

Lab6-Final

ติดตั้ง WSL แล้วเช็ค version

```
PS C:\Users\TUF> wsl --install
Installing: Ubuntu
Ubuntu has been installed.
Launching Ubuntu...
Installing, this may take a few minutes...
Please create a default UNIX user account. The username does not need to match your Windows username.
For more information visit: https://aka.ms/wslusers
Enter new UNIX username: Jirapat
err: Please enter a username matching the regular expression
      configured via the NAME_REGEX configuration variable. Use the
      '--allow-bad-names' option to relax this check or reconfigure
      NAME_REGEX in configuration.
Enter new UNIX username: jirapat
New password:
Retype new password:
passwd: password updated successfully
Installation successful!
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 5.15.153.1-microsoft-standard-WSL2 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

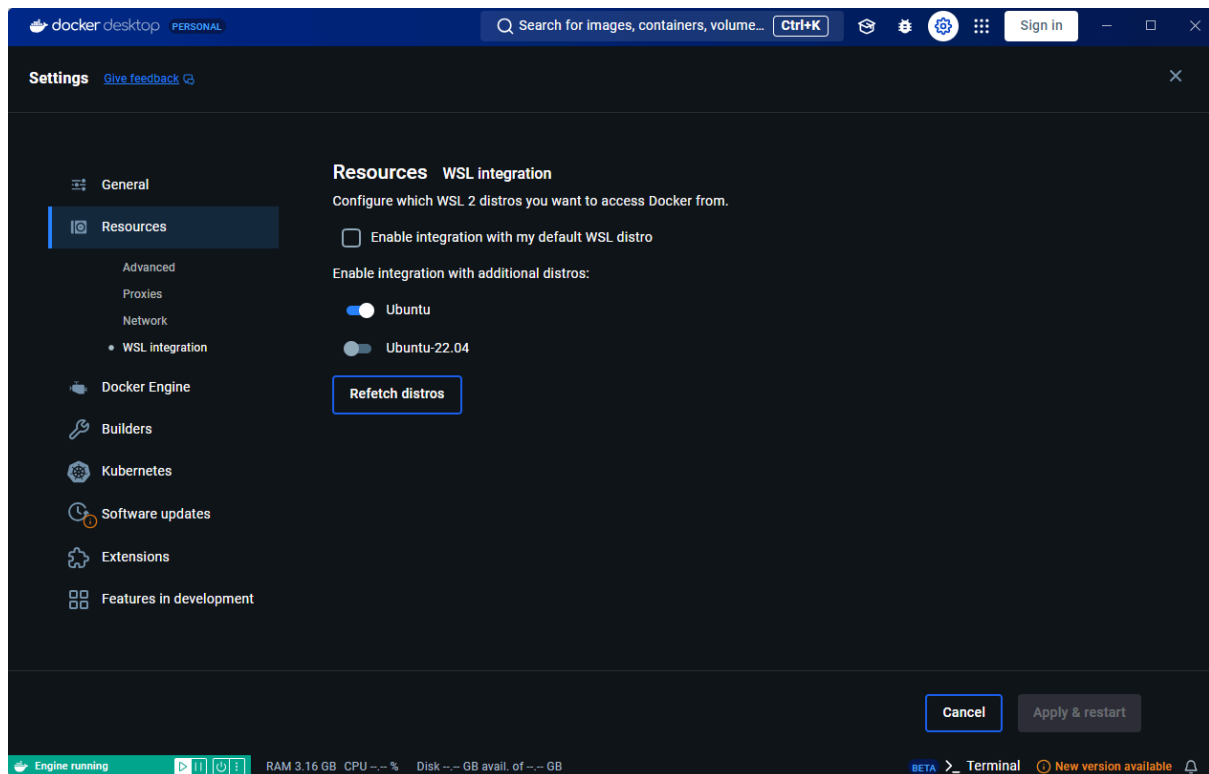
System information as of Sat Oct 19 01:22:51 +07 2024

System load:  0.15           Processes:            51
Usage of /:   0.1% of 1006.85GB Users logged in:       0
Memory usage: 12%           IPv4 address for eth0: 172.23.196.161
Swap usage:   0%

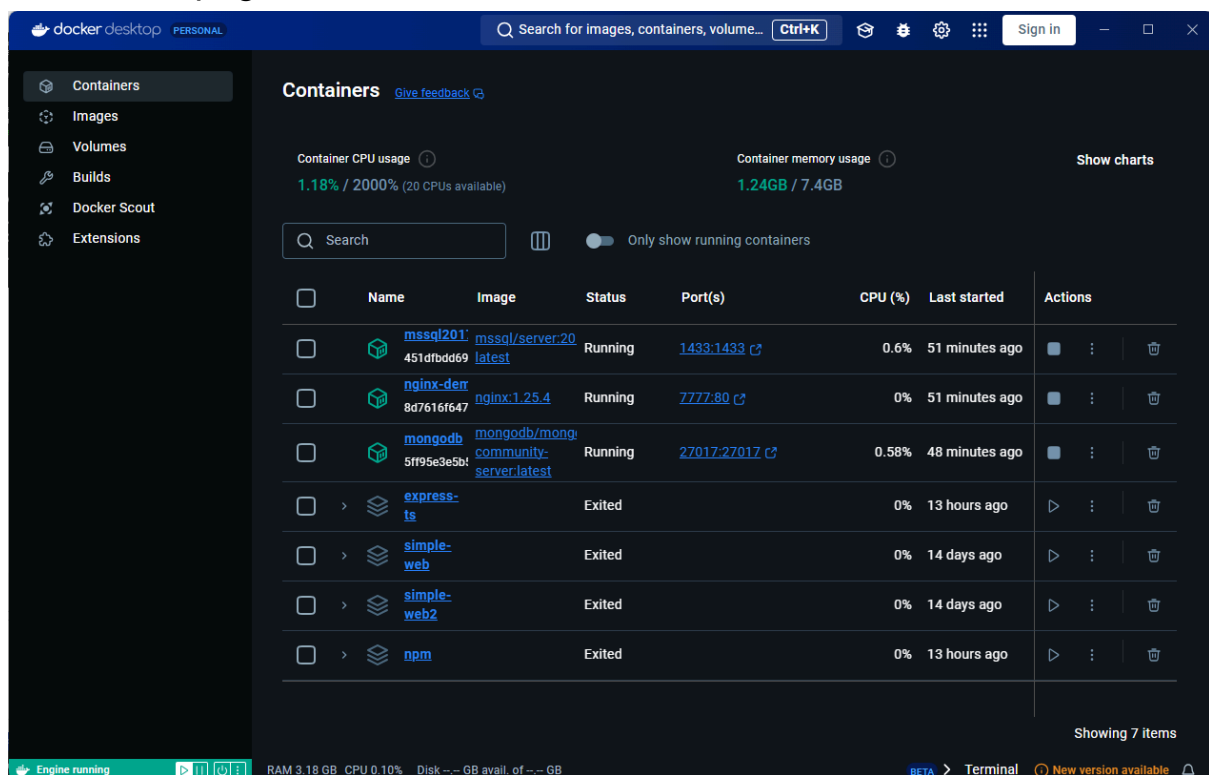
This message is shown once a day. To disable it please create the
/home/jirapat/.hushlogin file.
jirapat@Asus:~$ |
```

```
PS C:\Users\TUF> wsl -l -v
  NAME                STATE      VERSION
* Ubuntu-22.04        Running    2
  docker-desktop      Running    2
  Ubuntu               Running    2
PS C:\Users\TUF> |
```

