Project Report COMP.SE.140

Henri Rantala H293281

Instructions

To start the application, run following commands

```
$ git clone -b project https://github.com/tuniHR/DevOpsExercises.git
$ cd <created folder>
$ docker-compose build
$ docker-compose up -d
```

To start up the application locally, it is necessary that you have installed docker and docker-compose on your local machine. As per requirements, browser version can be accessed from port 8198 and API form port 8197. Note that any PUT calls through API gateway require basic auth, and as such should include -u admin:admin to authenticate the curl request.

To test the system it is possible to do with curl requests, or use the provided test script, by navigating to cloned directory and run: bash test/test_api.sh

Optional features

None.

Development Platform

Versions:
Docker version 27.3.1
Docker Compose version 2.31.0
Operating System: Linux (EndeavourOS), kernel: 6.12.4-arch1-1
CPU arhictecture: x86-64 bit

CI/CD pipeline

For version management, git was used, mainly operating only on single branch, project as there was no other contributors and work was relatively simple in scope. Thus it was more simple and easy to operate on single branch.

Docker and docker containers were used to build and deploy the system, as containers do provide easy solution for building and running systems across platforms.

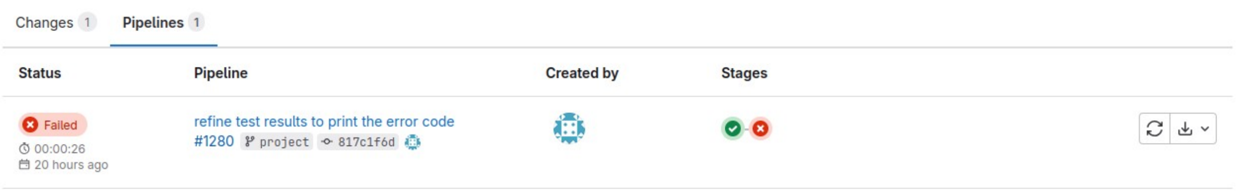
For testing, simple bash scripts were used, as the test themselves were very simple API calls and checking the answers. While first test were written with python and pytest, these were translated to bash scripts to reduce number of packages and because problems with installing these packages on my local system.

Test cases:

1. GET /state without Authentication
   Purpose: Check if /state is accessible without authentication.
   Validation: Status code 200 and response in [INIT, PAUSED, RUNNING, SHUTDOWN].
2. PUT /state without Authentication
   Purpose: Verify that state modification is restricted without authentication.
   Validation: Status code 401.
3. PUT /state with Authentication
   Purpose: Ensure authenticated users can change the state.
   Validation: Status code 200 and response "State changed to RUNNING".
4. PUT /state to PAUSED
   Purpose: Confirm state PAUSED disables /request and allow reversion to RUNNING.
   Validation: Status code 200 for state changes, /request inaccessible when PAUSED.
5. GET /request
   Purpose: Verify functionality of /request.
   Validation: Status code 200.
6. GET /run-log
   Purpose: Check /run-log reflects the current state.
   Validation: Status code 200 and log includes "RUNNING".

