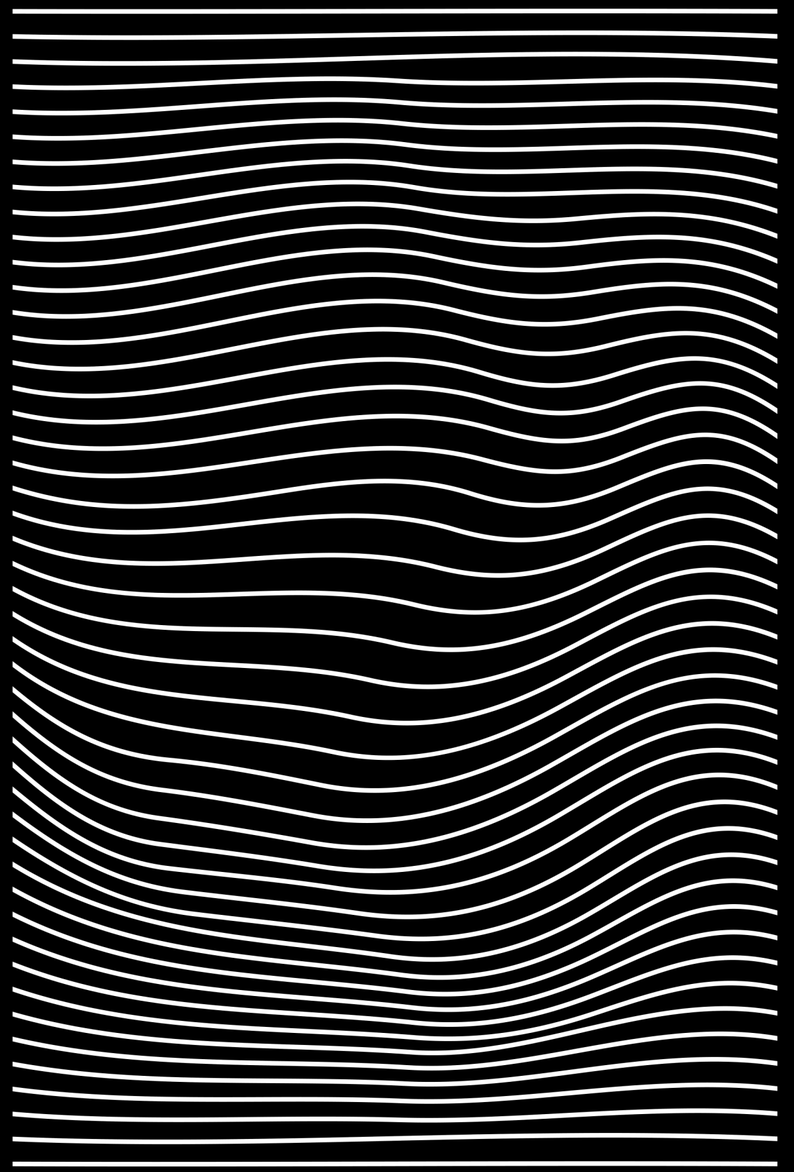
TQS Final Project



Bruno Páscoa 107418
Guilherme Lopes 103896
David Cobileac 102409

Agenda

1. Concept
2. Stories
3. Architecture
4. Deployment
5. Quality Assurance

Concept

- Apollo Care, system that facilitates scheduling and management of consultations for both clinic workers and patients alike
- Patient portal that allows the patient to schedule appointments and check past and future consultations
- Staff portal that manages consultations at the clinic
- Call screen for managing queues

