Ad-Campaign Recommenders

Task2 - Model Deployment

CAPSTONE1 – **SURBHI SINHA**

MS IN DATA SCIENCE

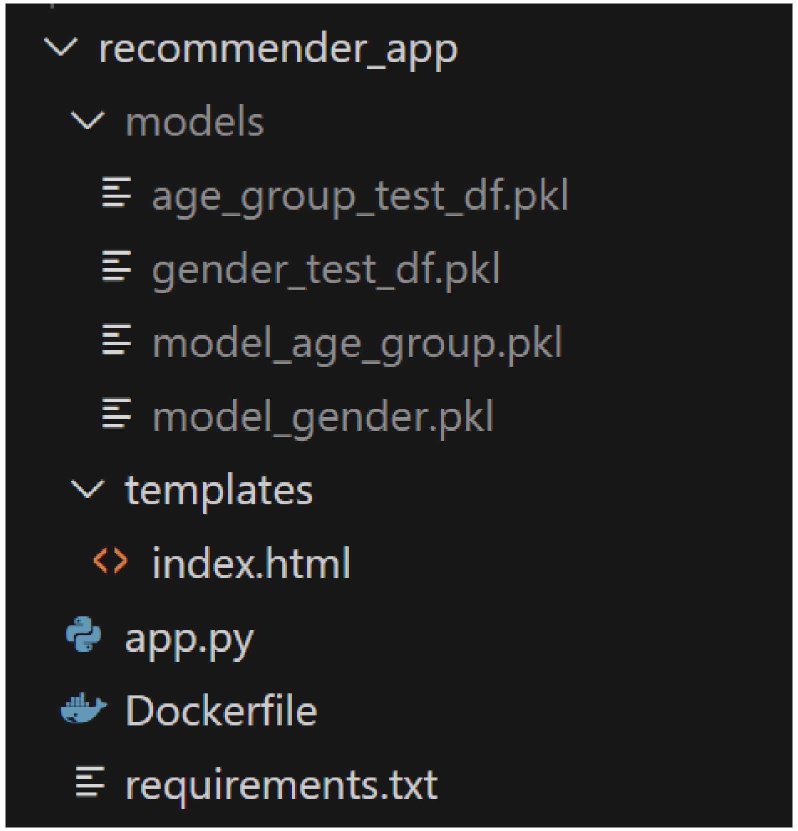
UNIVERSITY OF ARIZONA

Index

Source Code – Flask App

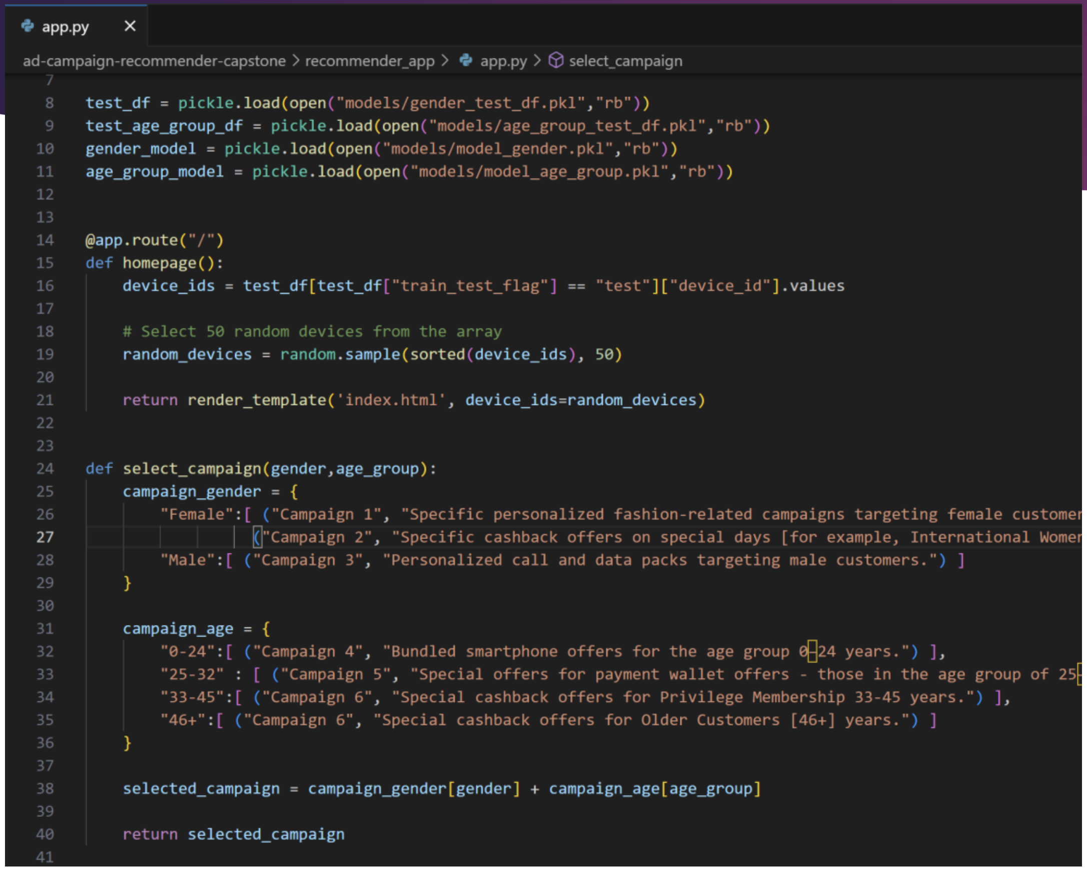
Folder structure of the source code for the ad-campaign recommender Flask app

- models/ - this directory contains 4 pickle files:
  - model_age_group and model_gender are the model pickle files for age_group and gender prediction respectively.
  - age_group_test_df and gender_test_df are the input dataframes for age and gender inferencing.
- templates/index.html – contains script which is used for rendering the UI of the application.
- app.py is the main file which hosts the flask application and has the business logic for predicting ad-campaign based on gender and age prediction.
- Dockerfile – This is the dockerfile for our application.
- requirements.txt – contains all the package dependencies for deploying this application.



```
recommender_app
  models
    age_group_test_df.pkl
    gender_test_df.pkl
    model_age_group.pkl
    model_gender.pkl
  templates
    index.html
  app.py
  Dockerfile
  requirements.txt
```

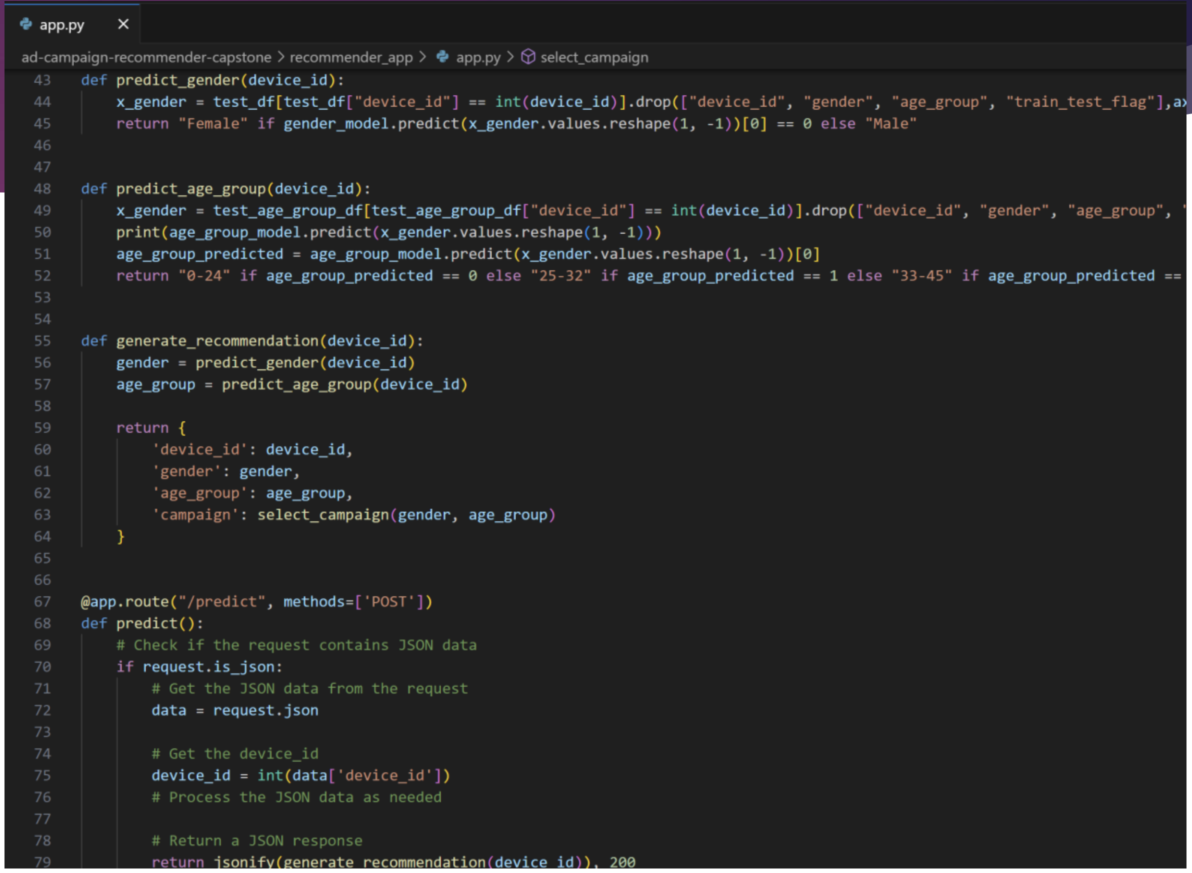
Source Code – Flask App (app.py)



app.py

ad-campaign-recommender-capstone > recommender_app > app.py > select_campaign

```python

test_df = pickle.load(open("models/gender_test_df.pkl","rb"))
test_age_group_df = pickle.load(open("models/age_group_test_df.pkl","rb"))
gender_model = pickle.load(open("models/model_gender.pkl","rb"))
age_group_model = pickle.load(open("models/model_age_group.pkl","rb"))


@app.route("/")
def homepage():
    device_ids = test_df[test_df["train_test_flag"] == "test"]["device_id"].values

    # Select 50 random devices from the array
    random_devices = random.sample(sorted(device_ids), 50)

    return render_template('index.html', device_ids=random_devices)


def select_campaign(gender,age_group):
    campaign_gender = {
        "Female":[ ("Campaign 1", "Specific personalized fashion-related campaigns targeting female customer
                    ("Campaign 2", "Specific cashback offers on special days [for example, International Women
        "Male":[ ("Campaign 3", "Personalized call and data packs targeting male customers.") ]
    }

    campaign_age = {
        "0-24":[ ("Campaign 4", "Bundled smartphone offers for the age group 0-24 years.") ],
        "25-32" : [ ("Campaign 5", "Special offers for payment wallet offers - those in the age group of 25
        "33-45":[ ("Campaign 6", "Special cashback offers for Privilege Membership 33-45 years.") ],
        "46+":[ ("Campaign 6", "Special cashback offers for Older Customers [46+] years.") ]
    }

    selected_campaign = campaign_gender[gender] + campaign_age[age_group]

    return selected_campaign