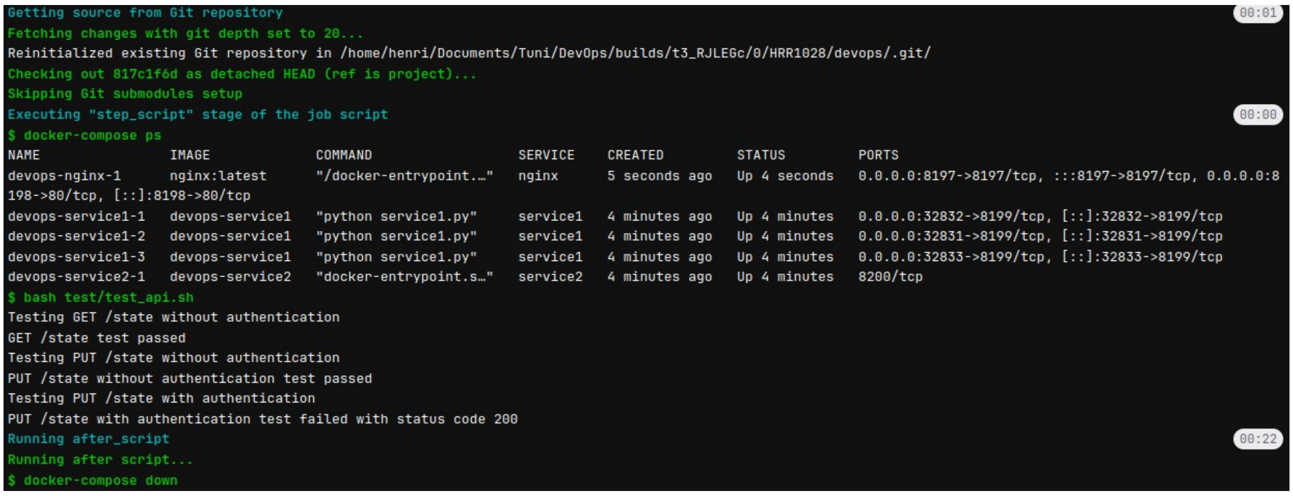
Examples of pipeline

Example from a commit

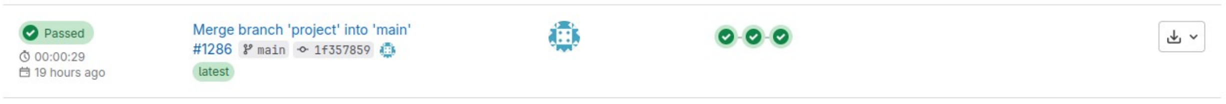

Changes 1 Pipelines 1

Status	Pipeline	Created by	Stages
Failed 00:00:26 20 hours ago	refine test results to print the error code #1280 project 817c1f6d		✓ ✗

Failure in tests

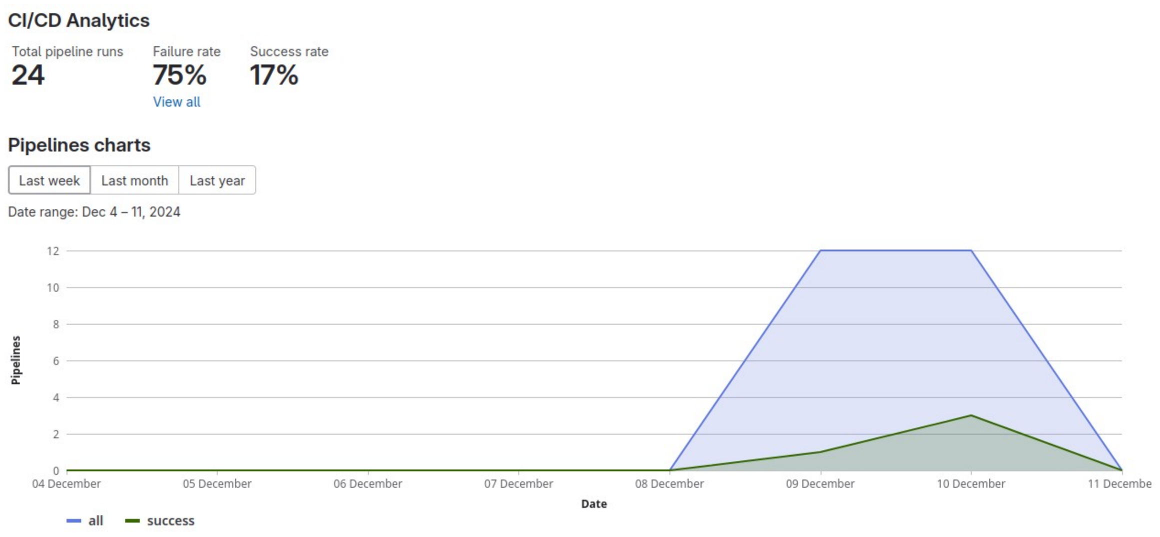

```
Getting source from Git repository                                                                      00:01
Fetching changes with git depth set to 20...
Reinitialized existing Git repository in /home/henri/Documents/Tuni/DevOps/builds/t3_RJLEGc/0/HRR1028/devops/.git/
Checking out 817c1f6d as detached HEAD (ref is project)...
Skipping Git submodules setup
Executing "step_script" stage of the job script                                                         00:00
$ docker-compose ps
NAME                IMAGE               COMMAND                  SERVICE     CREATED          STATUS          PORTS
devops-nginx-1      nginx:latest        "/docker-entrypoint.…"   nginx       5 seconds ago    Up 4 seconds    0.0.0.0:8197->8197/tcp, :::8197->8197/tcp, 0.0.0.0:8
198->80/tcp, [::]:8198->80/tcp
devops-service1-1   devops-service1     "python service1.py"     service1    4 minutes ago    Up 4 minutes    0.0.0.0:32832->8199/tcp, [::]:32832->8199/tcp
devops-service1-2   devops-service1     "python service1.py"     service1    4 minutes ago    Up 4 minutes    0.0.0.0:32831->8199/tcp, [::]:32831->8199/tcp
devops-service1-3   devops-service1     "python service1.py"     service1    4 minutes ago    Up 4 minutes    0.0.0.0:32833->8199/tcp, [::]:32833->8199/tcp
devops-service2-1   devops-service2     "docker-entrypoint.s…"   service2    4 minutes ago    Up 4 minutes    8200/tcp
$ bash test/test_api.sh
Testing GET /state without authentication
GET /state test passed
Testing PUT /state without authentication
PUT /state without authentication test passed
Testing PUT /state with authentication
PUT /state with authentication test failed with status code 200
Running after_script                                                                                    00:22
Running after script...
$ docker-compose down
```