Stories

Schedule an appointment

As a patient, I want to schedule a Cardiology appointment for next week

Check previous appointments

As a patient, I want to check the appointments I had last month

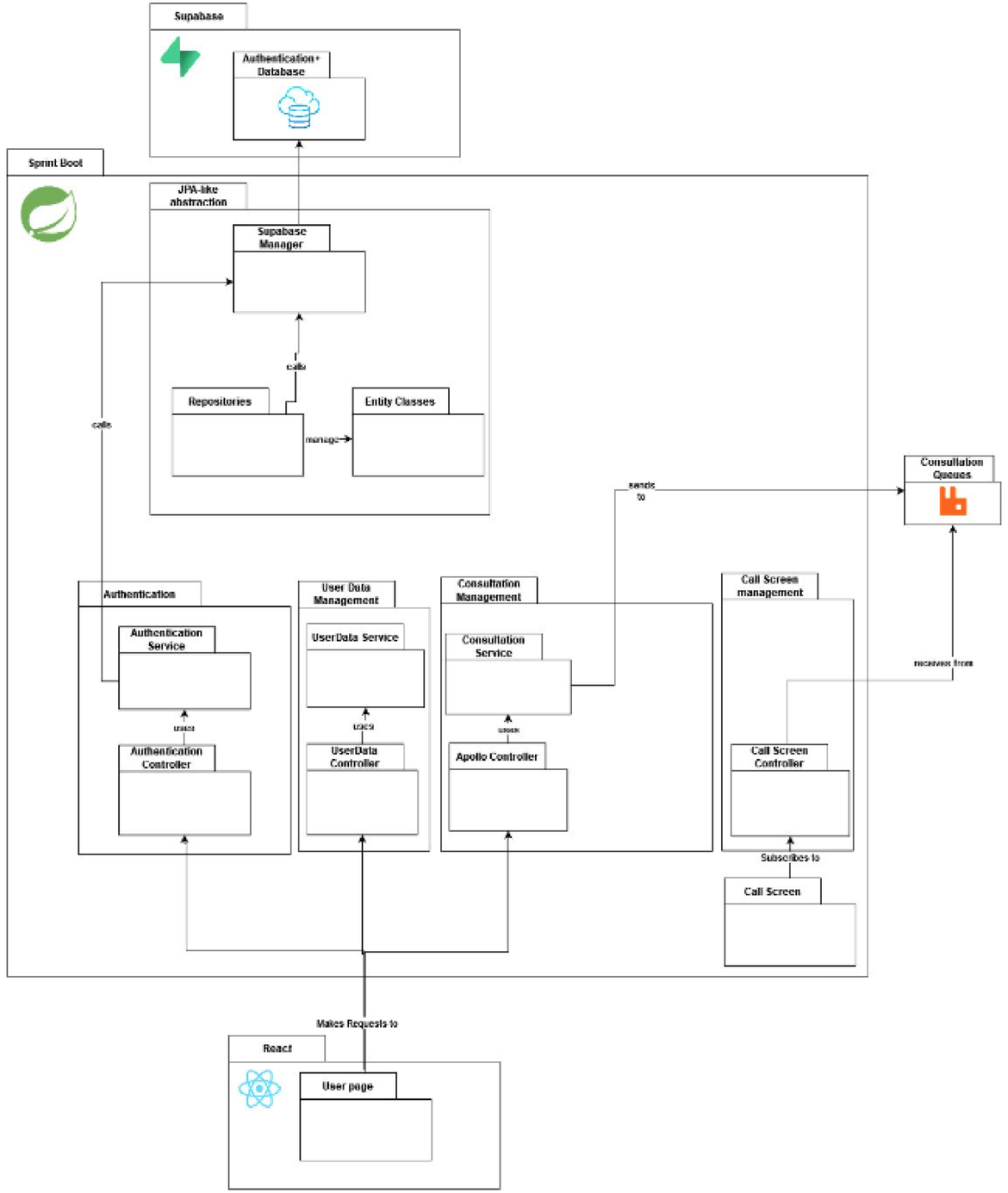
Checkin a patient