```



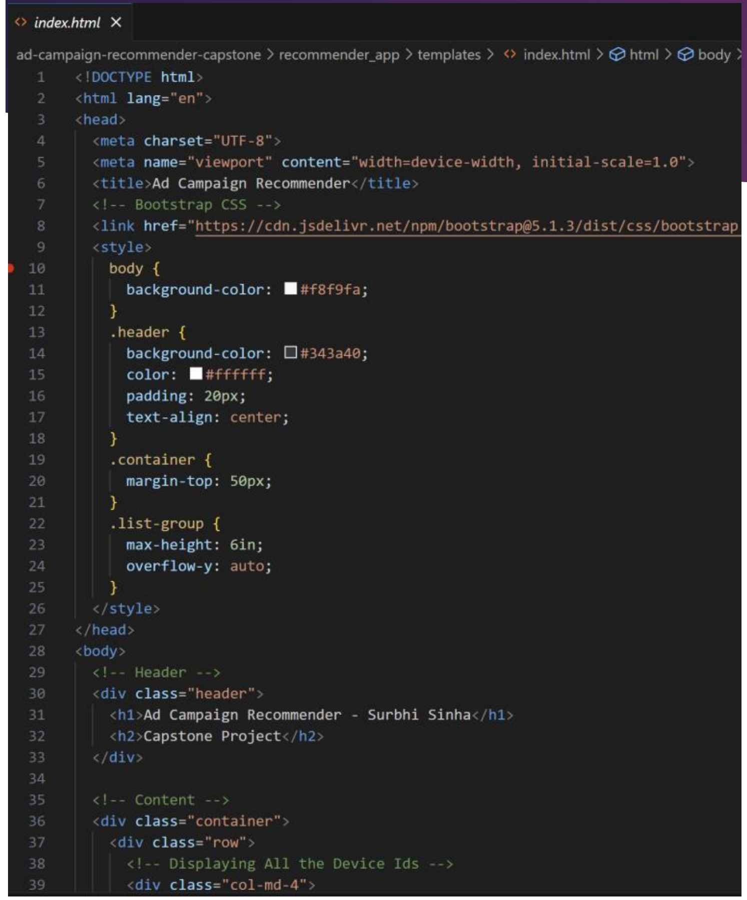
app.py

ad-campaign-recommender-capstone > recommender_app > app.py > select_campaign

```python
def predict_gender(device_id):
    x_gender = test_df[test_df["device_id"] == int(device_id)].drop(["device_id", "gender", "age_group", "train_test_flag"],ax
    return "Female" if gender_model.predict(x_gender.values.reshape(1, -1))[0] == 0 else "Male"


def predict_age_group(device_id):
    x_gender = test_age_group_df[test_age_group_df["device_id"] == int(device_id)].drop(["device_id", "gender", "age_group", '
    print(age_group_model.predict(x_gender.values.reshape(1, -1)))
    age_group_predicted = age_group_model.predict(x_gender.values.reshape(1, -1))[0]
    return "0-24" if age_group_predicted == 0 else "25-32" if age_group_predicted == 1 else "33-45" if age_group_predicted ==


def generate_recommendation(device_id):
    gender = predict_gender(device_id)
    age_group = predict_age_group(device_id)

    return {
        'device_id': device_id,
        'gender': gender,
        'age_group': age_group,
        'campaign': select_campaign(gender, age_group)
    }


@app.route("/predict", methods=['POST'])
def predict():
    # Check if the request contains JSON data
    if request.is_json:
        # Get the JSON data from the request
        data = request.json

        # Get the device_id
        device_id = int(data['device_id'])
        # Process the JSON data as needed

        # Return a JSON response
        return jsonify(generate_recommendation(device_id)), 200
```

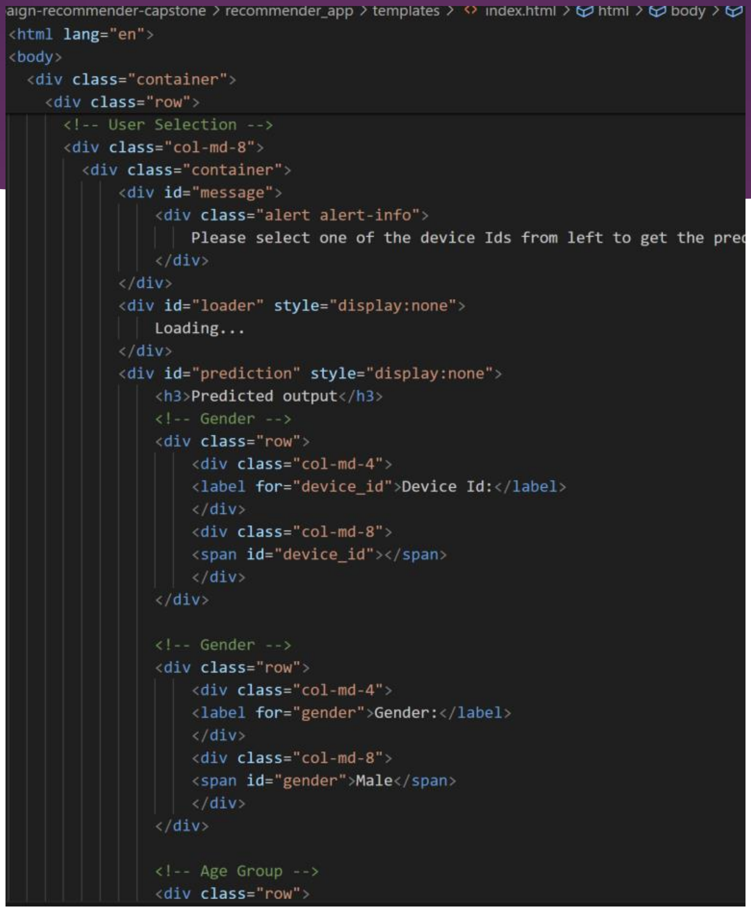
The app.py file is the main application script for the Ad Campaign Recommender system. Its primary purpose is to serve as the entry point for running the application, which includes loading necessary models, processing input data, making predictions, and serving the results through a web interface.

Source Code – Flask App (index.html)

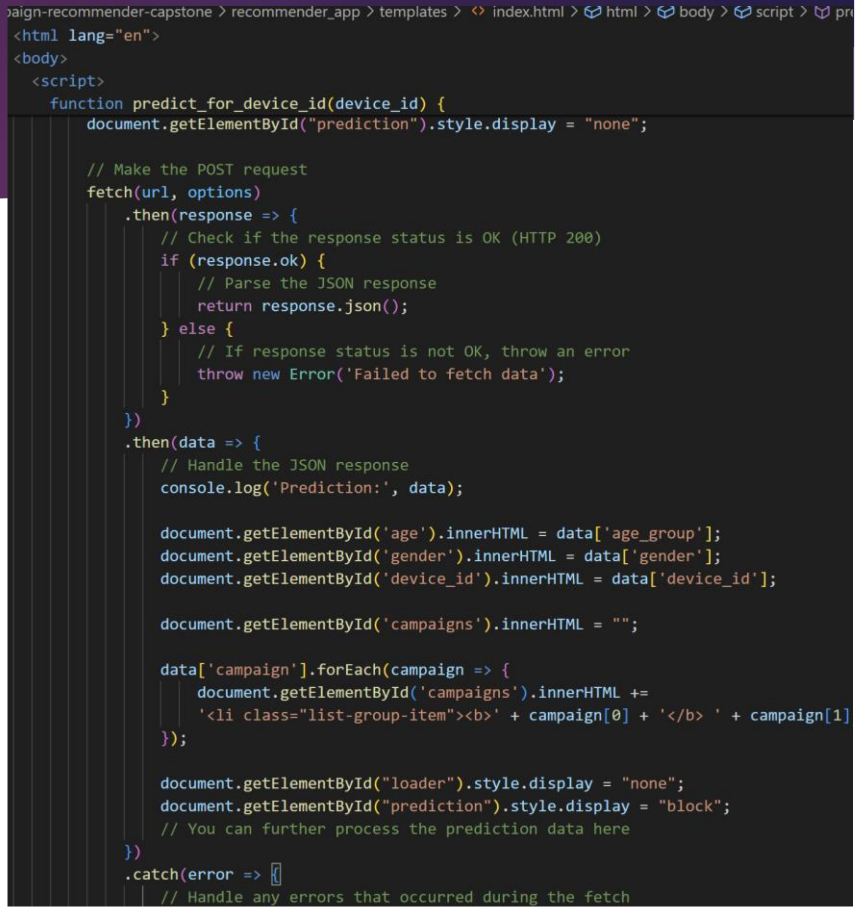

index.html ×

ad-campaign-recommender-capstone > recommender_app > templates > index.html > html > body >

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Ad Campaign Recommender</title>
  <!-- Bootstrap CSS -->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap
  <style>
    body {
      background-color: #f8f9fa;
    }
    .header {
      background-color: #343a40;
      color: #ffffff;
      padding: 20px;
      text-align: center;
    }
    .container {
      margin-top: 50px;
    }
    .list-group {
      max-height: 6in;
      overflow-y: auto;
    }
  </style>
</head>
<body>
  <!-- Header -->
  <div class="header">
    <h1>Ad Campaign Recommender - Surbhi Sinha</h1>
    <h2>Capstone Project</h2>
  </div>

  <!-- Content -->
  <div class="container">
    <div class="row">
      <!-- Displaying All the Device Ids -->
      <div class="col-md-4">
```


aign-recommender-capstone > recommender_app > templates > index.html > html > body >

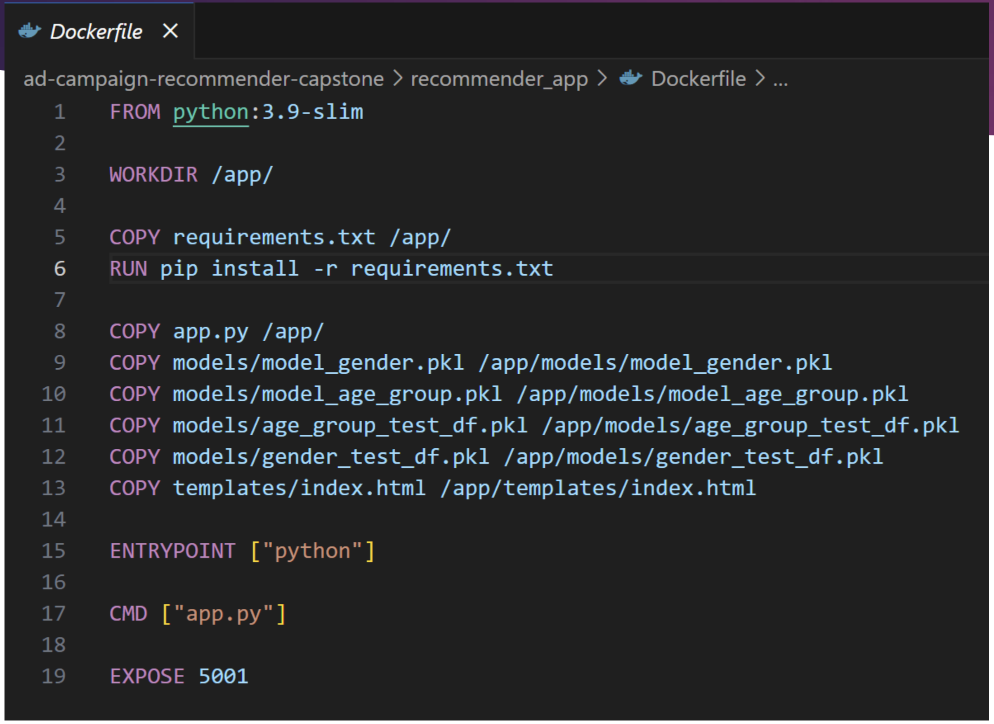
```
<html lang="en">
<body>
  <div class="container">
    <div class="row">
      <!-- User Selection -->
      <div class="col-md-8">
        <div class="container">
          <div id="message">
            <div class="alert alert-info">
              Please select one of the device Ids from left to get the pre
            </div>
          </div>
          <div id="loader" style="display:none">
            Loading...
          </div>
          <div id="prediction" style="display:none">
            <h3>Predicted output</h3>
            <!-- Gender -->
            <div class="row">
              <div class="col-md-4">
              <label for="device_id">Device Id:</label>
              </div>
              <div class="col-md-8">
              <span id="device_id"></span>
              </div>
            </div>

            <!-- Gender -->
            <div class="row">
              <div class="col-md-4">
              <label for="gender">Gender:</label>
              </div>
              <div class="col-md-8">
              <span id="gender">Male</span>
              </div>
            </div>

            <!-- Age Group -->
            <div class="row">
```


aign-recommender-capstone > recommender_app > templates > index.html > html > body > script > pr

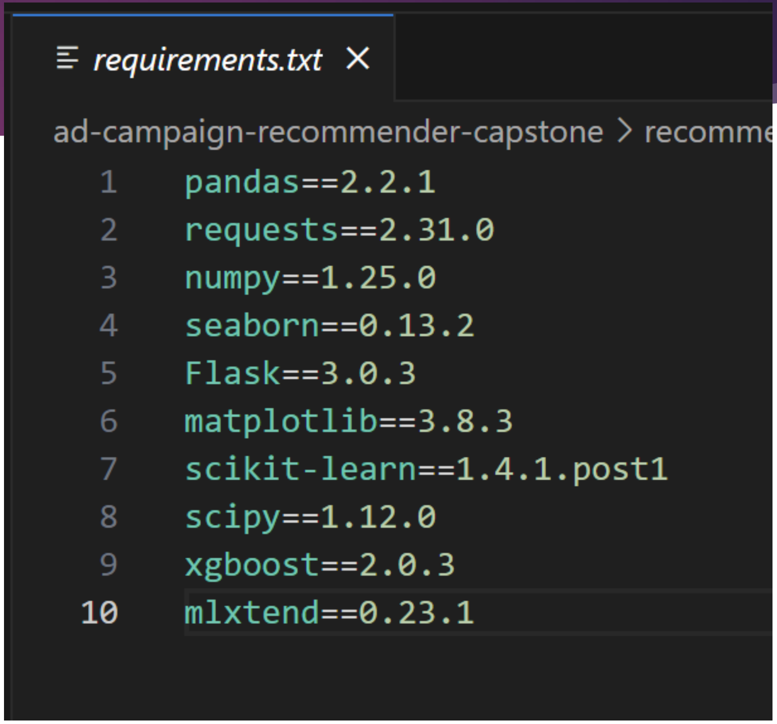
```
<html lang="en">
<body>
  <script>
    function predict_for_device_id(device_id) {
      document.getElementById("prediction").style.display = "none";

      // Make the POST request
      fetch(url, options)
        .then(response => {
          // Check if the response status is OK (HTTP 200)
          if (response.ok) {
            // Parse the JSON response
            return response.json();
          } else {
            // If response status is not OK, throw an error
            throw new Error('Failed to fetch data');
          }
        })
        .then(data => {
          // Handle the JSON response
          console.log('Prediction:', data);

          document.getElementById('age').innerHTML = data['age_group'];
          document.getElementById('gender').innerHTML = data['gender'];
          document.getElementById('device_id').innerHTML = data['device_id'];

          document.getElementById('campaigns').innerHTML = "";

          data['campaign'].forEach(campaign => {
            document.getElementById('campaigns').innerHTML +=
            '<li class="list-group-item"><b>' + campaign[0] + '</b> ' + campaign[1]
          });

          document.getElementById("loader").style.display = "none";
          document.getElementById("prediction").style.display = "block";
          // You can further process the prediction data here
        })
        .catch(error => {
          // Handle any errors that occurred during the fetch
```

The index.html file serves as the main interface for the Ad Campaign Recommender system. It allows users to select a device ID from a list and view the predicted age group, gender, and recommended ad campaigns based on the selected device ID.

Source Code – Flask App (Dockerfile and requirements.txt)


Dockerfile ×
ad-campaign-recommender-capstone > recommender_app > Dockerfile > ...

