MODEL DEPLOYMENT

Hrisav Bhowmick

9th May, 2021

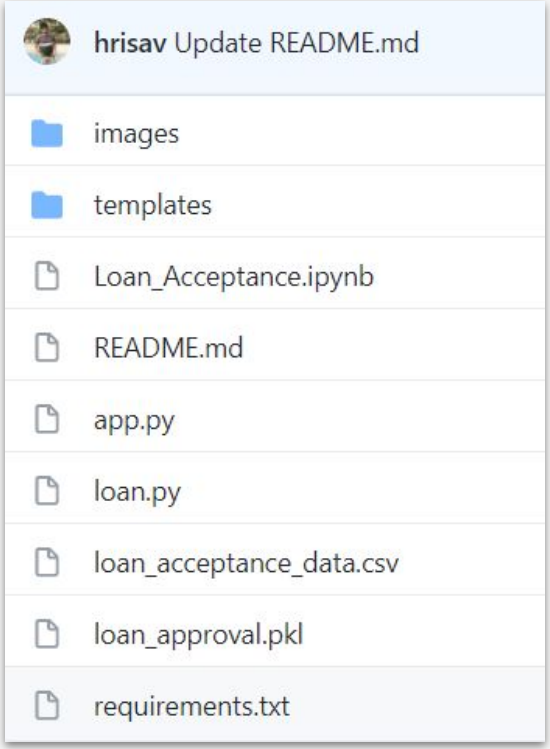
About Project

- *Dream Housing Finance Company* deals in all home loans.
- They have presence across all urban, semi urban and rural areas.
- Initially the customers apply for a home loan and then the company validates the customer if he/she is eligible for loan.
- Company wants to automate the loan eligibility process.
- To automate this process, they have given a problem to identify the customers that are eligible for loan, so that they can specifically target these them.

Project files

- Dataset
- Notebook
- Data folder

NOTE:
loan_approval.pkl file will get generated after executing loan.py. In app.py we have read the pkl file to make the Flask application.

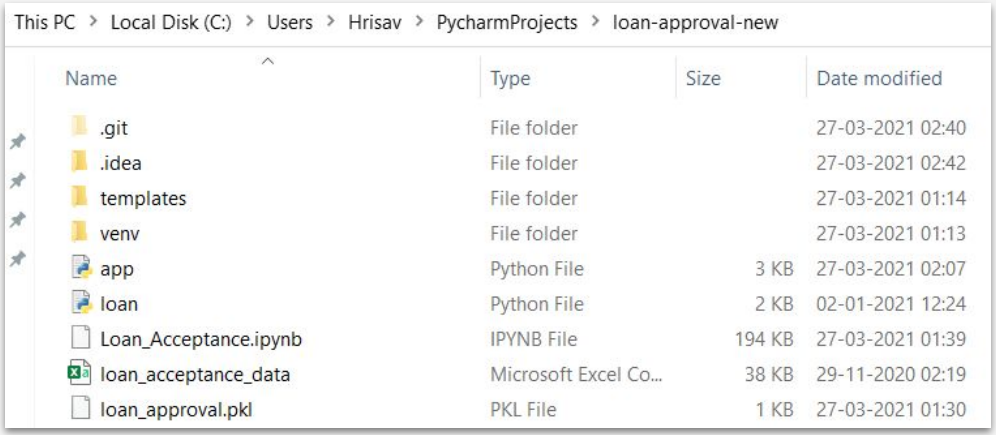


Running on LocalHost

(using Python localhost Server)

Opening PyCharm

- First check the local PyCharm directory (name: loan-approval-new).



- Open Pycharm and select the same folder to open.
- Make sure in app.py, deploy line is commented and local system line is uncommented. While deploying on AWS, we need to do the reverse.

```
70  #app.run(host="0.0.0.0")          # deploy
71  app.run(debug=True)               # run on local system
```

Installing dependencies

- In Pycharm terminal, install pip. Force reinstall if already installed.

  ```
  > python -m pip install -U --force-reinstall pip
  ```

- Install dependencies or libraries required to run the project.

  ```
  > python -m pip install Flask
  ```

  Like this do for all the libraries like sklearn, pandas, numpy, etc.

- Once all the libraries are installed make sure to generate a ‘requirements.txt’ file which will have name and version of all the libraries required to run the project.

  ```
  > pip freeze > requirements.txt
  ```

- Run below command to see list of libraries.

  ```
  > pip list
  ```

```
(venv) C:\Users\Hrisav\PycharmProjects\loan-approval-new>pip list
Package         Version
--------------- -------
click           7.1.2
Flask           1.1.2
itsdangerous    1.1.0
```

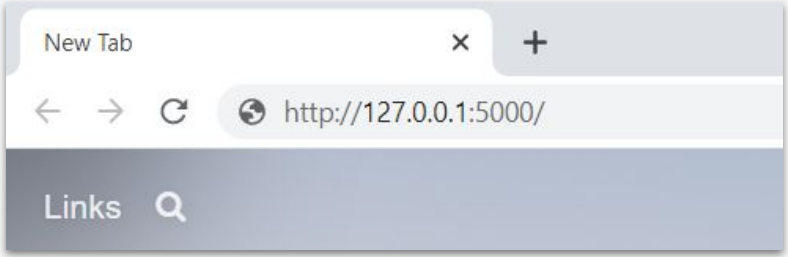
Running script

- Run app.py. If it does not throw error, it means all the libraries have been successfully installed.