ตั้งค่าใน Docker Desktop



Containers page



ทำการ pull nginx

```
jirapat@Asus:~/engse207/engse207-docker/docker-image$ docker image pull nginx:1.25.4
1.25.4: Pulling from library/nginx
Digest: sha256:9ff236ed47fe39cf1f0acf349d0e5137f8b8a6fd0b46e5117a401010e56222e1
Status: Image is up to date for nginx:1.25.4
docker.io/library/nginx:1.25.4
jirapat@Asus:~/engse207/engse207-docker/docker-image$
```

run container Nginx

```
jirapat@Asus:~/engse207/engse207-docker/docker-image$ docker container run -d nginx:1.25.4
dd2ceaed1702d6c7177d94a706f32843ff1f0b76e4bd22b68741bdefe5a532d4
```

```
jirapat@Asus:~/engse207-docker/docker-image$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
myapp	latest	3b1d98ee43ba	7 days ago	1.78GB
fifamyapp	latest	25bcd572f1ed	7 days ago	1.78GB
nginx	latest	d2eb56950b84	2 weeks ago	279MB
docker.frappet.com/training/oom_express	0.0.1	1780881eaf81	3 months ago	1.79GB
jc21/nginx-proxy-manager	latest	5bd2aae54ac0	3 months ago	1.58GB
nginx	1.25.4	9ff236ed47fe	8 months ago	272MB

port 7777

```
jirapat@Asus:~/engse207-docker/docker-image$ docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
6a6411d4f2de	nginx:1.25.4	"/docker-entrypoint..."	5 minutes ago	Up 5 minutes	0.0.0.0:7777->80/tcp	inspiring_blackburn

Clone ENGSE207 Docker git repository ลงมา

```
jirapat@Asus:~/engse207$ git clone https://github.com/se-rmutl/engse207-docker.git
Cloning into 'engse207-docker'...
remote: Enumerating objects: 92, done.
remote: Counting objects: 100% (92/92), done.
remote: Compressing objects: 100% (77/77), done.
remote: Total 92 (delta 7), reused 86 (delta 4), pack-reused 0 (from 0)
Receiving objects: 100% (92/92), 5.30 MiB | 6.14 MiB/s, done.
Resolving deltas: 100% (7/7), done.
jirapat@Asus:~/engse207$
```

ติดตั้ง MSSQL Image and Configuration

pull image

```
jirapat@Asus:~/engse207/engse207-docker/docker-image$ docker pull mcr.microsoft.com/mssql/server:2017-latest
2017-latest: Pulling from mssql/server
a6ec4b3b25ec: Download complete
ac40f1754a69: Download complete
699451820735: Download complete
Digest: sha256:896972291ff7bd8b7fc172df994fb8b35135b0a7614e97b9e892c18328bc1208
Status: Downloaded newer image for mcr.microsoft.com/mssql/server:2017-latest
mcr.microsoft.com/mssql/server:2017-latest
```

```
jirapat@Asus:~/engse207/engse207-docker/docker-image$ docker image ls
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
myapp	latest	3b1d98ee43ba	8 days ago	1.78GB
fifamyapp	latest	25bcd572f1ed	8 days ago	1.78GB
nginx	latest	d2eb56950b84	2 weeks ago	279MB
mcr.microsoft.com/mssql/server	2017-latest	896972291ff7	3 weeks ago	1.81GB
docker.frappet.com/training/oom_express	0.0.1	1780881eaf81	3 months ago	1.79GB
jc21/nginx-proxy-manager	latest	5bd2aae54ac0	3 months ago	1.58GB
nginx	1.25.4	9ff236ed47fe	8 months ago	272MB

```
jirapat@Asus:~/engse207/engse207-docker/docker-image$
```

Config SA Password

```
jirapat@Asus:~/engse207/engse207-docker/docker-image$ docker run -e "ACCEPT_EULA=Y" -e "MSSQL_SA_PASSWORD=P@ssw0rd" -p 1433:1433 --name mssql2017 -d mcr.microsoft.com/mssql/server:2017-latest c99484753fa99ef7a14b4bf89463b294cbb2b6793dcaa92af3a24c9d64164a49
```

ใครขึ้นแบบนี้ docker: Error response from daemon: Ports are not available: exposing port TCP 0.0.0.0:1433 -> 0.0.0.0:0: listen tcp 0.0.0.0:1433: bind: An attempt was made to access a socket in a way forbidden by its access permissions.

ให้เข้าไปที่ cmd(ต้องเข้าที่ run as administrator) ของ windows 11 แล้วให้ใช้คำสั่งนี้

net stop winnat

net start winnat

เข้าไปที่ mssql command line

```
jirapat@Asus:~/engse207/engse207-docker/docker-image$ docker exec -it mssql2017 "bash" root@451dfbdd69d8:/#
```

สร้าง Database

```
root@451dfbdd69d8:/# /opt/mssql-tools/bin/sqlcmd -S localhost -U sa -P
Password:
1> SELECT Name from sys.databases;
2> go
Name
-----
master
tempdb
model
msdb

(4 rows affected)

Network packet size (bytes): 4096
1 xact[s]:
Clock Time (ms.): total      1 avg      1.0 (1000.0 xacts per sec.)
1> CREATE DATABASE team7_web_labDB;
2> go

Network packet size (bytes): 4096
1 xact[s]:
Clock Time (ms.): total      206 avg     206.0 (4.9 xacts per sec.)
1> use team7_web_labDB;
2> go
Changed database context to 'team7_web_labDB'.

