

School of Computer Science & Statistics (SCSS)

Faculty Science, Technology, Engineering, and Mathematics (STEM)

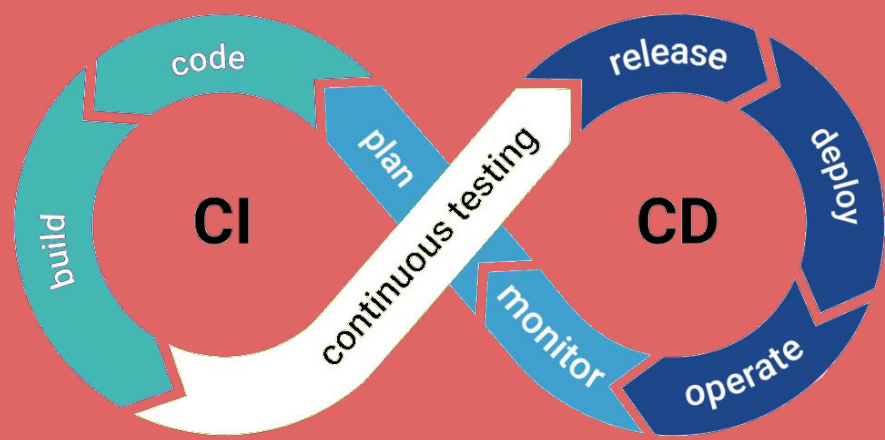
Kanban Board for Multiple Users

Students: Pavel Petrukhin, Peadar Kenny, Merlin Prasad, Aoife Simm, Leah O’Connor, Yongjia Li & Wen Geng Lin

Client-mentor: IBM (Mihai Criveti & Panpan Lin)

Overview

A kanban board is an agile project management tool. It is a shared space in which teams can asynchronously and visually manage their work.



Our client has put an emphasis on the importance of following the CI/CD development process and gaining development experience during our project.

Implementation

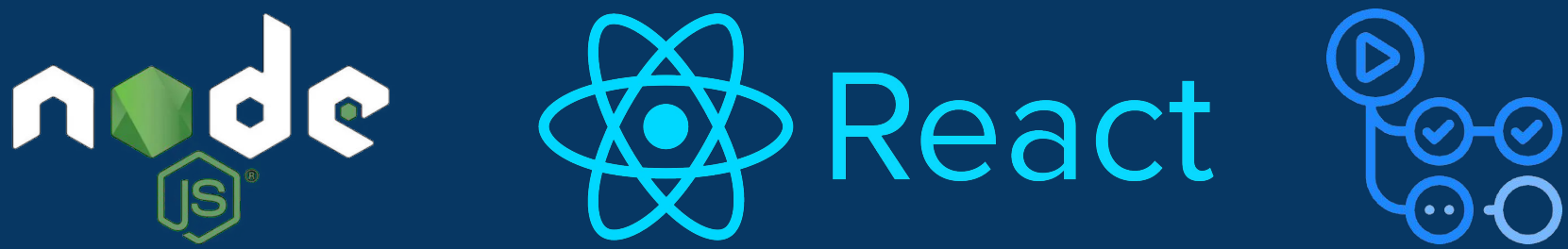
The aim of our project is to build a Kanban board that...

- supports multiple users,
- stores ticket data on a persistent NoSQL database,
- is built using open source development practices and tools
- is deployed on Openshift,
- has build containers based on the Red Hat Universal Base Image with Red Hat Quay and Clair as the container registry,
- uses CI/CD pipeline with code scanning, unit test coverage and static analysis stages, and
- uses Podman for the build image.



We decided to implement the frontend using React. In the backend we’re using Express for our REST API and MongoDB for our database.

Technologies Chosen



We chose...

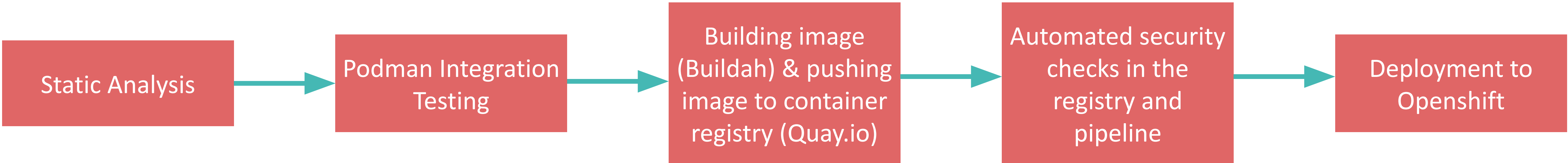
- NodeJS, Express and MongoDB as they’re easy to learn, and supported by most technologies currently available
- React due to many members having prior experience
- Github Actions for its easy set-up for CI/CD, pre-made workflows, and ease of integration with Red Hat technologies

Industry Practices Implemented

We successfully implemented...

