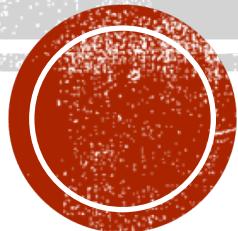


BOOKLY

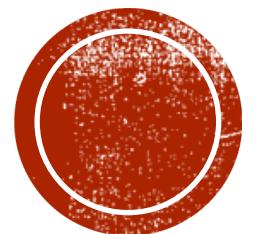
final presentation



OUTLINE

- Use Cases
- Demo
- Project Management
- Architecture
- Quality

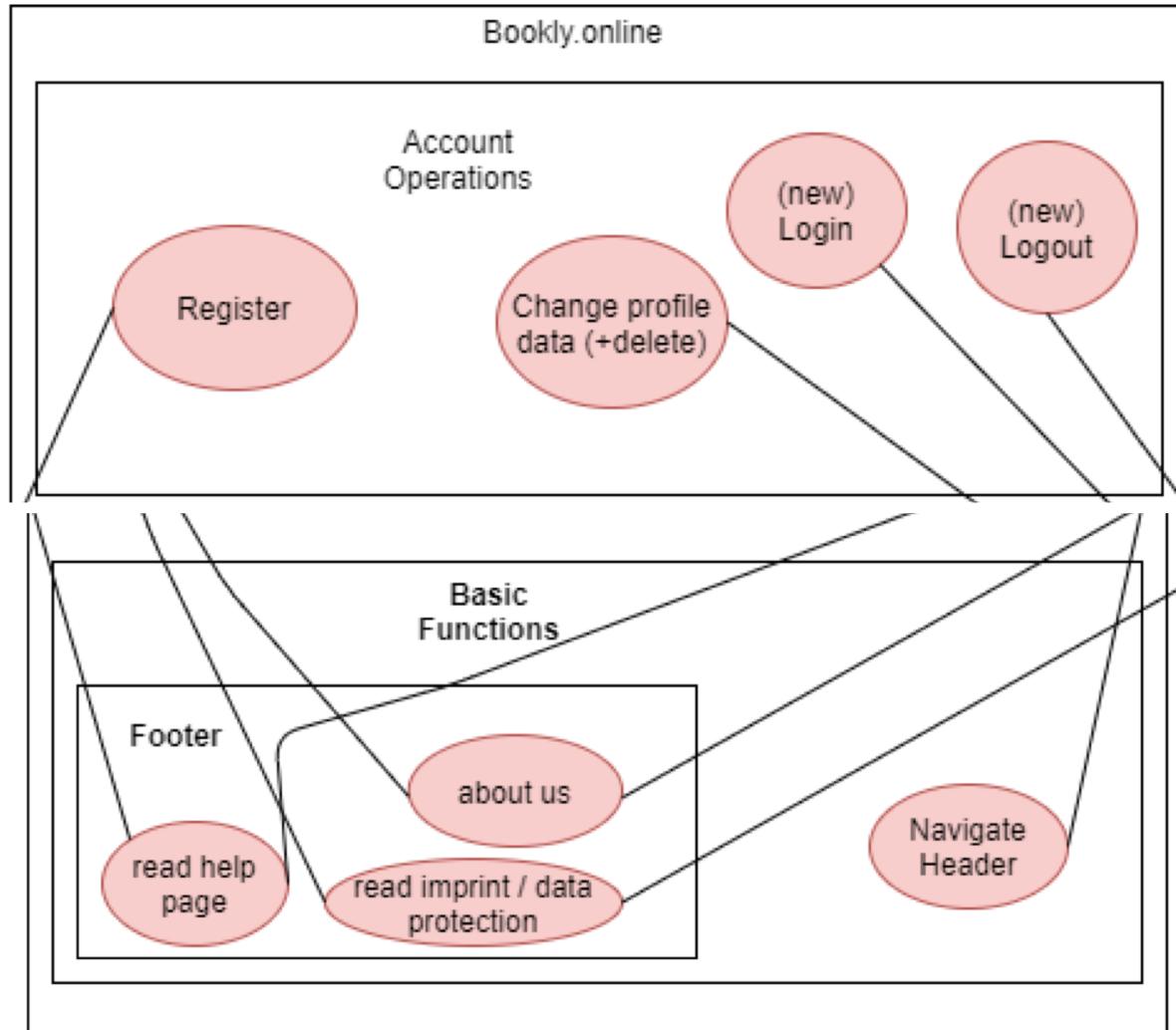




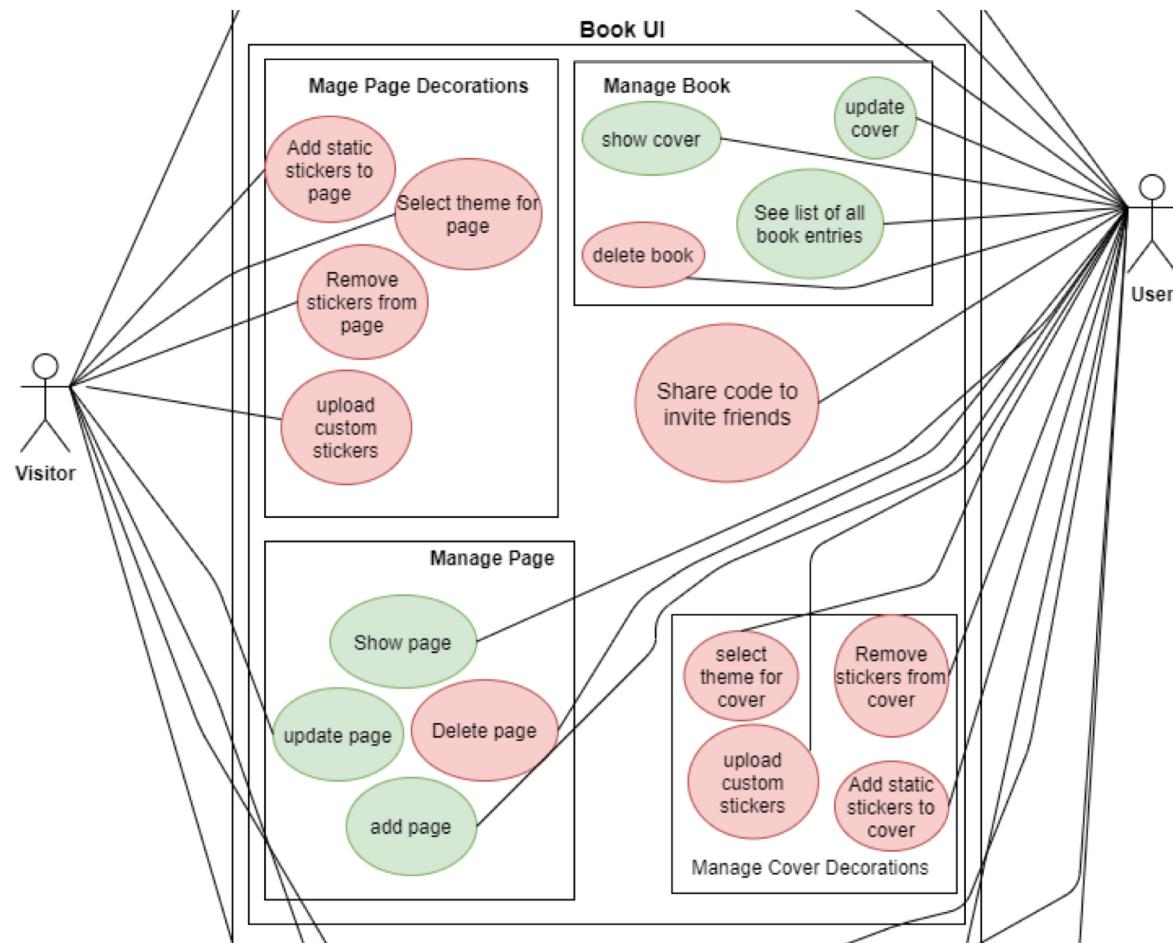
USE CASES



EXTEND OF FUNCTIONALITY



EXTEND OF FUNCTIONALITY



USE CASES

- Manage page
 - **C: create page entry (UC 1)**
 - **R: read an entry (UC 2)**
 - **U: update/edit an entry (UC 3)**
 - **D: delete an entry (part of account)**
- Manage book
 - **C: create a book (UC 4)**
 - **R: show cover/title (UC 5)**
 - **U: update cover/title (UC 6)**
 - **D: delete (part of account)**