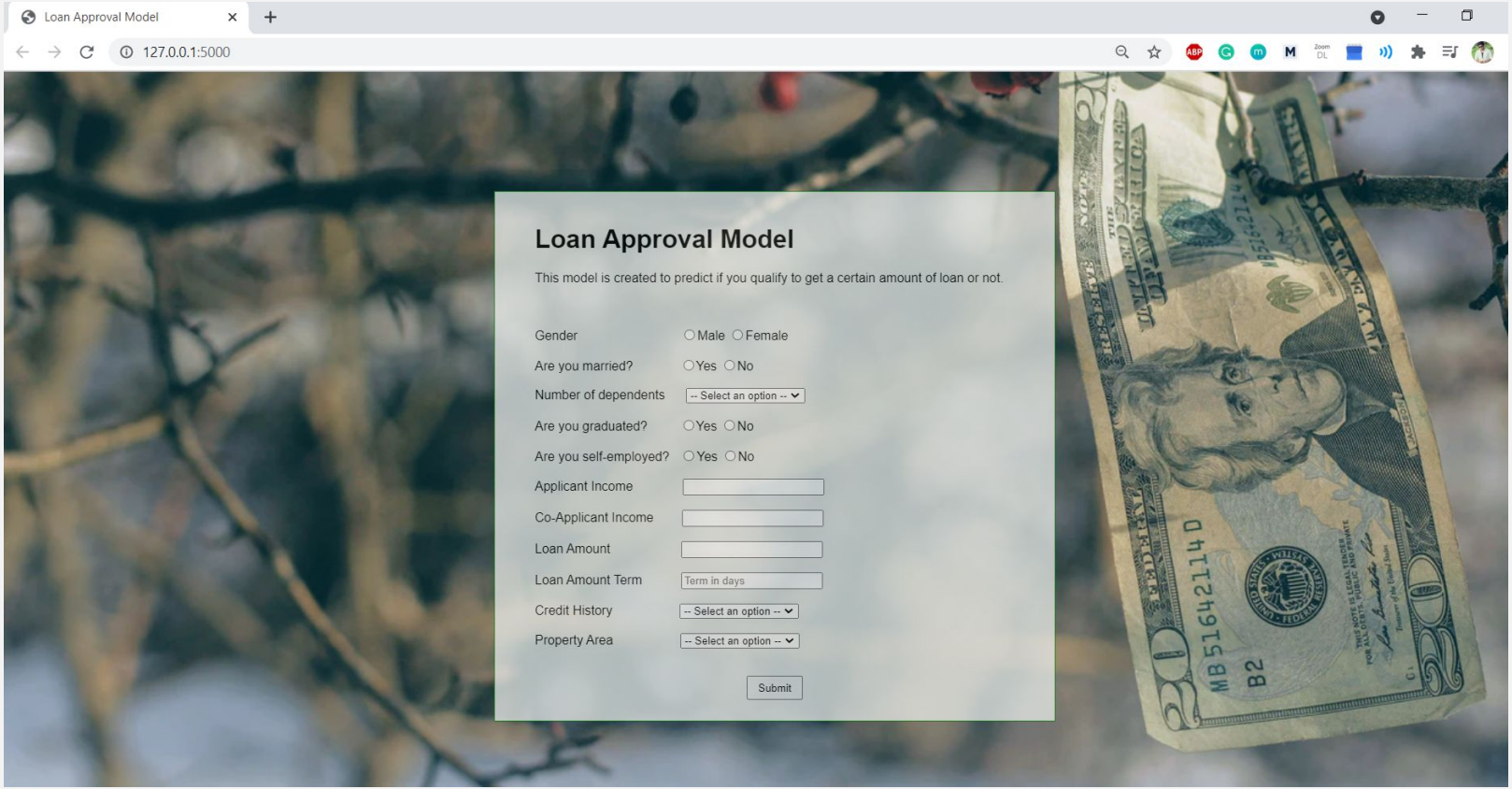
> python app.py

```
(venv) C:\Users\Hrisav\PycharmProjects\loan-approval-new>python app.py
 * Serving Flask app "app" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 853-201-556
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [09/May/2021 13:26:34] "←[37mGET / HTTP/1.1←[0m" 200 -
127.0.0.1 - - [09/May/2021 13:26:35] "←[33mGET /favicon.ico HTTP/1.1←[0m" 404 -
```

