TMDB Poster Download Project

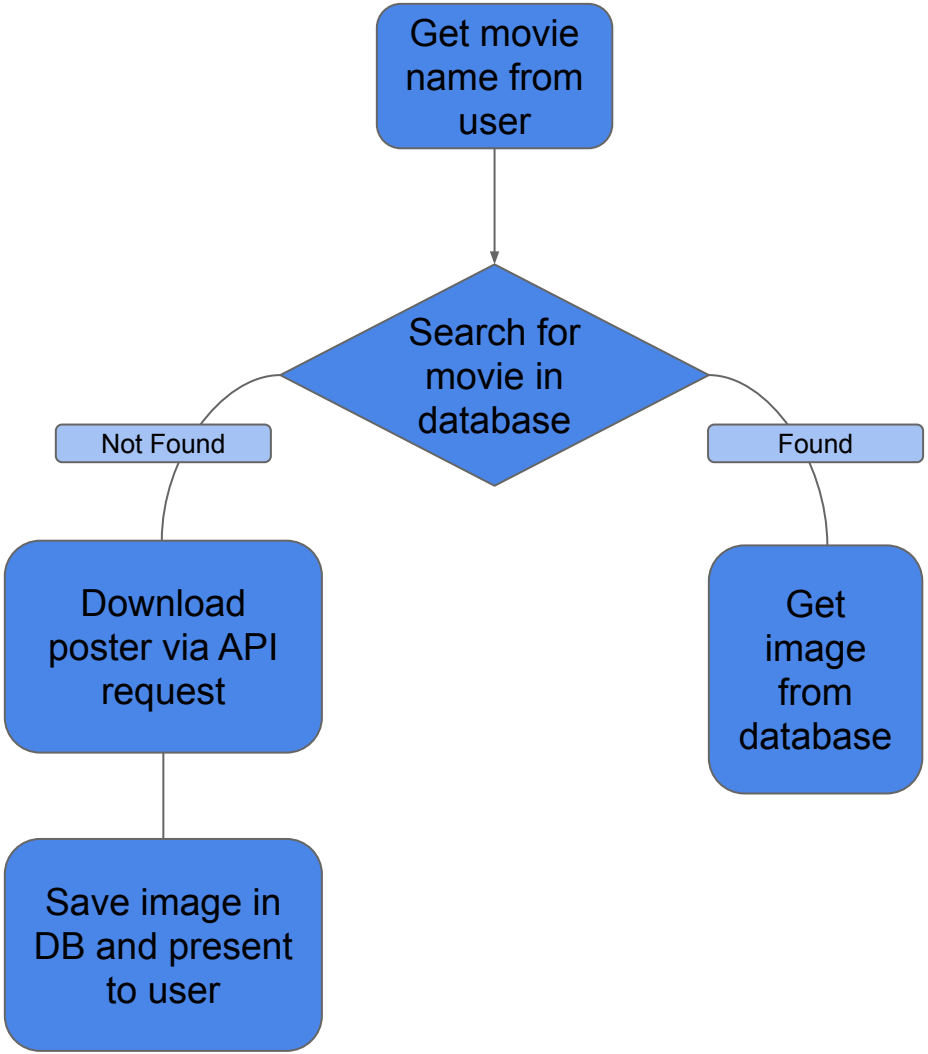
TEAM 7

Team Members

Project Objectives

- Get Movie Name
- Search Movie poster in TMDB Database
- Download poster and poster meta data and save it in MongoDB
- Construct a full Database API
- Write a simple web app to present the posters
- Dockerize the application
- Deploy application on AWS EC2 instance

Algorithm Design



Teamwork and responsibilities

Software Design

- OOP approach - Allows scalable code and services autonomy.
- **TMDB_Downloader.py** module - holds **TMDBDownloader** class and handles all the logic of downloading posters
- **MongoDBAPI.py** module - holds **MongoAPI** class and handles all the API management logic including all CRUD operations: reading records, creating records, updating records and deleting records.
- **mongo_tmdb_logic.py** module - holds the “Business logic”, integration between the two classes - if movie poster is not found in DB then download it from TMDB website and insert to DB.
- **api.py** module - hold web application logic using Flask, currently handles only presenting posters feature.