```
FROM python:3.9-slim

WORKDIR /app/

COPY requirements.txt /app/
RUN pip install -r requirements.txt

COPY app.py /app/
COPY models/model_gender.pkl /app/models/model_gender.pkl
COPY models/model_age_group.pkl /app/models/model_age_group.pkl
COPY models/age_group_test_df.pkl /app/models/age_group_test_df.pkl
COPY models/gender_test_df.pkl /app/models/gender_test_df.pkl
COPY templates/index.html /app/templates/index.html

ENTRYPOINT ["python"]

CMD ["app.py"]

EXPOSE 5001
```

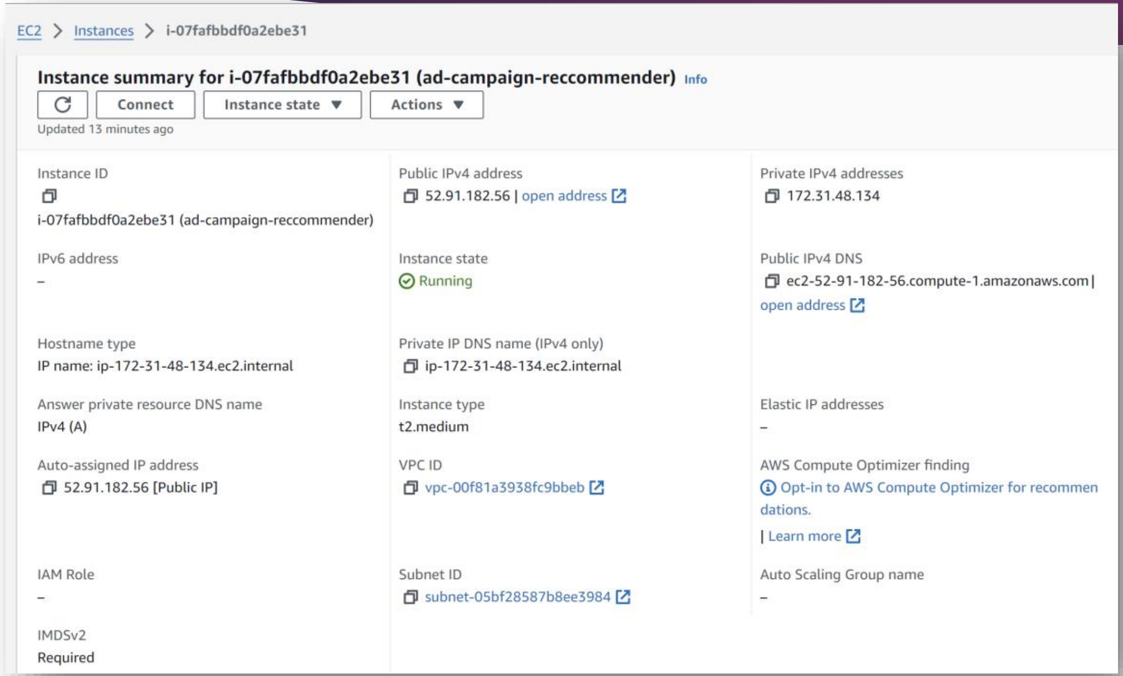


```
pandas==2.2.1
requests==2.31.0
numpy==1.25.0
seaborn==0.13.2
Flask==3.0.3
matplotlib==3.8.3
scikit-learn==1.4.1.post1
scipy==1.12.0
xgboost==2.0.3
mlxtend==0.23.1
```

The Dockerfile sets up a Python 3.9 environment, installs dependencies from requirements.txt, copies necessary application and model files, and runs the Flask app on port 5001.

The requirements.txt file lists essential Python libraries like pandas, Flask, scikit-learn, and xgboost, ensuring the application has all necessary dependencies for data manipulation, machine learning, and web functionality.

EC2 Instance and Security Group



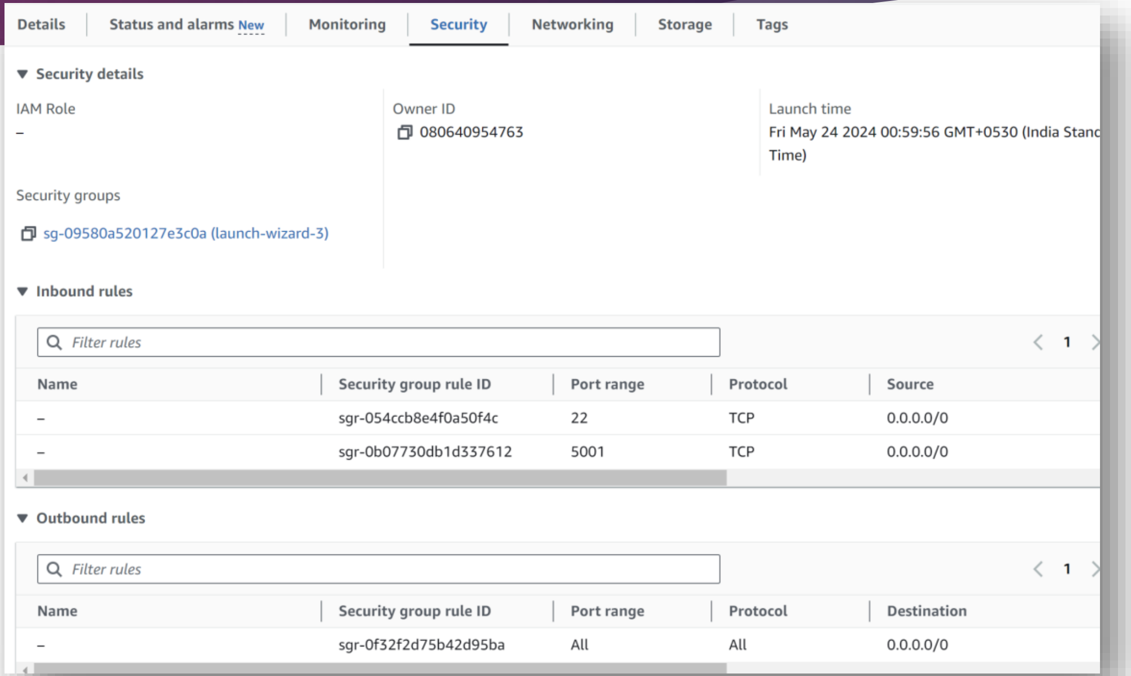
EC2 > Instances > i-07fafbbdf0a2ebe31

Instance summary for i-07fafbbdf0a2ebe31 (ad-campaign-reccommender) Info

Connect | Instance state ▼ | Actions ▼

Updated 13 minutes ago

Instance ID i-07fafbbdf0a2ebe31 (ad-campaign-reccommender)	Public IPv4 address 52.91.182.56 | open address	Private IPv4 addresses 172.31.48.134
IPv6 address –	Instance state Running	Public IPv4 DNS ec2-52-91-182-56.compute-1.amazonaws.com | open address
Hostname type IP name: ip-172-31-48-134.ec2.internal	Private IP DNS name (IPv4 only) ip-172-31-48-134.ec2.internal	
Answer private resource DNS name IPv4 (A)	Instance type t2.medium	Elastic IP addresses –
Auto-assigned IP address 52.91.182.56 [Public IP]	VPC ID vpc-00f81a3938fc9bbeb	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. | Learn more
IAM Role –	Subnet ID subnet-05bf28587b8ee3984	Auto Scaling Group name –
IMDSv2 Required		



Details | Status and alarms New | Monitoring | Security | Networking | Storage | Tags

▼ Security details

IAM Role –	Owner ID 080640954763	Launch time Fri May 24 2024 00:59:56 GMT+0530 (India Stand Time)
Security groups sg-09580a520127e3c0a (launch-wizard-3)		

▼ Inbound rules

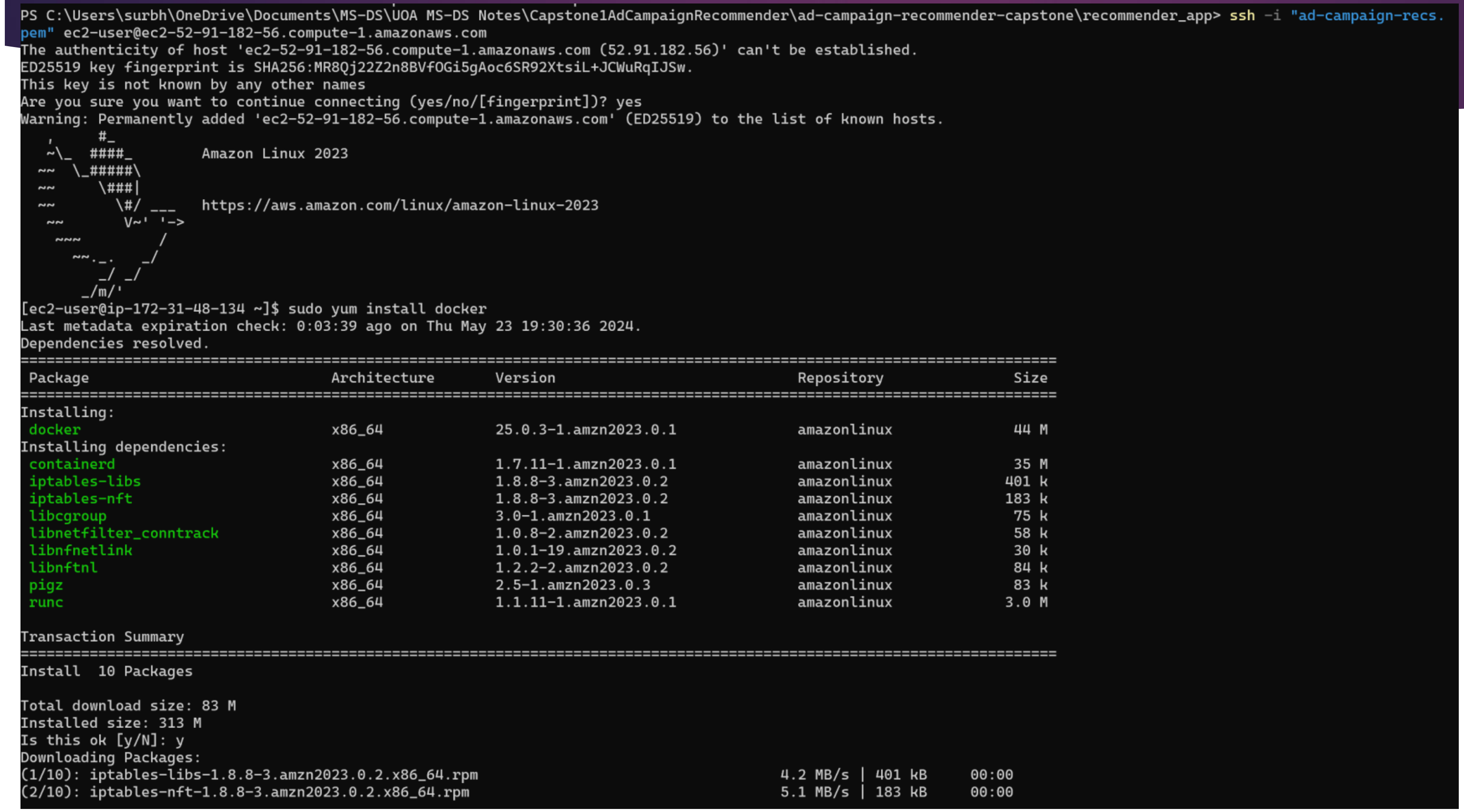
Name	Security group rule ID	Port range	Protocol	Source
–	sgr-054ccb8e4f0a50f4c	22	TCP	0.0.0.0/0
–	sgr-0b07730db1d337612	5001	TCP	0.0.0.0/0

▼ Outbound rules

Name	Security group rule ID	Port range	Protocol	Destination
–	sgr-0f32f2d75b42d95ba	All	All	0.0.0.0/0

Created an EC2 instance and added an inbound rule for port 5001 to enable traffic through that port.

Connect to EC2 Instance and install Docker

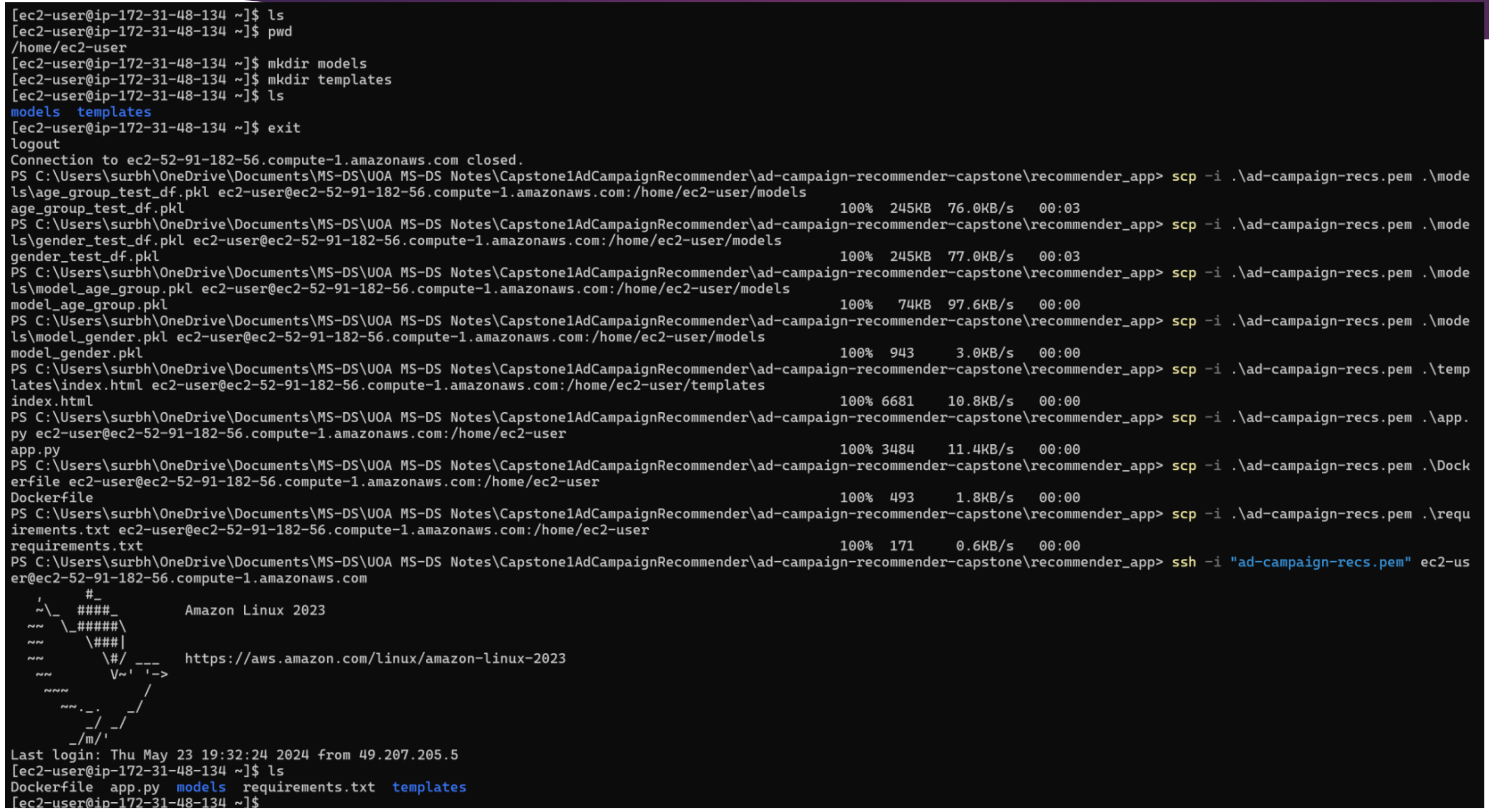