Network packet size (bytes): 4096
1 xact[s]:
Clock Time (ms.): total      1 avg      1.0 (1000.0 xacts per sec.)
1> CREATE TABLE dbo.Inventory ( id INT, name VARCHAR(50), quantity INT, PRIMARY KEY (id));
2> go

Network packet size (bytes): 4096
1 xact[s]:
Clock Time (ms.): total     11 avg     11.0 (90.9 xacts per sec.)
1> INSERT INTO dbo.Inventory VALUES (1, 'banana', 150);
2> INSERT INTO dbo.Inventory VALUES (2, 'orange', 154);
3> go

(1 rows affected)

(1 rows affected)

Network packet size (bytes): 4096
1 xact[s]:
Clock Time (ms.): total      9 avg      9.0 (111.1 xacts per sec.)
1> SELECT * FROM dbo.Inventory;
2> go
id          name          quantity
-----
1 banana          150
2 orange          154

(2 rows affected)

Network packet size (bytes): 4096
1 xact[s]:
Clock Time (ms.): total      3 avg      3.0 (333.3 xacts per sec.)
```

Import SQL Data

- Import data โดยเรียกโปรแกรม mssql

```
jirapat@Asus: ~/engse207/engse207-docker/docker-image$ docker exec -it mssql2017 "bash"
root@451dfbdd69d8:/# /opt/mssql-tools/bin/sqlcmd -S localhost -U sa -p -d team7_web_labDB -i /home/dbo.OnlineAgents.Table.sql
Password:
```

```
Network packet size (bytes): 4096
1 xact[s]:
Clock Time (ms.): total      1  avg   1.0 (1000.0 xacts per sec.)
```

```
Network packet size (bytes): 4096
1 xact[s]:
Clock Time (ms.): total      1  avg   1.0 (1000.0 xacts per sec.)
```

```
Network packet size (bytes): 4096
1 xact[s]:
Clock Time (ms.): total      6  avg   6.0 (166.7 xacts per sec.)
```

```
Network packet size (bytes): 4096
1 xact[s]:
Clock Time (ms.): total      81  avg   81.0 (12.3 xacts per sec.)
```

(1 rows affected)

(1 rows affected)

(1 rows affected)

(1 rows affected)

```
Network packet size (bytes): 4096
1 xact[s]:
Clock Time (ms.): total      5  avg   5.0 (200.0 xacts per sec.)
```

```
Network packet size (bytes): 4096
1 xact[s]:
Clock Time (ms.): total      2  avg   2.0 (500.0 xacts per sec.)
```

```
Network packet size (bytes): 4096
1 xact[s]:
Clock Time (ms.): total      2  avg   2.0 (500.0 xacts per sec.)
root@451dfbdd69d8:/#
```

```
root@451dfbdd69d8:/# /opt/mssql-tools/bin/sqlcmd -S localhost -U sa -p
Password:
1> use team7_web_labDB;
2> go
Changed database context to 'team7_web_labDB'.

Network packet size (bytes): 4096
1 xact[s]:
Clock Time (ms.): total      1 avg      1.0 (1000.0 xacts per sec.)
1> SELECT * FROM dbo.OnlineAgents;
2> go
OnlineAgent_id agent_id agent_code DIDQueueID StartOnline uuid
AgentStatus IsLogin LastUpdated
-----
1 1 NULL 9999 2023-08-31 13:56:26.033 cb40f193-1317-4b17-a360-506cd89700f7 Dev2
1 2 NULL 3891 2023-09-05 14:27:35.483 0a9c6366-5ec7-4727-8698-045b977afe61 Dev2
1 3 NULL 9998 2023-09-07 14:05:20.360 5ac956de-560b-4888-8b69-d77dfd3191d Dev2
1 4 NULL 9998 2023-09-08 10:08:16.063 b195cf34-b69a-46c8-8ef6-f4d31429bae8 Dev2
1 5 NULL 3723 2023-09-13 13:45:22.167 7cd54ecf-f29f-44cf-8e84-17a86151b3c2 Dev2
1 6 NULL 3748 2023-09-13 17:20:07.890 db3197c3-5624-41bd-8655-916a5a1040f7 Dev2
1 7 NULL 3746 2023-09-13 17:20:13.613 01b735a3-9bfb-4246-ae67-06d2177a76cd Dev2
1 8 NULL 3773 2023-09-13 17:20:55.317 ec1fbb94-fc47-4fe0-84bf-a8cf292e6a04 Dev2
1 9 NULL 3772 2023-09-13 17:21:01.337 2b61affc-0940-46dd-ad36-7122210208de Dev2
1 10 NULL 3822 2023-09-13 17:21:14.643 585bea48-8fc4-4e8f-964a-347f72e0649f Dev2
1 11 NULL 3881 2023-09-13 17:21:20.837 6b420e3c-f1c9-4d45-ace3-2755ae2075d6 Dev2
1 12 NULL 3797 2023-09-13 17:21:36.893 1da06967-168b-4380-b6fa-c8a35fb7e5f5 Dev2
```

ติดตั้ง MongoDB Image and Configuration

```
jirapat@Asus:~/engse207/engse207-docker/docker-image$ docker pull mongodb/mongodb-community-server:latest
latest: Pulling from mongodb/mongodb-community-server
6414378b6477: Download complete
ea9f76730df0: Download complete
4a38c01437f4: Download complete
4f4fb700ef54: Already exists
3561320b1d22: Download complete
6892f94b5cc5: Download complete
2579002e66c4: Download complete
09218eb43544: Download complete
dbf20f635577: Download complete
c43c894ea6e9: Download complete
923f72ee8d96: Download complete
Digest: sha256:9c913685357d8c373743097fdd7892a699cf1e42d4d2562f10a8afc72386b8c5
Status: Downloaded newer image for mongodb/mongodb-community-server:latest
docker.io/mongodb/mongodb-community-server:latest
```

```
jirapat@Asus:~/engse207/engse207-docker/docker-image$ docker exec -it mongodb mongosh
Current Mongosh Log ID: 67133b66d830f347e6fe6910
Connecting to:      mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.3.2
Using MongoDB:      7.0.14
Using Mongosh:      2.3.2

For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/

To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (https://www.mongodb.com/legal/privacy-policy).
You can opt-out by running the disableTelemetry() command.

-----
The server generated these startup warnings when booting
2024-10-19T04:53:47.265+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See http://dochub.mongodb.org/core/prodnotes-filesystem
2024-10-19T04:53:47.803+00:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
2024-10-19T04:53:47.803+00:00: /sys/kernel/mm/transparent_hugepage/enabled is 'always'. We suggest setting it to 'never' in this binary version
2024-10-19T04:53:47.803+00:00: vm.max_map_count is too low
-----
test> █
```

```
test> show dbs
admin    40.00 KiB
config   12.00 KiB
local    40.00 KiB
test> █
```

Create Database MongoDB

```
test> use wallboarddb
switched to db wallboarddb
wallboarddb> db.createUser({user: "wallboarduser",pwd: "WB1qazxsw2",roles: [ { role: "readWrite", db: "wallboarddb" } ]})
{ ok: 1 }
wallboarddb> show users
[
  {
    _id: 'wallboarddb.wallboarduser',
    userId: UUID('3dbda561-8cd6-400e-9b80-26fa9072cf6a'),
    user: 'wallboarduser',
    db: 'wallboarddb',
    roles: [ { role: 'readWrite', db: 'wallboarddb' } ],
    mechanisms: [ 'SCRAM-SHA-1', 'SCRAM-SHA-256' ]
  }
]
wallboarddb> █
```

show database list after created

