FileFighter

Fight against Chaos

FileFighter

Fight against Chaos

Agenda

Agenda

1. Vision

1.1 Vision

- Network-Attached-Storage solution
- Easy to set up for self-hosting
- Free and open source alternative for Google-Drive/Dropbox
- for home use

1.1 Vision

- fast
- simple
- secured via login
- well structured web interface
- supports all the common file types
- different user groups
- share files and folders
- upload, download, get rid of files
- find all files through search feature

1.2 Scope & Use Cases

- 1. Semester
  - CRUD users
  - View folder contents
  - Sort folder contents
  - Edit filesystem (rename, delete)
  - Install Script / Tool
- 2. Semester
  - Download
  - Upload
  - Search files
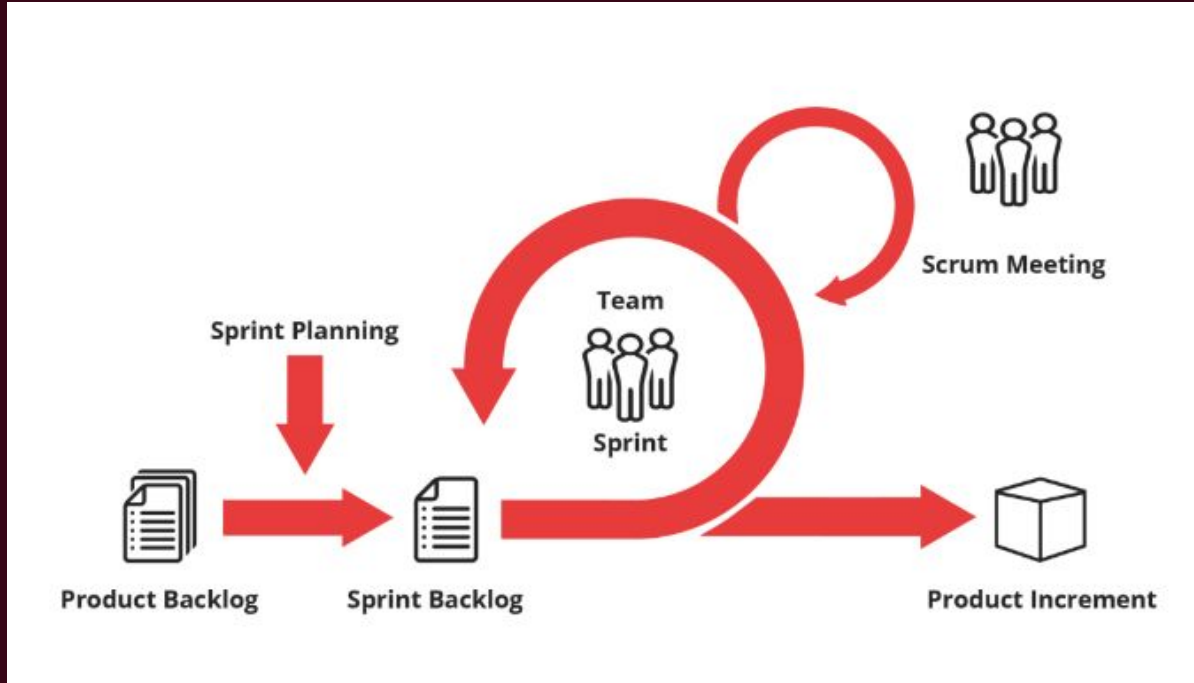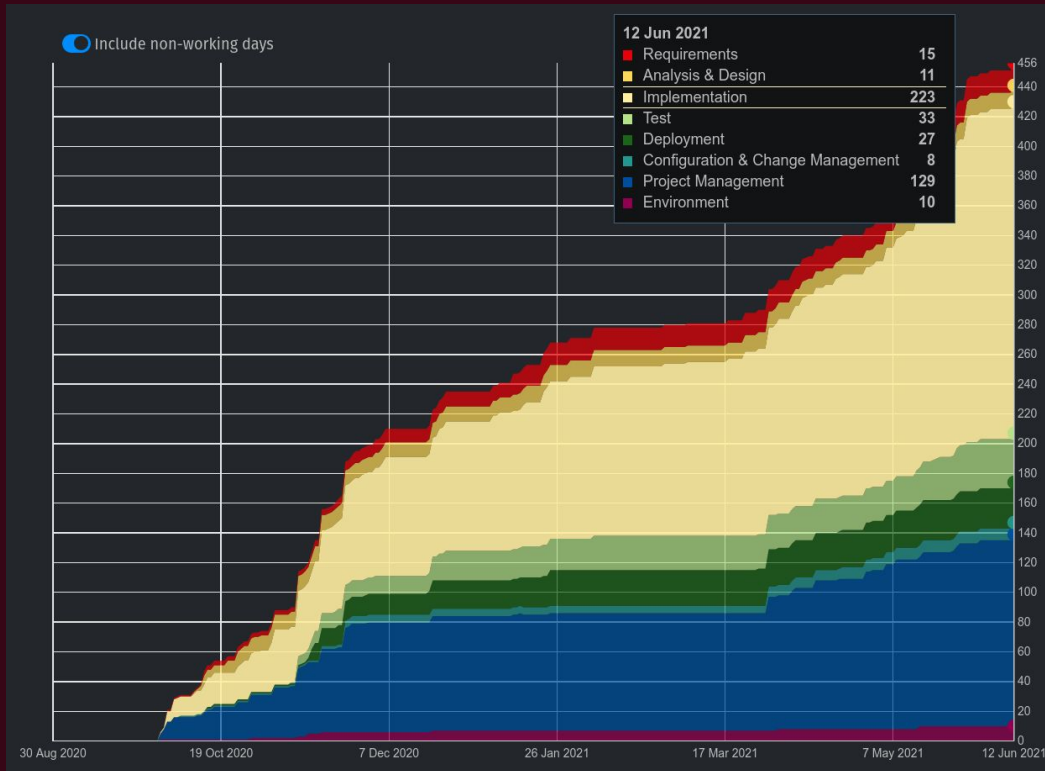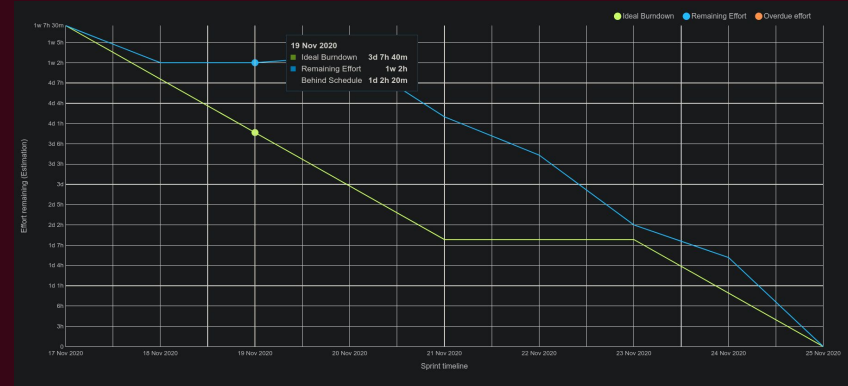 - Edit profile
  - Show profile
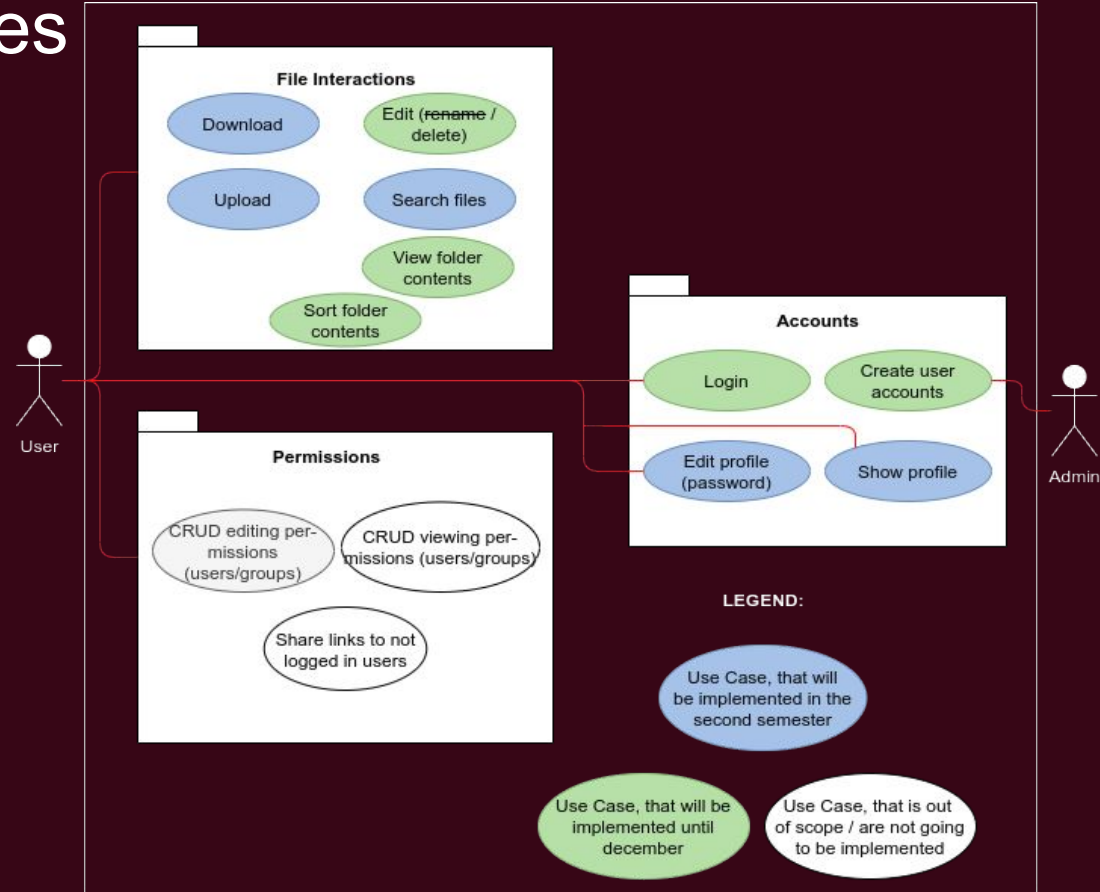
2. How we work
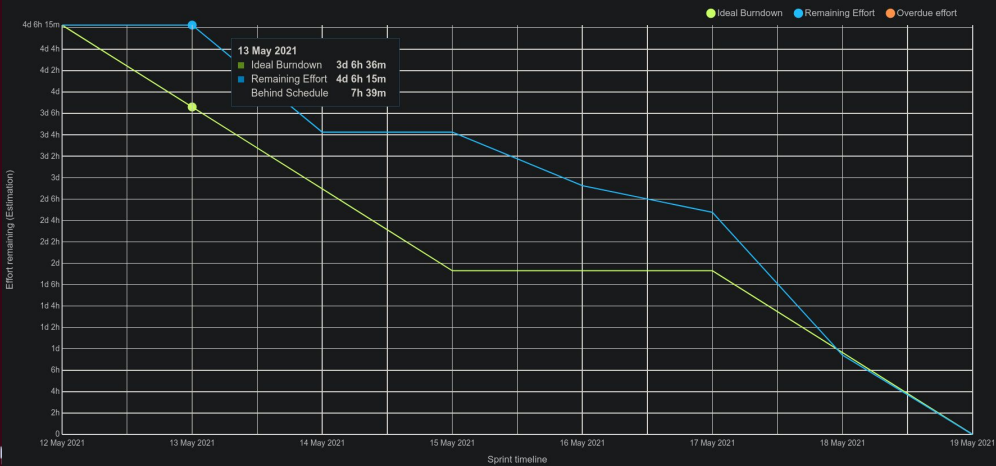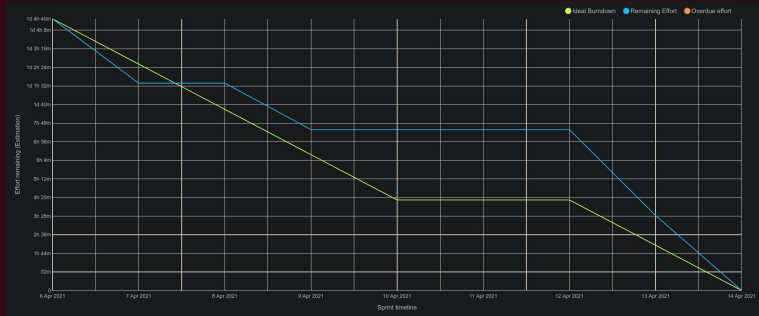
2.1 Scrum