- Click on the blue link to run the web app on browser. Or copy paste the link and run on browser: http://127.0.0.1:5000/



Web-App

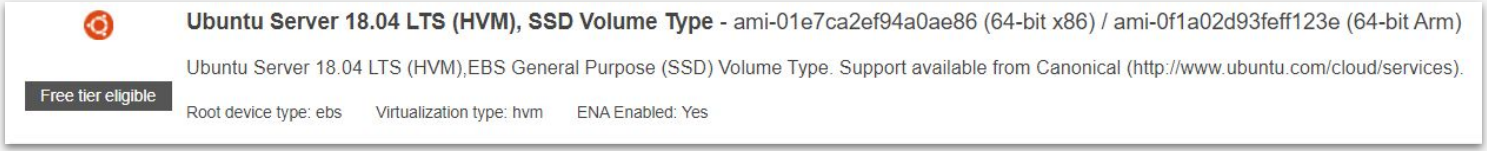

Loan Approval Model
127.0.0.1:5000
Loan Approval Model
This model is created to predict if you qualify to get a certain amount of loan or not.
Gender
Male
Female
Are you married?
Yes
No
Number of dependents
-- Select an option --
Are you graduated?
Yes
No
Are you self-employed?
Yes
No
Applicant Income
Co-Applicant Income
Loan Amount
Loan Amount Term
Term in days
Credit History
-- Select an option --
Property Area
-- Select an option --
Submit

Running on WebHost

(using Amazon-EC2)

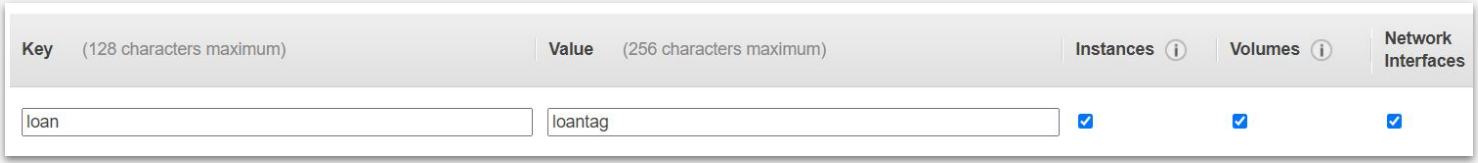
Creating Instance on EC2

- Go to EC2 Dashboard. (https://us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2)
- Click on Instances.
- Click on Launch Instances.
- In step 1, choose an AMI – select Ubuntu Server 18.04. Click Next.



- Choose instance type in step 2 – *'t2.micro'*. Click Next.

- Ignore step 3, 4 and on step 5 click Add Tag. Key: *'loan'*, Value: *'loantag'*. Click next.

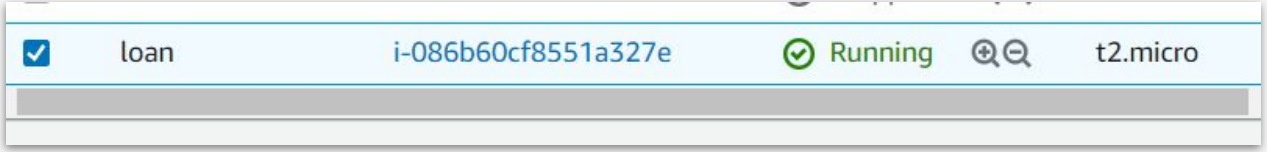


- In step 6, change security group name to *'sgloantag'*. Click Review and Launch.

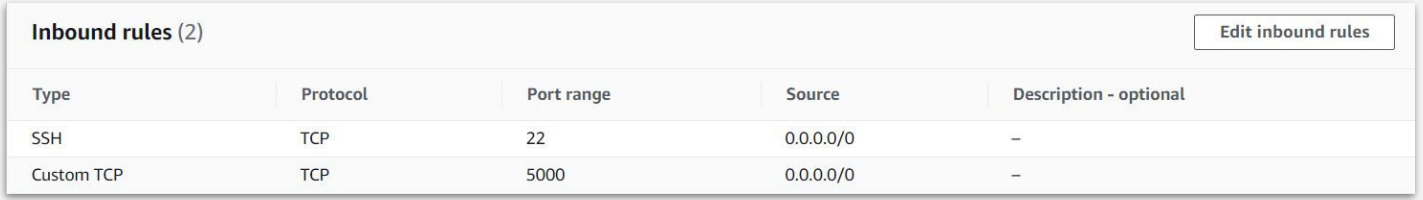
Launching Instance

- Click Launch. Create key pair name – *'kploantag'*. Download key pair. Save the PEM file in a new folder (name: deploy) on Desktop. Click Launch Instances.

- Go to view instances and add a name for the instance – *'loan'*. It should show Running.