```
wallboarddb> use admin
switched to db admin
admin> show dbs
admin    132.00 KiB
config   12.00 KiB
local    40.00 KiB
admin> █
```

create example collection

```
admin> use wallboarddb
switched to db wallboarddb
wallboarddb> db.createCollection("log", { capped : true, size : 5242880, max : 5000 } )
{ ok: 1 }
wallboarddb> db.log.insertOne({ capped : true, size : 5242880, max : 5000 })
{
  acknowledged: true,
  insertedId: ObjectId('67133c73d830f347e6fe6911')
}
wallboarddb> █
```

wallboarddb

```
wallboarddb> db.log.find()
[
  {
    _id: ObjectId('67133c73d830f347e6fe6911'),
    capped: true,
    size: 5242880,
    max: 5000
  }
]
wallboarddb> █
```

```
wallboarddb> db.log.find()
[
  {
    _id: ObjectId('67133c73d830f347e6fe6911'),
    capped: true,
    size: 5242880,
    max: 5000
  }
]
wallboarddb> show collections
log
wallboarddb> show dbs
admin          132.00 KiB
config         60.00 KiB
local          40.00 KiB
wallboarddb    40.00 KiB
wallboarddb> █
```

ติดตั้ง Nodejs Image and Configuration

```
jirapat@Asus:~/engse207/engse207-docker/docker-image$ docker pull node:20
20: Pulling from library/node
e07e01c8fd93: Download complete
8ba0b3d08b81: Download complete
898ee3ab8d89: Download complete
2c430d81b800: Download complete
7d98d813d54f: Download complete
5f57b3eca258: Download complete
27bc85537ac7: Download complete
d0c5f3b3f727: Download complete
Digest: sha256:196a5fcd13db4362fb9c0ec5391db36ec954c65d6b0d0e5d37f59c7dc9920690
Status: Downloaded newer image for node:20
docker.io/library/node:20
```



```
jirapat@Asus:~/engse207/engse207-docker/docker-image$ docker image ls
REPOSITORY                                TAG                IMAGE ID           CREATED           SIZE
mongodb/mongodb-community-server          latest             9c913685357d      23 hours ago     1.67GB
myapp                                      latest             3b1d98ee43ba      8 days ago       1.78GB
fifamypapp                                latest             25bcd572f1ed      8 days ago       1.78GB
node                                        20                 196a5fcd13db      2 weeks ago      1.58GB
nginx                                       latest             d2eb56950b84      2 weeks ago      279MB
mcr.microsoft.com/mssql/server            2017-latest       896972291ff7      3 weeks ago      1.81GB
docker.frappet.com/training/oom_express    0.0.1             1780881eaf81      3 months ago     1.79GB
jc21/nginx-proxy-manager                  latest             5bd2aae54ac0      3 months ago     1.58GB
nginx                                       1.25.4            9ff236ed47fe      8 months ago     272MB
jirapat@Asus:~/engse207/engse207-docker/docker-image$
```

git clone <https://github.com/se-rmutl/engse207.git> ลงมา

```
jirapat@Asus:~/engse207$ git clone https://github.com/se-rmutl/engse207.git
Cloning into 'engse207'...
remote: Enumerating objects: 738, done.
remote: Counting objects: 100% (77/77), done.
remote: Compressing objects: 100% (58/58), done.
remote: Total 738 (delta 26), reused 61 (delta 17), pack-reused 661 (from 1)
Receiving objects: 100% (738/738), 7.06 MiB | 7.76 MiB/s, done.
Resolving deltas: 100% (211/211), done.
jirapat@Asus:~/engse207$
```

cd เข้าไปที่

engse207/engse207/n-tier_architecture/backend/endpoint-api/

- npm install

```
jirapat@Asus:~/engse207/engse207/n-tier_architecture/backend/endpoint-api$ npm install
npm warn deprecated har-validator@5.1.5: this library is no longer supported
npm warn deprecated request@2.88.2: request has been deprecated, see https://github.com/request/request/issues/3142
npm warn deprecated uuid@3.4.0: Please upgrade to version 7 or higher. Older versions may use Math.random() in certain circumstances, which is known to be problematic. See https://v8.dev/blog/math-random for details.

added 242 packages, and audited 243 packages in 37s

24 packages are looking for funding
  run 'npm fund' for details

2 moderate severity vulnerabilities

Some issues need review, and may require choosing
a different dependency.