```
PS C:\Users\surbh\OneDrive\Documents\MS-DS\UOA MS-DS Notes\Capstone1AdCampaignRecommender\ad-campaign-recommender-capstone\recommender_app> ssh -i "ad-campaign-recs.
pem" ec2-user@ec2-52-91-182-56.compute-1.amazonaws.com
The authenticity of host 'ec2-52-91-182-56.compute-1.amazonaws.com (52.91.182.56)' can't be established.
ED25519 key fingerprint is SHA256:MR8Qj22Z2n8BVfOGi5gAoc6SR92XtsiL+JCWuRqIJSw.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-52-91-182-56.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-172-31-48-134 ~]$ sudo yum install docker
Last metadata expiration check: 0:03:39 ago on Thu May 23 19:30:36 2024.
Dependencies resolved.
========================================================================================================================
 Package                            Architecture        Version                            Repository             Size
========================================================================================================================
Installing:
 docker                             x86_64              25.0.3-1.amzn2023.0.1              amazonlinux            44 M
Installing dependencies:
 containerd                         x86_64              1.7.11-1.amzn2023.0.1              amazonlinux            35 M
 iptables-libs                      x86_64              1.8.8-3.amzn2023.0.2               amazonlinux           401 k
 iptables-nft                       x86_64              1.8.8-3.amzn2023.0.2               amazonlinux           183 k
 libcgroup                          x86_64              3.0-1.amzn2023.0.1                 amazonlinux            75 k
 libnetfilter_conntrack             x86_64              1.0.8-2.amzn2023.0.2               amazonlinux            58 k
 libnfnetlink                       x86_64              1.0.1-19.amzn2023.0.2              amazonlinux            30 k
 libnftnl                           x86_64              1.2.2-2.amzn2023.0.2               amazonlinux            84 k
 pigz                               x86_64              2.5-1.amzn2023.0.3                 amazonlinux            83 k
 runc                               x86_64              1.1.11-1.amzn2023.0.1              amazonlinux           3.0 M

Transaction Summary
========================================================================================================================
Install  10 Packages

Total download size: 83 M
Installed size: 313 M
Is this ok [y/N]: y
Downloading Packages:
(1/10): iptables-libs-1.8.8-3.amzn2023.0.2.x86_64.rpm                                  4.2 MB/s | 401 kB     00:00
(2/10): iptables-nft-1.8.8-3.amzn2023.0.2.x86_64.rpm                                   5.1 MB/s | 183 kB     00:00
```

- **Commands used:**
  - SSH into EC2 instance:
    ssh -i "ad-campaign-recs.pem" ec2-user@ec2-52-91-182-56.compute-1.amazonaws.com
  - Install Docker:
    sudo yum install docker

Note: "ad-campaign-recs.pem" is the EC2 key-pair used for ssh into EC2 instance.

Copy files from local to EC2 Instance

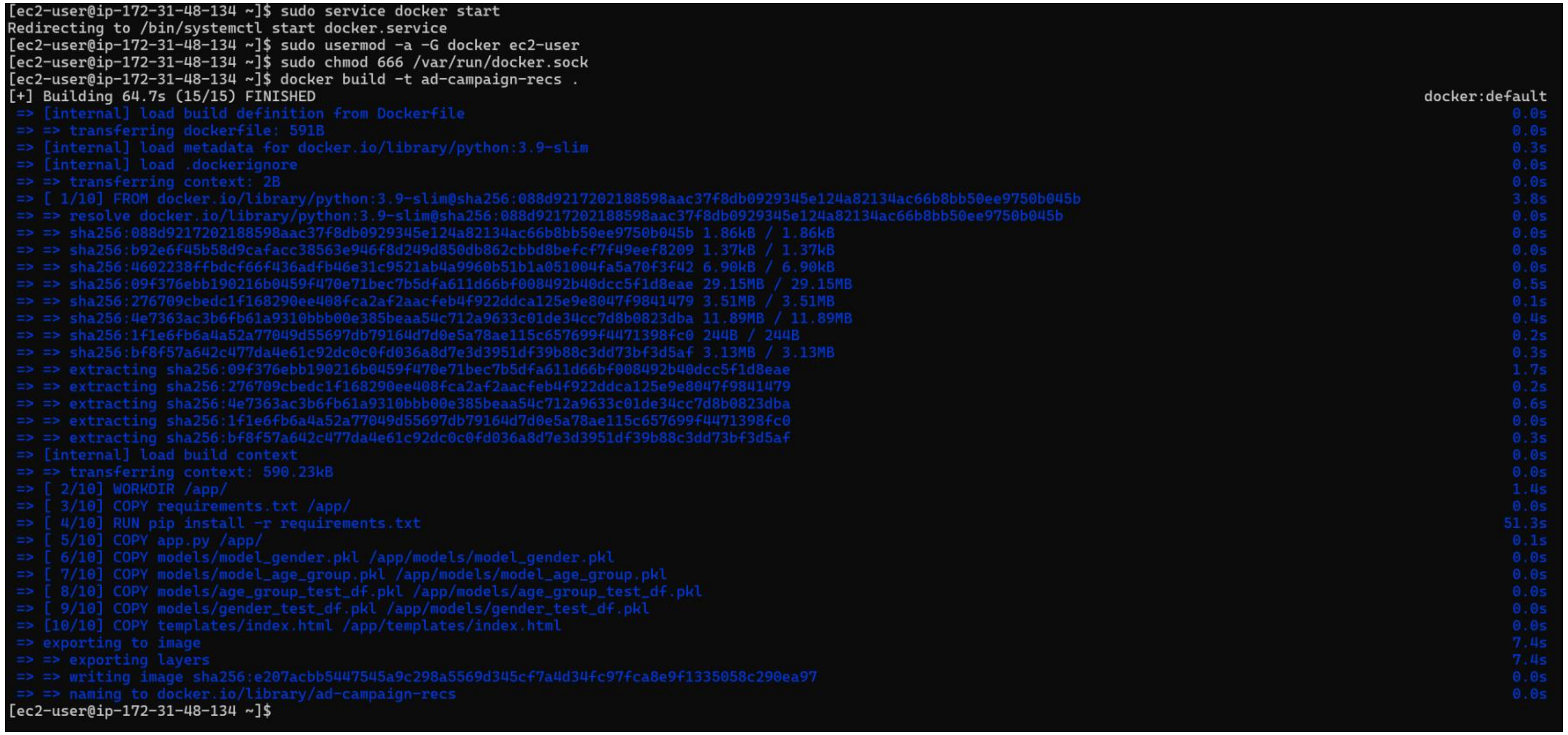


