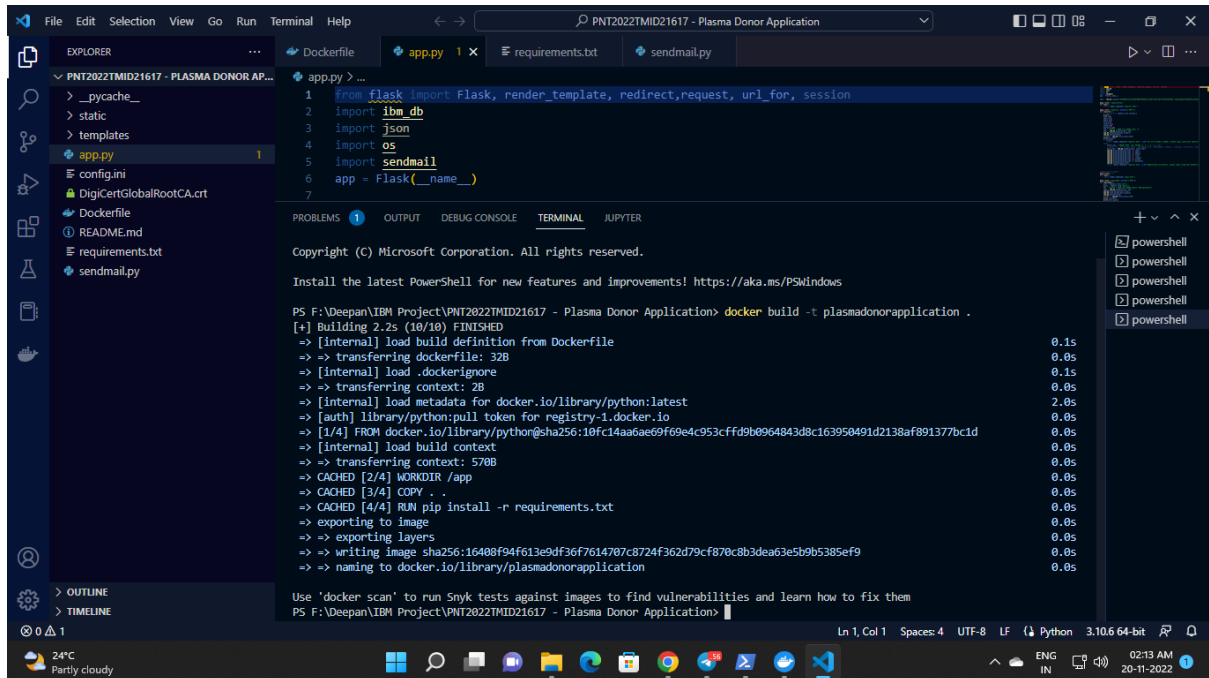


# CONTAINERIZE THE APP

Project	Plasma Donor Application
Team ID	PNT2022TMID21617

## 1, Building the image for the application

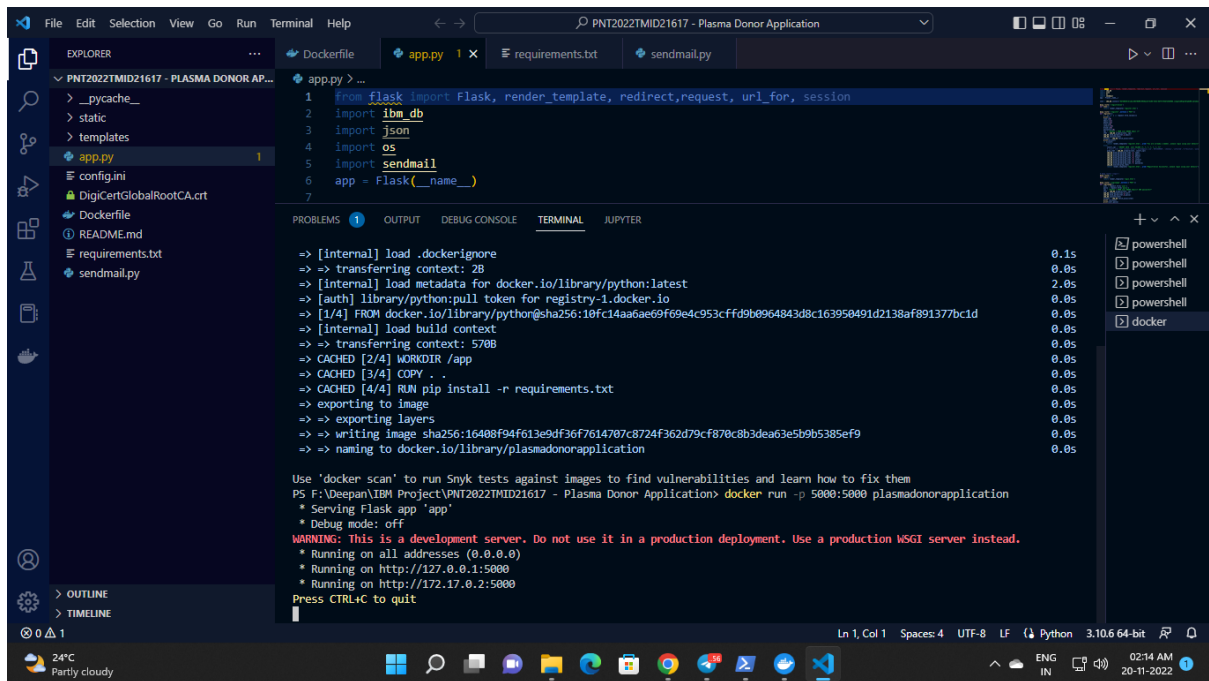


The screenshot shows the Visual Studio Code interface with the following components:

- EXPLORER:** Displays the file structure of the project 'PNT2022TMID21617 - PLASMA DONOR AP...'. Files include `_pycache_`, `static`, `templates`, `app.py` (selected), `config.ini`, `Dockerfile`, `README.md`, `requirements.txt`, and `sendmail.py`.
- EDITOR:** Shows the `app.py` file with the following code:

```
1 from flask import Flask, render_template, redirect, request, url_for, session
2 import ibm_db
3 import json
4 import os
5 import sendmail
6 app = Flask(__name__)
7
```
- TERMINAL:** Displays the output of the `docker build -t plasmadonorapplication .` command. The output shows the build process, including layer creation, context loading, and image export. The final image is named `docker.io/library/plasmadonorapplication`.
- STATUS BAR:** Shows the current file encoding as UTF-8, the language as Python, and the architecture as 64-bit.

## 2, Running the built image using the port 5000



The screenshot shows the Visual Studio Code interface with the Dockerfile editor open. The Dockerfile contains the following instructions:

```
1 FROM flask:latest
2 COPY . /app
3 RUN pip install -r requirements.txt
4 CMD ["python", "app.py"]
```

The terminal output shows the build process:

```
=> [internal] load .dockerignore
=> transferring context: 2B
=> [internal] load metadata for docker.io/library/python:latest
=> [auth] library/python:pull token for registry-1.docker.io
=> [1/4] FROM docker.io/library/python:latest
=> [internal] load build context
=> transferring context: 570B
=> CACHED [2/4] WORKDIR /app
=> CACHED [3/4] COPY . .
=> CACHED [4/4] RUN pip install -r requirements.txt
=> exporting to image
=> exporting layers
=> writing image sha256:16488f94f613e9df36f7614707c8724f362d79cf870c8b3dea63e5b9b5385ef9
=> naming to docker.io/library/plasmadonorapplication
```

Below the build output, there is a warning message:

```
Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
PS F:\DeePan\IBM Project\PNT2022TMID21617 - Plasma Donor Application> docker run -p 5000:5000 plasmadonorapplication
* Serving Flask app 'app'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://172.17.0.1:5000
* Running on http://172.17.0.2:5000
Press CTRL+C to quit
```

## 3, Viewing the container in the docker desktop

