20 02:54:36 [notice] 1#1: start worker process 86
2022/10/20 02:54:36 [notice] 1#1: start worker process 87
2022/10/20 02:54:36 [notice] 1#1: start worker process 88
2022/10/20 02:54:36 [notice] 1#1: start worker process 89
2022/10/20 02:54:36 [notice] 1#1: start worker process 90
2022/10/20 02:54:36 [notice] 1#1: start worker process 91
2022/10/20 02:54:36 [notice] 1#1: start worker process 92
2022/10/20 02:54:36 [notice] 1#1: start worker process 93
2022/10/20 02:54:36 [notice] 1#1: start worker process 94
2022/10/20 02:54:36 [notice] 1#1: start worker process 95
2022/10/20 02:54:36 [notice] 1#1: start worker process 96
2022/10/20 02:54:36 [notice] 1#1: start worker process 97
2022/10/20 02:54:36 [notice] 1#1: start worker process 98
2022/10/20 02:54:36 [notice] 1#1: start worker process 99
2022/10/20 02:54:36 [notice] 1#1: start worker process 100
2022/10/20 02:54:36 [notice] 1#1: start worker process 101
2022/10/20 02:54:36 [notice] 1#1: start worker process 102
20
```

CSYE 7220 Sentiment Analyser

Type your sentence.

SEND

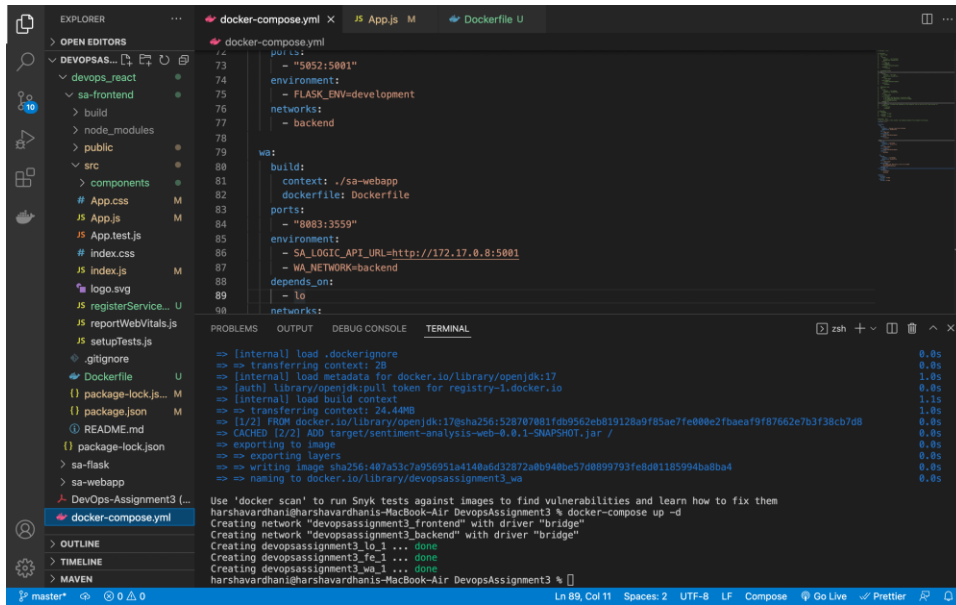
DOCKER-COMPOSE:

docker-compose build

```
docker-compose.yml
version: '3.8'
services:
  sa-webapp:
    build:
      context: ./sa-webapp
      dockerfile: Dockerfile
    ports:
      - "8083:3559"
    environment:
      - SA_LOGIC_API_URL=http://172.17.0.8:5001
      - WA_NETWORK=backend
    depends_on:
      - lo
    networks:
      - backend
```

```
harshavardhani@harshavardhani-MacBook-Air DevOpsAssignment3 % docker-compose build
ERROR! build path /Users/harshavardhani/Downloads/DevOpsAssignment3/sa-frontend either does not exist, is not accessible, or is not a valid URL.
harshavardhani@harshavardhani-MacBook-Air DevOpsAssignment3 % docker-compose build
Building fe
[+] Building 0.6s (7/7) FINISHED
=> [internal] load build definition from Dockerfile
=> transferring dockerfile: 170B
=> [internal] load .dockerignore
=> transferring context: 2B
=> [internal] load metadata for docker.io/library/nginx:latest
=> [internal] load build context
=> transferring context: 1.80MB
=> [1/2] FROM docker.io/library/nginx
=> CACHED [2/2] COPY build /usr/share/nginx/html
=> exporting to image
=> exporting layers
=> writing image sha256:2e4a894dfc81828dad42cbcd861da5f0e4f4923550a47540b263ed511c0513
=> naming to docker.io/library/devopsassignment3_fe
Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
```

docker-compose up -d



docker-compose stop