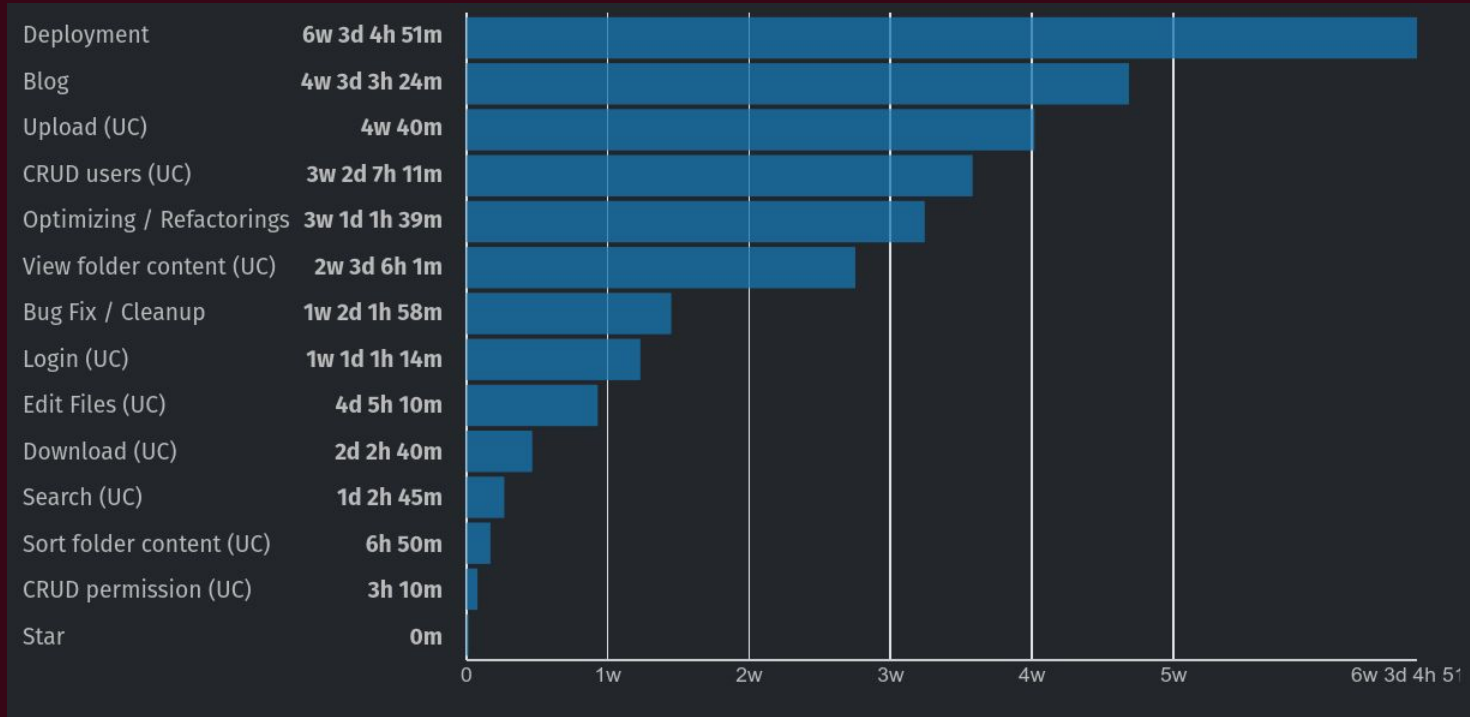# 2.2 RUP / long term planning

2.3 Burndown Chart

2.4 Cost Estimation

Use Case	Function Points (rounded)	Estimated Time in h	Spent Time in h
CRUD Users	234	-	134
View Folder Contents	165	-	110
Login	119	-	49
Upload	190	105	160,4
Download	120	45	20
Search and Sort Files	90	50	19

2.5 Time per Use Case

Use Case	Time
Deployment	6w 3d 4h 51m
Blog	4w 3d 3h 24m
Upload (UC)	4w 40m
CRUD users (UC)	3w 2d 7h 11m
Optimizing / Refactorings	3w 1d 1h 39m
View folder content (UC)	2w 3d 6h 1m
Bug Fix / Cleanup	1w 2d 1h 58m
Login (UC)	1w 1d 1h 14m
Edit Files (UC)	4d 5h 10m
Download (UC)	2d 2h 40m
Search (UC)	1d 2h 45m
Sort folder content (UC)	6h 50m
CRUD permission (UC)	3h 10m
Star	0m

2.6 Time Report

Users	Time estimated	Time spent
Total time	**596h 15m**	**681h 23m**
Gimleux		201h 41m
Open Schnick		272h 12m
Valentin		207h 30m