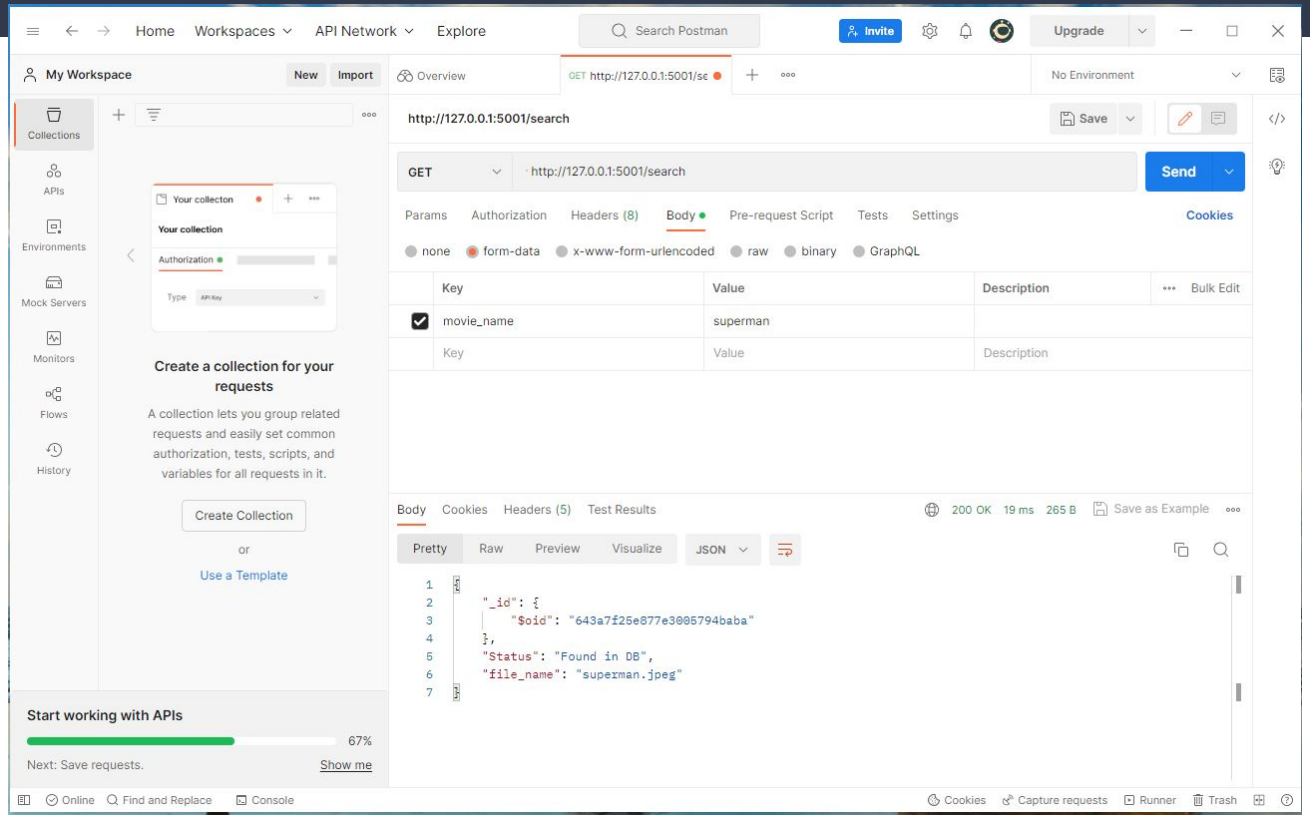
Dockerize the App and deploy on AWS

- Write Dockerfile
- Write docker-compose file
- Write user_data script to deploy the App on an EC2 instance

