**Framework and Technology used for DISQO Task**

Contents

[Functional testing framework 1](#_Toc75630490)

[About 1](#_Toc75630491)

[Code architecture 1](#_Toc75630492)

[DB log sample: 2](#_Toc75630493)

[Allure report sample 3](#_Toc75630494)

[Load testing 3](#_Toc75630495)

[Other tools 3](#_Toc75630496)

[Docker 3](#_Toc75630497)

# Functional testing framework

## About

Framework is custom made. Technologies and tools used in framework:

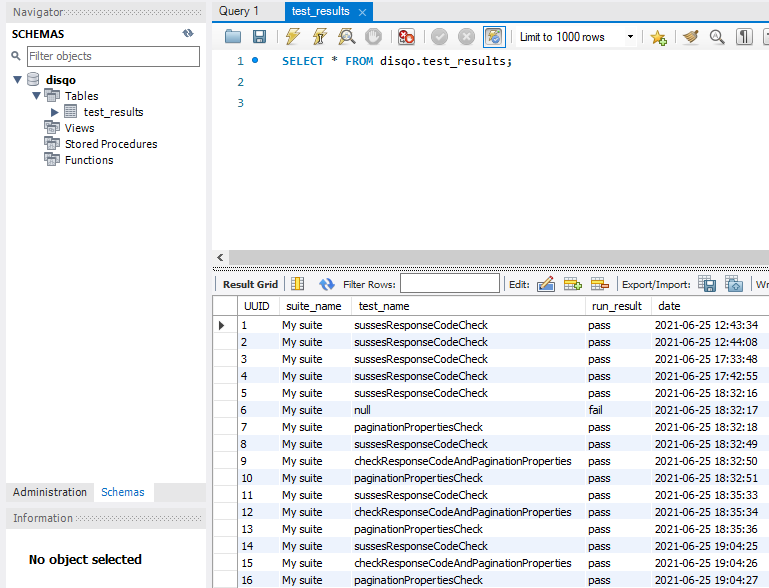
* Java – as main programming language,
* Gradle – as dependency and build tool,
* TestNG – as unit testing framework,
* RESTAssured – as API testing framework,
* Allure – for graphical tests run reporting,
* Log4j – for test run logging,
* AWS SDK – for connecting to AWS RDS Aurora DB with use of IAM,
* MySQL connector – for querying test result DB.

## Code architecture

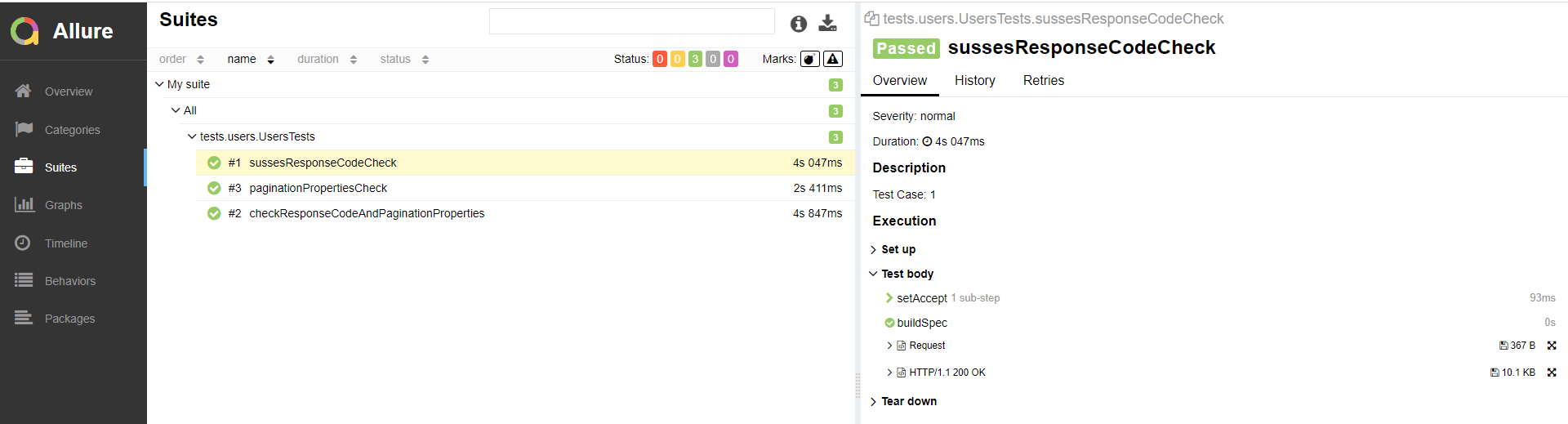
* Request specifications for each endpoint:
  + Page Object pattern like used for specifications,
  + Builder pattern used for specification construction.
* Pojos for each entity,
* Utile classes:
  + AwsRdsUtils – for connecting to AWS RDS DB,
  + DBLog – DB logging,
  + Log - Log4j logging,
  + TestListener – TestNG test run listener,
  + TestStatuses – enum for test run statuses,
  + Utils – random number and string generation.
* Tests:
  + Base test,
  + Endpoint tests.

Also, main resources have Log4j configuration and AWS us-east-2 PEM certificate. Test resources has only TestNG test suite configuration file.

## DB log sample:



## Allure report sample



# Load testing

Framework used – JMeter.

# Other tools

## Docker

Dockerfile.test - contains docker container definition for functional testing framework.

Docker-compose.yml – contains docker compose definition for docker image run and functional tests full suite execution.