CS 588 Assignment 5, Part 1

Bhavya Chawla (A20516957) bchawla@hawk.iit.edu

(PART I: 150 points) Requirements Specification:

1. Use **Go** languages, **Docker**, **GCP** (Google Cloud Platform), **Postgres** to create microservices to collect data of the **Posts** (Question/body and Answer/body) from **StackOverflow** and **Issues** from **GitHub** and store the data collected in two databases: **StackoverflowDB** and **GitHubDB** using **Postgres** database server.

2. Collect the data from **StackOverflow** and **GitHub** for the following frameworks/libraries:
   a. Prometheus
      i. https://stackoverflow.com/search?q=Prometheus
      ii. https://github.com/prometheus/prometheus
   b. Selenium
   c. OpenAI
   d. Docker
   e. Milvus
   f. Go
      i. https://stackoverflow.com/search?q=golang
      ii. https://github.com/golang/go

3. You will need to create **developer accounts** and **API KEYs** on **GitHub** and **StackExchange**

   - https://api.github.com
   - https://docs.github.com/en
   - Here is the URL to guide you on how to generate your GITHUB_TOKEN
     o https://help.github.com/articles/creating-an-access-token-for-command-line-use/
   - Create your GitHub personal access tokens from
     o https://github.com/settings/tokens
   - https://api.stackexchange.com/
   - https://api.stackexchange.com/docs
   - https://stackapps.com/apps/oauth/register

4. Use Prometheus to show how many API calls to GitHub and StackOverflow were made per second, and how much data collected per second.

5. Run 3 experiments using data collected for past 2 days, 7 days, and 45 days.

Overview:

This project aims to collect data from StackOverflow and GitHub related to specific frameworks/libraries and store the information in two separate databases, StackoverflowDB and GitHubDB, using Postgres as the database server. The chosen frameworks/libraries are Prometheus, Docker, and Go.

Technologies Used:

- Programming Language: Go
- Containerization: Docker
- Cloud Platform: Google Cloud Platform (GCP)
- Database: Postgres
- Monitoring Tool: Prometheus

Steps:

1. Created Developer Accounts and API Keys

GitHub
- Generated a GitHub personal access token
- Created a personal access token on GitHub.

StackOverflow
- Obtained the necessary API key.

2. Running the Microservices

Step 1: Clone the Repository

bash
Copy code
git clone https://github.com/your-username/stackoverflow-github-microservices.git
cd src

Step 2: Build and Run Docker Containers

docker-compose up --build

3. Data Collection

The microservices will start collecting data from StackOverflow and GitHub for the specified frameworks/libraries. The data will be stored in the StackoverflowDB and GitHubDB databases on the Postgres server.

4. Prometheus Monitoring

Prometheus is configured to monitor API calls to GitHub and StackOverflow. Metrics are shown at the end of the report

5. Run Experiments

Three experiments were conducted using data collected over the past 2 days, 7 days, and 45 days.