Succesful commit



Passed 00:00:29 19 hours ago	Merge branch 'project' into 'main' #1286 main 1f357859 latest		✓ ✓ ✓

Some analytics

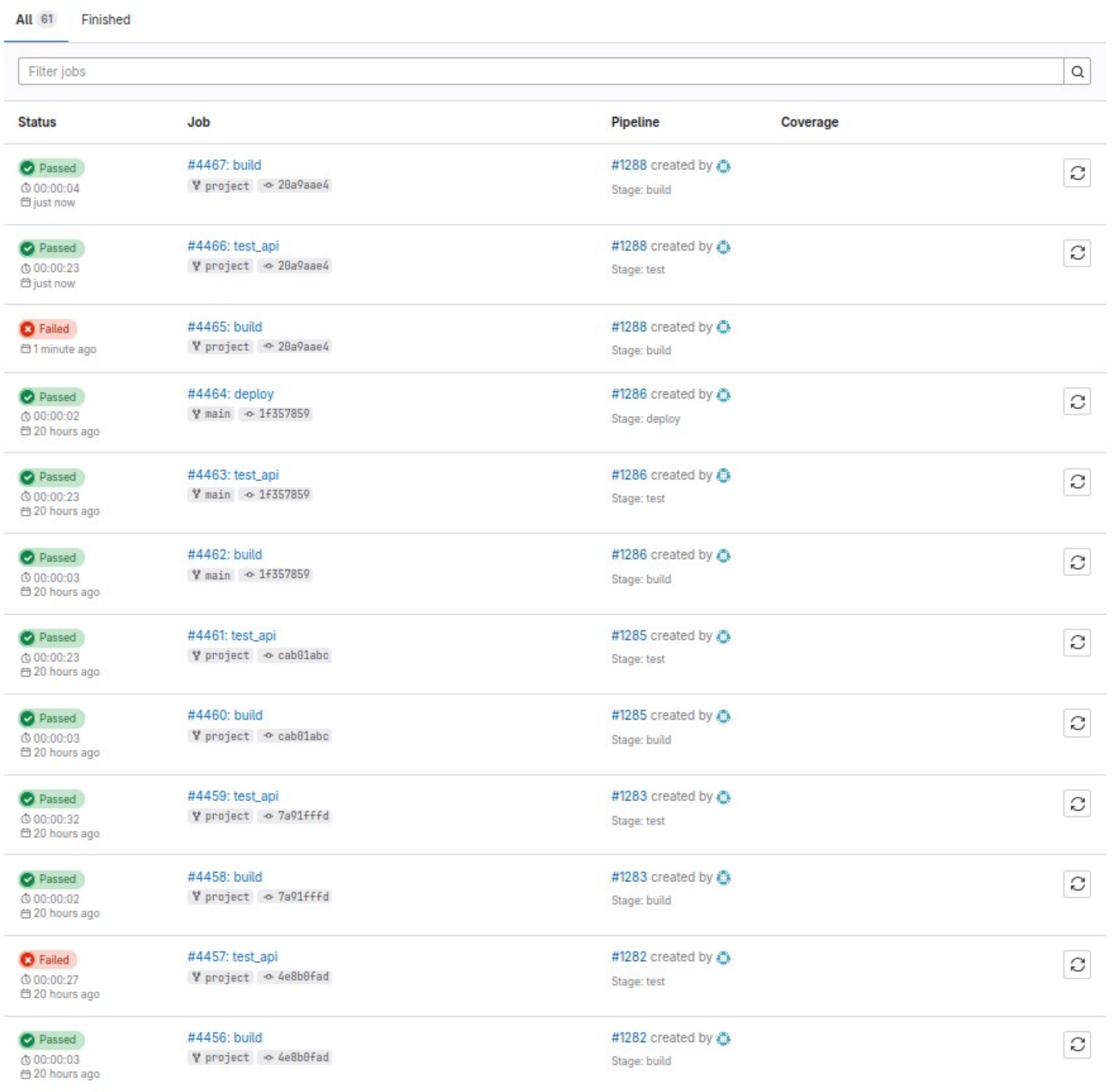

CI/CD Analytics

Total pipeline runs 24

Failure rate 75%

Success rate 17%

Some recent history



All 61 Finished

Filter jobs

Status	Job	Pipeline	Coverage
Passed 00:00:04 just now	#4467: build project 20a9aae4	#1288 created by Stage: build	
Passed 00:00:23 just now	#4466: test_api project 20a9aae4	#1288 created by Stage: test	
Failed 1 minute ago	#4465: build project 20a9aae4	#1288 created by Stage: build	
Passed 00:00:02 20 hours ago	#4464: deploy main 1f357859	#1286 created by Stage: deploy	
Passed 00:00:23 20 hours ago	#4463: test_api main 1f357859	#1286 created by Stage: test	
Passed 00:00:03 20 hours ago	#4462: build main 1f357859	#1286 created by Stage: build	
Passed 00:00:23 20 hours ago	#4461: test_api project cab01abc	#1285 created by Stage: test	
Passed 00:00:03 20 hours ago	#4460: build project cab01abc	#1285 created by Stage: build	
Passed 00:00:32 20 hours ago	#4459: test_api project 7a91fffd	#1283 created by Stage: test	
Passed 00:00:02 20 hours ago	#4458: build project 7a91fffd	#1283 created by Stage: build	
Failed 00:00:27 20 hours ago	#4457: test_api project 4e8b0fad	#1282 created by Stage: test	
Passed 00:00:03 20 hours ago	#4456: build project 4e8b0fad	#1282 created by Stage: build	

Reflections

The most difficult thing during this project was setting up the actual pipeline, form setting up the runner and getting first simple build pipeline to work. This was also the greatest learning that the project provided, to learn to setup and define a pipeline. Also test driven development while not new concept for me, was something that was new in practise for me and this also gave some experience and perspective.

Hour estimate: 30h

AI/LLM usage

This project was done with assistance from ChatGPT. ChatGPT was used to help construct individual functions, small scripts, components and troubleshooting. However, some or even most results given by ChatGPT were modified or at least refined with iterative exhances with ChatGPT as initial results were often not satisfactory. While ChatGPT was not always correct, it did significantly improve and speed up development.