Green

Red

Bold

implemented in 2019

implemented in 2020

Use Case



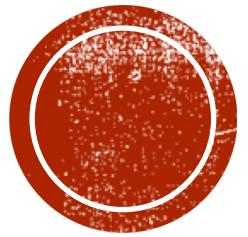
USE CASES

- **Share code to invite friends (UC 7)**
- **Manage Page Decorations (UC 8)**
 - C: create sticker
 - R: read sticker
 - U: update sticker
 - D: delete sticker
- **Navigate Footer (UC 10)**
- **Navigate Header (UC 11)**
- **Operate Account (UC 12)**

- **Manage Cover Decorations (UC 9)**
 - C: create sticker
 - R: read sticker/theme
 - U: update sticker/theme
 - D: delete sticker

Green	implemented in 2019
Red	implemented in 2020
Bold	Use Case



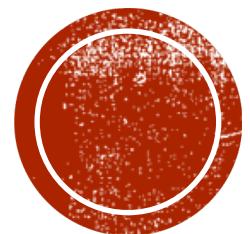


DEMO

► [Video](#)

BOOKLY - friendship book





PROJECT MANAGEMENT



RUP- RATIONAL UNIFIED PROCESS

- **Alexandra**

- **Requirements Specifier:** Describe and phrase the features / design of the website with the team
- **Project Manager:** Phrase acceptance criteria of the features + Coordinate sprints + Manage YouTrack + Assign Tags

- **Jeanne**

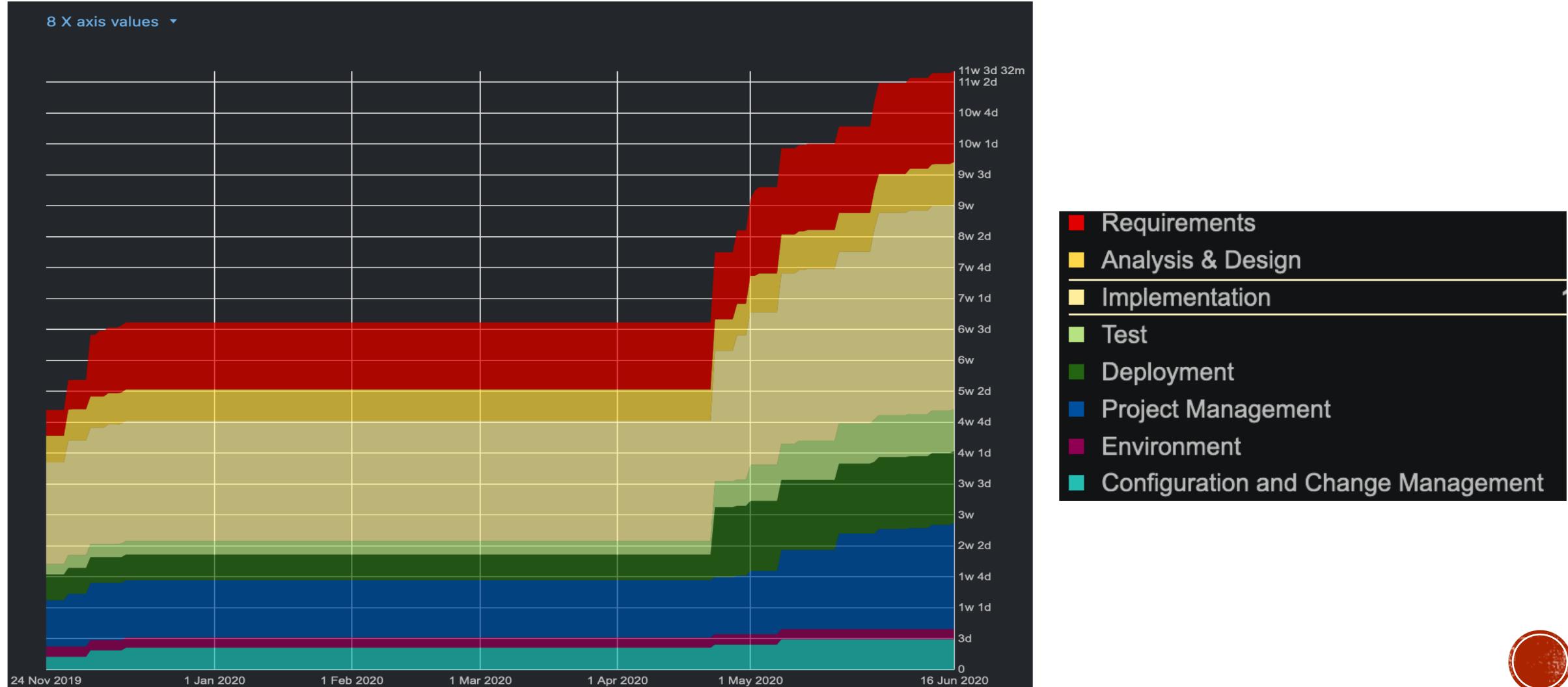
- **Implementer:** Create database structure, logic, and interfaces (API)
- **Tester:** Test the interfaces (APIs) between Frontend, Backend, Database and the functionality of the Frontend