```
[ec2-user@ip-172-31-48-134 ~]$ ls
[ec2-user@ip-172-31-48-134 ~]$ pwd
/home/ec2-user
[ec2-user@ip-172-31-48-134 ~]$ mkdir models
[ec2-user@ip-172-31-48-134 ~]$ mkdir templates
[ec2-user@ip-172-31-48-134 ~]$ ls
models  templates
[ec2-user@ip-172-31-48-134 ~]$ exit
logout
Connection to ec2-52-91-182-56.compute-1.amazonaws.com closed.
PS C:\Users\surbh\OneDrive\Documents\MS-DS\UOA MS-DS Notes\Capstone1AdCampaignRecommender\ad-campaign-recommender-capstone\recommender_app> scp -i .\ad-campaign-recs.pem .\mode
ls\age_group_test_df.pkl ec2-user@ec2-52-91-182-56.compute-1.amazonaws.com:/home/ec2-user/models
age_group_test_df.pkl                                                                                  100%  245KB  76.0KB/s   00:03
PS C:\Users\surbh\OneDrive\Documents\MS-DS\UOA MS-DS Notes\Capstone1AdCampaignRecommender\ad-campaign-recommender-capstone\recommender_app> scp -i .\ad-campaign-recs.pem .\mode
ls\gender_test_df.pkl ec2-user@ec2-52-91-182-56.compute-1.amazonaws.com:/home/ec2-user/models
gender_test_df.pkl                                                                                     100%  245KB  77.0KB/s   00:03
PS C:\Users\surbh\OneDrive\Documents\MS-DS\UOA MS-DS Notes\Capstone1AdCampaignRecommender\ad-campaign-recommender-capstone\recommender_app> scp -i .\ad-campaign-recs.pem .\mode
ls\model_age_group.pkl ec2-user@ec2-52-91-182-56.compute-1.amazonaws.com:/home/ec2-user/models
model_age_group.pkl                                                                                    100%   74KB  97.6KB/s   00:00
PS C:\Users\surbh\OneDrive\Documents\MS-DS\UOA MS-DS Notes\Capstone1AdCampaignRecommender\ad-campaign-recommender-capstone\recommender_app> scp -i .\ad-campaign-recs.pem .\mode
ls\model_gender.pkl ec2-user@ec2-52-91-182-56.compute-1.amazonaws.com:/home/ec2-user/models
model_gender.pkl                                                                                       100%  943     3.0KB/s   00:00
PS C:\Users\surbh\OneDrive\Documents\MS-DS\UOA MS-DS Notes\Capstone1AdCampaignRecommender\ad-campaign-recommender-capstone\recommender_app> scp -i .\ad-campaign-recs.pem .\temp
lates\index.html ec2-user@ec2-52-91-182-56.compute-1.amazonaws.com:/home/ec2-user/templates
index.html                                                                                             100% 6681    10.8KB/s   00:00
PS C:\Users\surbh\OneDrive\Documents\MS-DS\UOA MS-DS Notes\Capstone1AdCampaignRecommender\ad-campaign-recommender-capstone\recommender_app> scp -i .\ad-campaign-recs.pem .\app.
py ec2-user@ec2-52-91-182-56.compute-1.amazonaws.com:/home/ec2-user
app.py                                                                                                 100% 3484    11.4KB/s   00:00
PS C:\Users\surbh\OneDrive\Documents\MS-DS\UOA MS-DS Notes\Capstone1AdCampaignRecommender\ad-campaign-recommender-capstone\recommender_app> scp -i .\ad-campaign-recs.pem .\Dock
erfile ec2-user@ec2-52-91-182-56.compute-1.amazonaws.com:/home/ec2-user
Dockerfile                                                                                             100%  493     1.8KB/s   00:00
PS C:\Users\surbh\OneDrive\Documents\MS-DS\UOA MS-DS Notes\Capstone1AdCampaignRecommender\ad-campaign-recommender-capstone\recommender_app> scp -i .\ad-campaign-recs.pem .\requ
irements.txt ec2-user@ec2-52-91-182-56.compute-1.amazonaws.com:/home/ec2-user
requirements.txt                                                                                       100%  171     0.6KB/s   00:00
PS C:\Users\surbh\OneDrive\Documents\MS-DS\UOA MS-DS Notes\Capstone1AdCampaignRecommender\ad-campaign-recommender-capstone\recommender_app> ssh -i "ad-campaign-recs.pem" ec2-us
er@ec2-52-91-182-56.compute-1.amazonaws.com
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Thu May 23 19:32:24 2024 from 49.207.205.5
[ec2-user@ip-172-31-48-134 ~]$ ls
Dockerfile  app.py  models  requirements.txt  templates
[ec2-user@ip-172-31-48-134 ~]$
```

Commands used:

```
scp -i .\ad-campaign-recs.pem .\models\age_group_test_df.pkl ec2-user@ec2-52-91-182-56.compute-1.amazonaws.com:/home/ec2-user/models
scp -i .\ad-campaign-recs.pem .\models\gender_test_df.pkl ec2-user@ec2-52-91-182-56.compute-1.amazonaws.com:/home/ec2-user/models
 scp -i .\ad-campaign-recs.pem .\models\model_age_group.pkl ec2-user@ec2-52-91-182-56.compute-1.amazonaws.com:/home/ec2-user/models
scp -i .\ad-campaign-recs.pem .\models\model_gender.pkl ec2-user@ec2-52-91-182-56.compute-1.amazonaws.com:/home/ec2-user/models

 scp -i .\ad-campaign-recs.pem .\templates\index.html ec2-user@ec2-52-91-182-56.compute-1.amazonaws.com:/home/ec2-user/templates

scp -i .\ad-campaign-recs.pem .\app.py ec2-user@ec2-52-91-182-56.compute-1.amazonaws.com:/home/ec2-user

scp -i .\ad-campaign-recs.pem .\Dockerfile ec2-user@ec2-52-91-182-56.compute-1.amazonaws.com:/home/ec2-user

scp -i .\ad-campaign-recs.pem .\requirements.txt ec2-user@ec2-52-91-182-56.compute-1.amazonaws.com:/home/ec2-user
```

Start docker service on EC2 and Create docker image



```
[ec2-user@ip-172-31-48-134 ~]$ sudo service docker start
Redirecting to /bin/systemctl start docker.service
[ec2-user@ip-172-31-48-134 ~]$ sudo usermod -a -G docker ec2-user
[ec2-user@ip-172-31-48-134 ~]$ sudo chmod 666 /var/run/docker.sock
[ec2-user@ip-172-31-48-134 ~]$ docker build -t ad-campaign-recs .
[+] Building 64.7s (15/15) FINISHED                                                                                                                              docker:default
 => [internal] load build definition from Dockerfile                                                                                                                       0.0s
 => => transferring dockerfile: 591B                                                                                                                                       0.0s
 => [internal] load metadata for docker.io/library/python:3.9-slim                                                                                                         0.3s
 => [internal] load .dockerignore                                                                                                                                          0.0s
 => => transferring context: 2B                                                                                                                                            0.0s
 => [ 1/10] FROM docker.io/library/python:3.9-slim@sha256:088d9217202188598aac37f8db0929345e124a82134ac66b8bb50ee9750b045b                                                 3.8s
 => => resolve docker.io/library/python:3.9-slim@sha256:088d9217202188598aac37f8db0929345e124a82134ac66b8bb50ee9750b045b                                                   0.0s
 => => sha256:088d9217202188598aac37f8db0929345e124a82134ac66b8bb50ee9750b045b 1.86kB / 1.86kB                                                                             0.0s
 => => sha256:b92e6f45b58d9cafacc38563e946f8d249d850db862cbbd8befcf7f49eef8209 1.37kB / 1.37kB                                                                             0.0s
 => => sha256:4602238ffbdcf66f436adfb46e31c9521ab4a9960b51b1a051004fa5a70f3f42 6.90kB / 6.90kB                                                                             0.0s
 => => sha256:09f376ebb190216b0459f470e71bec7b5dfa611d66bf008492b40dcc5f1d8eae 29.15MB / 29.15MB                                                                           0.5s
 => => sha256:276709cbedc1f168290ee408fca2af2aacfeb4f922ddca125e9e8047f9841479 3.51MB / 3.51MB                                                                             0.1s
 => => sha256:4e7363ac3b6fb61a9310bbb00e385beaa54c712a9633c01de34cc7d8b0823dba 11.89MB / 11.89MB                                                                           0.4s
 => => sha256:1f1e6fb6a4a52a77049d55697db79164d7d0e5a78ae115c657699f4471398fc0 244B / 244B                                                                                 0.2s
 => => sha256:bf8f57a642c477da4e61c92dc0c0fd036a8d7e3d3951df39b88c3dd73bf3d5af 3.13MB / 3.13MB                                                                             0.3s
 => => extracting sha256:09f376ebb190216b0459f470e71bec7b5dfa611d66bf008492b40dcc5f1d8eae                                                                                  1.7s
 => => extracting sha256:276709cbedc1f168290ee408fca2af2aacfeb4f922ddca125e9e8047f9841479                                                                                  0.2s
 => => extracting sha256:4e7363ac3b6fb61a9310bbb00e385beaa54c712a9633c01de34cc7d8b0823dba                                                                                  0.6s
 => => extracting sha256:1f1e6fb6a4a52a77049d55697db79164d7d0e5a78ae115c657699f4471398fc0                                                                                  0.0s
 => => extracting sha256:bf8f57a642c477da4e61c92dc0c0fd036a8d7e3d3951df39b88c3dd73bf3d5af                                                                                  0.3s
 => [internal] load build context                                                                                                                                          0.0s
 => => transferring context: 590.23kB                                                                                                                                      0.0s
 => [ 2/10] WORKDIR /app/                                                                                                                                                  1.4s
 => [ 3/10] COPY requirements.txt /app/                                                                                                                                    0.0s
 => [ 4/10] RUN pip install -r requirements.txt                                                                                                                           51.3s
 => [ 5/10] COPY app.py /app/                                                                                                                                              0.1s
 => [ 6/10] COPY models/model_gender.pkl /app/models/model_gender.pkl                                                                                                      0.0s
 => [ 7/10] COPY models/model_age_group.pkl /app/models/model_age_group.pkl                                                                                                0.0s
 => [ 8/10] COPY models/age_group_test_df.pkl /app/models/age_group_test_df.pkl                                                                                            0.0s
 => [ 9/10] COPY models/gender_test_df.pkl /app/models/gender_test_df.pkl                                                                                                  0.0s
 => [10/10] COPY templates/index.html /app/templates/index.html                                                                                                            0.0s
 => exporting to image                                                                                                                                                     7.4s
 => => exporting layers                                                                                                                                                    7.4s
 => => writing image sha256:e207acbb5447545a9c298a5569d345cf7a4d34fc97fca8e9f1335058c290ea97                                                                               0.0s
 => => naming to docker.io/library/ad-campaign-recs                                                                                                                        0.0s
[ec2-user@ip-172-31-48-134 ~]$
```

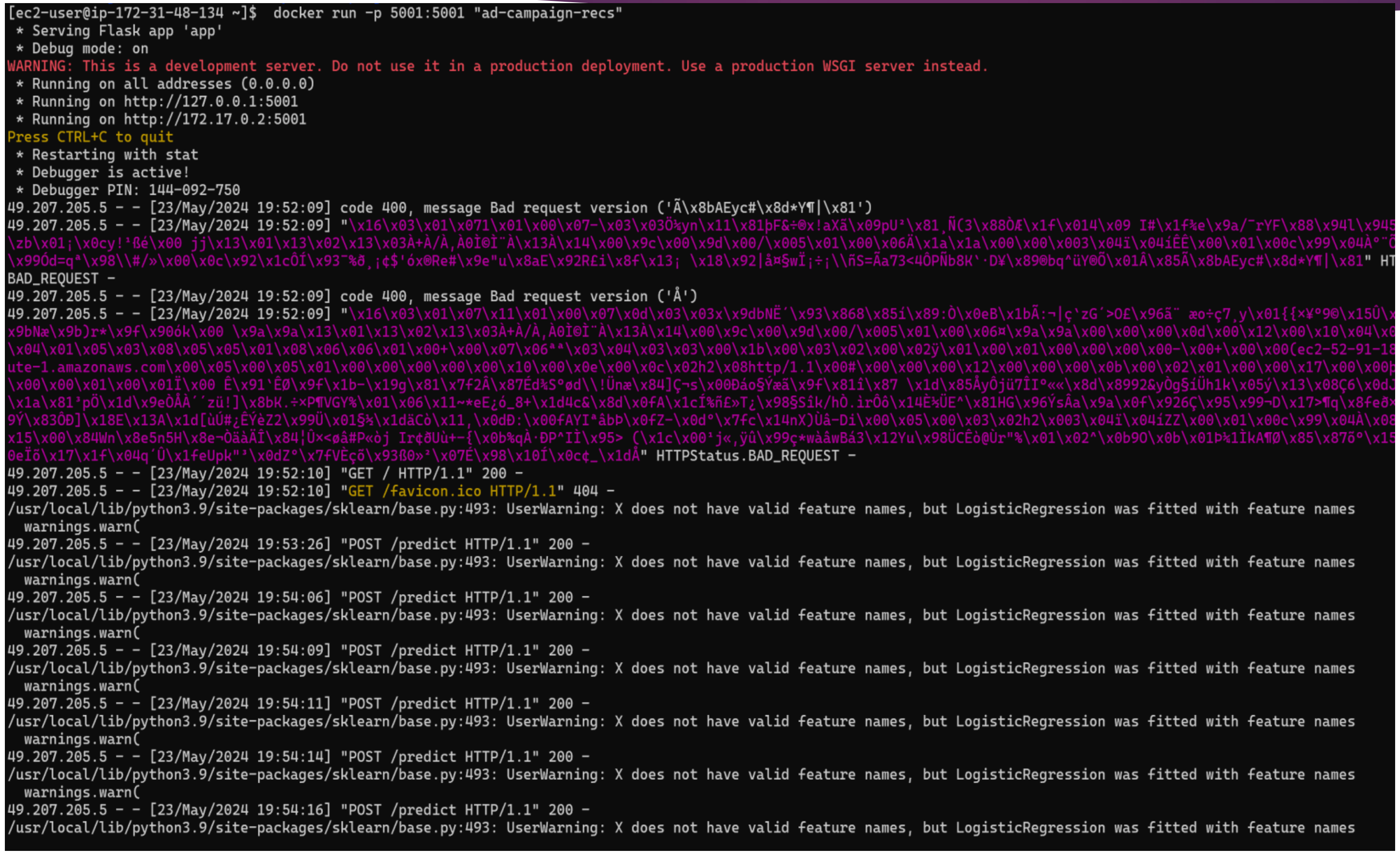
- **Commands used:**

- Starting the docker service:
  sudo service docker start

- Enable permissions:
  sudo usermod -a -G docker ec2-user
  sudo chmod 666 /var/run/docker.sock

- Build docker image:
  docker build –t ad-campaign-recs .

Start Docker Container

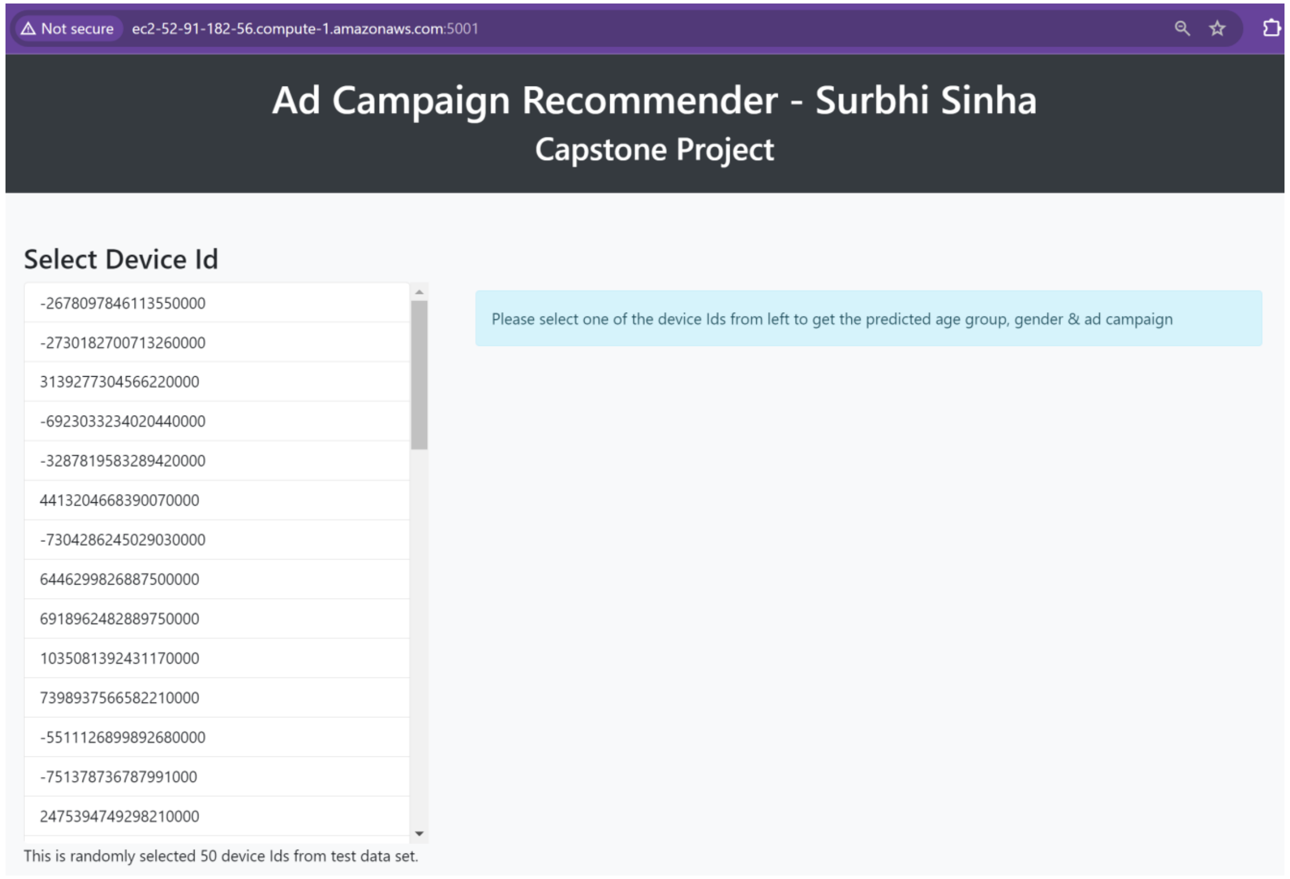