Run 'npm audit' for details.
jirapat@Asus:~/engse207/engse207/n-tier_architecture/backend/endpoint-api$
```

Azure data studio

The screenshot displays the Azure Data Studio interface. On the left, the 'CONNECTIONS' pane shows a tree view of servers and databases. The 'team7_web_labDB' database is selected, and its 'Tables' folder is expanded, showing a list of tables and views. The main pane shows the 'Home' view for the selected database, displaying version information (14.0.3480.1) and computer name (451dfbdd69d8). Below this, the 'Backup Status' section shows the last updated time (12:15:18 PM 10/21/2024) and a summary of backup counts: 0 within 24 hours, 0 older than 24 hours, and 1 no backup found. To the right, the 'Database Size (MB)' section shows a bar chart with four bars, each representing a different database or table, with the total size being 127.0.0.1 MB. The bottom status bar indicates 'Restricted Mode' and 'Choose SQL Language 127.0.0.1 : team7_web_labDB'.

File Edit View Help

DB_Script

CONNECTIONS

SERVERS

- 192.168.56.104, <default> (sa)
- 10.21.43.203,1433, <default> (sa)
- 127.0.0.1, <default> (sa)
 - Databases
 - System Databases
 - team7_web_labDB
 - Tables
 - dbo.Inventory
 - dbo.OnlineAgents
 - Views
 - Synonyms
 - Programmability
 - External Resources
 - Service Broker
 - Storage
 - Security
 - Security
 - Server Objects

Home > 127.0.0.1

New Query New Notebook Refresh Learn More

Home

Databases

Version: 14.0.3480.1

Computer Name: 451dfbdd69d8

Edition: Developer Edition (64-bit)

OS Version: Ubuntu 18.04

Backup Status

Last Updated: 12:15:18 PM 10/21/2024

0 Within 24hrs

0 Older than 24hrs

1 No backup found

Database Size (MB)

team7_web_labDB

127.0.0.1

Restricted Mode 0 0 0 Choose SQL Language 127.0.0.1 : team7_web_labDB

build container endpoint-api

- docker build -t endpoint-api .

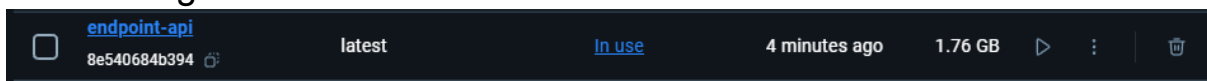
```
jirapat@Asus:~/engse207/Lab6-Final/endpoint-api$ docker build -t endpoint-api .
[+] Building 44.8s (12/12) FINISHED                                docker:default
=> [internal] load build definition from Dockerfile                0.0s
=> => transferring dockerfile: 259B                                0.0s
=> [internal] load metadata for docker.io/library/node:20         3.5s
=> [internal] load .dockerignore                                   0.0s
=> => transferring context: 2B                                       0.0s
=> [1/7] FROM docker.io/library/node:20@sha256:a5e0ed56f2c20b9689e0f7dd498cac7e08d2a3a283e92d9304e7 11.4s
=> => resolve docker.io/library/node:20@sha256:a5e0ed56f2c20b9689e0f7dd498cac7e08d2a3a283e92d9304e7b 0.0s
=> => sha256:8bd6b85cfa4a45f95b351eb9675dcca707a9825e06852293d11f45cc01673bf2 48.22MB / 48.22MB 2.1s
=> => sha256:ee2a0351a4986d763ba45f06f7490ae672b0979fa664761cd82021b18f943a8c 3.32kB / 3.32kB 0.7s
=> => sha256:8f86048bbb7710b3d45488537965686bf525dc92ac033e1baff319769647b697 1.25MB / 1.25MB 1.1s
=> => sha256:57cfff12bcf233d48aed6c2b24f032a02efdae43afb12d7c2b86657ae1e6361da 446B / 446B 1.1s
=> => sha256:ad1c7cfc347f5c86fc2678b58f6a8fb6c6003471405760532fc3240b9eb1b343 211.27MB / 211.27MB 6.2s
=> => sha256:7aad5092c3b7a865666b14bef3d4d038282b19b124542f1a158c98ea8c1ed1b 64.39MB / 64.39MB 3.1s
=> => sha256:da802df85c965baeca9d39869f9e2cbb3dc844d4627f413bfb2f2c3d6055988 24.05MB / 24.05MB 1.6s
=> => extracting sha256:da802df85c965baeca9d39869f9e2cbb3dc844d4627f413bfb2f2c3d6055988 0.5s
=> => extracting sha256:7aad5092c3b7a865666b14bef3d4d038282b19b124542f1a158c98ea8c1ed1b 1.6s
=> => extracting sha256:ad1c7cfc347f5c86fc2678b58f6a8fb6c6003471405760532fc3240b9eb1b343 3.2s
=> => extracting sha256:ee2a0351a4986d763ba45f06f7490ae672b0979fa664761cd82021b18f943a8c 0.0s
=> => extracting sha256:8bd6b85cfa4a45f95b351eb9675dcca707a9825e06852293d11f45cc01673bf2 1.2s
=> => extracting sha256:8f86048bbb7710b3d45488537965686bf525dc92ac033e1baff319769647b697 0.0s
=> => extracting sha256:57cfff12bcf233d48aed6c2b24f032a02efdae43afb12d7c2b86657ae1e6361da 0.0s
=> [internal] load build context                                   0.0s
=> => transferring context: 3.09kB                                   0.0s
=> [2/7] WORKDIR /app                                             5.2s
=> [3/7] RUN git clone https://github.com/jirapatfifa/endpoint-api.git /app 1.4s
=> [4/7] RUN git pull                                             1.0s
=> [5/7] RUN npm install                                          18.2s
=> [6/7] COPY ./server.key /app                                   0.1s
=> [7/7] COPY ./server.crt /app                                   0.0s
=> exporting to image                                             3.8s
=> => exporting layers                                             2.7s
=> => exporting manifest sha256:2693414cc8350c1c64b7d69fd6a0c4ae66760dabd16657313de456c0cf4e5d5 0.0s
=> => exporting config sha256:050fedeb078528fde335b9975303d786a1edc35433c1b4d2b3264339256d7ea2 0.0s
=> => exporting attestation manifest sha256:ecb971d0b252ac6137c9f954e2107a3736711f8646fcd03b91b83f6e 0.0s
=> => exporting manifest list sha256:8e540684b39401b4b37dd6d45f44597725a92c59111938b1f475d40f744424b 0.0s
=> => naming to docker.io/library/endpoint-api:latest             0.0s
=> => unpacking to docker.io/library/endpoint-api:latest          1.0s
```

docker run container and port

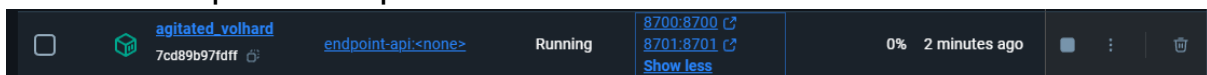
- docker run -p 8700:8700 -p 8701:8701 endpoint-api

```
jirapat@Asus:~/engse207/Lab6-Final/endpoint-api$ docker run -p 8700:8700 -p 8701:8701 endpoint-api
```

create image dockerfile



Docker compose with port



Dockerfile

```
Dockerfile
1 FROM node:20
2 WORKDIR /app
3 RUN git clone https://github.com/jirapatfifa/endpoint-api.git /app
4 RUN git pull
5 RUN npm install
6 COPY ./server.key /app
7 COPY ./server.crt /app
8 EXPOSE 8700 8701
9 CMD ["node", "webreport-api-https"]
```

Test api

GET https://localhost:8700/

The screenshot shows a REST client interface with the following details:

- Method:** GET
- URL:** https://localhost:8700
- Response Status:** 200 OK
- Response Time:** 21 ms
- Response Size:** 375 B
- Response Content:** Test Hello, from Endpoint Web Report API.

The interface includes tabs for Params, Authorization, Headers (7), Body, Scripts, Tests, Settings, and Cookies. The Body tab is currently selected, showing the response content in HTML format.

GET getOnlineAgentByAgentCode

ENGSE207 / Final / getOnlineAgentByAgentCode

Save Share

GET

https://localhost:8700/api/v1/getOnlineAgentByAgentCode?agentcode=9998

Send

Params

Authorization

Headers (7)

Body

Scripts

Tests

Settings

Cookies

Query Params

<input checked="" type="checkbox"/>	Key	Value	Description	Bulk Edit
<input checked="" type="checkbox"/>	agentcode	9998		
	Key	Value	Description	

Body

Cookies

Headers (10)

Test Results

200 OK · 97 ms · 635 B · Save Response

Pretty

Raw

Preview

Visualize

JSON

