Docker, CI, Travis

Patka Zsolt-András

2020.03.16

Sapientia - Computer Science BSc.

What is this presentation about?

- Continous Integration
  - CI CD
- Docker
  - Dockerfile
  - Docker image
  - Docker container
- How to: CI
  - Travis
  - Github Actions
- Live demo
  - Backend
  - Frontend
- Advices

Continous Integration

Continous Integration: CI CD

Continous Integration Continous Delivery

1. Plan
2. Code
3. Build
4. Test
5. Release
6. Deploy
7. Operate
8. Measure
9. Repeat

Docker

Docker: Dockerfile

1. How should our image look like?
2. Analog: Source code for Windows

Docker: Docker image

1. Result of building the Dockerfile
2. Contains everything needed to spin up your application
3. Analog: CD containing Windows

Docker: Docker container

1. Result of starting a docker image
2. Is alive, runs your application
3. Analog: Your PC after you've installed Windows

How to: CI

How to: CI: Travis

1. Free (has a paid version as well)
2. Can easily be integrated with Github
3. Uses Docker containers internally
4. .travis.yml: contains build steps to be executed on each push

How to: CI: Github Actions

1. Free
2. It's trivial to integrate it with Github
3. Uses Docker containers internally
4. Easy to use

Live demo

Live demo: Backend

1. Spring boot app
2. Has two simple endpoints
3. Travis automatically (on each push):
    3.1 Runs the build
    3.2 Runs the tests
    3.3 Creates a .jar file
    3.4 Build the Dockerfile
    3.5 Pushes the Docker image to DockerHub

Live demo: Frontend

1. React
2. Can send a request to the backend
3. Travis automatically (on each push):
    3.1 Runs the build
    3.2 Runs the tests
    3.3 Build the Dockerfile (two stage build)
    3.4 Pushes the Docker image to DockerHub

Advices

Advices: Docker best practices

Advices: Docker commands cheatsheet

Thank you for your attention!

Sources

- https://mherman.org/blog/dockerizing-a-react-app/