```
[ec2-user@ip-172-31-48-134 ~]$  docker run -p 5001:5001 "ad-campaign-recs"
 * Serving Flask app 'app'
 * Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:5001
 * Running on http://172.17.0.2:5001
Press CTRL+C to quit
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 144-092-750
49.207.205.5 - - [23/May/2024 19:52:09] code 400, message Bad request version ('Ã\x8bAEyc#\x8d*Y¶|\x81')
49.207.205.5 - - [23/May/2024 19:52:09] "\x16\x03\x01\x071\x01\x00\x07-\x03\x03Ö¾yn\x11\x81þF&÷®x!aXã\x09pU²\x81¸Ñ(3\x88ÒÆ\x1f\x014\x09 I#\x1f¾e\x9a/¯rYF\x88\x94l\x945
\zb\x01¡\x0cy!¹ßé\x00 jj\x13\x01\x13\x02\x13\x03À+À/À,À0Ì©Ì¨À\x13À\x14\x00\x9c\x00\x9d\x00/\x005\x01\x00\x06Ä\x1a\x1a\x00\x00\x003\x04ï\x04íÊÊ\x00\x01\x00c\x99\x04À°¨Õ
\x99Ód=qª\x98\\#/»\x00\x0c\x92\x1cÔÍ\x93¯%ð¸¡¢$'óx®Re#\x9e"u\x8aE\x92R£i\x8f\x13¡ \x18\x92|å¤§wÏ¡÷¡\\ñS=Ãa73<4ÔPÑb8K`·D¥\x89®bq^üY®Õ\x01Â\x85Ã\x8bAEyc#\x8d*Y¶|\x81" HT
BAD_REQUEST -
49.207.205.5 - - [23/May/2024 19:52:09] code 400, message Bad request version ('Å')
49.207.205.5 - - [23/May/2024 19:52:09] "\x16\x03\x01\x07\x11\x01\x00\x07\x0d\x03\x03x\x9dbNË´\x93\x868\x85í\x89:Ò\x0eB\x1bÃ:¬|ç`zG´>O£\x96ã¨ æo÷ç7¸y\x01{{×¥º9©\x15Û\x
x9bNæ\x9b)r*\x9f\x90ók\x00 \x9a\x9a\x13\x01\x13\x02\x13\x03À+À/À,À0Ì©Ì¨À\x13À\x14\x00\x9c\x00\x9d\x00/\x005\x01\x00\x06¤\x9a\x9a\x00\x00\x00\x0d\x00\x12\x00\x10\x04\x0
\x04\x01\x05\x03\x08\x05\x05\x01\x08\x06\x06\x01\x00+\x00\x07\x06ªª\x03\x04\x03\x03\x00\x1b\x00\x03\x02\x00\x02ÿ\x01\x00\x01\x00\x00\x00\x00-\x00+\x00\x00(ec2-52-91-18
ute-1.amazonaws.com\x00\x05\x00\x05\x01\x00\x00\x00\x00\x00\x10\x00\x0e\x00\x0c\x02h2\x08http/1.1\x00#\x00\x00\x00\x12\x00\x00\x00\x0b\x00\x02\x01\x00\x00\x17\x00\x00þ
\x00\x00\x01\x00\x01Ï\x00 Ê\x91`ÊØ\x9f\x1b-\x19g\x81\x7f2Â\x87Éd¾S°ød\\!Ünæ\x84]Ç¬s\x00Ðáo§Ýæã\x9f\x81î\x87 \x1d\x85ÅyÔjü7ÎI°««\x8d\x8992&yÒg§íÜh1k\x05ý\x13\x08Ç6\x0dJ
\x1a\x81³pÖ\x1d\x9eÒÅÀ´´zü!]\x8bK.÷×P¶VGY%\x01\x06\x11~*eE¿ó_8+\x1d4c&\x8d\x0fA\x1cÍ%ñ£»T¿\x98§Sîk/hÒ.ìrÔô\x14È¾ÜE^\x81HG\x96ÝsÂa\x9a\x0f\x926Ç\x95\x99¬D\x17>¶q\x8feð×
9Ý\x83ÔÐ]\x18E\x13A\x1d[ùÚ#¿ÊÝèZ2\x99Ü\x01§½\x1däCò\x11,\x0dÐ:\x00fAYIªâbÞ\x0fZ-\x0d°\x7fc\x14nX)Ùâ-Di\x00\x05\x00\x03\x02h2\x003\x04ï\x04íZZ\x00\x01\x00c\x99\x04À\x08
x15\x00\x84Wn\x8e5n5H\x8e¬ÒäàÄÎ\x84¦Û×<øâ#P«òj Ir¢ðUù+-{\x0b%qÀ·ÐP^IÌ\x95> (\x1c\x00¹j«,ÿû\x99ç*wàâwBá3\x12Yu\x98ÜCÊò@Ùr"%\x01\x02^\x0b9O\x0b\x01Þ¼1ÌkA¶Ø\x85\x87õ°\x15
0eÏõ\x17\x1f\x04q´Û\x1feUpk"³\x0dZ°\x7fVÈçõ\x93ß0»²\x07É\x98\x10Í\x0c¢_\x1dÅ" HTTPStatus.BAD_REQUEST -
49.207.205.5 - - [23/May/2024 19:52:10] "GET / HTTP/1.1" 200 -
49.207.205.5 - - [23/May/2024 19:52:10] "GET /favicon.ico HTTP/1.1" 404 -
/usr/local/lib/python3.9/site-packages/sklearn/base.py:493: UserWarning: X does not have valid feature names, but LogisticRegression was fitted with feature names
  warnings.warn(
49.207.205.5 - - [23/May/2024 19:53:26] "POST /predict HTTP/1.1" 200 -
/usr/local/lib/python3.9/site-packages/sklearn/base.py:493: UserWarning: X does not have valid feature names, but LogisticRegression was fitted with feature names
  warnings.warn(
49.207.205.5 - - [23/May/2024 19:54:06] "POST /predict HTTP/1.1" 200 -
/usr/local/lib/python3.9/site-packages/sklearn/base.py:493: UserWarning: X does not have valid feature names, but LogisticRegression was fitted with feature names
  warnings.warn(
49.207.205.5 - - [23/May/2024 19:54:09] "POST /predict HTTP/1.1" 200 -
/usr/local/lib/python3.9/site-packages/sklearn/base.py:493: UserWarning: X does not have valid feature names, but LogisticRegression was fitted with feature names
  warnings.warn(
49.207.205.5 - - [23/May/2024 19:54:11] "POST /predict HTTP/1.1" 200 -
/usr/local/lib/python3.9/site-packages/sklearn/base.py:493: UserWarning: X does not have valid feature names, but LogisticRegression was fitted with feature names
  warnings.warn(
49.207.205.5 - - [23/May/2024 19:54:14] "POST /predict HTTP/1.1" 200 -
/usr/local/lib/python3.9/site-packages/sklearn/base.py:493: UserWarning: X does not have valid feature names, but LogisticRegression was fitted with feature names
  warnings.warn(
49.207.205.5 - - [23/May/2024 19:54:16] "POST /predict HTTP/1.1" 200 -
/usr/local/lib/python3.9/site-packages/sklearn/base.py:493: UserWarning: X does not have valid feature names, but LogisticRegression was fitted with feature names
```

Command used for starting the docker container:

docker run –p 5001:5001 “ad-campaign-recs”