```
1 {
2   "error": false,
3   "statusCode": 200,
4   "data": {
5     "OnlineAgent_id": 3,
6     "agent_id": null,
7     "agent_code": "9998",
8     "DIDQueueID": null,
9     "StartOnline": "2023-09-07T14:05:20.360Z",
10    "uuid": "5ac956de-560b-4888-8b69-d77dfdf3191d",
11    "AgentName": "Dev2",
12    "AgentStatus": "1",
13    "IsLogin": "1",
14    "LastUpdated": "2023-09-07T14:05:20.360Z"
15  }
16 }
```

Postbot
Ctrl Alt P

Postbot

Runner

Start Proxy

Cookies

Vault

Trash

POSE <https://localhost:8700/api/v1/postOnlineAgentStatus>

ENGSE207 / Final / postOnlineAgentStatus

POST

https://localhost:8700/api/v1/postOnlineAgentStatus

Send

Params

Authorization

Headers (9)

Body

Scripts

Tests

Settings

Cookies

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

	Key		Value	Description		Bulk Edit
<input checked="" type="checkbox"/>	AgentCode	Text	1234			
<input checked="" type="checkbox"/>	AgentName	Text	Test Agent			
<input checked="" type="checkbox"/>	IsLogin	Text	0	1=Login, 0=Logout		
<input checked="" type="checkbox"/>	AgentStatus	Text	1	1=Available, 2=Active, 3=Wrap, 4=Not Ready		
	Key	Text	Value	Description		

Body

Cookies

Headers (9)

Test Results

200 OK

91 ms

405 B

Save Response

Pretty

Raw

Preview

Visualize

JSON

```
1 {
2   "error": false,
3   "statusCode": 200,
4   "data": "Agent was inserted, status has been set also"
5 }
```

Postbot

Ctrl Alt P

Postbot

Runner

Start Proxy

Cookies

Vault

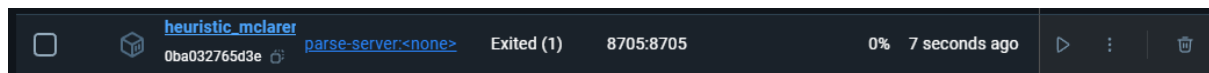
Trash

Parse-server

```
jirapat@Asus:~/engse207/Lab6-Final/Parse-server-api$ docker build -t parse-server .
[+] Building 175.9s (14/14) FINISHED                                docker:default
=> [internal] load build definition from Dockerfile                 0.0s
=> => transferring dockerfile: 262B                                0.0s
=> [internal] load metadata for docker.io/library/node:20          0.0s
=> [internal] load .dockerignore                                    0.0s
=> => transferring context: 2B                                       0.0s
=> [1/7] FROM docker.io/library/node:20@sha256:a5e0ed56f2c20b9689e0f7dd498cac7e08d2a3a283e92d9304e7 61.0s
=> => resolve docker.io/library/node:20@sha256:a5e0ed56f2c20b9689e0f7dd498cac7e08d2a3a283e92d9304e7 61.0s
=> [internal] load build context                                    0.0s
=> => transferring context: 3.09kB                                    0.0s
=> [auth] library/node:pull token for registry-1.docker.io        0.0s
=> [auth] library/node:pull token for registry-1.docker.io        0.0s
=> CACHED [2/7] WORKDIR /app                                       0.0s
=> [3/7] RUN git clone https://github.com/jirapatfifa/Parse-server-Api.git /app 1.4s
=> [4/7] RUN git pull                                              1.1s
=> [5/7] RUN npm install                                           89.2s
=> [6/7] COPY ./server.key /app                                    0.3s
=> [7/7] COPY ./server.crt /app                                    0.0s
=> exporting to image                                              22.6s
=> => exporting layers                                             16.0s
=> => exporting manifest sha256:296ca2e64276ce333f39a4a8f46107f40c0aa8170c72be3b81001a1b6cc56b4e 0.0s
=> => exporting config sha256:b6f16a0445b29a199c506af2eac52d98284cb754e9a1dfe50c5d72c87ae696667 0.0s
=> => exporting attestation manifest sha256:368cd978253618b30c64569b5902e2d74dfba493ac67441bb9ff8549 0.0s
=> => exporting manifest list sha256:9b8a661f9fbd2f77ac0956b52048aaae29a5a29d08d0ca70801fa5481108b84 0.0s
=> => naming to docker.io/library/parse-server:latest             0.0s
=> => unpacking to docker.io/library/parse-server:latest          6.6s
```

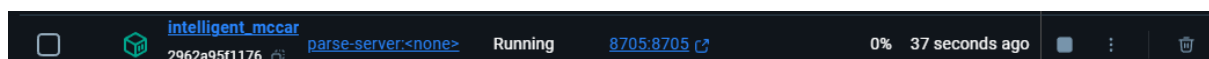
docker run -d -p 8705:8705 parse-server

```
jirapat@Asus:~/engse207/Lab6-Final/Parse-server-api$ docker run -d -p 8705:8705 parse-server
2962a95f1176b0ddb63db8c8b9b20b59d701830915962d63c21435048901597
```

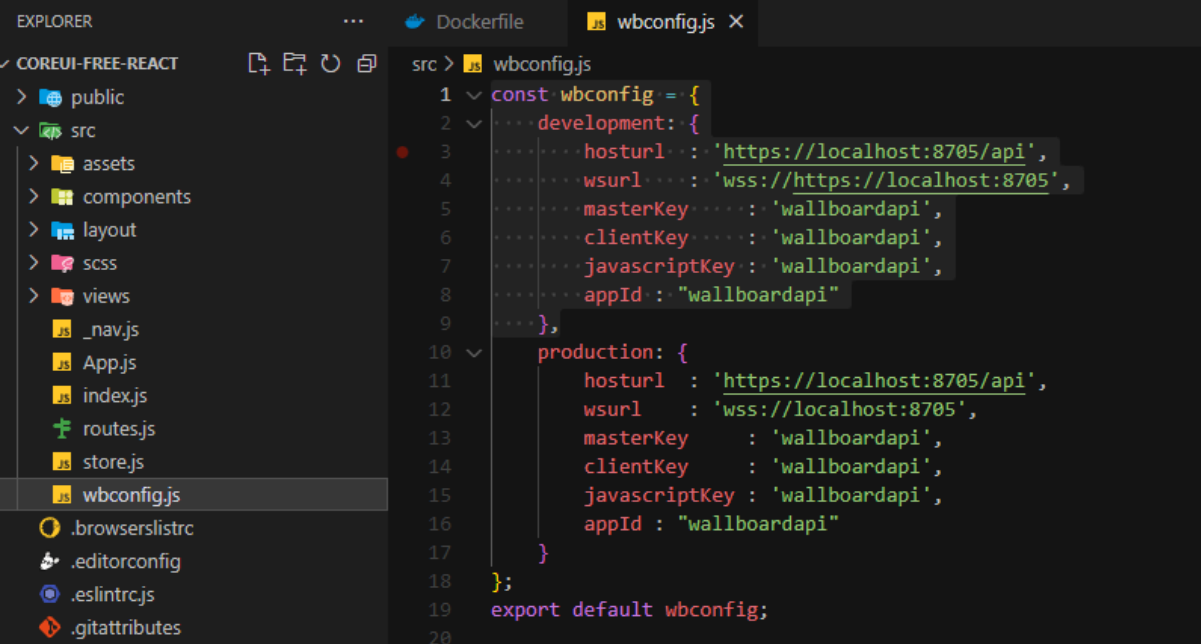


Dockerfile