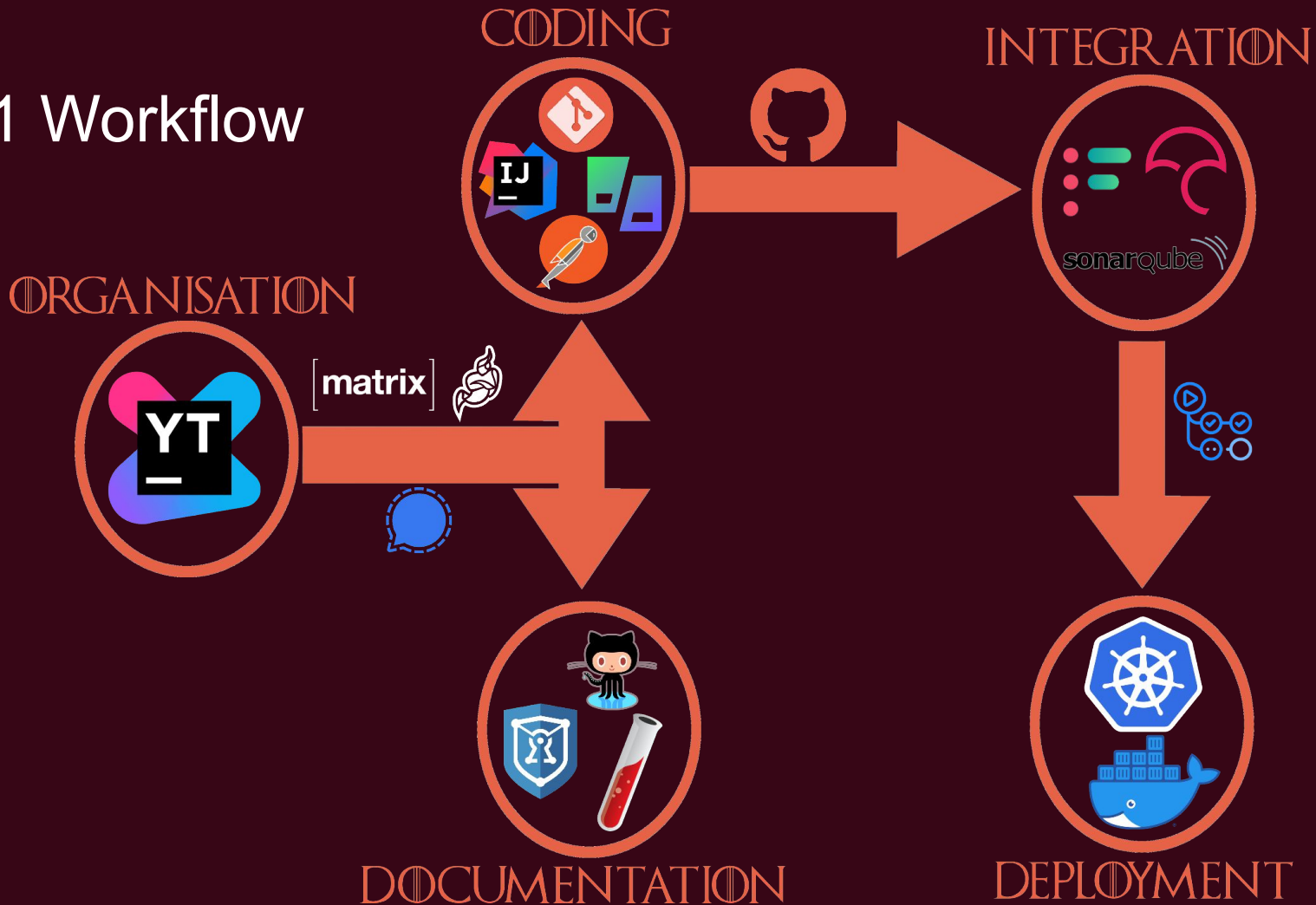# 3. Quality

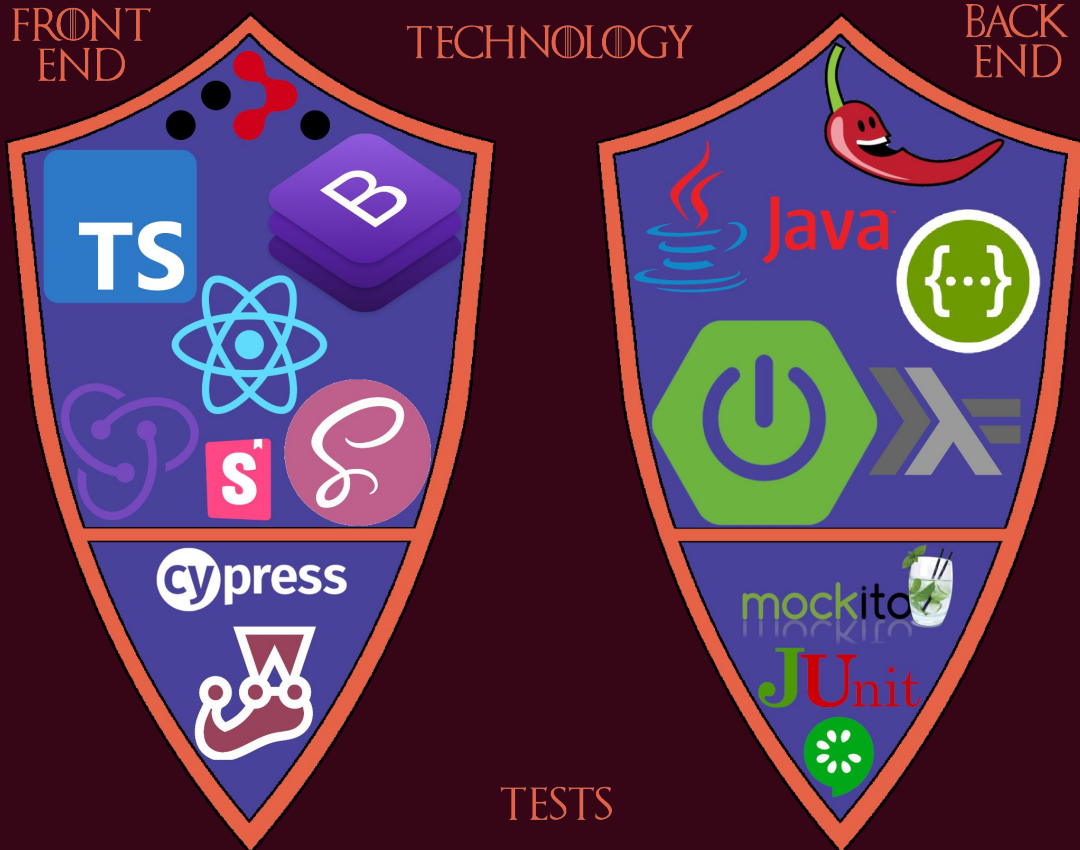3.1 Workflow

CODING

INTEGRATION

ORGANISATION
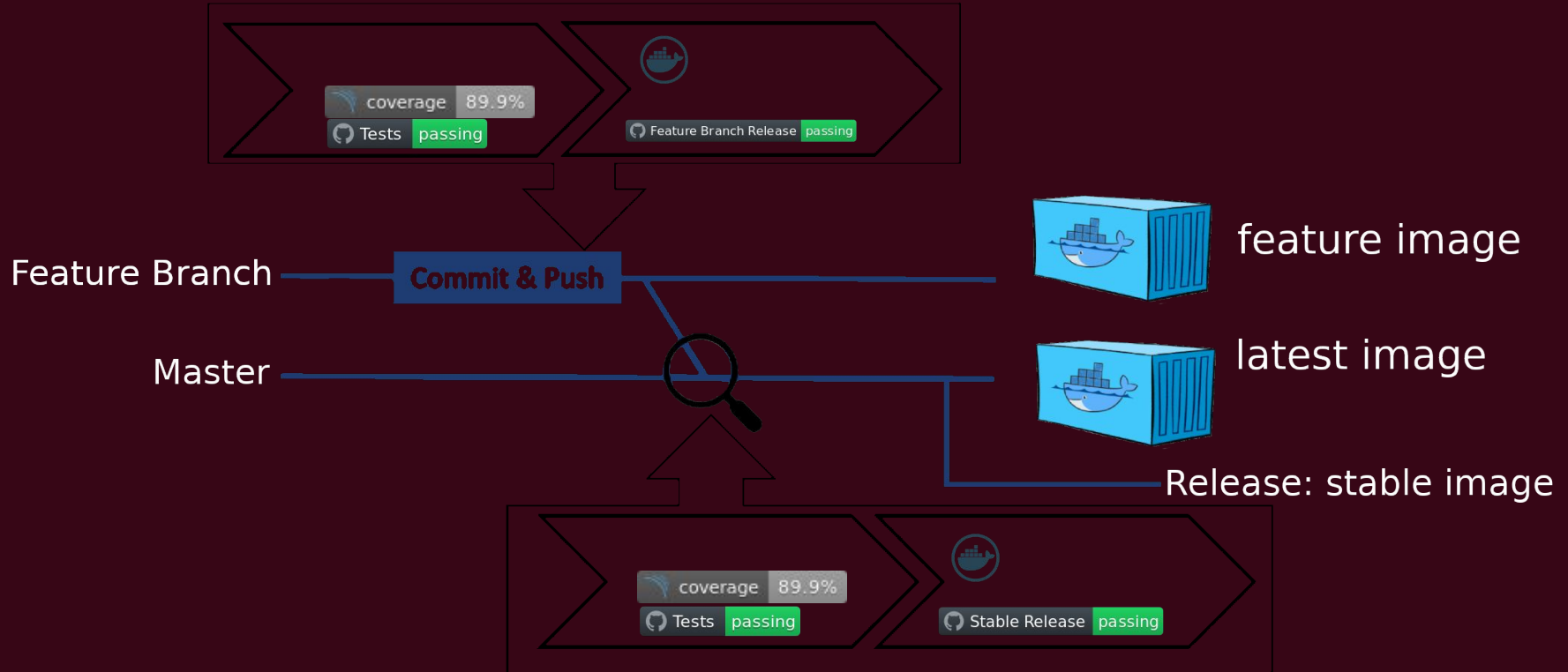
DOCUMENTATION

DEPLOYMENT

3.2 Technologies

3.3 Architecture

- Microservice architecture