Metrics:

```
# HELP github_api_calls_2_days_total Total number of API calls made to GitHub in the past 2 days
# TYPE github_api_calls_2_days_total counter
github_api_calls_2_days_total 762
# HELP github_api_calls_45_days_total Total number of API calls made to GitHub in the past 45 days
# TYPE github_api_calls_45_days_total counter
github_api_calls_45_days_total 546
# HELP github_api_calls_7_days_total Total number of API calls made to GitHub in the past 7 days
# TYPE github_api_calls_7_days_total counter
github_api_calls_7_days_total 623
# HELP github_api_calls_per_second Rate of API calls made to GitHub per second
# TYPE github_api_calls_per_second summary
github_api_calls_per_second_sum{endpoint="github_endpoint"} 7.856890268377948
github_api_calls_per_second_count{endpoint="github_endpoint"} 18
# HELP go_gc_duration_seconds A summary of the pause duration of garbage collection cycles.
# TYPE go_gc_duration_seconds summary
go_gc_duration_seconds{quantile="0"} 6.3042e-05
go_gc_duration_seconds{quantile="0.25"} 0.0001115
go_gc_duration_seconds{quantile="0.5"} 0.000134416
go_gc_duration_seconds{quantile="0.75"} 0.000229501
go_gc_duration_seconds{quantile="1"} 0.000559417
go_gc_duration_seconds_sum 0.008311496
go_gc_duration_seconds_count 44
```

```
# HELP promhttp_metric_handler_requests_in_flight current number of scrapes being served.
# TYPE promhttp_metric_handler_requests_in_flight gauge
promhttp_metric_handler_requests_in_flight 1
# HELP promhttp_metric_handler_requests_total Total number of scrapes by HTTP status code.
# TYPE promhttp_metric_handler_requests_total counter
promhttp_metric_handler_requests_total{code="200"} 9
promhttp_metric_handler_requests_total{code="500"} 0
promhttp_metric_handler_requests_total{code="503"} 0
# HELP stackoverflow_api_calls_2_days_total Total number of API calls made to StackOverflow in the past 2 days
# TYPE stackoverflow_api_calls_2_days_total counter
stackoverflow_api_calls_2_days_total 0
# HELP stackoverflow_api_calls_45_days_total Total number of API calls made to StackOverflow in the past 45 days
# TYPE stackoverflow_api_calls_45_days_total counter
stackoverflow_api_calls_45_days_total 0
# HELP stackoverflow_api_calls_7_days_total Total number of API calls made to StackOverflow in the past 7 days
# TYPE stackoverflow_api_calls_7_days_total counter
stackoverflow_api_calls_7_days_total 0
# HELP stackoverflow_api_calls_per_second Rate of API calls made to StackOverflow per second
# TYPE stackoverflow_api_calls_per_second summary
stackoverflow_api_calls_per_second_sum{endpoint="stackoverflow_endpoint"} 343.2951616792917
stackoverflow_api_calls_per_second_count{endpoint="stackoverflow_endpoint"} 18
# HELP total_github_api_calls Total number of API calls made to GitHub
# TYPE total_github_api_calls counter
total_github_api_calls 18
# HELP total_stackoverflow_api_calls Total number of API calls made to StackOverflow
# TYPE total_stackoverflow_api_calls counter
total_stackoverflow_api_calls 18
```

```
# HELP github_api_calls_2_days_total Total number of API calls made to GitHub in the past 2 days
# TYPE github_api_calls_2_days_total counter
github_api_calls_2_days_total 762
# HELP github_api_calls_45_days_total Total number of API calls made to GitHub in the past 45 days
# TYPE github_api_calls_45_days_total counter
github_api_calls_45_days_total 546
# HELP github_api_calls_7_days_total Total number of API calls made to GitHub in the past 7 days
# TYPE github_api_calls_7_days_total counter
github_api_calls_7_days_total 623
# HELP github_api_calls_per_second Rate of API calls made to GitHub per second
# TYPE github_api_calls_per_second summary
github_api_calls_per_second_sum{endpoint="github_endpoint"} 7.856890268377948
github_api_calls_per_second_count{endpoint="github_endpoint"} 18
# HELP go_gc_duration_seconds A summary of the pause duration of garbage collection cycles.
# TYPE go_gc_duration_seconds summary
go_gc_duration_seconds{quantile="0"} 6.3042e-05
go_gc_duration_seconds{quantile="0.25"} 0.0001115
go_gc_duration_seconds{quantile="0.5"} 0.000134416
go_gc_duration_seconds{quantile="0.75"} 0.000229501
go_gc_duration_seconds{quantile="1"} 0.000559417
go_gc_duration_seconds_sum 0.008311496
go_gc_duration_seconds_count 44
# HELP go_goroutines Number of goroutines that currently exist.
# TYPE go_goroutines gauge
go_goroutines 12
# HELP go_info Information about the Go environment.
```

```
# TYPE go_info gauge
go_info{version="go1.21.4"} 1
# HELP go_memstats_alloc_bytes Number of bytes allocated and still in use.
# TYPE go_memstats_alloc_bytes gauge
go_memstats_alloc_bytes 1.9816e+06
# HELP go_memstats_alloc_bytes_total Total number of bytes allocated, even if freed.
# TYPE go_memstats_alloc_bytes_total counter
go_memstats_alloc_bytes_total 7.7657376e+07
# HELP go_memstats_buck_hash_sys_bytes Number of bytes used by the profiling bucket hash table.
# TYPE go_memstats_buck_hash_sys_bytes gauge
go_memstats_buck_hash_sys_bytes 5191
# HELP go_memstats_frees_total Total number of frees.
# TYPE go_memstats_frees_total counter
go_memstats_frees_total 301514
# HELP go_memstats_gc_sys_bytes Number of bytes used for garbage collection system metadata.
# TYPE go_memstats_gc_sys_bytes gauge
go_memstats_gc_sys_bytes 4.47256e+06
# HELP go_memstats_heap_alloc_bytes Number of heap bytes allocated and still in use.
# TYPE go_memstats_heap_alloc_bytes gauge
go_memstats_heap_alloc_bytes 1.9816e+06
# HELP go_memstats_heap_idle_bytes Number of heap bytes waiting to be used.
# TYPE go_memstats_heap_idle_bytes gauge
go_memstats_heap_idle_bytes 3.743744e+06
# HELP go_memstats_heap_inuse_bytes Number of heap bytes that are in use.
# TYPE go_memstats_heap_inuse_bytes gauge
go_memstats_heap_inuse_bytes 4.05504e+06
# HELP go_memstats_heap_objects Number of allocated objects.
# TYPE go_memstats_heap_objects gauge
go_memstats_heap_objects 5545
# HELP go_memstats_heap_released_bytes Number of heap bytes released to OS.
# TYPE go_memstats_heap_released_bytes gauge
go_memstats_heap_released_bytes 1.490944e+06
# HELP go_memstats_heap_sys_bytes Number of heap bytes obtained from system.
# TYPE go_memstats_heap_sys_bytes gauge
go_memstats_heap_sys_bytes 7.798784e+06
# HELP go_memstats_last_gc_time_seconds Number of seconds since 1970 of last garbage collection.
# TYPE go_memstats_last_gc_time_seconds gauge
go_memstats_last_gc_time_seconds 1.7012269834825652e+09
# HELP go_memstats_lookups_total Total number of pointer lookups.
# TYPE go_memstats_lookups_total counter
go_memstats_lookups_total 0
# HELP go_memstats_mallocs_total Total number of mallocs.
# TYPE go_memstats_mallocs_total counter
go_memstats_mallocs_total 307059
```