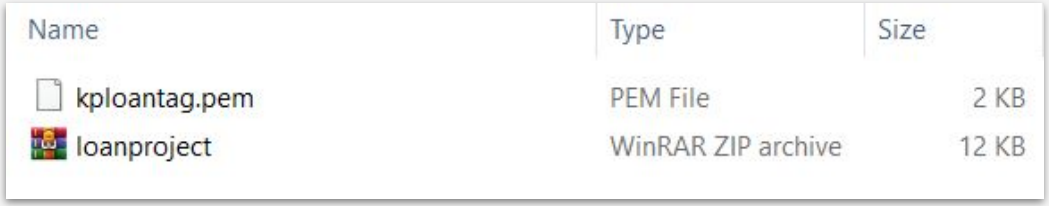
- Now click on instance id. Go to security tab. Click on Security Group link. Edit inbound rules -> Add rule: Custom TCP – Port : 5000 – Source: 0.0.0.0/0 - Save the rule.



Getting directory ready

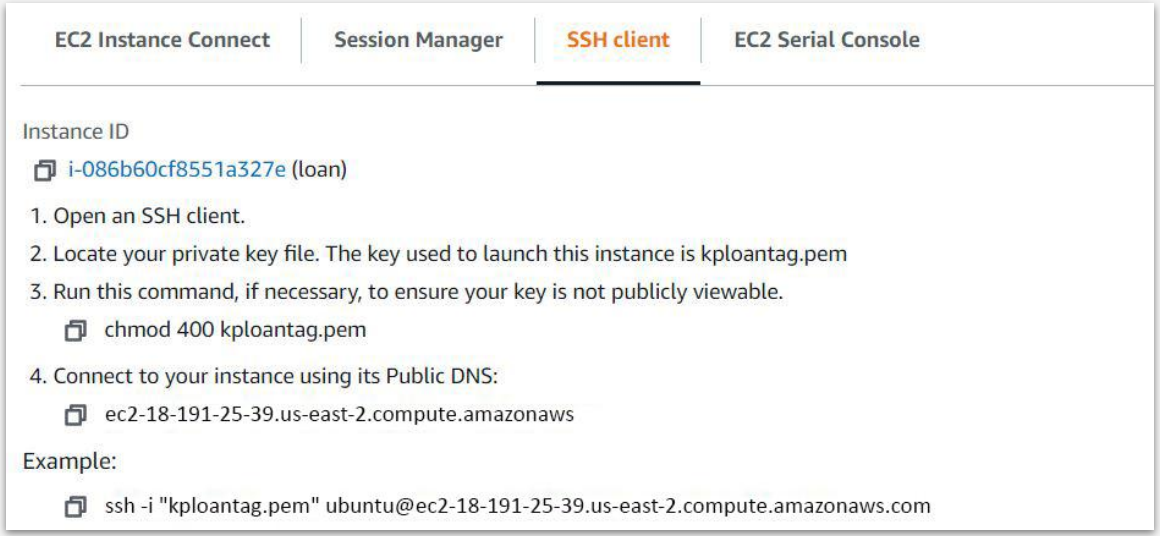
- Go the the folder (name: loanproject) containing project files. It should contain: dataset, app.py, requirements.txt, pkl file, templates folder having the html file. Make a zip file of the project folder as loanproject.zip.

- Go to the folder (name: deploy) where PEM file was saved. Copy loanproject.zip and paste it in same folder where PEM file lies.



Checking connection links

- Go to the instance page on EC2, select instance and select Connect. Below screen will show up.