```
Dockerfile > ...
1 FROM node:20
2 WORKDIR /app
3 RUN git clone https://github.com/jirapatfifa/Parse-server-Api.git /app
4 RUN git pull
5 RUN npm install
6 COPY ./server.key /app
7 COPY ./server.crt /app
8 EXPOSE 8705
9 CMD ["node", "api-wallboard-https.js"]
10
11
```



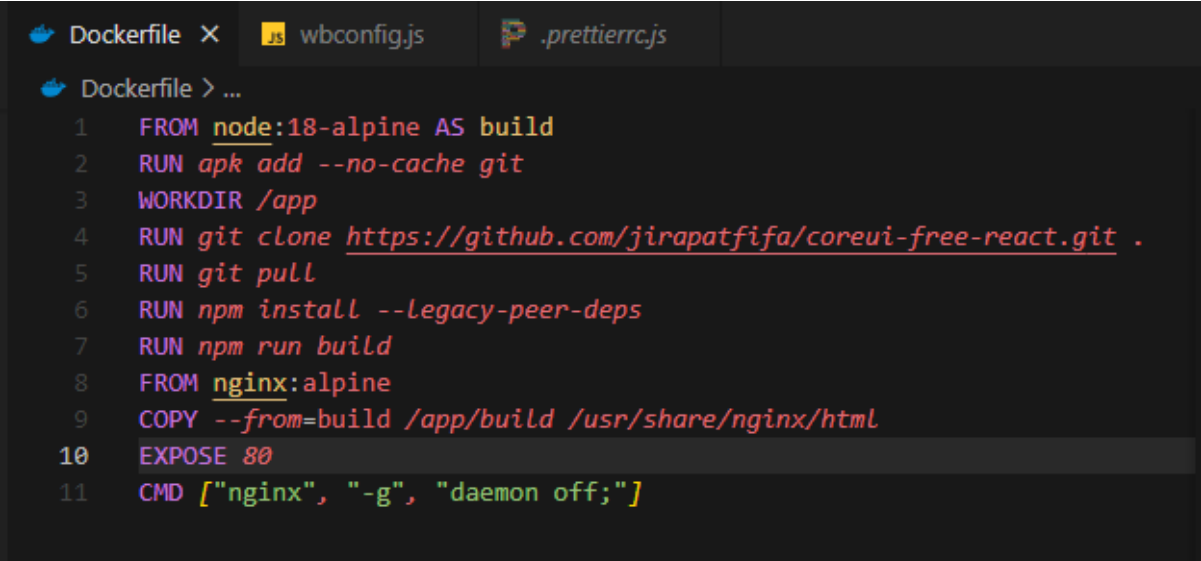
React



The screenshot shows the VS Code interface. On the left, the 'EXPLORER' sidebar displays the project structure for 'COREUI-FREE-REACT'. The 'src' directory is expanded, showing files like _nav.js, App.js, index.js, routes.js, store.js, and wbconfig.js. The 'wbconfig.js' file is selected. The main editor area shows the code for 'wbconfig.js' with line numbers 1 through 20. The code defines a 'wbconfig' object with 'development' and 'production' environments, both pointing to 'https://localhost:8705/api'.

```
1  const wbconfig = {
2    development: {
3      hosturl: 'https://localhost:8705/api',
4      wsurl: 'wss://https://localhost:8705',
5      masterKey: 'wallboardapi',
6      clientKey: 'wallboardapi',
7      javascriptKey: 'wallboardapi',
8      appId: "wallboardapi"
9    },
10   production: {
11     hosturl: 'https://localhost:8705/api',
12     wsurl: 'wss://localhost:8705',
13     masterKey: 'wallboardapi',
14     clientKey: 'wallboardapi',
15     javascriptKey: 'wallboardapi',
16     appId: "wallboardapi"
17   }
18 };
19 export default wbconfig;
20
```

Dockerfile



The screenshot shows the VS Code interface with the 'Dockerfile' tab active. The editor displays the Dockerfile content with line numbers 1 through 11. The Dockerfile sets up a container based on 'node:18-alpine', clones the 'coreui-free-react' repository, installs dependencies, builds the application, and serves it using 'nginx' on port 80.

```
1  FROM node:18-alpine AS build
2  RUN apk add --no-cache git
3  WORKDIR /app
4  RUN git clone https://github.com/jirapatfifa/coreui-free-react.git .
5  RUN git pull
6  RUN npm install --legacy-peer-deps
7  RUN npm run build
8  FROM nginx:alpine
9  COPY --from=build /app/build /usr/share/nginx/html
10 EXPOSE 80
11 CMD ["nginx", "-g", "daemon off;"]
```

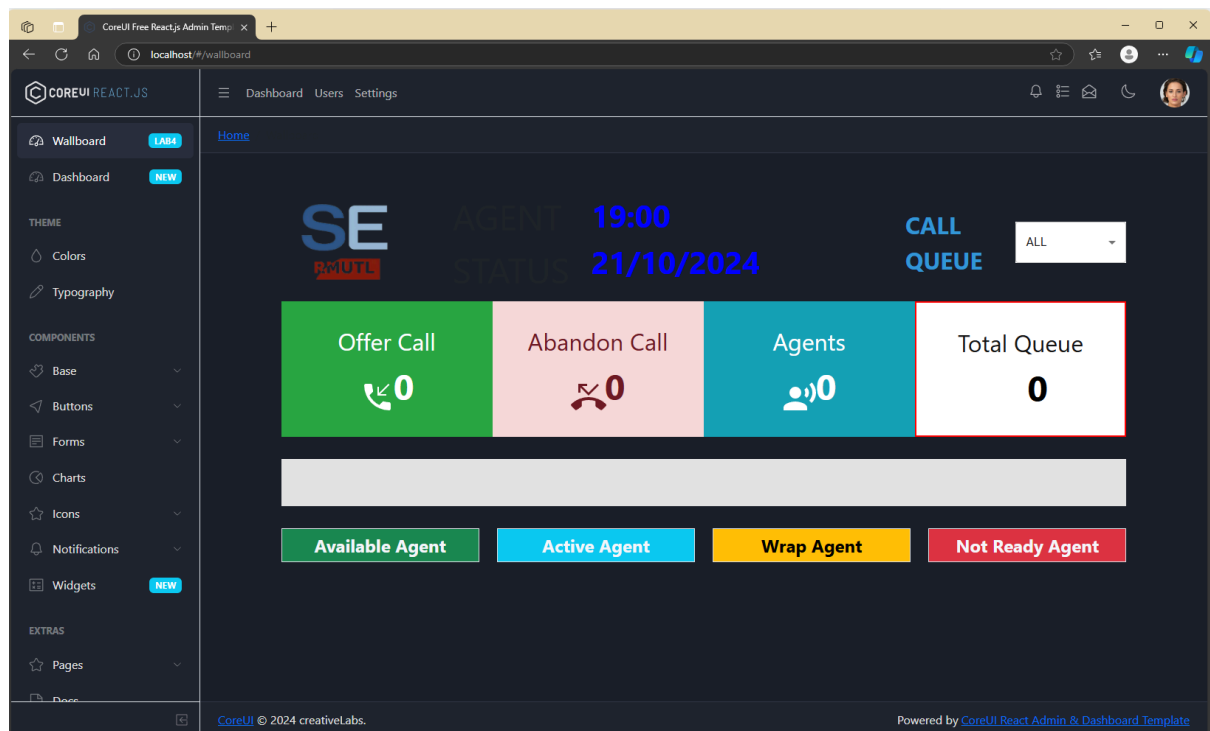

Docker build -t react