- Code maintained by us:
  - MVC backend.
  - Frontend
  - File upload/download handler
- Third party code:
  - Database
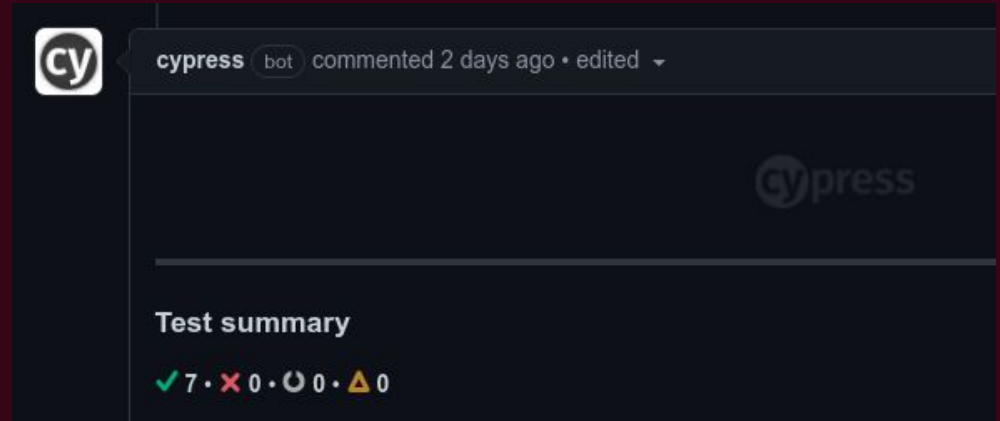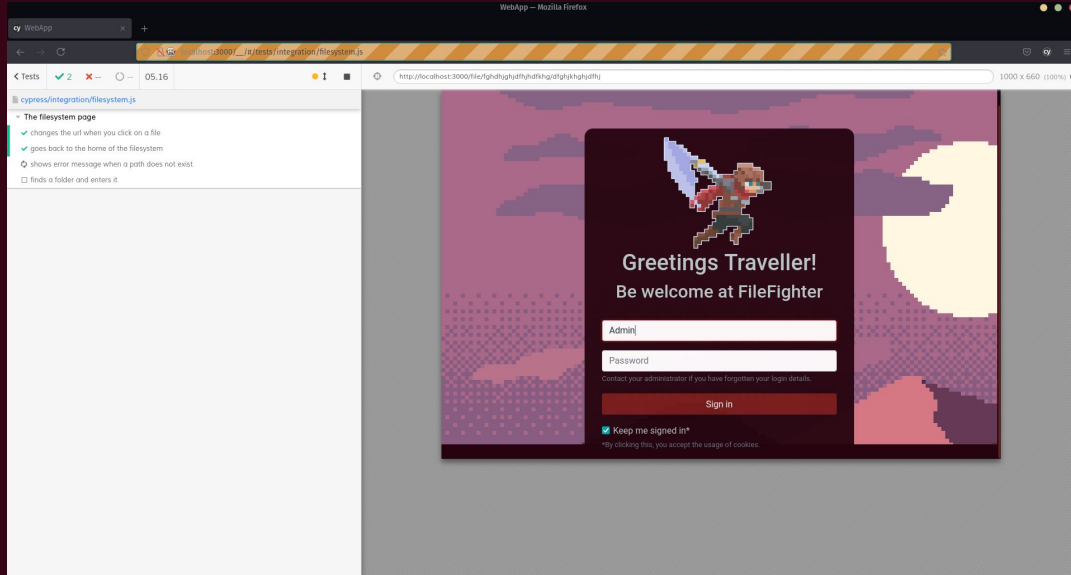 - Reverse Proxy for easier usage and security