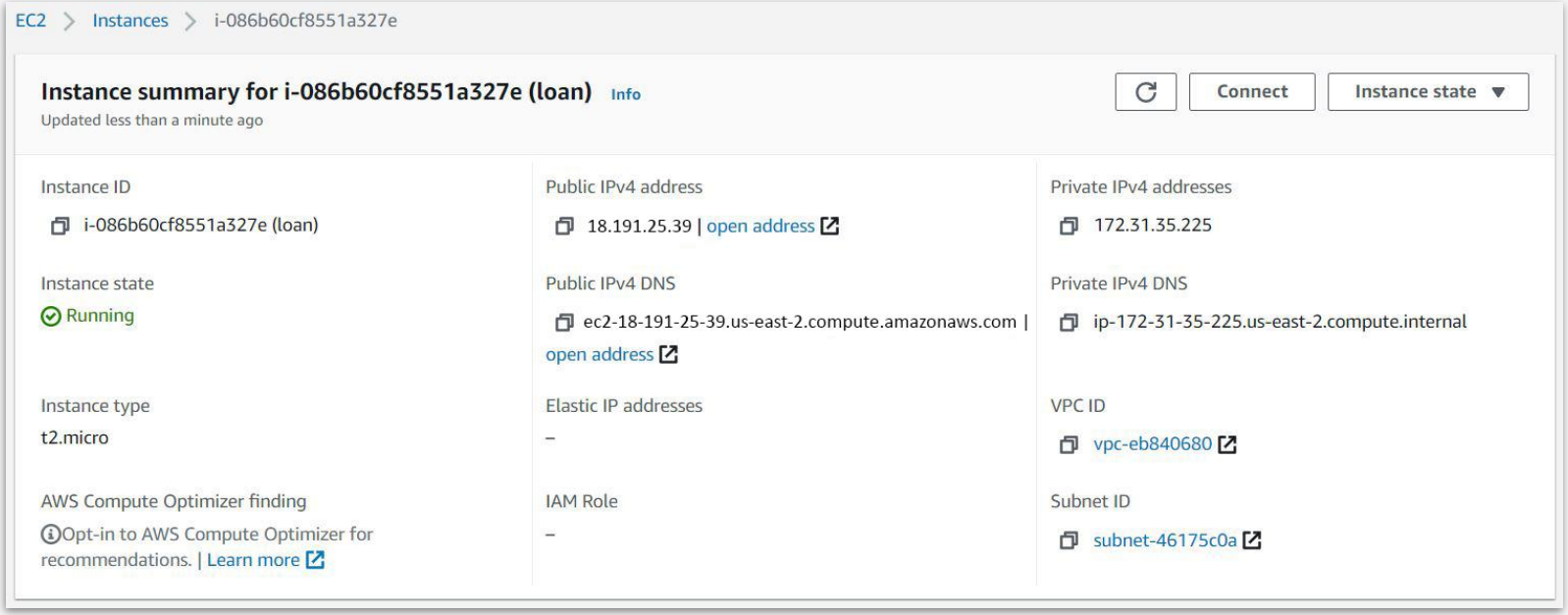


Last line of code will be used to connect to server.

Fourth line of code will be used to view the project UI on browser.

Checking instance summary

- Click on instance name and check if instance summary details are fine (like shown in the figure).



Connecting to Instance

- Open powershell from that folder by going to address bar and typing powershell and after it starts run the following commands:

> scp -i kploantag.pem loanproject.zip ubuntu@ec2-18-191-25-39.us-east-2.compute.amazonaws.com:

> ssh -i "kploantag.pem" ubuntu@ec2-18-191-25-39.us-east-2.compute.amazonaws.com

```
PS C:\Users\Hrisav\Desktop\deploy> scp -i kploantag.pem loanproject.zip ubuntu@ec2-18-191-25-39.us-east-2.compute.amazon
aws.com:
The authenticity of host 'ec2-18-191-25-39.us-east-2.compute.amazonaws.com (18.191.25.39)' can't be established.
ECDSA key fingerprint is SHA256:yf2BfsqgIIVJI3tpmh4ayNSc7Xh/imQWiYOAfko6wzk.
Are you sure you want to continue connecting (yes/no)?
Warning: Permanently added 'ec2-18-191-25-39.us-east-2.compute.amazonaws.com,18.191.25.39' (ECDSA) to the list of known
hosts.
loanproject.zip                                                                100%   11KB  31.6KB/s   00:00
PS C:\Users\Hrisav\Desktop\deploy> ssh -i "kploantag.pem" ubuntu@ec2-18-191-25-39.us-east-2.compute.amazonaws.com
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-1038-aws x86_64)Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-1038-aws x
86_64)

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

  System information as of Sat May  1 06:51:33 UTC 2021

  System load:  0.04              Processes:           93
  Usage of /:   14.8% of 7.69GB   Users logged in:     0
  Memory usage: 22%               IP address for eth0: 172.31.35.225
  Swap usage:   0%
```

Getting files on server

> dir

> sudo apt install zip unzip

> unzip loanproject.zip

> cd loanproject

```
ubuntu@ip-172-31-35-225:~$ dir
loanproject.zip
ubuntu@ip-172-31-35-225:~$ sudo apt install zip unzip
Reading package lists... Done
Selecting previously unselected package unzip.
(Reading database ... 57101 files and directories currently installed.)
Unpacking unzip (6.0-21ubuntu1.1) ...
Selecting previously unselected package zip...
Preparing to unpack .../zip_3.0-11build1_amd64.deb ...
Unpacking zip (3.0-11build1) ...
Setting up unzip (6.0-21ubuntu1.1) ...
Setting up zip (3.0-11build1) ...
Processing triggers for mime-support (3.60ubuntu1) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...bionic/main amd64 zip amd64 3.0-11build1 [167 kB]
ubuntu@ip-172-31-35-225:~$ unzip loanproject.zip
Archive:  loanproject.zip
   creating: loanproject/
  inflating: loanproject/app.py
  inflating: loanproject/loan_acceptance_data.csv
  inflating: loanproject/loan_approval.pkl
  inflating: loanproject/requirements.txt
   creating: loanproject/templates/
  inflating: loanproject/templates/index.html
ubuntu@ip-172-31-35-225:~$ dir
loanproject  loanproject.zip
ubuntu@ip-172-31-35-225:~$ cd loanproject
ubuntu@ip-172-31-35-225:~/loanproject$ dir
app.py  loan_acceptance_data.csv  loan_approval.pkl  requirements.txt  templates
ubuntu@ip-172-31-35-225:~/loanproject$
```