- Code review and pair programming sessions
- CI/CD pipelines to ensure reliability of the production deployment
- Static code analysis to catch idiomatic errors and to guarantee consistent style
- Microservices design to decouple different parts of the projects and provide better reliability

CI/CD Pipeline Architecture



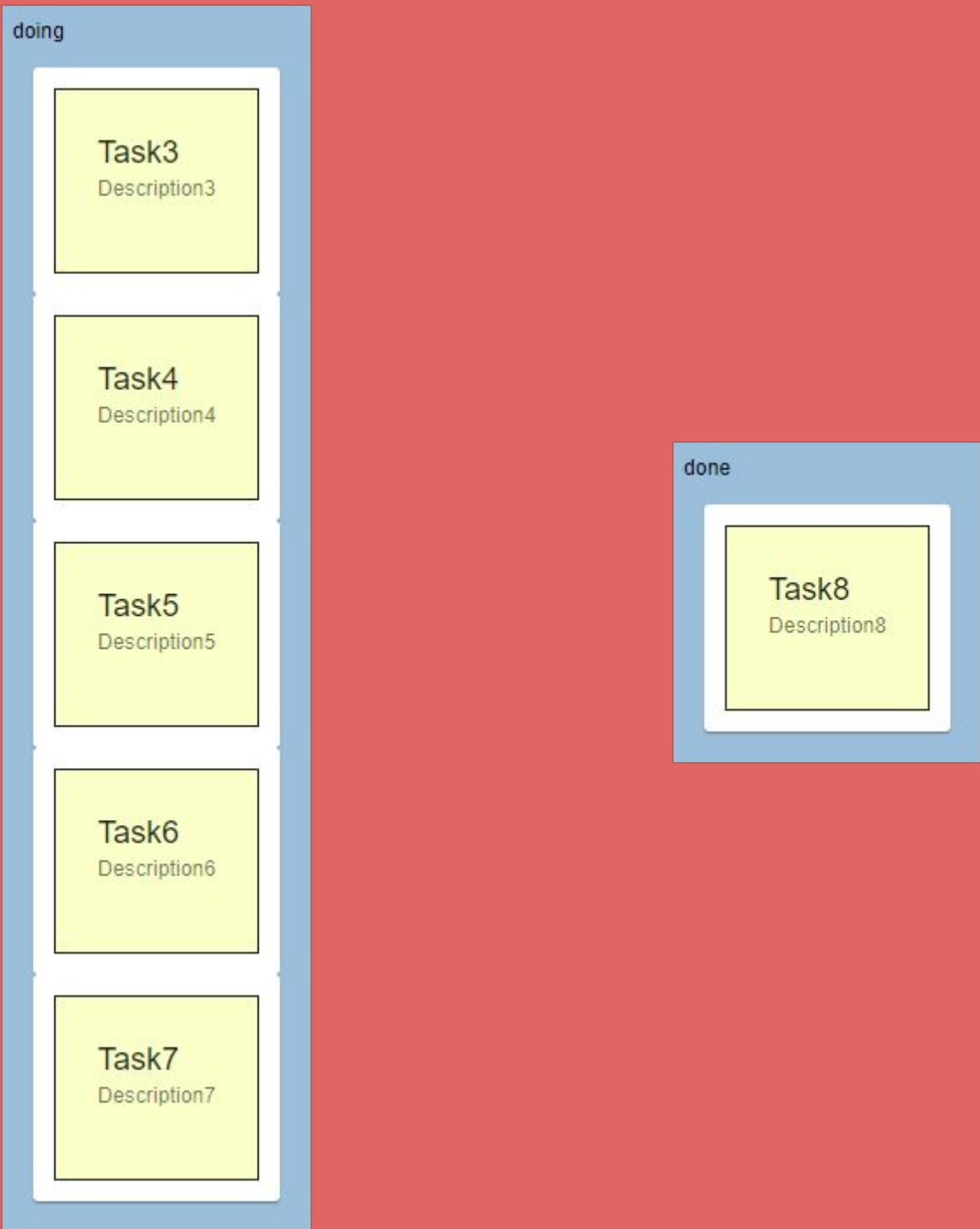
- Each stage is triggered upon the successful completion of the previous stage
- There are two pipelines in the project: development and production
- The difference between the two pipelines is in the container registry repositories and where each image is deployed
- Development pipeline is triggered upon a pull request and push
- Production pipeline is triggered upon pushed to the main branch only

Organisation

- Grouped team according to prior expertise
- Facilitated intra-group mentoring through regular check-in messages and meetings, usually in the form of paired programming or Q&As
- Tracked issues using Github, and set-up automation to increase transparency
- Planned agile sprints and adjusted the backlog according to our progress

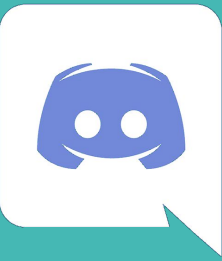


UI Mock-Ups



Communication

- Everyday updates through Discord
- Working-group meetings at least twice a week
- Client-mentor meetings every two weeks to pivot our development with additional interim communication through Slack



Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin



Human Capital Initiative

