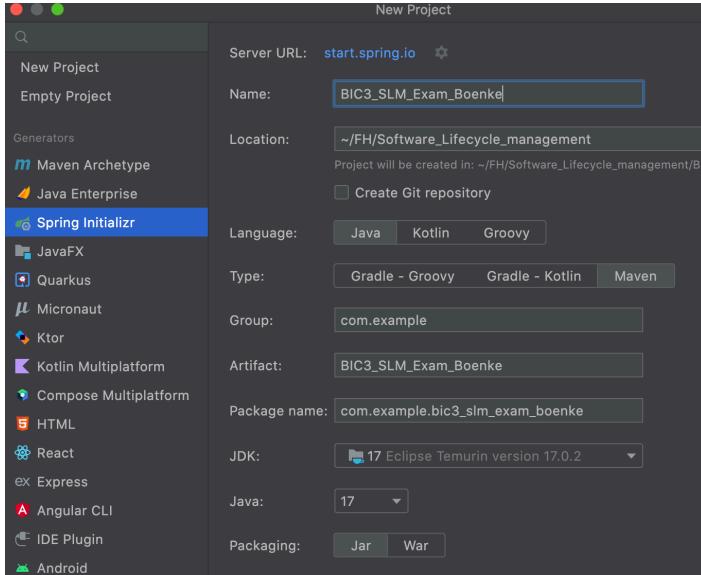


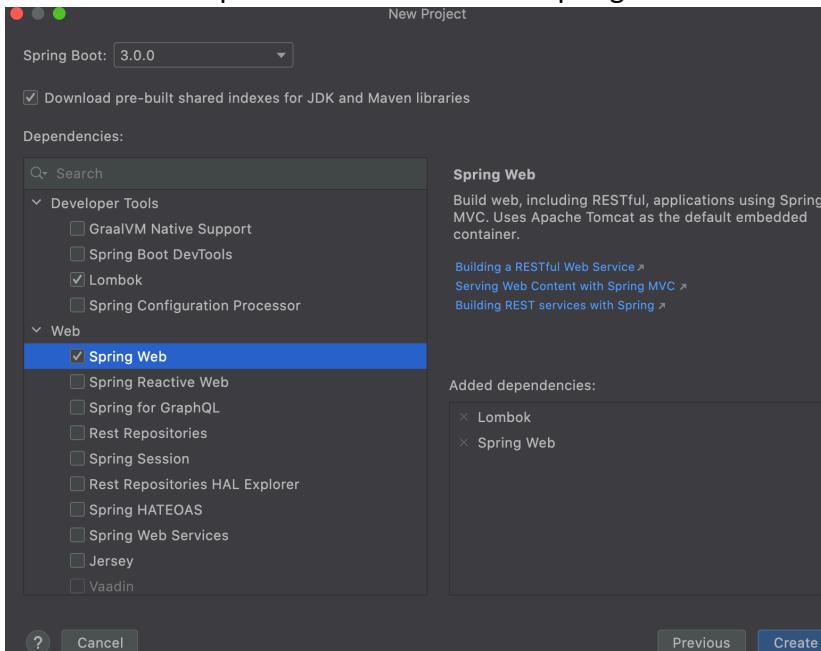
21.12.2022 Software Lifecycle Management Exam

Thorina Boenke ic21b060

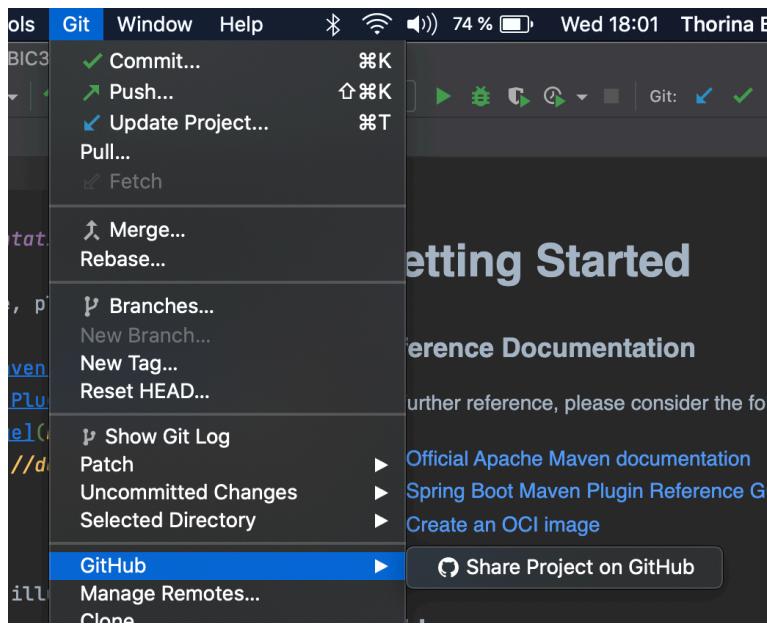
1. IntelliJ new project erstellen, mit Spring Initializer, Java, Maven, JDK 17 und Jar Packaging