Getting environment ready

```
> sudo apt-get update
```

```
> sudo apt-get install -y python3-pip
```

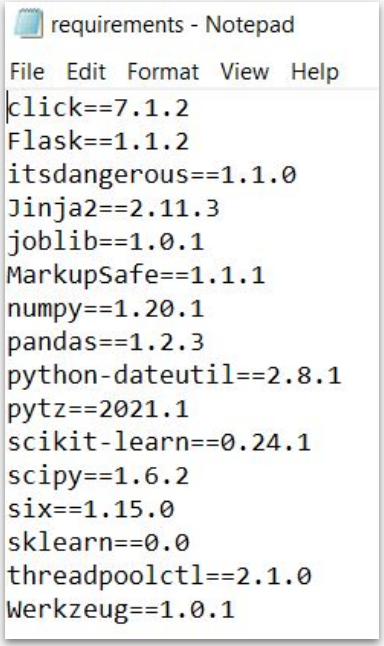
```
> pip3 install -r requirements.txt
```

The 'requirements.txt' contains all the libraries required to run the project.

If while running the last code any error shows up, check which library failed to install and install them manually like below:

```
> pip3 install numpy
```

In my case, I had to install flask and sklearn also manually along with numpy. It was mainly due to version compatibility issue.



Running script

Finally run the python file app.py:

> python3 app.py

```
ubuntu@ip-172-31-35-225:~/loanproject$ dir
app.py  loan_acceptance_data.csv  loan_approval.pkl  requirements.txt  templates
ubuntu@ip-172-31-35-225:~/loanproject$ python3 app.py
/home/ubuntu/.local/lib/python3.6/site-packages/sklearn/base.py:315: UserWarning: Trying to unpickle estimator Lo
ng version 0.24.2. This might lead to breaking code or invalid results. Use at your own risk.
  UserWarning)
 * Serving Flask app "app" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
45.112.242.216 - - [01/May/2021 06:59:05] "GET / HTTP/1.1" 200 -
45.112.242.216 - - [01/May/2021 06:59:06] "GET /favicon.ico HTTP/1.1" 404 -
/home/ubuntu/.local/lib/python3.6/site-packages/sklearn/utils/validation.py:63: FutureWarning: Arrays of bytes/st
f dtype='numeric'. This behavior is deprecated in 0.24 and will be removed in 1.1 (renaming of 0.26). Please conv
stead.
  return f(*args, **kwargs)
45.112.242.216 - - [01/May/2021 07:00:32] "POST /predict HTTP/1.1" 200 -
```

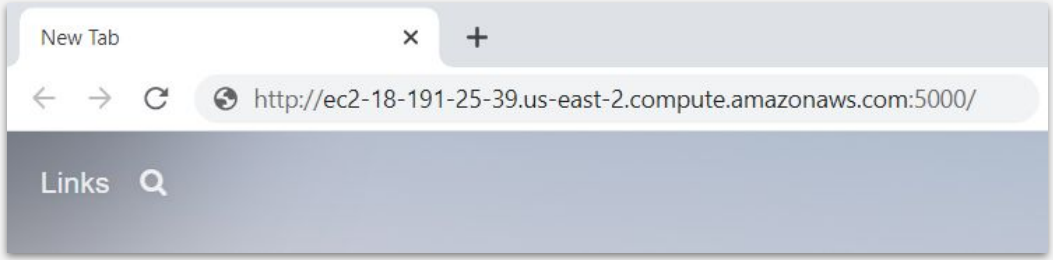
The server is up and running. Had it failed, it would have thrown up error.