- **Nico**

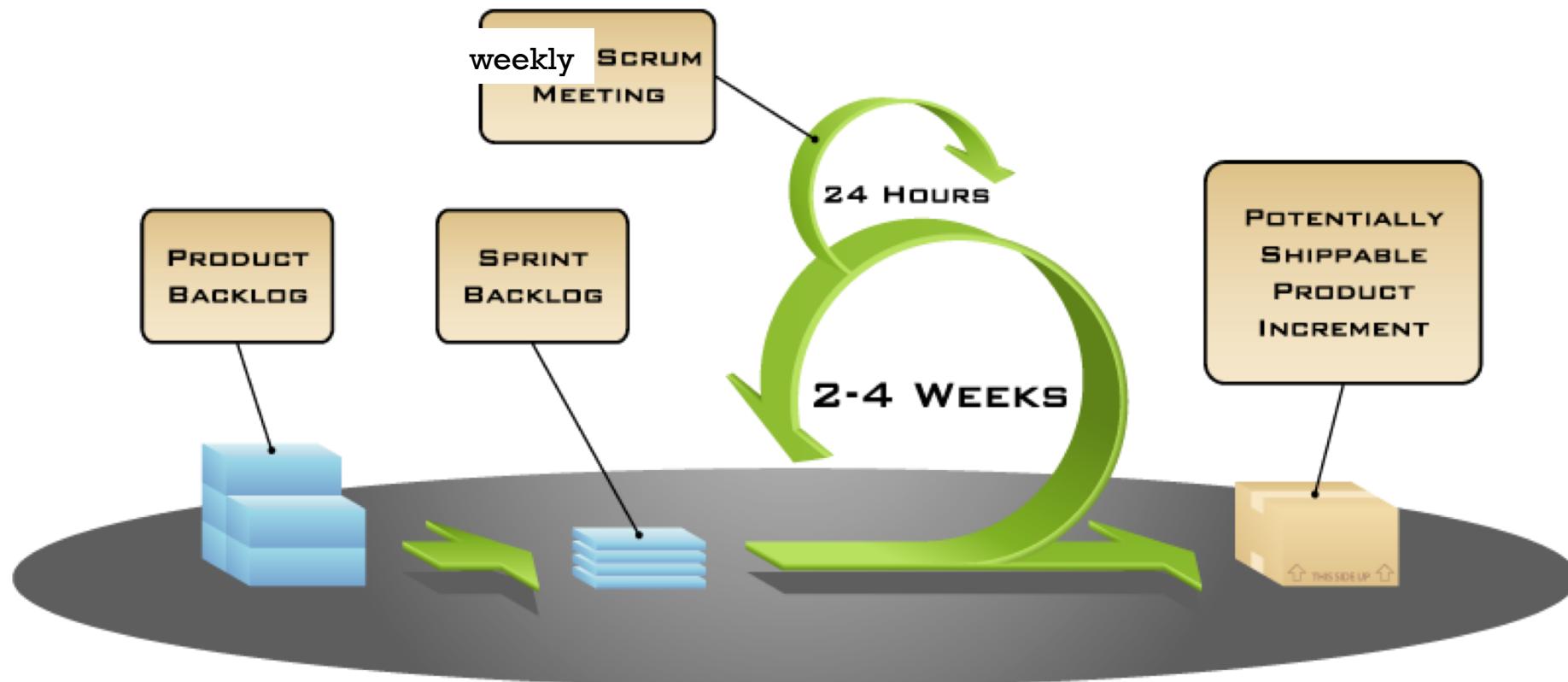
- **Deployer:** Host domain and integrate CI (and Docker)
- **Designer:** Implement the design described in the Use-Cases + Web tests



RUP- WORKFLOW - COMPLETE



SCRUM AND YOUTRACK



COPYRIGHT © 2005, MOUNTAIN GOAT SOFTWARE



SCRUM AND YOUTRACK

The screenshot shows the YouTrack Agile Boards interface. At the top, there is a navigation bar with links for Issues, Dashboards, Agile Boards, Reports, Projects, Create, and a user icon. Below the navigation is a search bar with the placeholder "Filter cards on the board" and a search icon. The main area displays a Scrum board for the project "bookly Project Develop...". The board has three columns: "Sprint 1" (14 — 28 Nov), "Code Review", and "Testing". The "Code Review" column has a progress bar with one blue segment and one green dot. Below the columns are status filters: Open, In Progress, Code Review, Testing, and Done. The "Done" status shows a green circle with the text "1w2d1h10m". The "Uncategorized Cards" section is expanded, showing three empty columns with plus signs for adding new cards. To the right, two cards are listed: "BOOKLY-45 Update use case diagramm" and "BOOKLY-14 create Read.md". Each card includes a user icon (JH), a green dot (Normal), a blue dot (Task), a red dot (Not estimated), and a time estimate (10m for the first, 1h for the second). A red circular badge is visible in the bottom right corner.

Issues | Dashboards Agile Boards Reports Projects Create

bookly Project Develop... Filter cards on the board

Sprint 1 14 — 28 Nov

Open In Progress 0m Code Review 0m Testing 0m Done 1w2d1h10m

Uncategorized Cards 14 cards

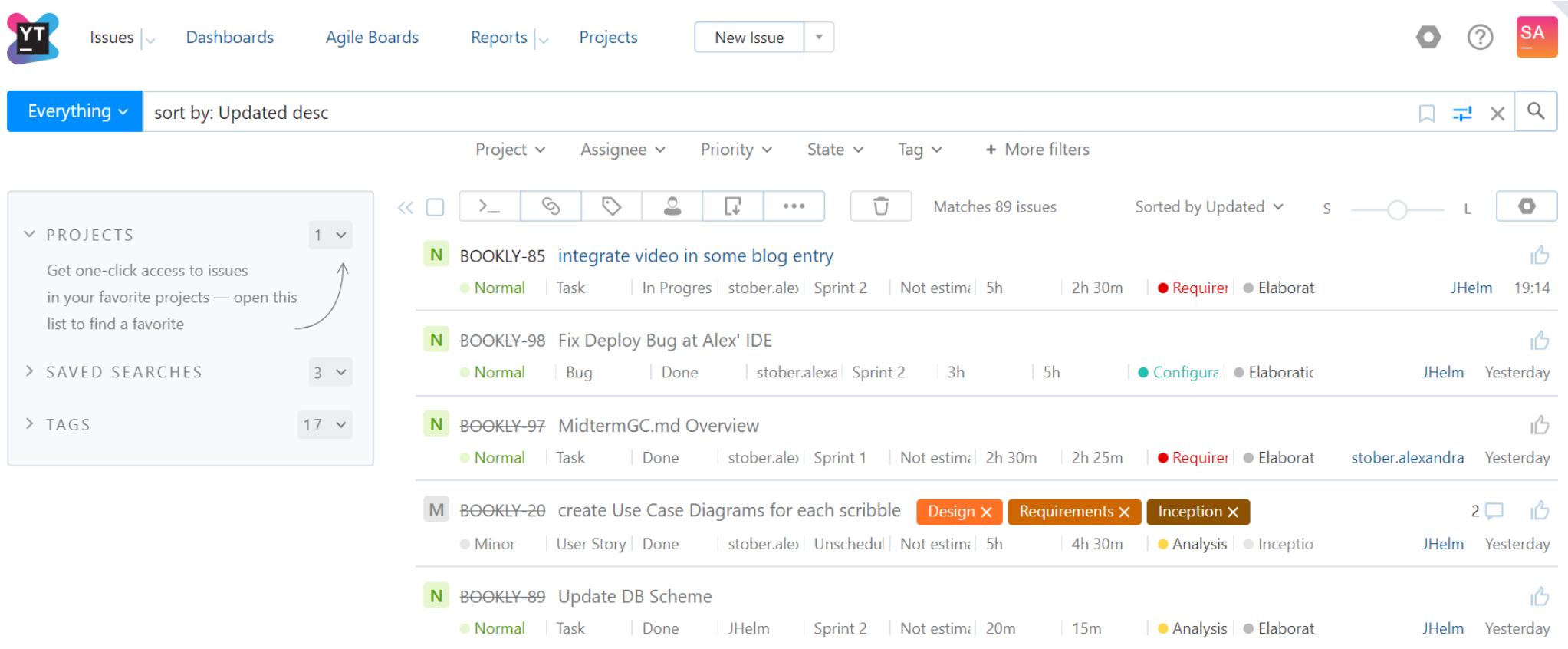
BOOKLY-45 Update use case diagramm

JH Normal Task Not estimated 10m

BOOKLY-14 create Read.md

JH Normal Task Not estimated 1h

SCRUM AND YOUTRACK



The screenshot shows the YouTrack interface with the following details:

