MP6 Overview Session

CS 340 - Introduction to Computer Systems
Goals

In this MP:

- Create a web-based **microservice** using your **MP2 code**
- Build a **Flask** application that extracts hidden GIFs and maintains the state of all the hidden GIFs the application has extracted
- Package the application into a **Docker** container (extra credit)
Microservice Overview

- **Extract GIF:** If there is a POST request on the route `/extract`, return the extracted GIF from the PNG file, else return error code 500 with some message if not `uiuc` chunk is found.

  ![Image 1](image1.png) → ![Image 2](image2.png)

  ![Image 3](image3.png) → No Hidden GIF Found!, 500
Return Nth GIF: If there is a GET request on the route /extract/<image_num>, return the nth extracted GIF, else return error code 500 with some message if the nth GIF doesn’t exist.

GIF 100 Not Found!, 500
Flask Overview

- **Provided Code:** The provided code can be found in `app.py` and will get you two routes you need to define.

```python
from flask import Flask, render_template, send_file, request
import os

app = Flask(__name__)

# Route for "/" for a web-based interface to this micro-service:
@app.route('/

def index():
    return render_template("index.html")

# Extract a hidden "uiuc" GIF from a PNG image:
@app.route('/extract', methods=['POST'])

def extract_hidden_gif():
    # ...your code here...

# Get the nth saved "uiuc" GIF:
@app.route('/extract/<int:image_num>', methods=['GET'])

def extract_image(image_num):
    # ...your code here...
```

- Found the app using `python -m flask run`
  - Set the `FLASK_DEBUG` environment variable to 1 to run in debug mode.
- Visit [https://127.0.0.1:5000/](https://127.0.0.1:5000/) to the application.
Helpful Functions & Modules

Programming in Python

- **os.makedirs**: Creates a directory
- **request.files[‘png’]**: Access the PNG from the POST request
- **os.system**: Can be used to run the MP2 extractGIF function
- **send_file**: Send the contents of a file as a response
- **return “error message”, error code**: Can send an error message and error code if GIF is not found
MP6 Extra Credit

Packaging Application into Docker Container
Package as a Docker Container

- **Dockerfile**: contains the instructions for building the Docker image
- **.env**: contains the commands to build the Docker image
- **docker/entrypoint.sh**: contains the instructions that will be run when the container is initiated
Useful Dockerfile Instructions

- **FROM**: initialize the build stage/base image of the *Docker* image
- **RUN**: executes commands and commits the results to the *Docker* image
- **CMD**: command the *container* executes by default when launching the built image
- **COPY**: copies files from src to dest path in the *container*
- **ENV**: sets the environment variables
- **ENTRYPOINT**: used to set the *executables* that will always run when the container is initiated
- **WORKDIR**: sets the working directory for the *Docker* image
MP6 Testing
MP6 Testing

- Run the tests by using `pytest`
  - You can specify a file after pytest to only run the tests for the Flask app or for Docker
- Use `Docker container ls` to list all the Docker containers
- Use `Docker stop <container_name>` to stop the container