Running on browser

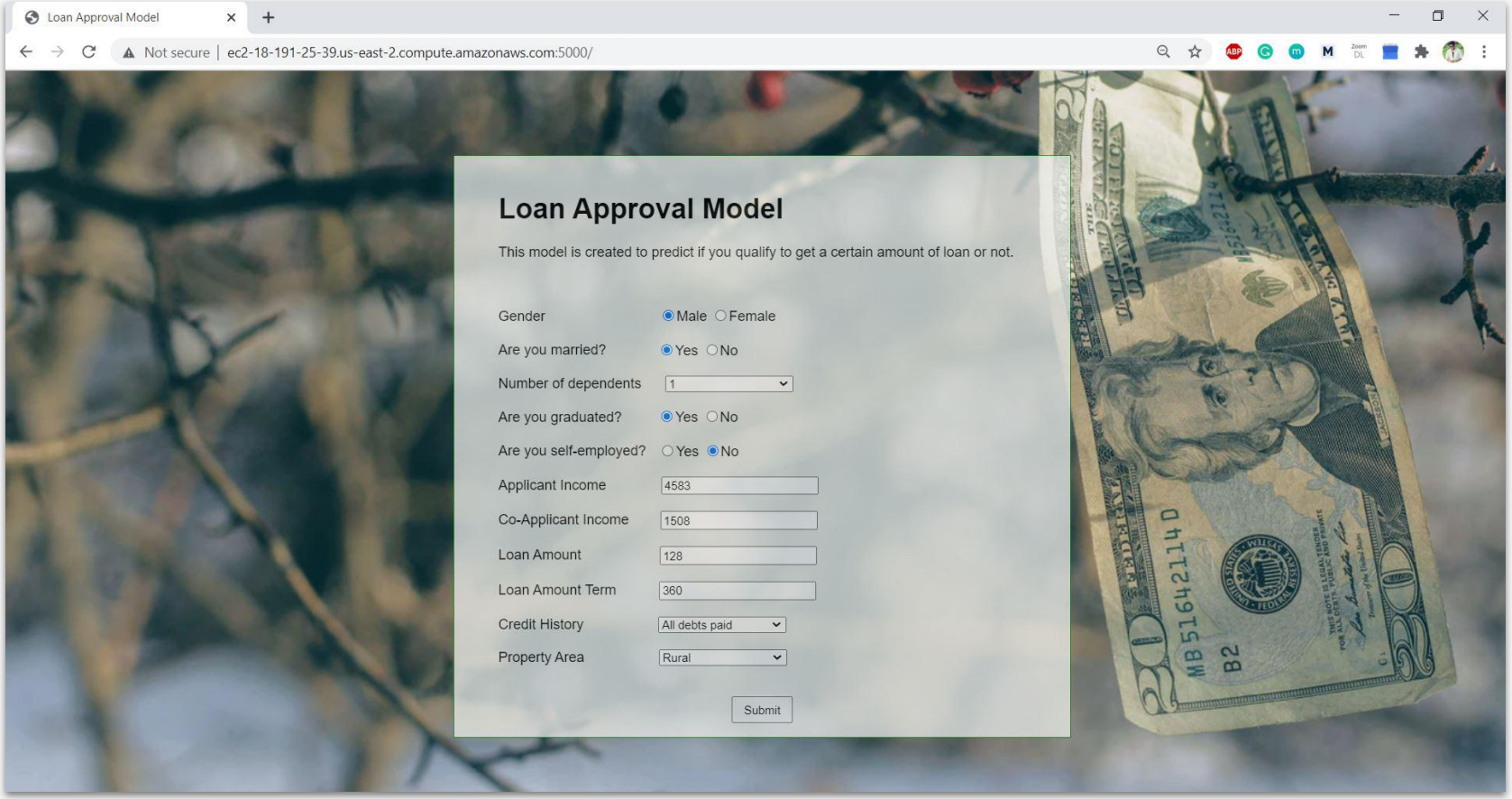
Go to the SSH client page by connecting to instance, and copy the link to run.

Go to browser and paste the link which we got after connecting to instance. Note that we need to append port at the end -> :5000/

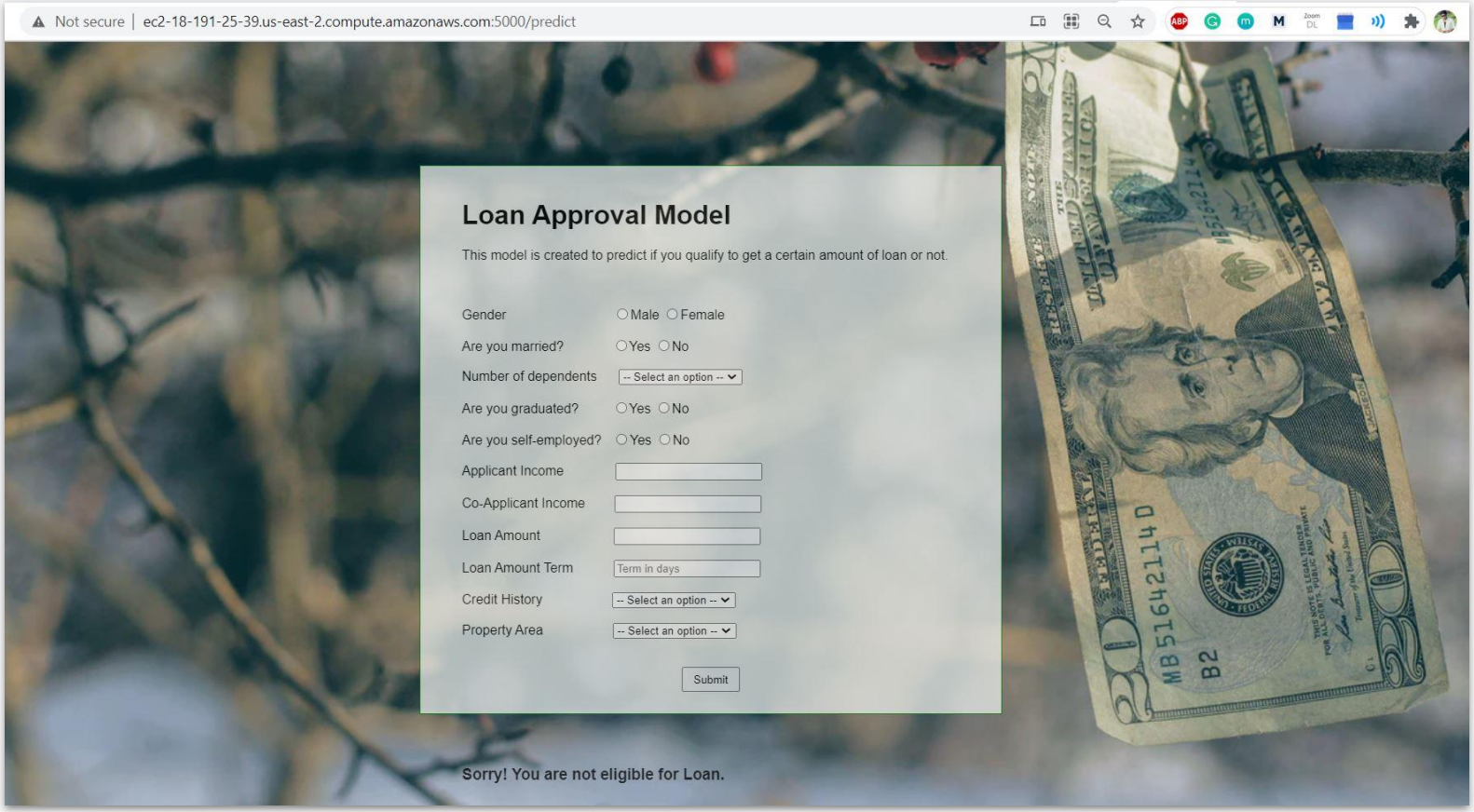
Paste and run on browser: http://ec2-18-191-25-39.us-east-2.compute.amazonaws.com:5000/



