CS552: Major Project Presentation

“Comprehensive Movie Review Analysis System using Microservice-architecture”

(CS-552: Introduction to Cloud Computing)

Instructor: Prof. Hui Lu

Presentation by –

- Pavan Gangareddy
- Prashanth Devineni
- Nikita Mandlik
- Kanishk Barhanpurkar

List of Contents

- Motivation
- Goals
- Block Diagram
- System Architecture
- Major Challenges Faced
- Conclusions
- Future Scope
- References

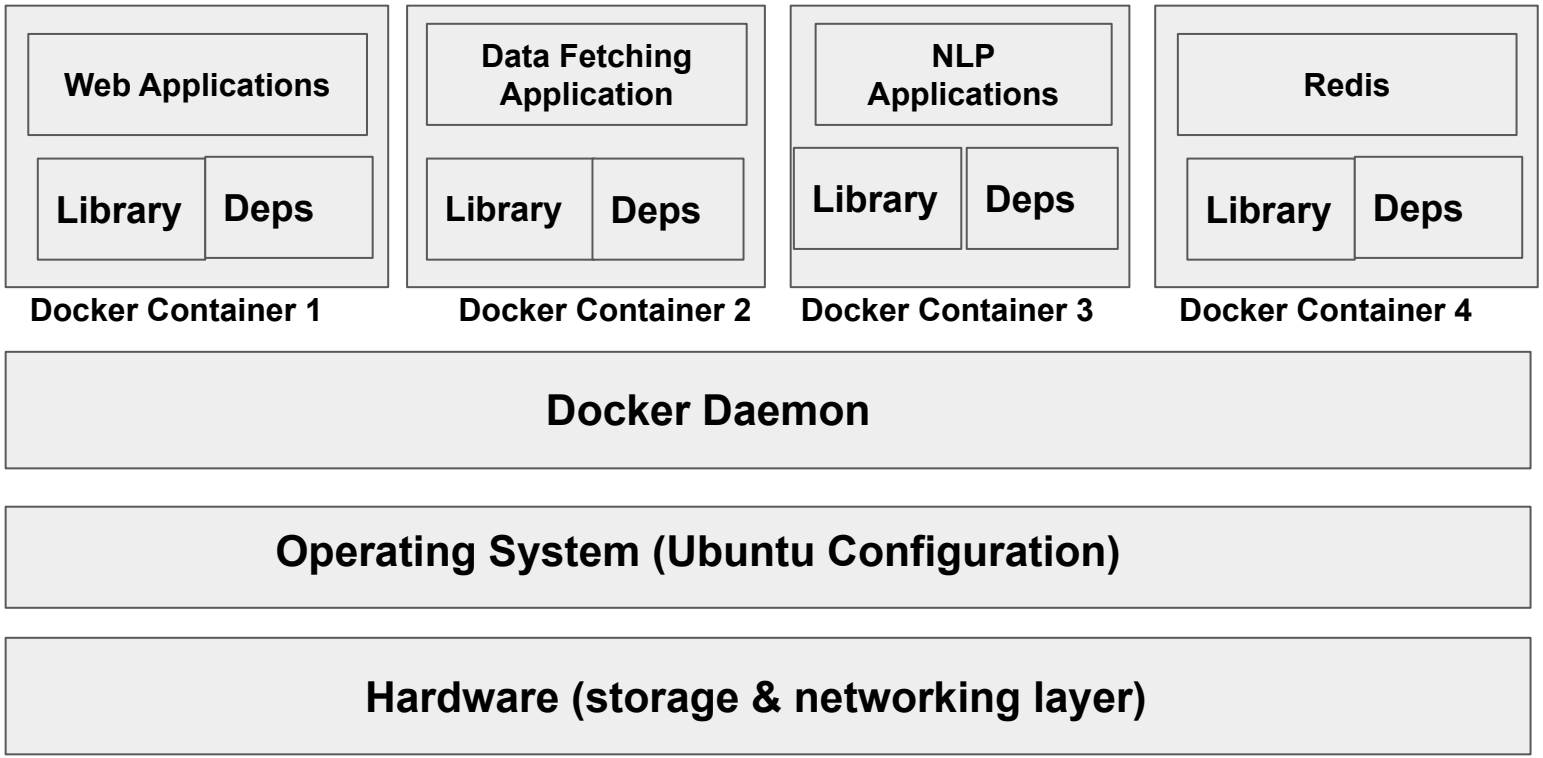
Motivation

- Understanding the working of text-analytical system deployed on cloud.
- Deployment of several layer microservices architecture for NLP-based application.

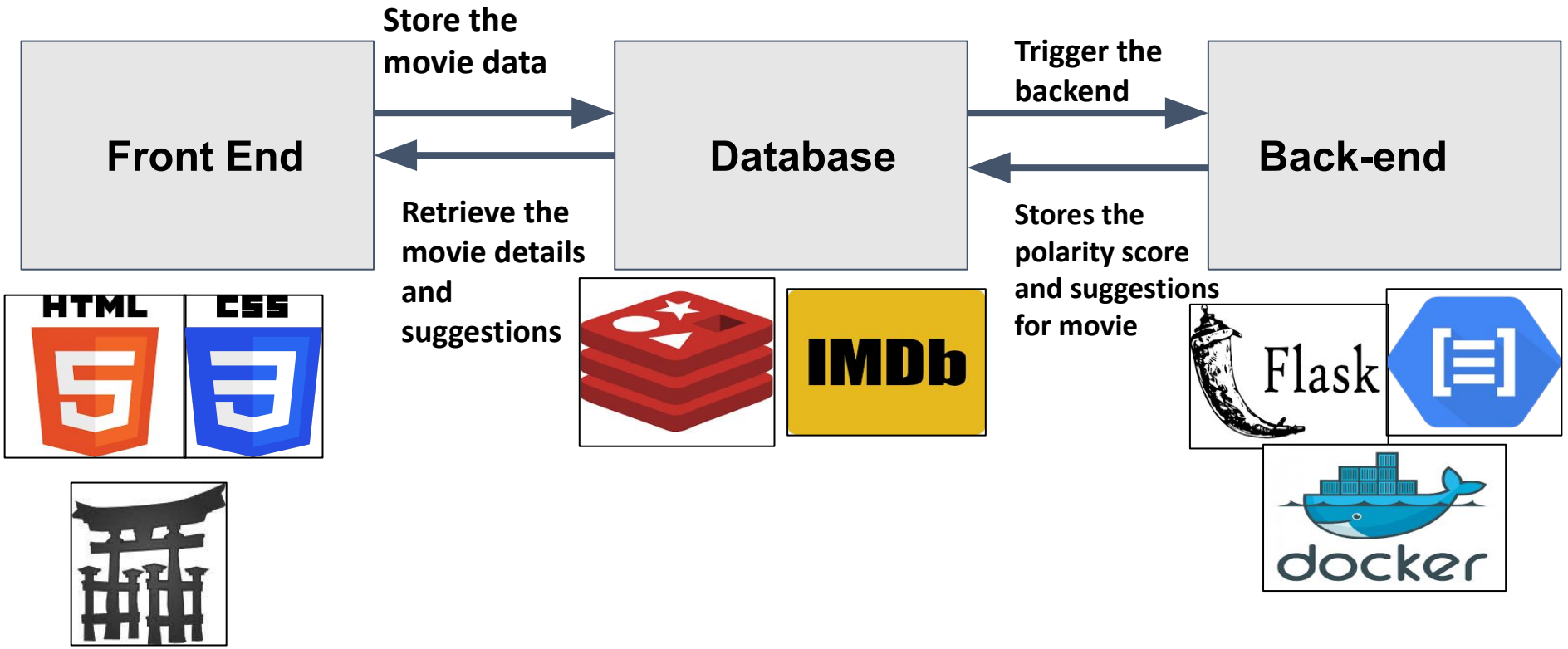
Goals

- Develop a movie reviews analysis system based on the services provided by Google cloud.
- To understand the working of microservices for process of data scraping, data storage and NLP-based applications.