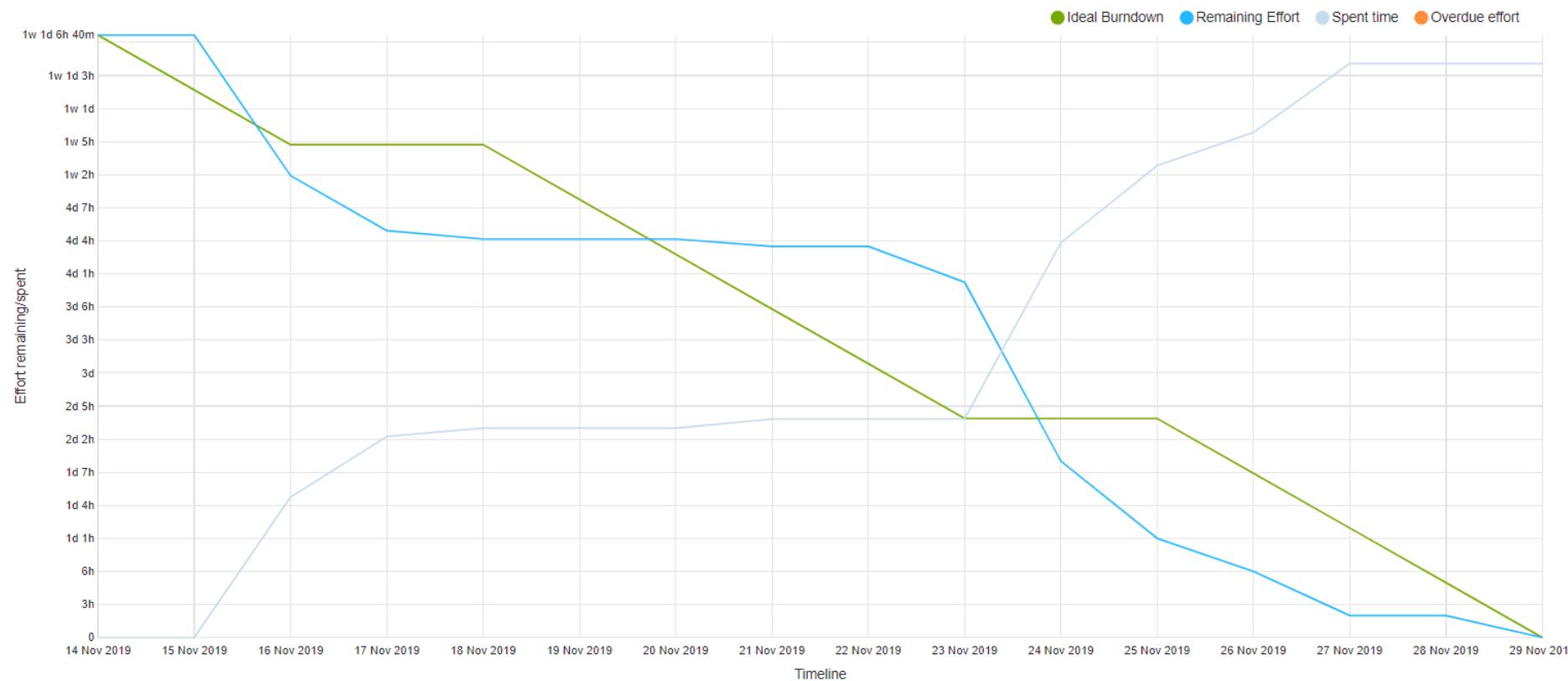
- Header:** Issues, Dashboards, Agile Boards, Reports, Projects, New Issue, and various icons for settings and help.
- Search Bar:** Everything, sort by: Updated desc, and a search bar with filters (Project, Assignee, Priority, State, Tag, More filters).
- Left Sidebar:** Projects (1), Saved Searches (3), and Tags (17).
- Issue List:** Matches 89 issues, Sorted by Updated. The list includes the following items:

 - BOOKLY-85** integrate video in some blog entry: Normal Task, In Progress, stober.alex, Sprint 2, Not estimated, 5h, 2h 30m, Requirer, Elaboration, JHelm, 19:14.
 - BOOKLY-98** Fix Deploy Bug at Alex' IDE: Normal Bug, Done, stober.alex, Sprint 2, 3h, 5h, Configura, Elaborative, JHelm, Yesterday.
 - BOOKLY-97** MidtermGC.md Overview: Normal Task, Done, stober.alex, Sprint 1, Not estimated, 2h 30m, 2h 25m, Requirer, Elaboration, stober.alexandra, Yesterday.
 - BOOKLY-20** create Use Case Diagrams for each scribble: Minor User Story, Done, stober.alex, Unscheduled, Not estimated, 5h, 4h 30m, Analysis, Inception, JHelm, Yesterday. Labels: Design X, Requirements X, Inception X.
 - BOOKLY-89** Update DB Scheme: Normal Task, Done, JHelm, Sprint 2, Not estimated, 20m, 15m, Analysis, Elaboration, JHelm, Yesterday.