As a staff member, I want to be able to checkin a patient for their appointment

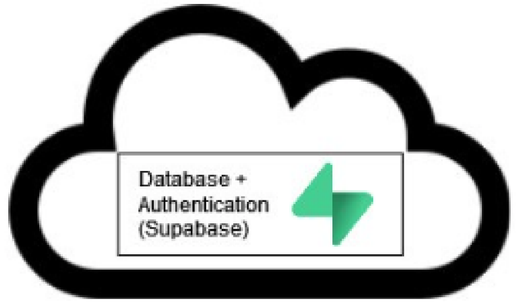
Schedule an appointment for a patient

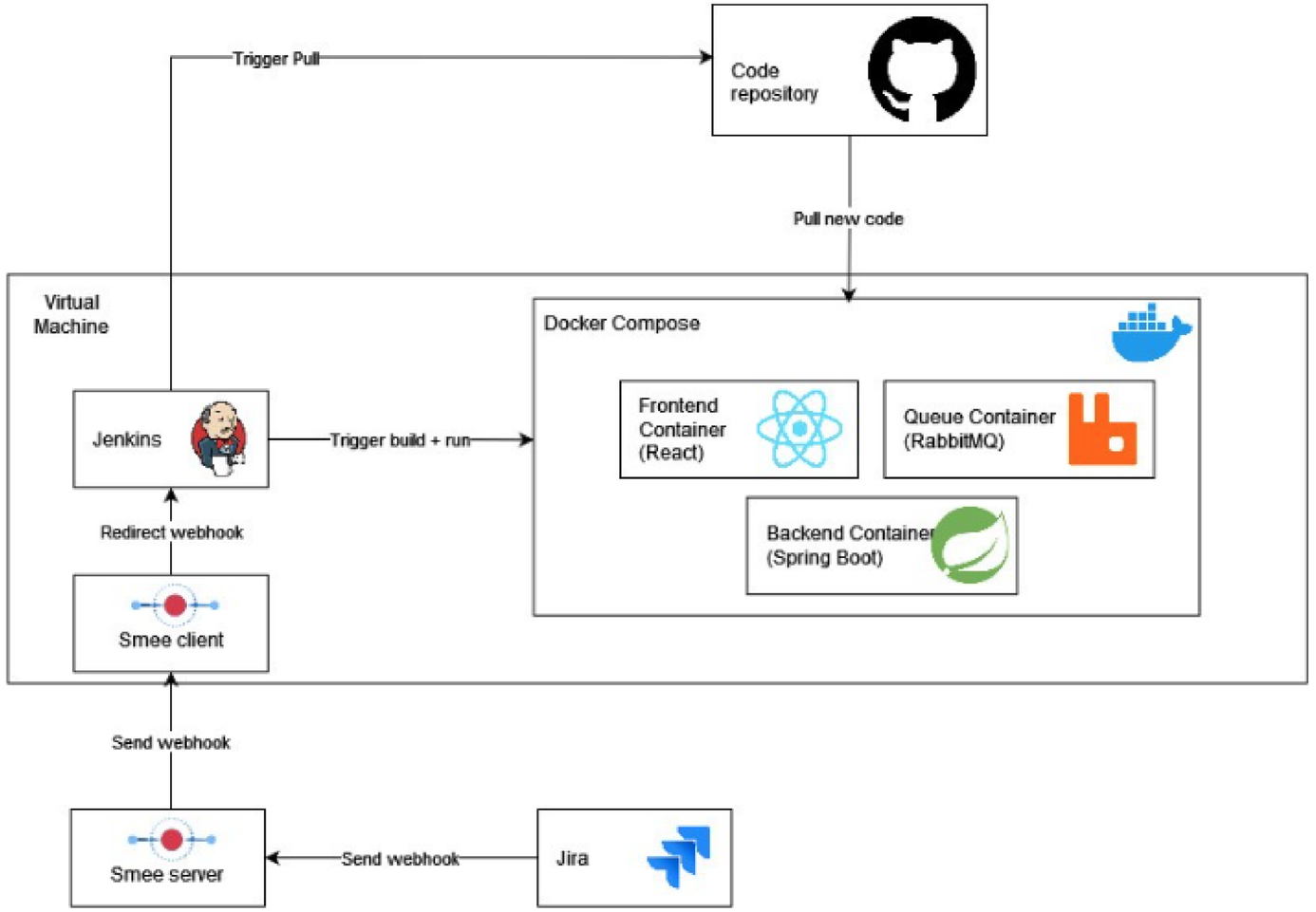
As a staff member, I want to schedule an appointment for a patient that is in the clinic at the moment