3.4 CI/CD & Automation

3.5 Code Quality & Tests

- Unit Tests in backend
- Integration tests with Cucumber
- Snapshot Tests in frontend
- E2E test with Cypress
- CI/CD with GitHub Actions, Maven and SonarQube
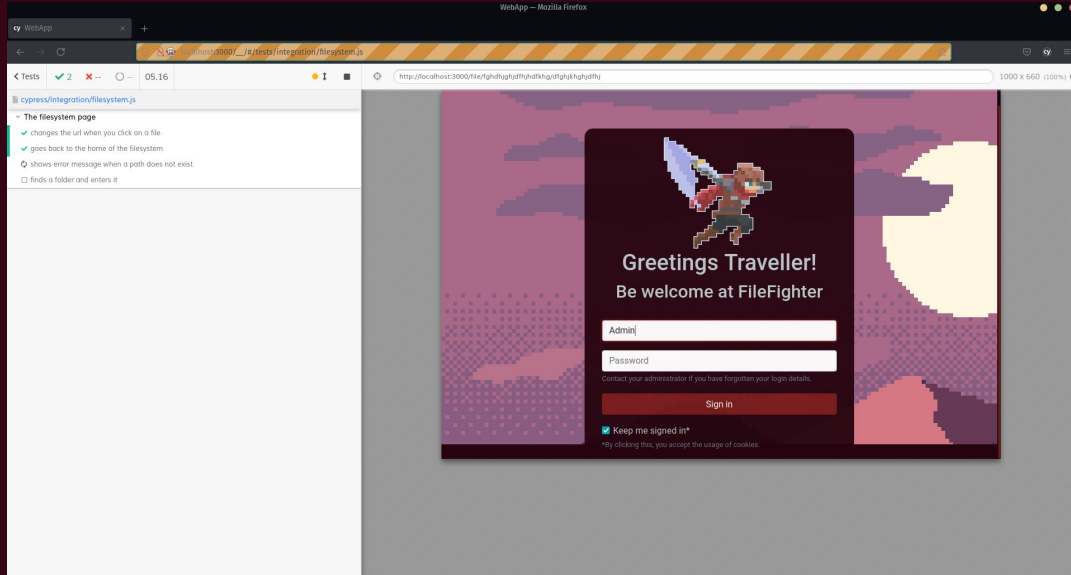
3.5 Code Quality & Tests

All checks have passed
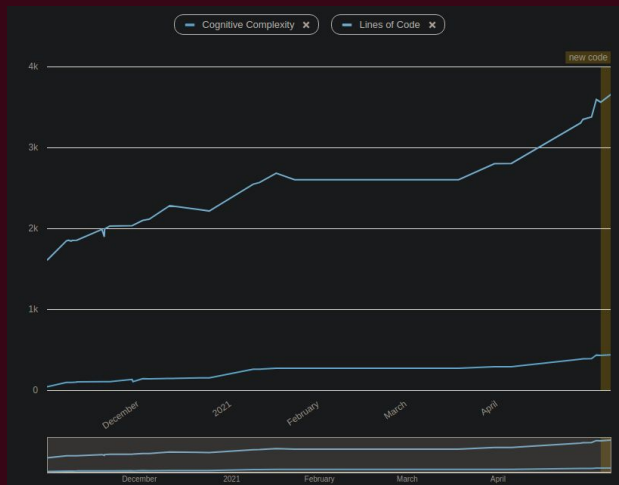6 successful checks

✓	End-to-end tests / runTests (push)		Details
✓	Feature Branch Release / Build_Docker_Image_on_Pu...		Details
✓	Storybook Release / Build_Docker_Image (pull_reque...		Details
✓	CodeFactor — No issues found.		Details
✓	cypress: default-group — 6 tests passed in 00:25		Details
✓	security/snyk (FileFighter) — 3 security tests have pa...		Details

WebApp — Mozilla Firefox

cy WebApp

localhost:3000/_/s/tests/integration/filesystem.js

http://localhost:3000/file/fghdhjghjdfhjhdfkhg/dfghjkhghjdfhj

< Tests ✓ 2 ✗ -- ◯ --   05.16   ● ! ■   ⊕   1000 x 660 (100%) ⊕

cypress/integration/filesystem.js

The filesystem page
✓ changes the url when you click on a file
✓ goes back to the home of the filesystem
◯ shows error message when a path does not exist
☐ finds a folder and enters it

Greetings Traveller!
Be welcome at FileFighter

Admin

Password

Contact your administrator if you have forgotten your login details.

Sign in

☑ Keep me signed in*

*By clicking this, you accept the usage of cookies.