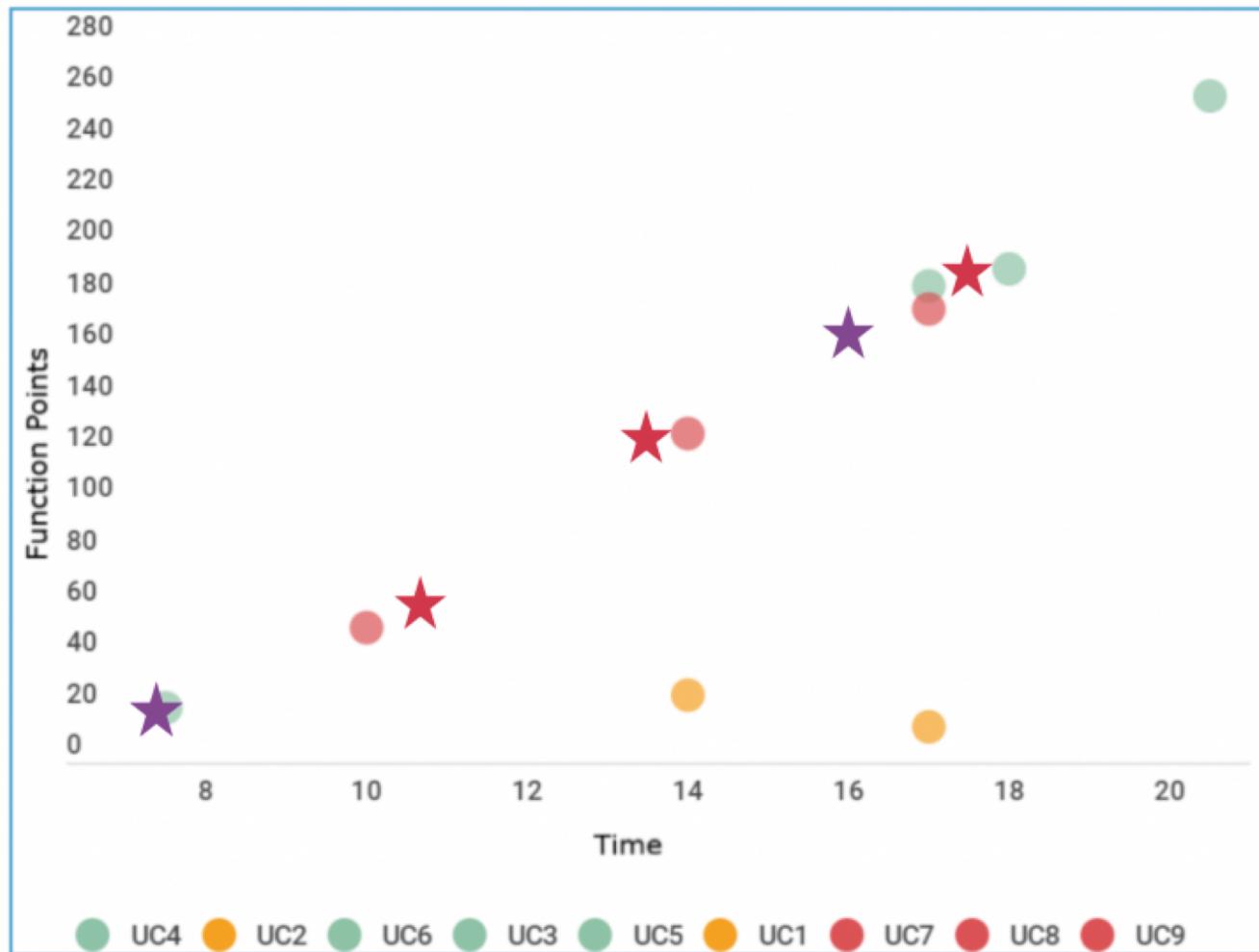
BURN-DOWN-CHART

► [more](#)



COST ESTIMATION

- ★ Real time spent on new UCs without estimated FPs
- ★ Real time spent on new UCs with estimated FPs
- UCs 2020
- disregarded UCs
- UCs 2019

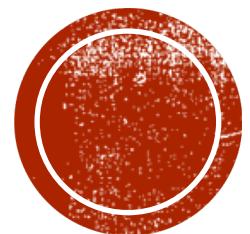


RISK MANAGEMENT

		1 to 100	1 to 10			
Risk Name	Risk Description	Risk Probability of Occurrence	Risk Impact	Risk Factor	Risk Mitigation	Person in Charge of Tracking
Time	not enough time to fullfill every task	50%	9	4,5	finish work ahead of time for the next week; keep up to date with team members	Everyone
Technical difficulties	difficult bugs or laptop gets lost	65%	6	3,9	back up laptop/ code; get support from team members; look for alternative implementation	Everyone
Code quality	bad coding slows or breaks the application	30%	4	1,2	Code reviews; Ide code checks; Testing	Nico
Internet Provider	no internet connection due to unitymedia	25%	3	0,75	conntact Internet Provider to fix issue	Alexandra
Corona	2 Weeks of sick leave because of quarantine	10%	2	0,2	stay at home; team picks up tasks of sick person	Everyone
Hacked Server	server gets hacked by third party	9%	6	0,54	HTTPS; secure passwords	Jeanne
Database Corruption	Database gets accessed by third party	5%	4	0,2	Secured by strong password; setup new database if data gets corrupted	Alexandra

Risk Management ► [here](#)