Access the Web Application from the browser


Not secure ec2-52-91-182-56.compute-1.amazonaws.com:5001

Ad Campaign Recommender - Surbhi Sinha
Capstone Project

Select Device Id

-2678097846113550000
-2730182700713260000
3139277304566220000
-6923033234020440000
-3287819583289420000
4413204668390070000
-7304286245029030000
6446299826887500000
6918962482889750000
1035081392431170000
7398937566582210000
-5511126899892680000
-751378736787991000
2475394749298210000

This is randomly selected 50 device Ids from test data set.

Please select one of the device Ids from left to get the predicted age group, gender & ad campaign

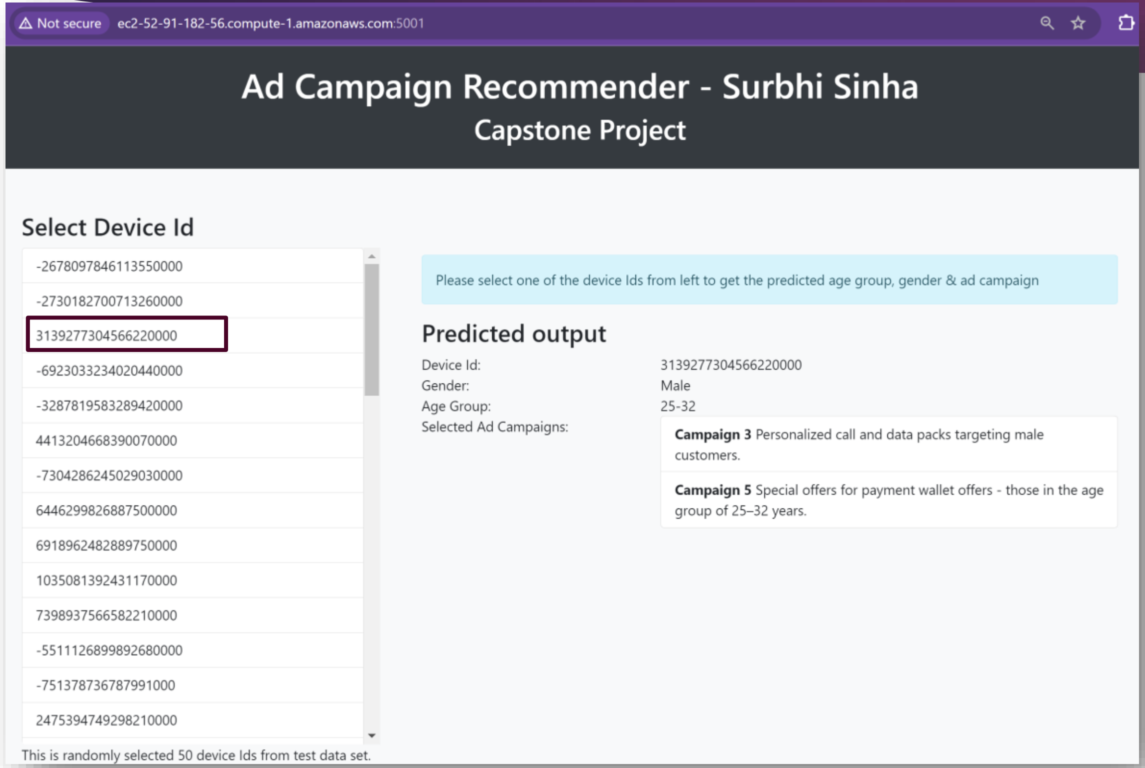
Access the web application from the browser using below address:

<public IPV4 DNS>:5001

Note: Retrieve the public IPv4 DNS of the EC2 instance, and note that port 5001 is the one exposed by our application.

The screenshot shows our application running. On the left side, we can choose a device ID from the 50 randomly populated device IDs in the test dataset. Once a device is selected, we will receive the predicted age group, gender, and campaign.

Ad-Campaign Recommender Application page



⚠ Not secure ec2-52-91-182-56.compute-1.amazonaws.com:5001

Ad Campaign Recommender - Surbhi Sinha
Capstone Project

Select Device Id

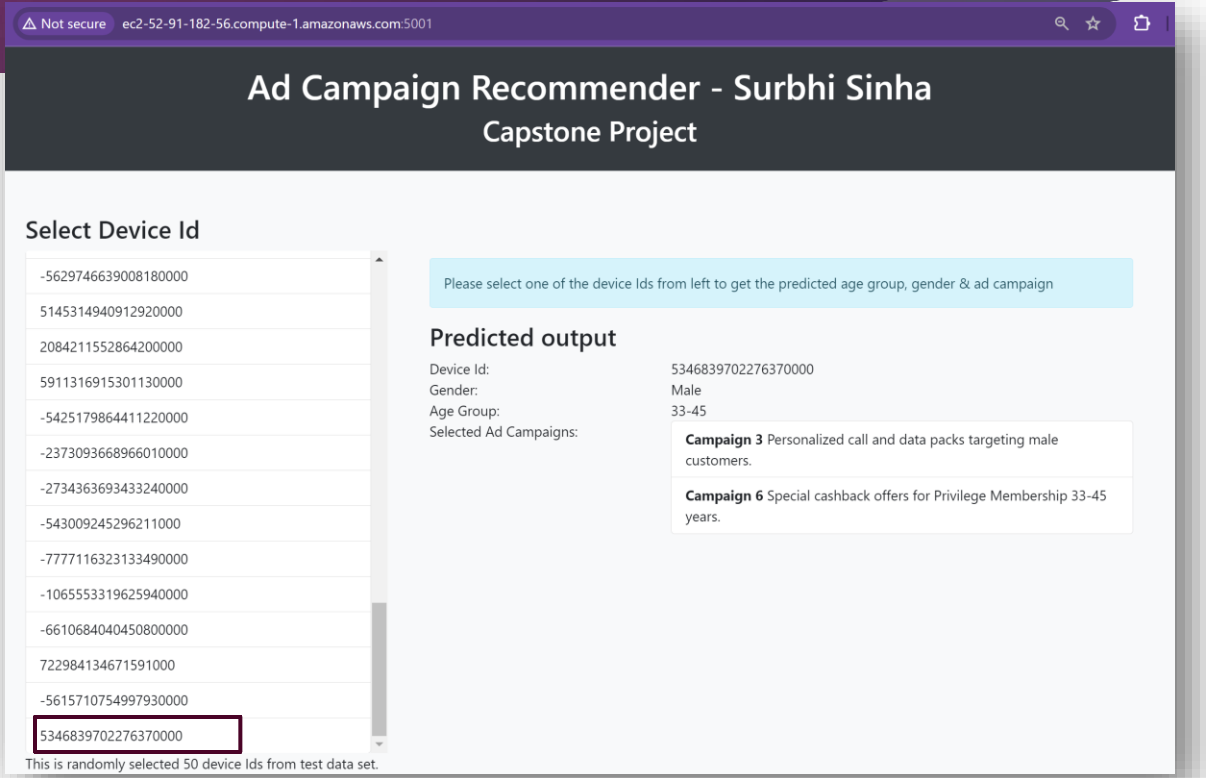
-2678097846113550000
-2730182700713260000
3139277304566220000
-6923033234020440000
-3287819583289420000
4413204668390070000
-7304286245029030000
6446299826887500000
6918962482889750000
1035081392431170000
7398937566582210000
-5511126899892680000
-751378736787991000
2475394749298210000

This is randomly selected 50 device Ids from test data set.

Please select one of the device Ids from left to get the predicted age group, gender & ad campaign

Predicted output

Device Id:	3139277304566220000
Gender:	Male
Age Group:	25-32
Selected Ad Campaigns:	**Campaign 3** Personalized call and data packs targeting male customers. **Campaign 5** Special offers for payment wallet offers - those in the age group of 25–32 years.



⚠ Not secure ec2-52-91-182-56.compute-1.amazonaws.com:5001

Ad Campaign Recommender - Surbhi Sinha
Capstone Project

Select Device Id

-5629746639008180000
5145314940912920000
2084211552864200000
5911316915301130000
-5425179864411220000
-2373093668966010000
-2734363693433240000
-543009245296211000
-7777116323133490000
-1065553319625940000
-6610684040450800000
722984134671591000
-5615710754997930000
5346839702276370000

This is randomly selected 50 device Ids from test data set.

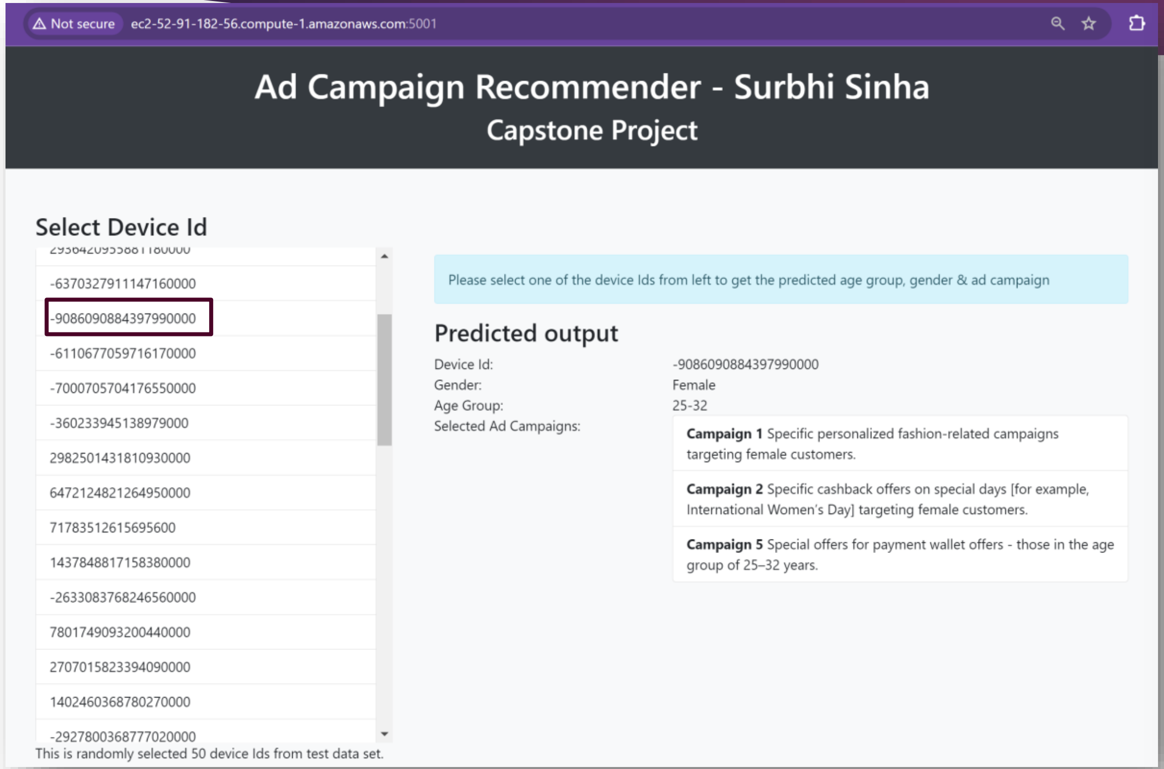
Please select one of the device Ids from left to get the predicted age group, gender & ad campaign

Predicted output

Device Id:	5346839702276370000
Gender:	Male
Age Group:	33-45
Selected Ad Campaigns:	**Campaign 3** Personalized call and data packs targeting male customers. **Campaign 6** Special cashback offers for Privilege Membership 33-45 years.

The screenshots above show the predictions for the selected device ID. As the predicted age group and gender change, the suggested campaign also updates accordingly.

Ad-Campaign Recommender Application page



Not secure ec2-52-91-182-56.compute-1.amazonaws.com:5001

Ad Campaign Recommender - Surbhi Sinha

Capstone Project

Select Device Id

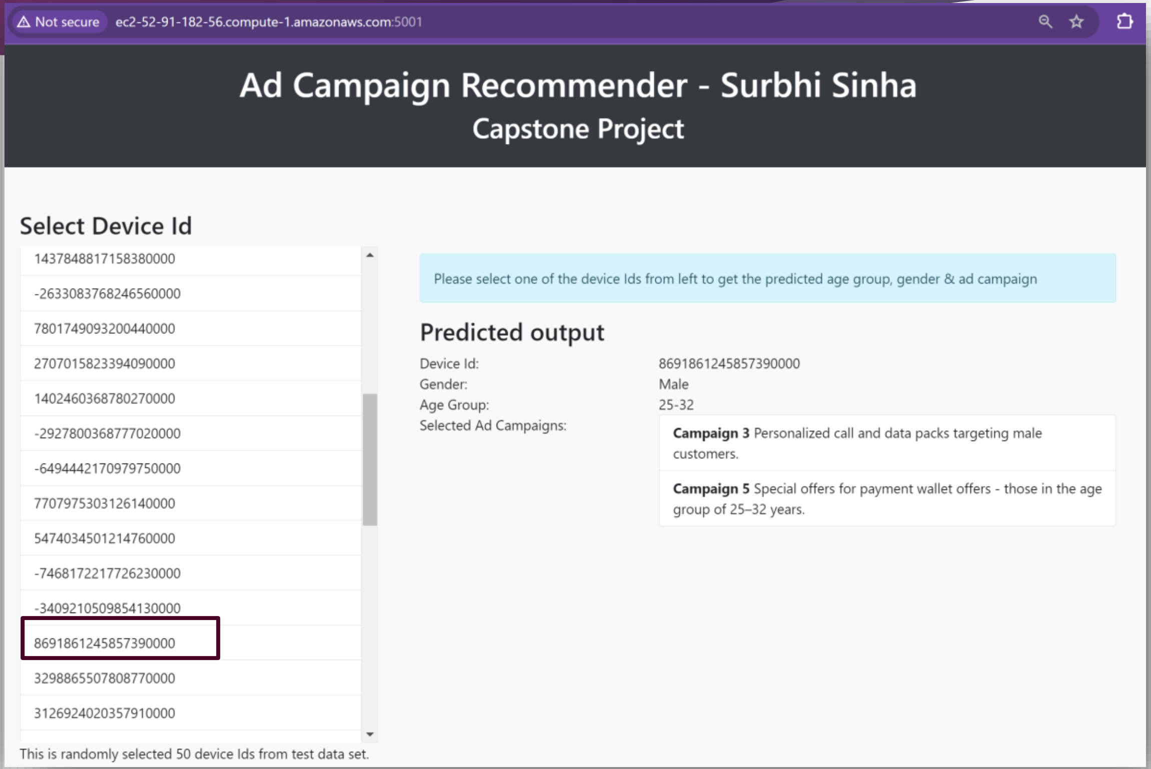
```
2936420955881180000
-6370327911147160000
-9086090884397990000
-6110677059716170000
-7000705704176550000
-3602339451389790000
2982501431810930000
6472124821264950000
717835126156956000
1437848817158380000
-2633083768246560000
7801749093200440000
2707015823394090000
1402460368780270000
-2927800368777020000
```

This is randomly selected 50 device Ids from test data set.

Please select one of the device Ids from left to get the predicted age group, gender & ad campaign

Predicted output

Device Id:	-9086090884397990000
Gender:	Female
Age Group:	25-32
Selected Ad Campaigns:	**Campaign 1** Specific personalized fashion-related campaigns targeting female customers. **Campaign 2** Specific cashback offers on special days [for example, International Women's Day] targeting female customers. **Campaign 5** Special offers for payment wallet offers - those in the age group of 25–32 years.



Not secure ec2-52-91-182-56.compute-1.amazonaws.com:5001

Ad Campaign Recommender - Surbhi Sinha

Capstone Project

Select Device Id