Architecture

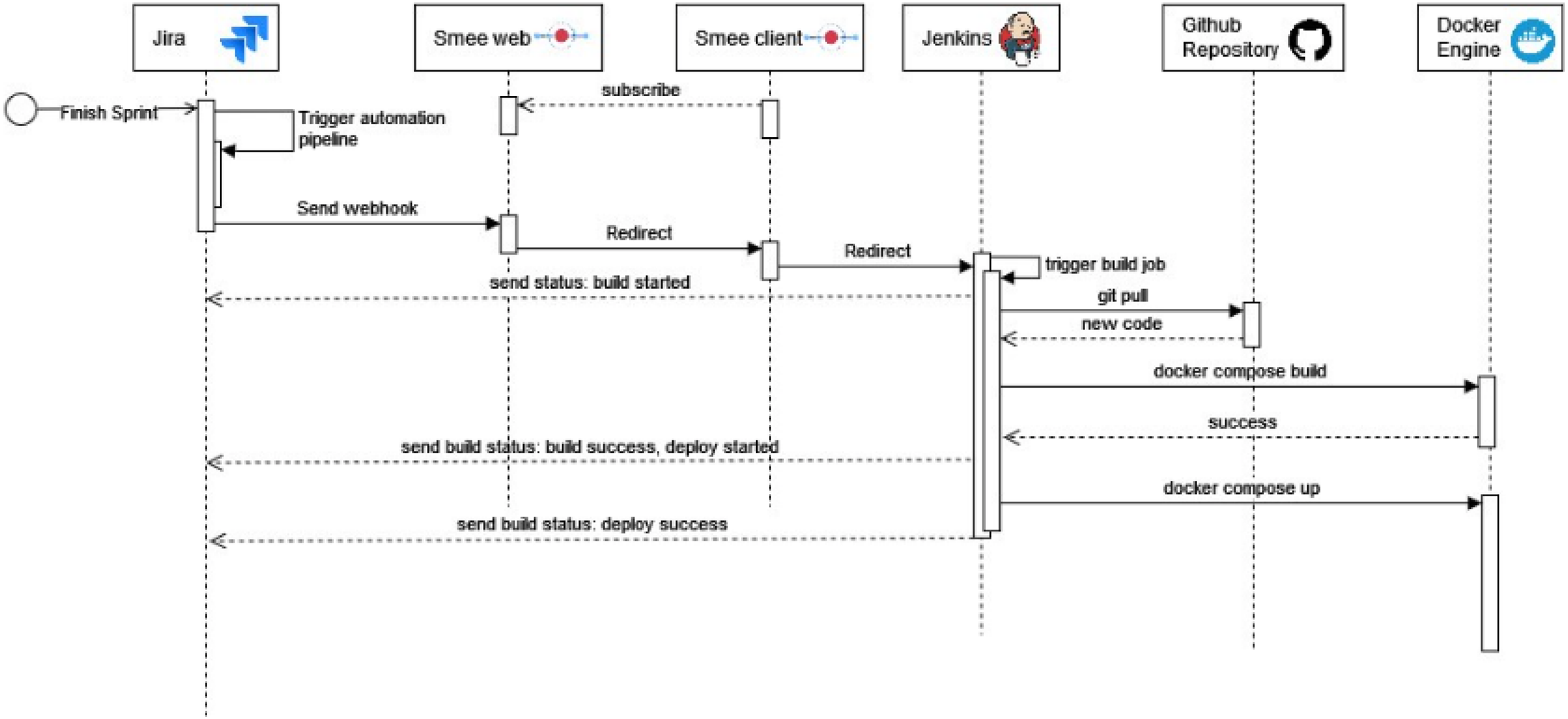


Deployment


Database +
Authentication
(Supabase)


Trigger Pull
Code
repository
Pull new code
Virtual
Machine
Docker Compose
Jenkins
Trigger build + run
Frontend
Container
(React)
Queue Container
(RabbitMQ)
Backend Container
(Spring Boot)
Redirect webhook
Smee client
Send webhook
Smee server
Send webhook
Jira

Deployment


Jira
Smee web
Smee client
Jenkins
Github Repository
Docker Engine
Finish Sprint
Trigger automation pipeline
subscribe
Send webhook
Redirect
Redirect
trigger build job
send status: build started
git pull
new code
docker compose build
success
send build status: build success, deploy started
docker compose up
send build status: deploy success

Quality Assurance

Code Quality Management

Coding Style Guidelines

- Follows Oracle’s Coding style
- Similar coding styles by devs

Code Quality Metrics

- VSCode plugins
- SonarCloud

Development workflow

Code Review Process

- Reviewed by other team member.
- Focus on spotting issues missed by SonarCloud.
- Manual acceptance tests for code cleanliness and functionality.

Definition of Done

1. Issue must have a score and description
2. Seperate branch related to the issue
3. Functional code with all unit tests passing
4. Pass SonarCloud's quality gate
5. Pull request must be created
6. Reviewed and accepted by other team member

CI/CD pipeline and tools

CI Pipeline

- Done on GitHub Actions
- Runs tests and reports to Jira and SonarClound
- It fails if quality gate doesn’t pass
- Jira created relevants tests
- SonarCloud reported in pull requests

CD Pipeline

- Use Smee to handle VM restrictions
- Integration with Jenkins for fetching, rebuilding, and rerunning
- Dockererzied components
- Containers orchestrated with docker compose

Software Testing

Functional testing

Unit tests

System and integration testing

Performance testing

Questions