```
jirapat@Asus: ~/engse207/Lab6-Final/react$ docker build -t react .
[+] Building 4.6s (16/16) FINISHED                                docker:default
=> [internal] load build definition from Dockerfile              0.0s
=> => transferring dockerfile: 356B                             0.0s
=> [internal] load metadata for docker.io/library/nginx:alpine  4.4s
=> [internal] load metadata for docker.io/library/node:18-alpine 4.4s
=> [auth] library/nginx:pull token for registry-1.docker.io     0.0s
=> [auth] library/node:pull token for registry-1.docker.io      0.0s
=> [internal] load .dockerignore                                 0.0s
=> => transferring context: 2B                                    0.0s
=> [stage-1 1/2] FROM docker.io/library/nginx:alpine@sha256:2140dad235c130ac861018a4e13a6bc8aea3a35f 0.0s
=> => resolve docker.io/library/nginx:alpine@sha256:2140dad235c130ac861018a4e13a6bc8aea3a35f3a40e20c 0.0s
=> [build 1/7] FROM docker.io/library/node:18-alpine@sha256:02376a266c84acbf45bd19440e08e48b1c8b9803 0.0s
=> => resolve docker.io/library/node:18-alpine@sha256:02376a266c84acbf45bd19440e08e48b1c8b9803741733 0.0s
=> CACHED [build 2/7] RUN apk add --no-cache git                 0.0s
=> CACHED [build 3/7] WORKDIR /app                               0.0s
=> CACHED [build 4/7] RUN git clone https://github.com/jirapatfifa/coreui-free-react.git . 0.0s
=> CACHED [build 5/7] RUN git pull                              0.0s
=> CACHED [build 6/7] RUN npm install --legacy-peer-deps        0.0s
=> CACHED [build 7/7] RUN npm run build                          0.0s
=> CACHED [stage-1 2/2] COPY --from=build /app/build /usr/share/nginx/html 0.0s
=> exporting to image                                           0.1s
=> => exporting layers                                           0.0s
=> => exporting manifest sha256:4cb324b4fbf7eb7668ef9f25be62bc722b1301b2aeecf3d299964731afb11f8a 0.0s
=> => exporting config sha256:5670abfb9e8b50d0a31b669183d2d35125a54e546debc07da9f25a4c609afa19 0.0s
=> => exporting attestation manifest sha256:b61478f2b909afdc679eb5c344fe77de3a2145adaa14500cfed89291 0.0s
=> => exporting manifest list sha256:3aa9802990a0be1af4d74af1dabf4d970abb0228d8a9108367b5a06c3f8f636 0.0s
=> => exporting to docker.io/library/react:latest               0.0s
=> => unpacking to docker.io/library/react:latest               0.1s
```




































run container port 80:80

```
jirapat@Asus: ~/engse207/Lab6-Final/react$ docker run -d -p 80:80 react
196ed5dbad15dc5b0fca933b366c2f01a9110ab8123b8ef875b01e24e58b6130
```

Web Page



Docker container

<input type="checkbox"/>	 jolly_swanson 196ed5dbad15 	react<none>	Running	80:80 	0%	1 hour ago	  
<input type="checkbox"/>	 upbeat_elion 409d5f689612 	endpoint-api<none>	Running	8700:8700  Show all ports (2)	0%	1 hour ago	  
<input type="checkbox"/>	 mssql2017 451dfbdd69d8 	mssql/server:2017-latest	Running	1433:1433 	1.36%	1 hour ago	  
<input type="checkbox"/>	 nginx-demo 8d7616f647df 	nginx:1.25.4	Exited (255)	7777:80	0%	2 hours ago	  
<input type="checkbox"/>	 mongodb c8e234c975b9 	mongodb/mongodb-community-server:latest	Running	27017:27017 	0.38%	1 hour ago	  
<input type="checkbox"/>	 intelligent_mccar 2962a95f1176 	parse-server<none>	Running	8705:8705 	0.05%	31 minutes ago	  

Dockerfile

```
Dockerfile X
Dockerfile
1 FROM node:18-alpine AS build
2 RUN apk add --no-cache git
3 WORKDIR /app
4 RUN git clone https://github.com/jirapatfifa/coreui-free-react.git .
5 RUN git pull
6 RUN npm install --legacy-peer-deps
7 RUN npm run build
8 FROM nginx:alpine
9 COPY --from=build /app/build /usr/share/nginx/html
10 EXPOSE 80
11 CMD ["nginx", "-g", "daemon off;"]
```

config Agent Notification

```
//--Staging Server-----  
var APIServer = "https://localhost:8700/api/v1";  
var WebSecketServer = "ws://localhost:8701";  
  
var ConcurrentCall = [];  
var connectSocket = false;  
var IsLogin = 0;  
var wsChecks = "false";  
  
var c;  
let ws;  
  
let checked;  
  
//-----
```

```
success: function (response) {  
    console.log("Status updated successfully:", response);  
    $.ajax({  
        url: "https://localhost:8705/api/functions/postOnlineAgentListByTeam",  
        type: "post",  
  
        contentType: "application/json",  
        headers: {  
            "X-Parse-Application-Id": "wallboardapi",  
            "X-Parse-Master-Key": "wallboardapi",  
            "Content-Type": "application/json",  
        },  
        data: JSON.stringify({  
            AgentCode: $("#agent").val(),  
            AgentName: "test agent",  
            Queue: 5, // ใส่ TeamCode ที่ต้องการ  
            AgentStatus: agentStatusValue, // ใช้ค่า agentStatusValue จากการเลือกสถานะ  
            AgentStatusCode: "5",  
            IsLogin :1,  
        })),  
    });  
}
```

Codeium: Refactor | Explain | Generate Function Comment | X

Agent Notification