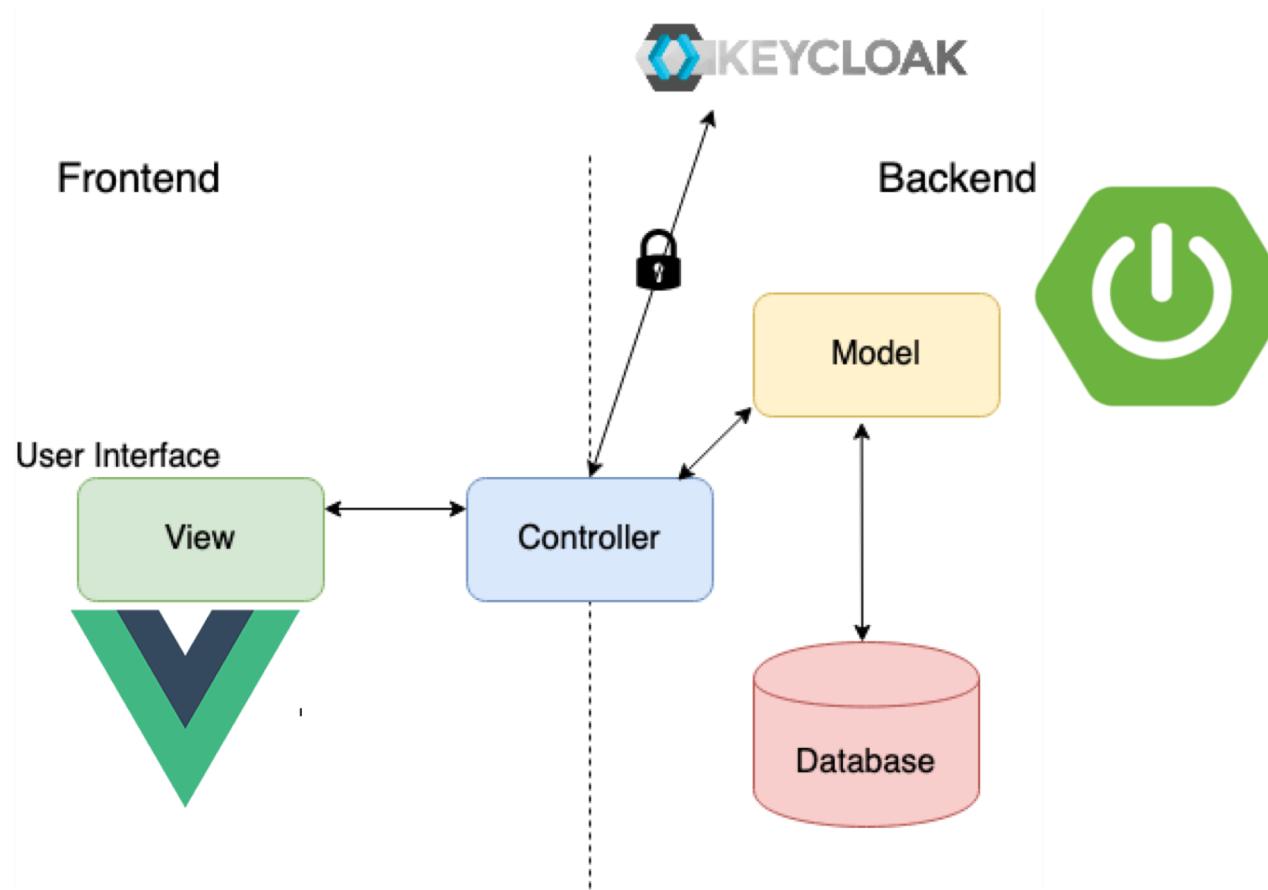




ARCHITECTURE



OVERVIEW



TECHNOLOGY

Backend



PostgreSQL



Maven™

Frontend



Maven™

Testing



JUnit

Maven™

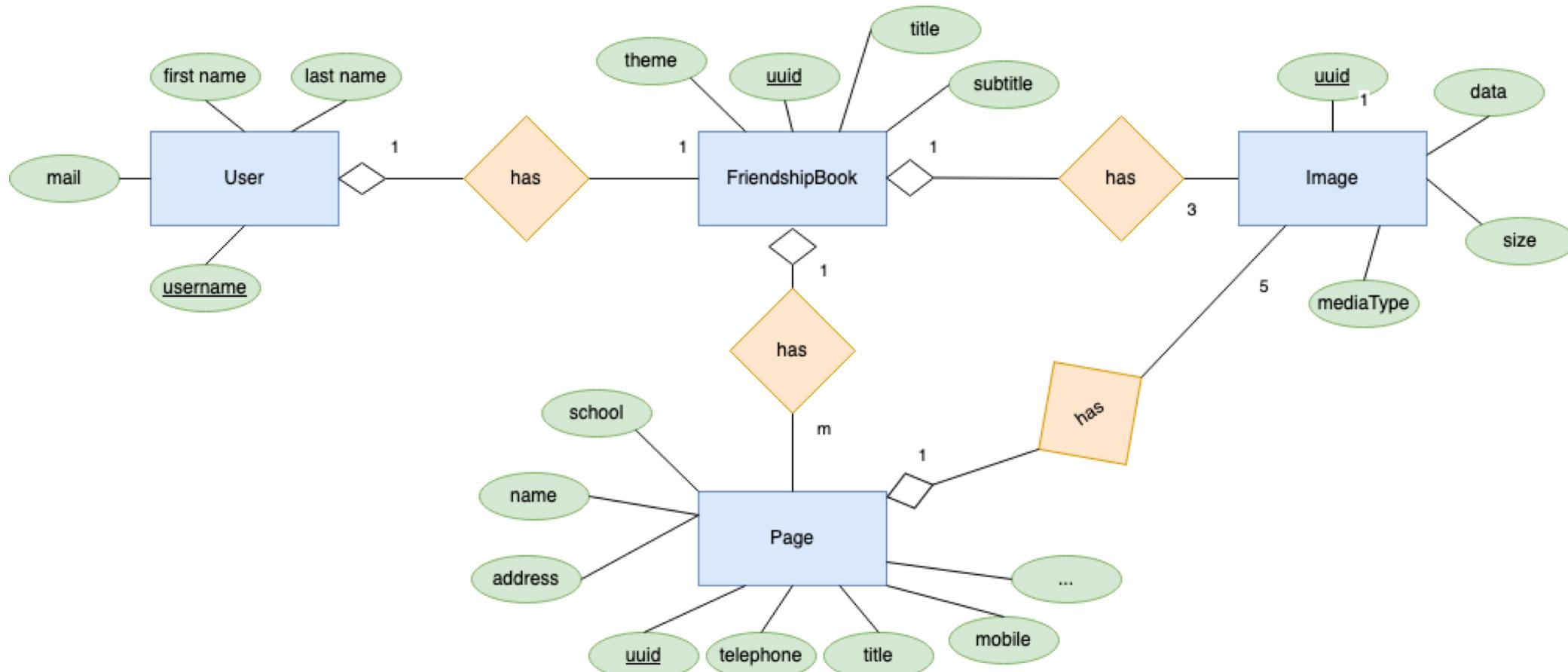
Deployment



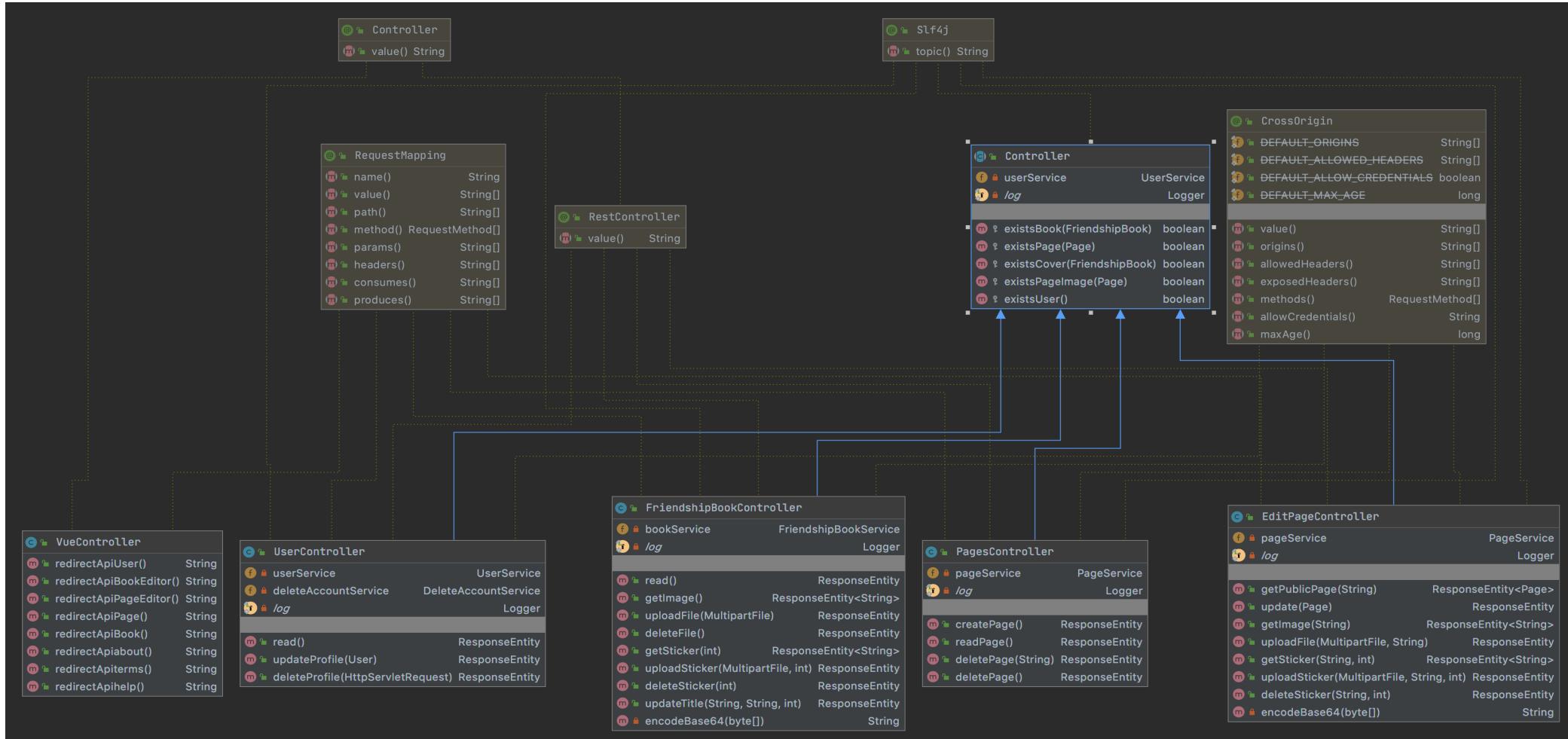
Maven™



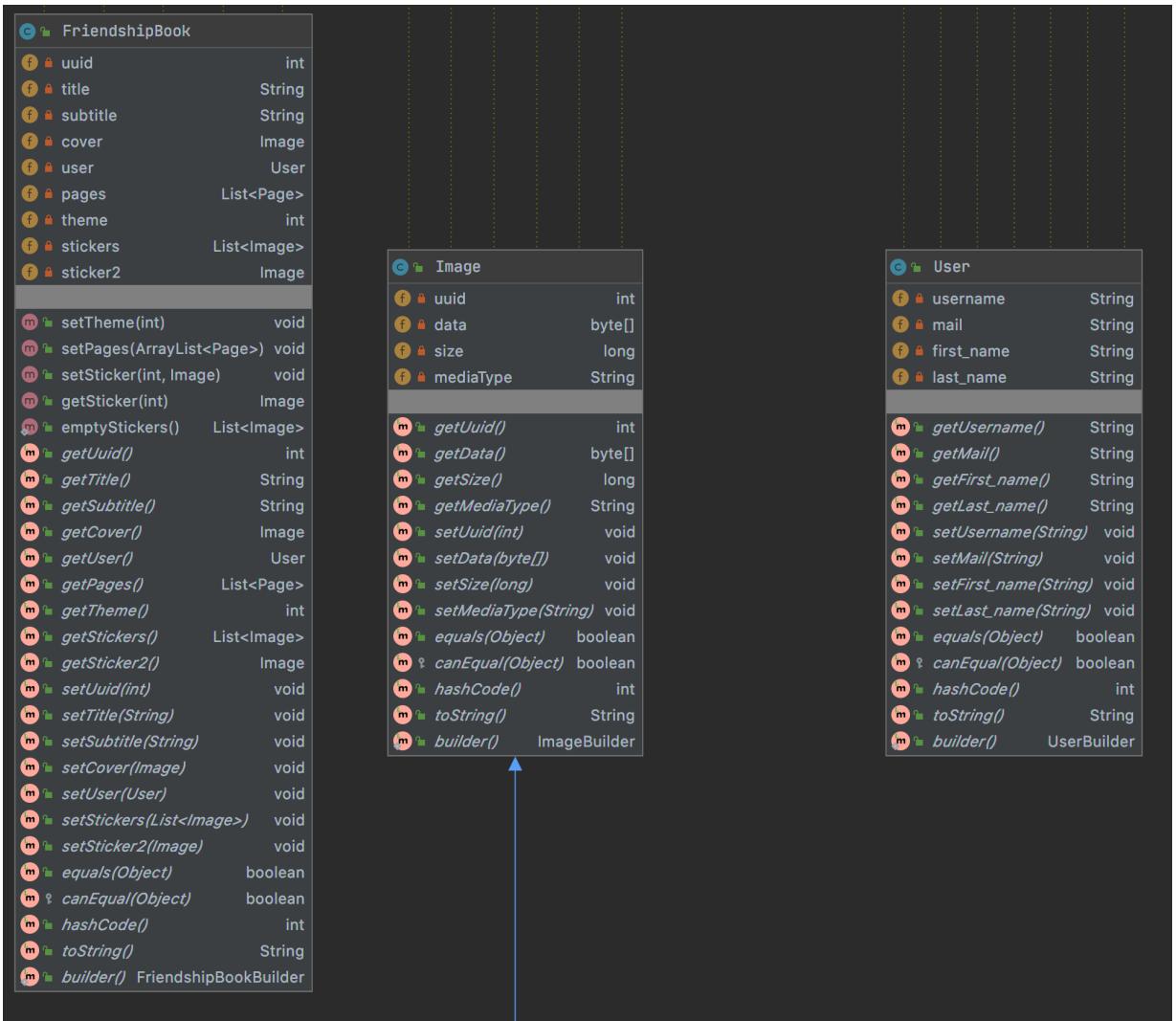
DATABASE SCHEME



CLASS DIAGRAM - CONTROLLER



C D M
 L I O
 A A D
 S G E
 S R L
 A
 M

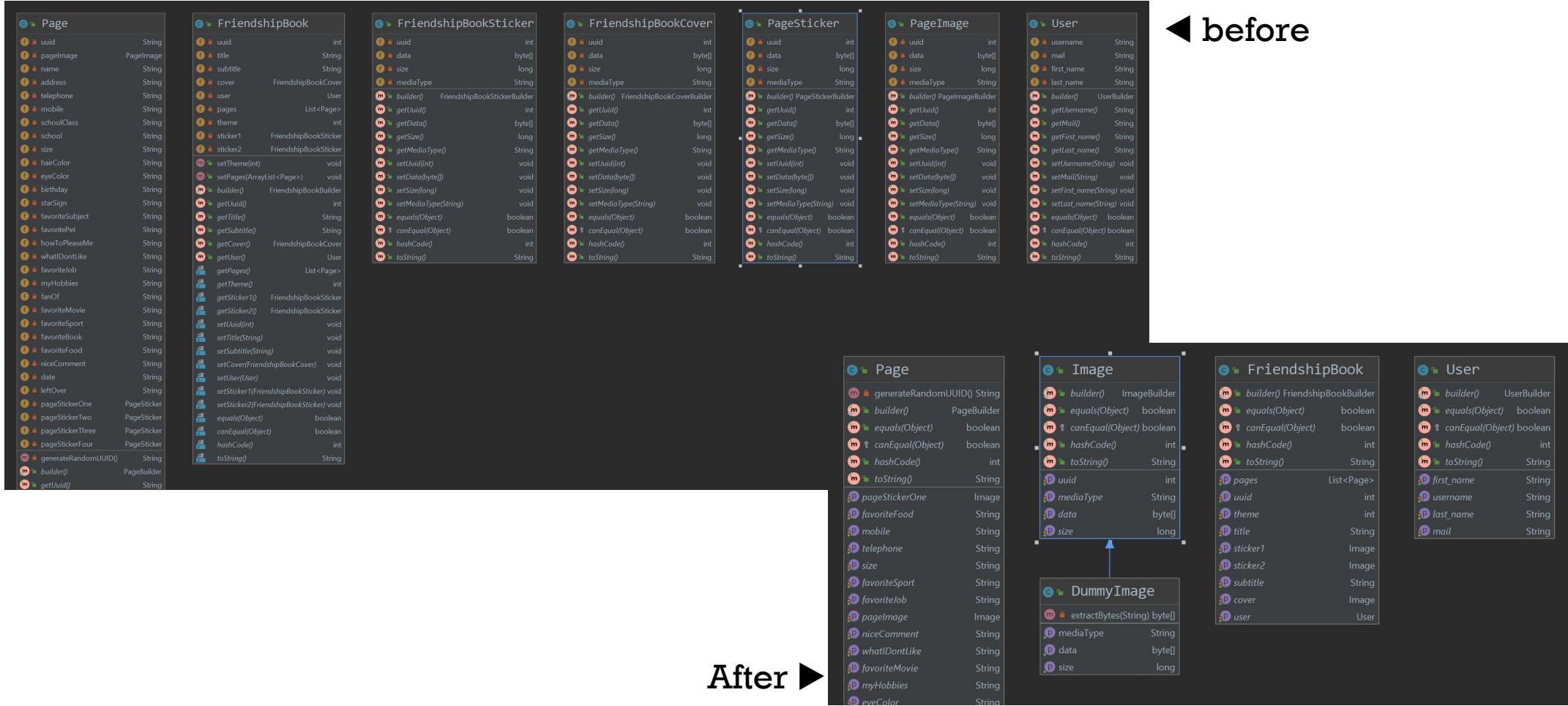


► Whole view

Page
 uid String
 pageImage Image
 name String
 address String
 telephone String
 mobile String
 schoolClass String
 school String
 size String
 hairColor String
 eyeColor String
 birthday String
 starSign String
 favoriteSubject String
 favoritePet String
 howToPleaseMe String
 whatIWantLike String
 favoriteJob String
 myHobbies String
 favoriteString String
 favoriteMovie String
 favoriteSport String
 favoriteBook String
 favoriteFood String
 niceComment String
 date String
 leftOver String
 stickers List<Image>
 generateRandomUUID() String
 setSticker(int, Image) void
 getSticker(int) Image
 emptyStickers() List<Image>
 getUuid() String
 getPageImage() Image
 getName() String
 getAddress() String
 getTelephone() String
 getMobile() String
 getSchoolClass() String
 getSchool() String