∨ ✓ <default package>	16 sec 636 ms
∨ ✓ JUnit Jupiter	2 sec 676 ms
∨ ✓ FileSystemUploadSteps	1 sec 18 ms
✓ contextLoads()	1 sec 18 ms
∨ ✓ FileSystemUploadServiceUnitTest	106 ms
✓ createNewFolderWorks()	97 ms
✓ createNewFolderThrows()	9 ms
> ✓ UserRegistrationSteps	11 ms
> ✓ UserDtoServiceUnitTest	9 ms
> ✓ UserEditInformationSteps	9 ms
> ✓ SystemHealthSteps	37 ms
∨ ✓ FileSystemRepositoryUnitTest	78 ms
✓ findAllByFileSyste	78 ms

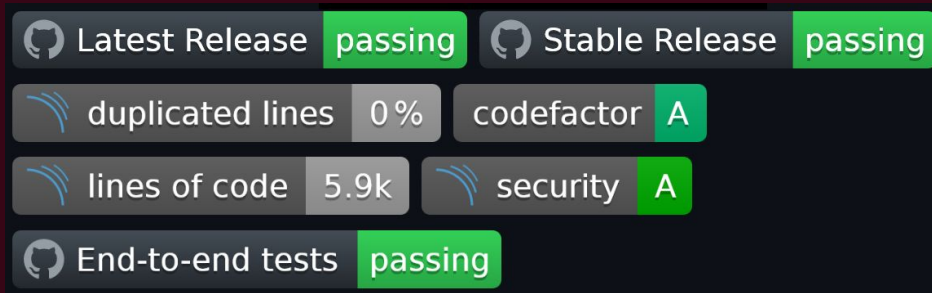Tests passed: 243

3.5 Code Quality & Tests

3.6 Metrics

- SonarQube since almost the beginning

Lines of Code/Complexity

Frontend

Latest Release `passing`	Stable Release `passing`	
duplicated lines `0%`	codefactor `A`	
lines of code `5.9k`	security `A`	
End-to-end tests `passing`		

Backend

quality gate `passed`	coverage `91.9%`	
lines of code `3.7k`	security `A`	
Latest Release `passing`	Stable Release `passing`	
Tests (Master) `passing`		
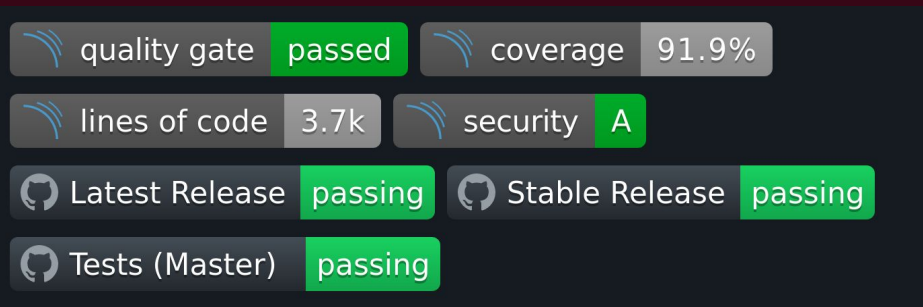
3.7 Design Patterns

- MVC
- Builder Pattern
- Reducer Pattern
- Microservices
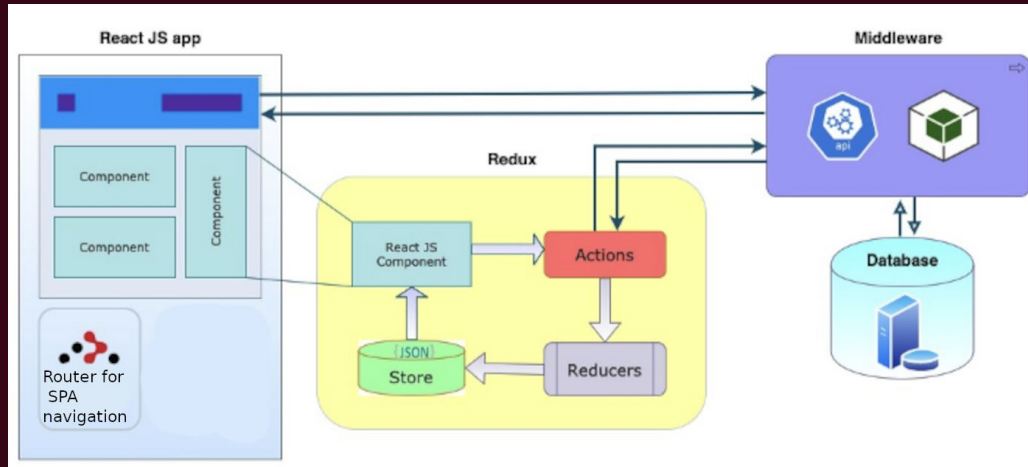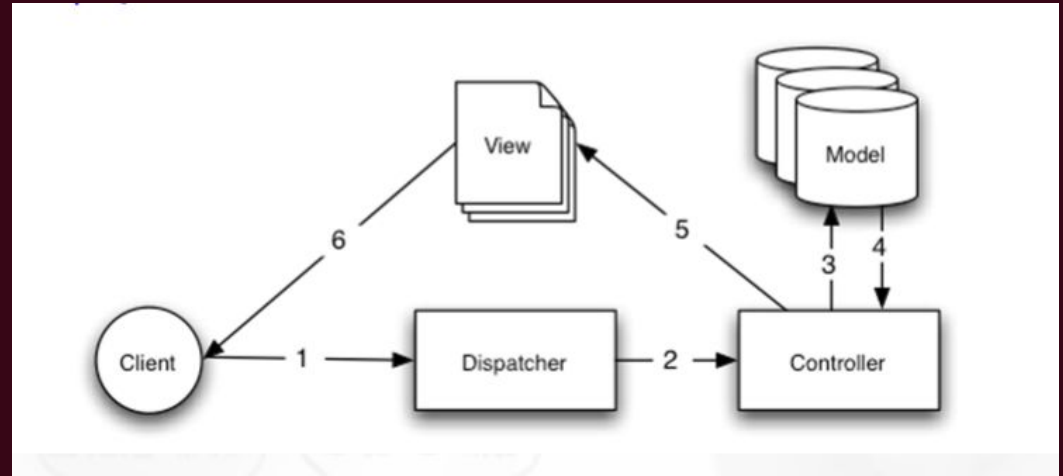
4. Technical Ability