Block Diagram



System Architecture



Implementation Steps

- **Step-1:** Set-up Redis in first docker container.
- **Step-2:** Set-up docker container for front-end.
- **Step-3:** Set-up container for Google Cloud NLP services.
- **Step-4:** Accept movie name and display movie details & verdict.

Setup 1: Set-up Redis in first container

- The data is in memory, which enables low latency.
- Flexible Data Structure used for data-storage.
- Entire movie-data is stored in this file.

Setup 2: Set-up docker container for front-end

- To enter the movie name.

Setup 3: Set-up container for Google NLP services

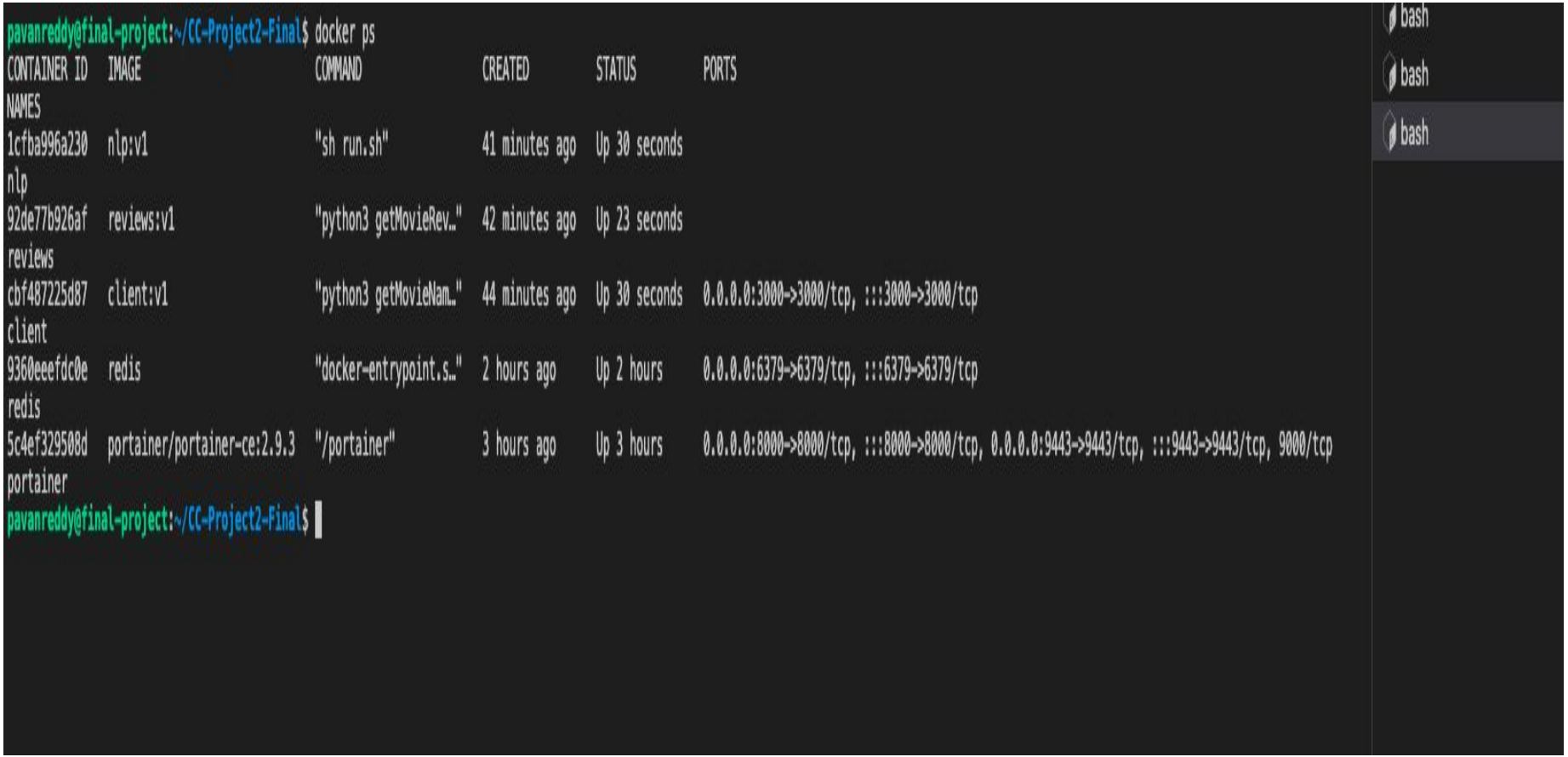
- **Containerize the Google Cloud NLP services.**
- **Polarity-** determines the sentimental aspect of an opinion.