 getSize() String
 getHairColor() String
 getEyeColor() String
 getBirthday() String
 getStarSign() String
 getFavoriteSubject() String
 getFavoritePet() String
 getHowToPleaseMe() String
 getWhatIWantLike() String
 getMyHobbies() String
 getFan() String
 getFavoriteMovie() String
 getFavoriteSport() String
 getFavoriteBook() String
 getFavoriteFood() String
 niceComment() String
 getDate() String
 leftOver() String
 stickers() List<Image>
 setUuid(String) void
 setPageImage(Image) void
 setName(String) void
 setAddress(String) void
 setTelephone(String) void
 setMobile(String) void
 setSchoolClass(String) void
 setSchool(String) void
 setSize(String) void
 setEyeColor(String) void
 setBirthday(String) void
 setStarSign(String) void
 setFavoriteSubject(String) void
 setFavoritePet(String) void
 setHowToPleaseMe(String) void
 setWhatIWantLike(String) void
 setFavoriteJob(String) void
 setMyHobbies(String) void
 setFan() void
 setFavoriteMovie(String) void
 setFavoriteSport(String) void
 setFavoriteBook(String) void
 setFavoriteFood(String) void
 niceComment(String) void
 setDate(String) void
 leftOver(String) void
 stickers(List<Image>) void
 equals(Object) boolean
 canEqual(Object) boolean
 hashCode() int
 toString() String
 builder() PageBuilder



