live-viewer-backend

Git commit with doc: df6705d

HTML version

1. Testing

1.1. Running unit tests

1. It's simple (and these tests are also called when building the Docker container). Just type:

```
$ ./gradlew clean build
```

> ℹ Maybe `KafkaConsumerServiceTest` should be considered an integration test. But since it runs by automatically creating an `EmbeddedKafkaBroker` by `SpringBootTest`, it does not require an active server. So, this test runs like any other ordinary unit test.

1.2. Running integration tests

1. Currently, these are the available integration test classes:

```
$ find . -type f -name '*IntegrationTest.java'
./src/test/java/com/snowplowanalytics/liveviewerprofile/service/KafkaConsumerServiceIntegrationTest.java
./src/test/java/com/snowplowanalytics/liveviewerprofile/repository/VideoEventRepositoryIntegrationTest.java
```

1.2.1. Testing `KafkaConsumerServiceIntegrationTest`

1. Start Kafka:

```
$ ../docker/compose.sh up kafka-services
```

2. Run the following command:

```
$ test_cases=KafkaConsumerServiceIntegrationTest ./run-integration-tests.sh
```

> 💡 You can pass the `--info` parameter to this script in order to get more details while running it.

1.2.2. Testing `VideoEventRepositoryIntegrationTest`

1. Start LocalStack:

```
$ ../docker/compose.sh up localstack
```

2. Run the following command:

```
$ test_cases=VideoEventRepositoryIntegrationTest ./run-integration-tests.sh
```

1.2.3. Testing all

1. If you already started the services required for each test, you can also type:

```
$ ./run-instegration-tests.sh
```