  -> -1: extreme negative score

  ->+1: extreme positive score

  ->0: neutral sentiment

- **Magnitude-** how much it is differing from the absolute neutral (=0).



List of all docker container running on Google Cloud Platform.

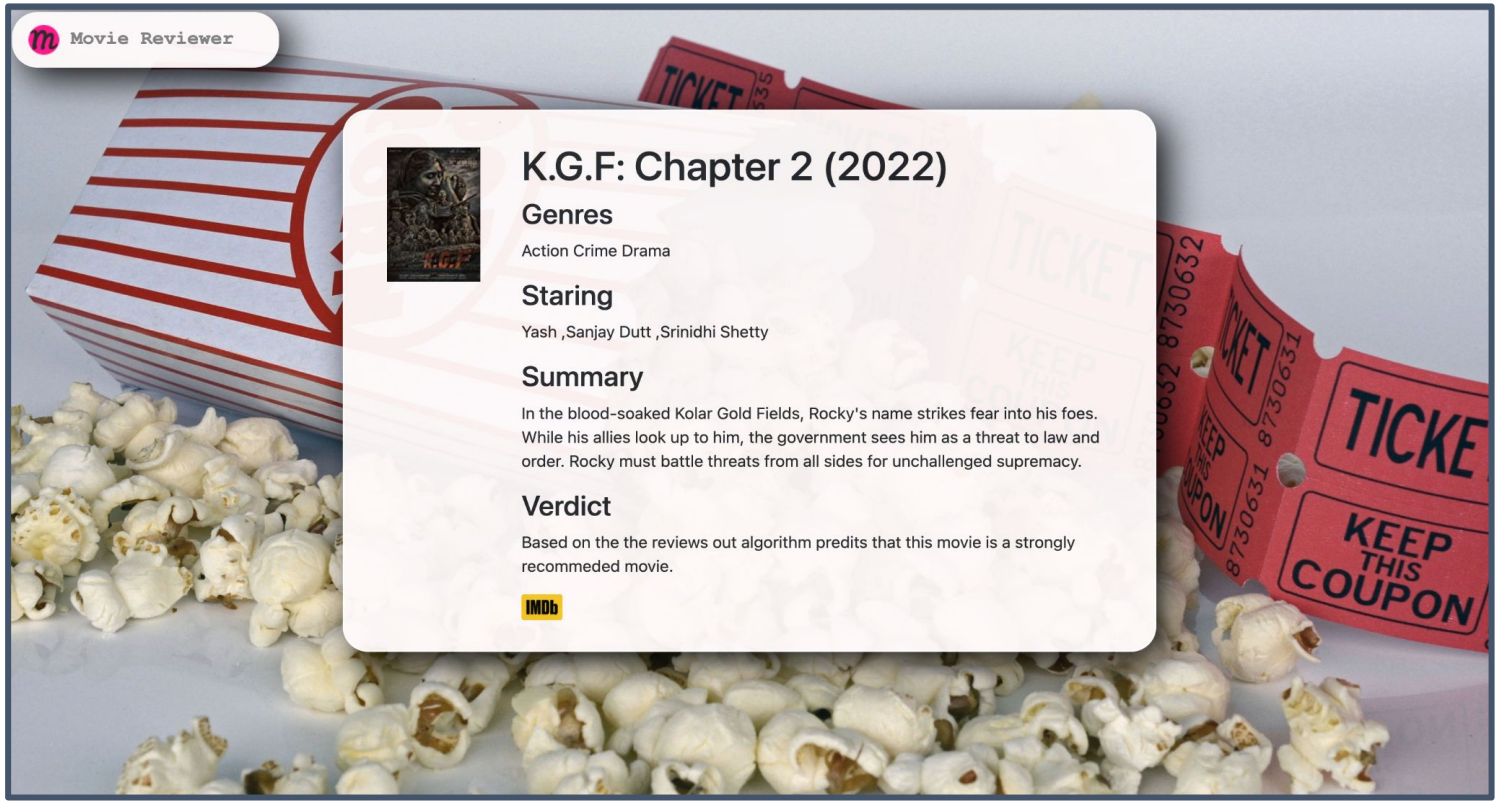
Scoring mechanism

- n:= number of reviews in one API calls.
- p:= polarity score for every review.
- M:=average score calculated for movie

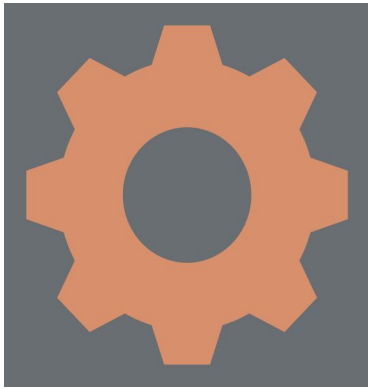
$$M=\Sigma p/n$$

On the basis of M, the movie verdict will be calculated.

Setup 4: Display movie details & display movie verdict.



Major Problem Faced



- **Synchronously working of Docker Containers.**

-Used Flags/Triggers for setting up other containers.

- **Integrating Google Cloud NLP Services and IMDB API.**

-Created separate containers for both.

Conclusion

- The current system is developed to suggest the movie recommendation based on the user comments on IMDB platform.
- We are using APIs to collect the movie reviews and movie trivia based on the IMDB public APIs.
- Used Redis an in-memory data structure store for storing comments and movie information.

Future Scope

- To create a recommendation system based on the users previous search history.
- To include and apply other NLP-based APIs for different purposes like segmentation speech recognition.
- To increase the number of reviews for movie review analysis system
- To add more statistical parameters to improve the rating mechanism.
- We can use orchestration as it creates replica for the containers and load balancing.

References

[1] Tamrakar, S., Madhavi, B. K., & Mohan, V. (2022). Democratizing Sentiment Analysis of Twitter Data Using Google Cloud Platform and BigQuery. Handbook of Intelligent Computing and Optimization for Sustainable Development, 287-304.

[2] Ghorbani, M., Bahaghighat, M., Xin, Q., & Özen, F. (2020). ConvLSTMConv network: a deep learning approach for sentiment analysis in cloud computing. Journal of Cloud Computing, 9(1), 1-12.

[3] Brar, G. S., & Sharma, A. (2018). Sentiment analysis of movie review using supervised machine learning techniques. International Journal of Applied Engineering Research, 13(16), 12788-12791.

[4] Redis Documentation. https://redis.io/docs/

[5] Docker Documentation. https://docs.docker.com/get-started/overview/

[6] Google Natural API Documentation. https://cloud.google.com/natural-language/docs/apis


Thank You!

Demo Session
Q & A Session