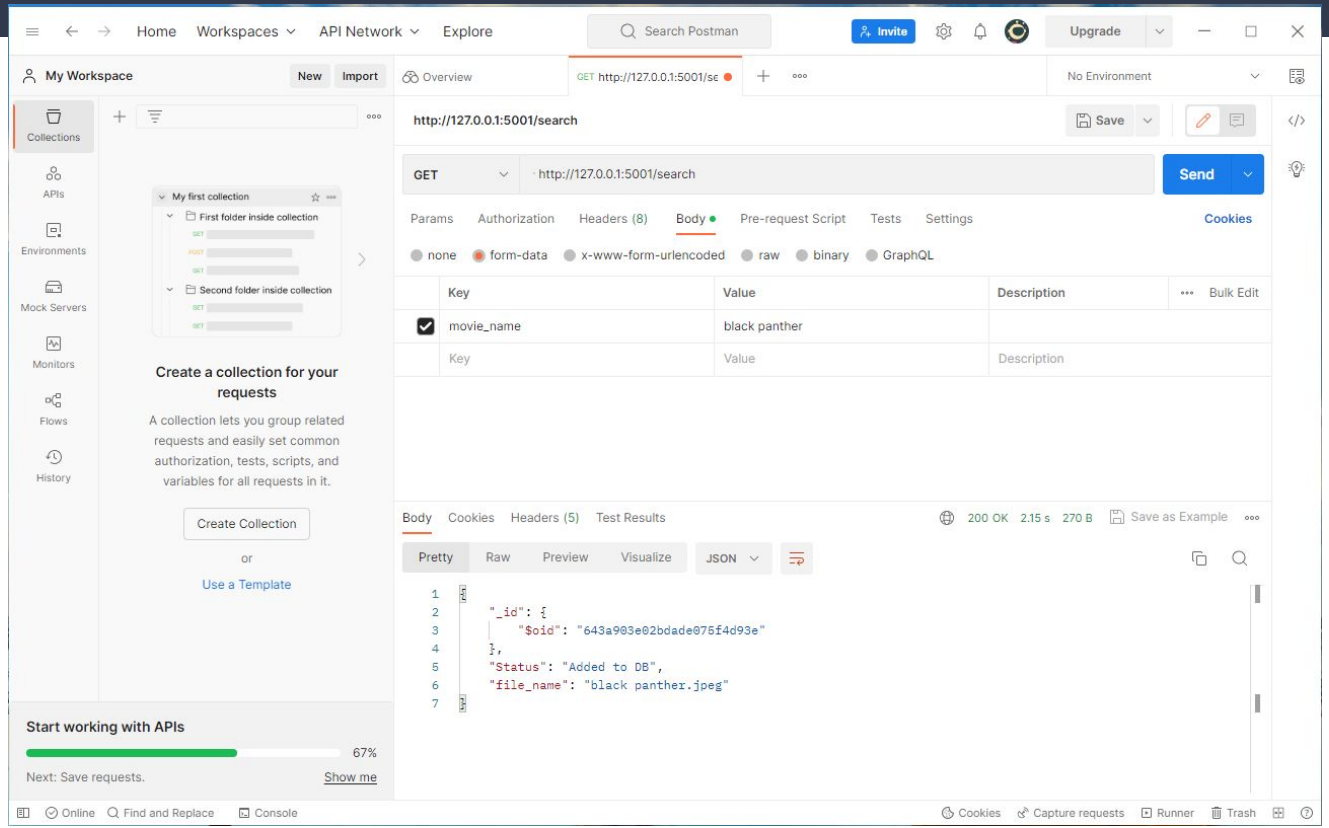
```
#! /bin/bash
sudo yum update -y
sudo amazon-linux-extras install docker -y
sudo systemctl start docker
sudo usermod -a -G docker ec2-user
sudo systemctl enable docker
sudo curl -SL https://github.com/docker/compose/releases/download/v2.4.1/docker-compose-linux-x86_64 -o /usr/local/bin/docker-compose
sudo chmod +x /usr/local/bin/docker-compose
sudo ln -s /usr/local/bin/docker-compose /usr/bin/docker-compose
sudo yum install git -y
sudo git clone https://github.com/AmirKatorza/AWS_reStart_Docker-Flask-Mongo_App.git
# wget https://github.com/AmirKatorza/AWS_reStart_Docker-Flask-Mongo_App/archive/refs/heads/master.zip
#sudo yum install unzip -y
#sudo unzip master.zip
sudo cd ./AWS_reStart_Docker-Flask-Mongo_App/
sudo docker-compose build
sudo docker-compose up
```

Reference:
https://gist.github.com/npearce/6f3c7826c7499587f00957fee62f8ee9

GET requests screenshots - Postman



GET requests screenshots - Postman



What can be improved?

- Improved UI (User Interface) Design - currently a simple HTML form with only search feature.
- Full API coverage by the web application including updating records and deleting records.
- Scale UP - currently downloading only the first movie from search results and only the first poster image.

  We want to save all movies from search results and all the posters related.

Challenges

- Learning new technologies in a very short amount of time.
- Learn how to integrate and orchestrate all the technologies to work together.
- Not enough ManPower to handle a project in that scale.

Thank You

For Listening!