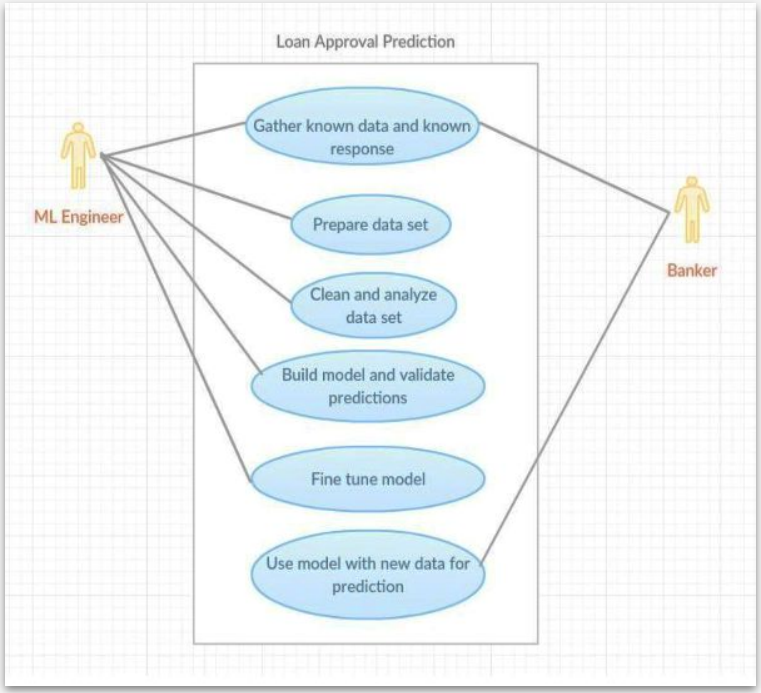
Web-App


Loan Approval Model
Not secure | ec2-18-191-25-39.us-east-2.compute.amazonaws.com:5000/
Loan Approval Model
This model is created to predict if you qualify to get a certain amount of loan or not.
Gender
Male
Female
Are you married?
Yes
No
Number of dependents
1
Are you graduated?
Yes
No
Are you self-employed?
Yes
No
Applicant Income
4583
Co-Applicant Income
1508
Loan Amount
128
Loan Amount Term
360
Credit History
All debts paid
Property Area
Rural
Submit

Prediction


Not secure | ec2-18-191-25-39.us-east-2.compute.amazonaws.com:5000/predict
Loan Approval Model
This model is created to predict if you qualify to get a certain amount of loan or not.
Gender
Male
Female
Are you married?
Yes
No
Number of dependents
-- Select an option --
Are you graduated?
Yes
No
Are you self-employed?
Yes
No
Applicant Income
Co-Applicant Income
Loan Amount
Loan Amount Term
Term in days
Credit History
-- Select an option --
Property Area
-- Select an option --
Submit
Sorry! You are not eligible for Loan.

Conclusion



- We have built a robust model which can predict whether to provide loan or not to an applicant.
- And then we have deployed the model using Amazon EC2.