HELP go_memstats_mcache_inuse_bytes Number of bytes in use by mcache structures.
TYPE go_memstats_mcache_inuse_bytes gauge
go_memstats_mcache_inuse_bytes 9600
HELP go_memstats_mcache_sys_bytes Number of bytes used for mcache structures obtained from system.
TYPE go_memstats_mcache_sys_bytes gauge
go_memstats_mcache_sys_bytes 15600
HELP go_memstats_mspan_inuse_bytes Number of bytes in use by mspan structures.
TYPE go_memstats_mspan_inuse_bytes gauge
go_memstats_mspan_inuse_bytes 161952
HELP go_memstats_mspan_sys_bytes Number of bytes used for mspan structures obtained from system.
TYPE go_memstats_mspan_sys_bytes gauge
go_memstats_mspan_sys_bytes 179256
HELP go_memstats_next_gc_bytes Number of heap bytes when next garbage collection will take place.
TYPE go_memstats_next_gc_bytes gauge
go_memstats_next_gc_bytes 4.194304e+06
HELP go_memstats_other_sys_bytes Number of bytes used for other system allocations.
TYPE go_memstats_other_sys_bytes gauge
go_memstats_other_sys_bytes 1.654705e+06
HELP go_memstats_stack_inuse_bytes Number of bytes in use by the stack allocator.
TYPE go_memstats_stack_inuse_bytes gauge
go_memstats_stack_inuse_bytes 589824
HELP go_memstats_stack_sys_bytes Number of bytes obtained from system for stack allocator.
TYPE go_memstats_stack_sys_bytes gauge
go_memstats_stack_sys_bytes 589824
HELP go_memstats_sys_bytes Number of bytes obtained from system.
TYPE go_memstats_sys_bytes gauge
go_memstats_sys_bytes 1.471592e+07
HELP go_threads Number of OS threads created.
TYPE go_threads gauge
go_threads 12
HELP promhttp_metric_handler_requests_in_flight Current number of scrapes being served.
TYPE promhttp_metric_handler_requests_in_flight gauge
promhttp_metric_handler_requests_in_flight 1
HELP promhttp_metric_handler_requests_total Total number of scrapes by HTTP status code.
TYPE promhttp_metric_handler_requests_total counter
promhttp_metric_handler_requests_total{code="200"} 9
promhttp_metric_handler_requests_total{code="500"} 0
promhttp_metric_handler_requests_total{code="503"} 0
HELP stackoverflow_api_calls_2_days_total Total number of API calls made to StackOverflow in the past 2 days
TYPE stackoverflow_api_calls_2_days_total counter
stackoverflow_api_calls_2_days_total 0

HELP stackoverflow_api_calls_45_days_total Total number of API calls made to StackOverflow in the past 45 days
TYPE stackoverflow_api_calls_45_days_total counter
stackoverflow_api_calls_45_days_total 0
HELP stackoverflow_api_calls_7_days_total Total number of API calls made to StackOverflow in the past 7 days
TYPE stackoverflow_api_calls_7_days_total counter
stackoverflow_api_calls_7_days_total 0
HELP stackoverflow_api_calls_per_second Rate of API calls made to StackOverflow per second
TYPE stackoverflow_api_calls_per_second summary
stackoverflow_api_calls_per_second_sum{endpoint="stackoverflow_endpoint"} 343.2951616792917
stackoverflow_api_calls_per_second_count{endpoint="stackoverflow_endpoint"} 18
HELP total_github_api_calls Total number of API calls made to GitHub
TYPE total_github_api_calls counter
total_github_api_calls 18
HELP total_stackoverflow_api_calls Total number of API calls made to StackOverflow
TYPE total_stackoverflow_api_calls counter
total_stackoverflow_api_calls 18