4.1 Class Diagram
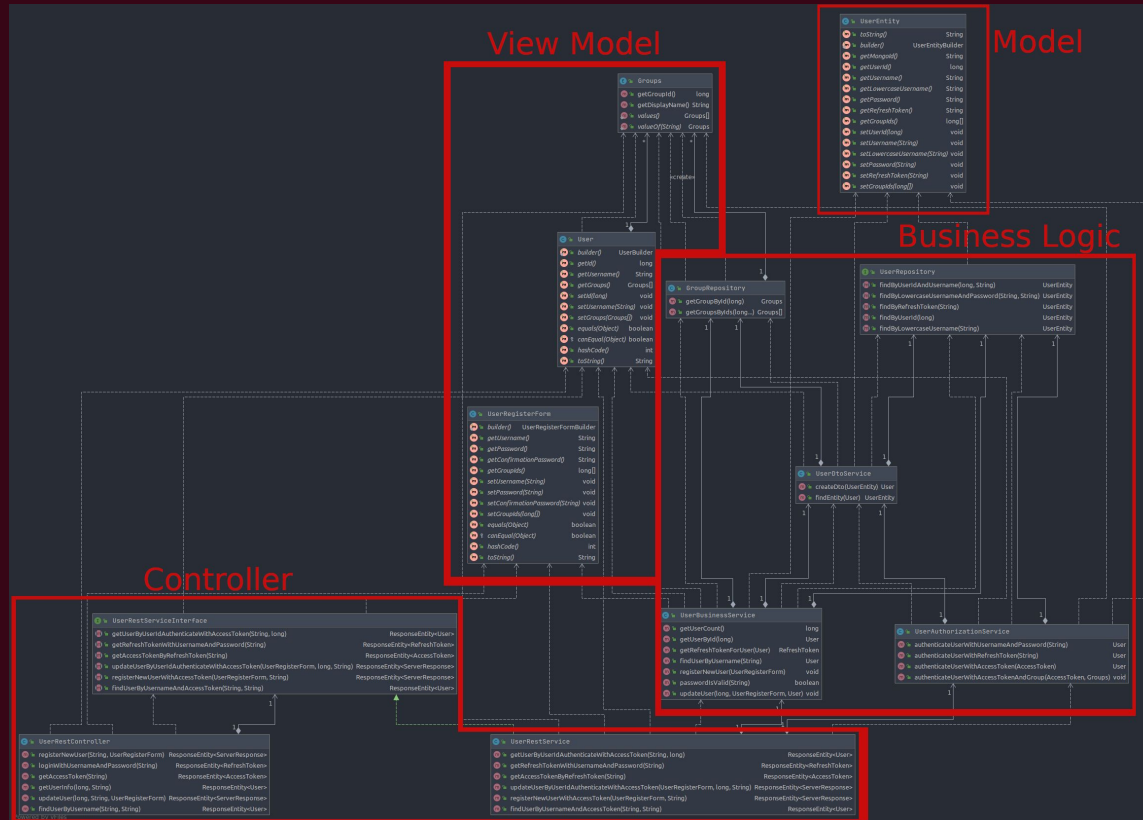
4.2 Deployment View
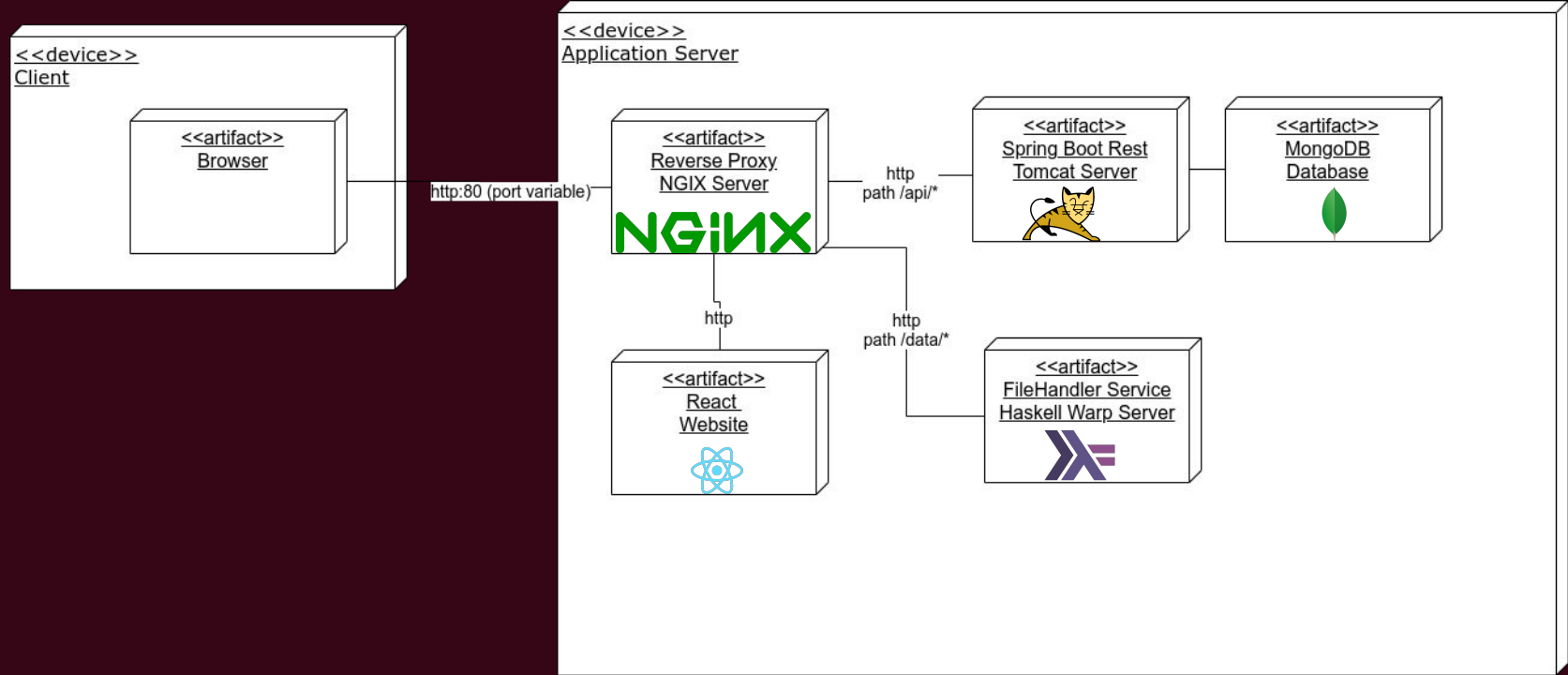
4.3 Demo