Team (6) BlockBuster

Project Top Movie API

Project Team: Elizabeth Tyree, Josh Shilts, Ricky Zheng, Alex Flinchum, James Zheng, and Rekik Mengstu

Doc updates

Previous work (iteration 0):

- Determined and configured framework and dependent tools
- Created and documented high-level architecture design document
- Create UI mockup document
- Determined our datasource (Kaggle)
- Created commit documentation
- Set up branch protections
- Compiled assignment documentation

Doc Updates (current work)

Changes (iteration 1):

- Created UML database design diagram
- Decision (the team will use pycharm)
- Testing github actions, microservice flow
- We started discussion on javascript IDE
- TBD with respect to Security Requirements

Accomplishments

- Database design diagram
- Created the movies database based on the schema design
  - The code creates all tables
  - A test code was created to verify table creation
  - The code drops tables after test, in order to the test database again if needed
- Decided on each name on the data table to have an unique key

Accomplishment(s)

Created and Dockerized Frontend and Backend MVP

Backend Story
Frontend Story

Connected and tested two Microservices with Rest API

Backend Story
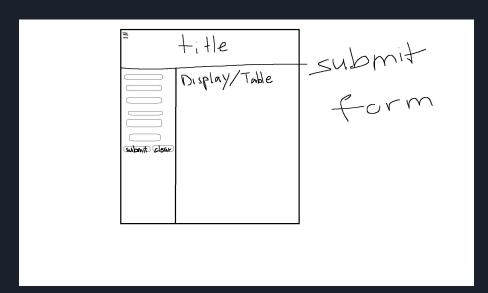
Created Pipeline with three job

Docker-and-Python-Flask-Service (Build and Runs Python backend)

Node-Service (Install)

Docker-and-Node Service (Build and Runs Python backend)

UI new functionality

- A home page
  - The button on the home page links to the search page
- Changed the format of the submit or search page
- Adding a hamburger button to the top left

Project Tools

Tools used for this iteration 1 for this project are:

- Docker
- Python
- SQLite
- HTML
- CSS
- Pycharm
- Git
- Nodejs
- Flask
- Podman
- VmWare Workstation 17

Work in Progress

- Frontend UI layout
- Github actions
  - Pipeline
- Scaling-up Frontend with NodeJS
- Loading data into database
- Containerizing SQLite database as a microservice
- Researching security tools

Next steps for iteration 2

- Incorporate database microservice into pipeline
- Create security process
- Continue Github Actions progress
- Finish UI layout and start incorporating CSS and Reactjs for organization