```
1437848817158380000
-2633083768246560000
7801749093200440000
2707015823394090000
1402460368780270000
-2927800368777020000
-6494442170979750000
7707975303126140000
5474034501214760000
-7468172217726230000
-3409210509854130000
8691861245857390000
3298865507808770000
3126924020357910000
```

This is randomly selected 50 device Ids from test data set.

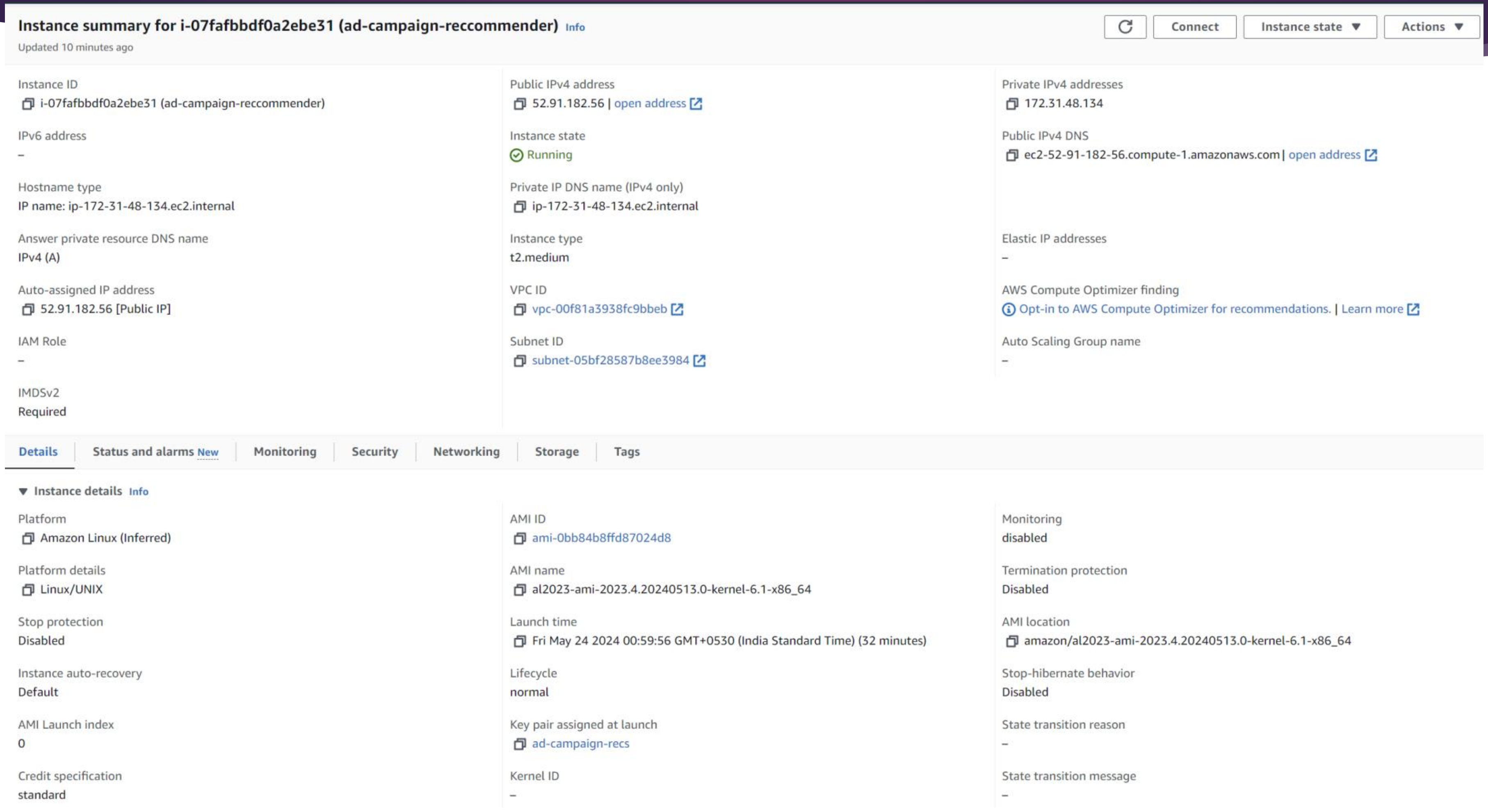
Please select one of the device Ids from left to get the predicted age group, gender & ad campaign

Predicted output

Device Id:	8691861245857390000
Gender:	Male
Age Group:	25-32
Selected Ad Campaigns:	**Campaign 3** Personalized call and data packs targeting male customers. **Campaign 5** Special offers for payment wallet offers - those in the age group of 25–32 years.

The screenshots above show the predictions for the selected device ID. As the predicted age group and gender change, the suggested campaign also updates accordingly.

AWS EC2 Instance Stats



Instance summary for i-07fafbbdf0a2ebe31 (ad-campaign-reccommender) Info

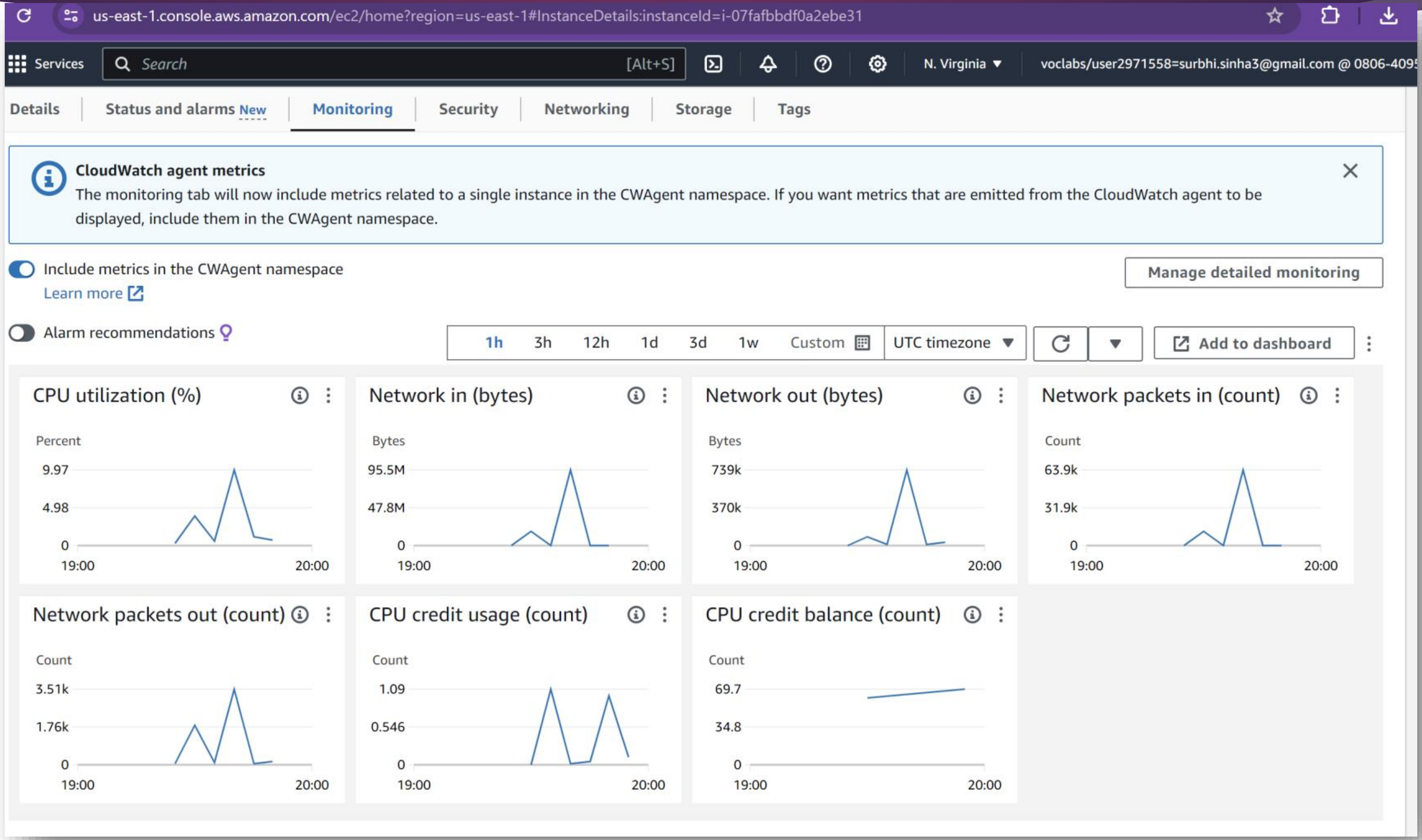
Updated 10 minutes ago

Instance ID i-07fafbbdf0a2ebe31 (ad-campaign-reccommender)	Public IPv4 address 52.91.182.56 | open address	Private IPv4 addresses 172.31.48.134
IPv6 address –	Instance state Running	Public IPv4 DNS ec2-52-91-182-56.compute-1.amazonaws.com | open address
Hostname type IP name: ip-172-31-48-134.ec2.internal	Private IP DNS name (IPv4 only) ip-172-31-48-134.ec2.internal	
Answer private resource DNS name IPv4 (A)	Instance type t2.medium	Elastic IP addresses –
Auto-assigned IP address 52.91.182.56 [Public IP]	VPC ID vpc-00f81a3938fc9bbeb	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. | Learn more
IAM Role –	Subnet ID subnet-05bf28587b8ee3984	Auto Scaling Group name –
IMDSv2 Required		

▼ **Instance details** Info

Platform Amazon Linux (Inferred)	AMI ID ami-0bb84b8ffd87024d8	Monitoring disabled
Platform details Linux/UNIX	AMI name al2023-ami-2023.4.20240513.0-kernel-6.1-x86_64	Termination protection Disabled
Stop protection Disabled	Launch time Fri May 24 2024 00:59:56 GMT+0530 (India Standard Time) (32 minutes)	AMI location amazon/al2023-ami-2023.4.20240513.0-kernel-6.1-x86_64
Instance auto-recovery Default	Lifecycle normal	Stop-hibernate behavior Disabled
AMI Launch index 0	Key pair assigned at launch ad-campaign-recs	State transition reason –
Credit specification standard	Kernel ID –	State transition message –

AWS EC2 Instance Stats


us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#InstanceDetails:instanceId=i-07fafbbdf0a2ebe31

Services | Search [Alt+S] | N. Virginia | voclabs/user2971558=surbhi.sinha3@gmail.com @ 0806-409

Details | Status and alarms New | Monitoring | Security | Networking | Storage | Tags

CloudWatch agent metrics
The monitoring tab will now include metrics related to a single instance in the CWAgent namespace. If you want metrics that are emitted from the CloudWatch agent to be displayed, include them in the CWAgent namespace.

Include metrics in the CWAgent namespace
Learn more

Manage detailed monitoring

Alarm recommendations

1h | 3h | 12h | 1d | 3d | 1w | Custom | UTC timezone | Add to dashboard

THANK YOU