Introduction

This page contains a guide on how to build and deploy the Quick Check software.

Project Repository

Repository service

- Controllers: handle http requests and responses
- DTOs (Data Transfer Objects): received from and send back to client, help us to create client specific views
- DB:
    - Entities: represent a table in the DB
    - Repositories: interface to communicate with DB
- Services: orchestra data transfer between client and DB (whatever needs to happen, happens in one of the services)

Build Process

Deployment for Development

Prerequisites

Docker is used to isolate each service (i.e. mysql database, springboot application) that we are working on and to be able to develop across different environments.

1. Install Docker
2. Start Docker on your machine

Launch development container

1. Open terminal and cd in root directory
2. Run command docker compose up

By executing the last command, docker will create the following containers.

- Container "database" is based on an image of a mysql:8
- Container "service" is based on an image of openjdk:11
- Container "webserver" is based on an image of nginx

Extending the service

To be able to deploy your changes to the service you need to fire a build command within your IDE.

To fire a build command in Intellij press command + F9 on Mac or from Menu Build --> Build Project

The build command causes springboot to restart within the container and apply your changes.

Testing Backend

1. Launch development container
2. Open another tab in your terminal
3. Run the following commands
    1. docker exec -it service bash (enter docker container service with a bash script)
    2. cd app (directory app contains pom.xml, needed to run mvn command)
    3. choose one of the following:
        1. mvn test to run all tests
        2. mvn test -Dtest=classname to run all tests within classname
        3. mvn test -Dtest=classname#method to run a single method within classname

CORS Configuration

We used a CORS configuration for defining rules for identification of origins. These origins are allowing access to our buckets