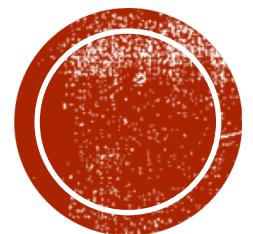
DESIGN PATTERNS: NULL OBJECT



CONTINUOUS INTEGRATION

What	Code	Push to GitLab	Test Application	Metrics	Build Application	Deploy to Server
When	continuously	Code is ready	after	after	after	after + we decide
Where			    	   	dev-system or stage-system	





QUALITY



TESTING

- Automatic testing
- Test results visible on GitLab per commit
- BDD tests with cucumber
- JUnit tests
- Database tests
- Field Test
- **Test Coverage**

coverage 30.00%

```
Feature: As a registered User
  I want to manage my friendship book

  Scenario: successfully logged in and getting user data
    When I login with "test-user""resu"
    Then The response is the test user

  Scenario: Delete friendship book cover
    When I login with "test-user""resu"
    And I navigate to friendship book
    When User friendship book cover is loaded
    Then I delete friendship book cover

  Scenario: Browse through friendship book
    When I login with "test-user""resu"
    And I navigate to friendship book
    When User friendship book cover is loaded
    Given User pages are loaded

  Scenario: Create new friendship book cover
    When I login with "test-user""resu"
    And I navigate to profile
    When I can create a friendship book cover

  Scenario: Edit friendship book cover
    When I login with "test-user""resu"
    And I navigate to friendship book
    When I can create a friendship book cover
```

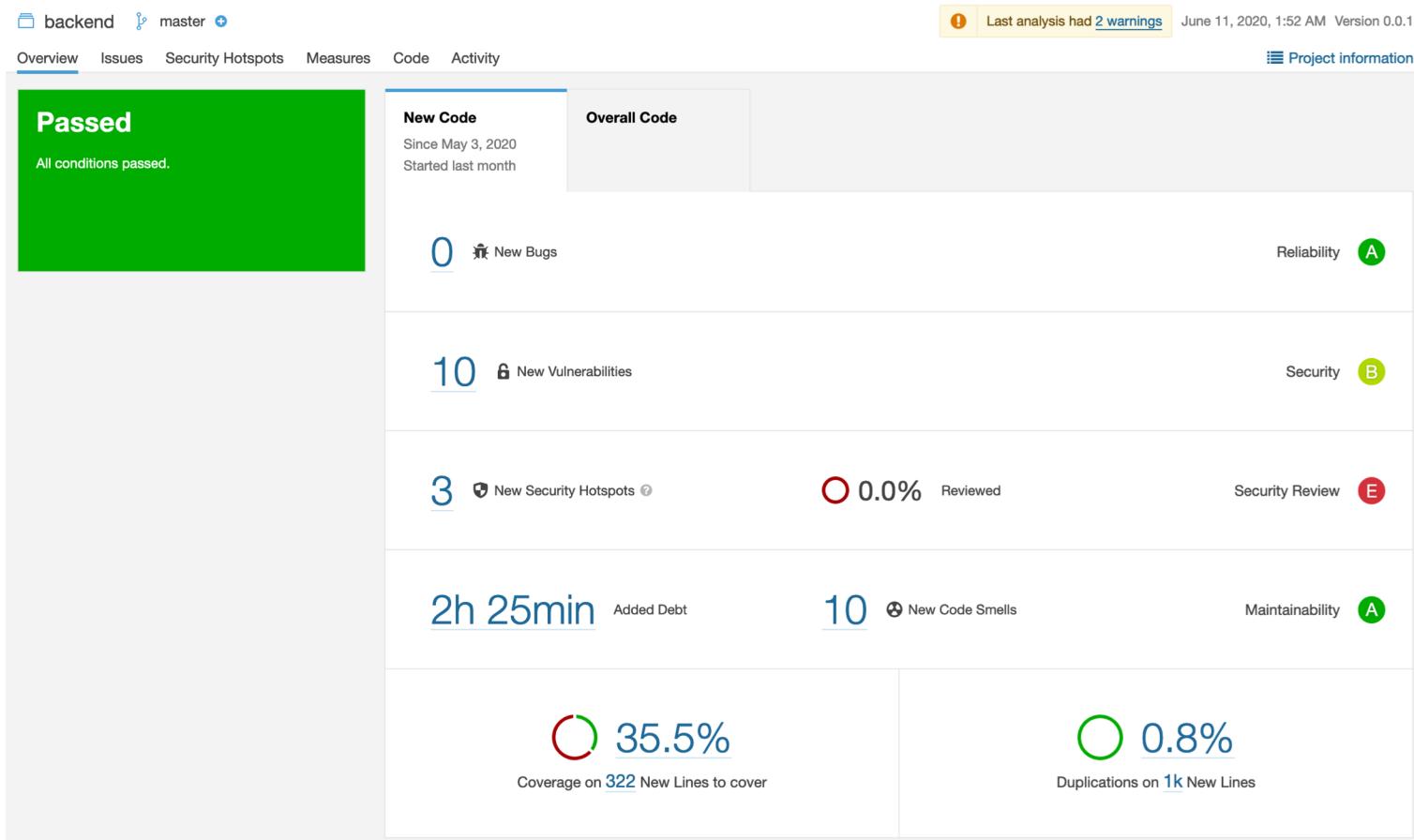


METRICS

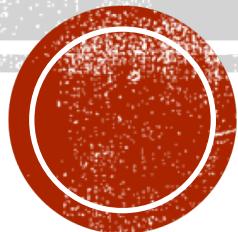


Project Overview

- > Reliability
- > Security
- > Security Review
- > Maintainability
- > Coverage
- > Duplications**
 - Overview
 - On new code
 - Density 0.8%
 - Duplicated Lines 8
 - Duplicated Blocks 2
 - Overall
 - Density 1.0%
 - Duplicated Lines 22
 - Duplicated Blocks 2
 - Duplicated Files 1
- > Size
- > Complexity**
 - Cyclomatic Complexity 178
 - Cognitive Complexity 107
- > Issues



FURTHER QUESTIONS?



BOOKLY DOCUMENTATION

- Stage System
 - bookly.online
- Documentation
 - [SRS](#)
 - [SAD](#)
 - [Swagger](#)
 - [Testplan](#)
- Project Management and Quality
 - [YouTrack](#)
 - [Blog](#)
 - [GitLab](#)
 - [SonarQube](#)
 - [Gantt Chart Time Estimation](#)
- [Demo](#)