2. Select dependencies Lombok and Spring Web



3. Publish on github



In Github:

A screenshot of a GitHub repository page. The URL is https://github.com/thorinaboenke/Boenke_BIC3_SLM_Exam. The repository is public. The commit history shows a single commit from 'thorinaboenke' titled 'Initial commit' at 'd85a393' (now). The commit details are as follows:

File	Message	Time
.mvn/wrapper	Initial commit	now
src	Initial commit	now
.gitignore	Initial commit	now
mvnw	Initial commit	now
mvnw.cmd	Initial commit	now
pom.xml	Initial commit	now

Below the commit history, there's a note: 'Help people interested in this repository understand your project by adding a README.' and a green button 'Add a README'.

Github URL: https://github.com/thorinaboenke/Boenke_BIC3_SLM_Exam

Create project to create a Kanban Board:

Welcome to the all-new projects

Built like a spreadsheet, project tables give you a live canvas to filter, sort, and group issues and pull requests. Tailor them to your needs with custom fields and saved views.

Learn more

Q is:open

Link a project

New project

Create a new project for this repository

Create User Stories on the Kanban Board.
The status an issue can have are "To Do" "In Progress", "Done".

Starting view of issues in the KanbanBoard:

@thorinaboenke's untitled project

View 1 + New view

Filter by keyword or by field

Todo	In Progress	Done
<p>As a user I want to call the API with a percentage value and want to receive back the calculated grade.</p> <p>Boenke_BIC3_SLM_Exam #2</p> <p>As a developer I want to test an then implement the API path /api/mark</p> <p>Boenke_BIC3_SLM_Exam #3</p> <p>As a developer I want to implement a continous integration and continous delivery pipeline (CI/CD) with Maven and Github Actions</p> <p>Boenke_BIC3_SLM_Exam #4</p> <p>As a developer i want to create a documentation for the whole project</p>	+ Add item	+ Add item

User Stories in Detail:

Boenke_BIC3_SLM_Exam #1

As a user I want to call the API with a percentage value and want to receive back the calculated grade. #1

Open thorinaboenke opened about 1 hour ago

thorinaboenke 50 minutes ago

Edit

Assignees Add assignees...

Labels Add labels...

Milestone Add milestone...

Status Done

Show all fields

REQ-Definition:

- A message is always returned when the user opens the URI path `api/mark?percentage=<inputValue>`.
- The message content is the grade corresponding to the entered percentage value
- The message must not be empty
- The message content must be in a human readable format
- If invalid parameters are provided, the user should be informed with the message "invalid input"

Complete Grading Scheme:

= 88% 1
= 75% and < 88% 2
= 63% and < 75% 3
= 50% and < 63% 4
< 50% 5

Example:

Open `api/mark?percentage=49` => "5"

or

`api/mark?percentage=80` => "2"

Boenke_BIC3_SLM_Exam #2

Open in new tab

Copy link

Archive

Delete from project

As a developer I want to test an then implement the API path /api/mark #2

Open thorinaboenke opened about 1 hour ago

thorinaboenke 49 minutes ago

Edit

Assignees Add assignees...

Labels Add labels...

Milestone Add milestone...

Status Done

Show all fields

Testcases:

`GetMark` => Returns Status OK and calculated grade as content
`GetMarkWithInvalidParamter` => Returns Status OK and content 'invalid input'



Set issues #1 and #2 auf In Progress

The screenshot shows a Jira board with two columns: "Todo" and "In Progress".

Todo Column:

- Boenke_BIC3_SLM_Exam #3
As a developer I want to implement a continuous integration and continuous delivery pipeline (CI/CD) with Maven and Github Actions
- Boenke_BIC3_SLM_Exam #4
As a developer i want to create a documentation for the whole project

In Progress Column:

- Boenke_BIC3_SLM_Exam #1
As a user I want to call the API with a percentage value and want to receive back the calculated grade.
- Boenke_BIC3_SLM_Exam #2
As a developer I want to test an then implement the API path /api/mark

In IntelliJ, switch to feature branch “feat/implement-and-test-getMark”
create new package controller and class MarkController to implement the REST Route
implement Rest Route:

```
import org.springframework.web.bind.annotation.RestController;
@RestController
public class MarkController {
    @GetMapping("/api/mark")
    public String returnMark(@RequestParam String percentage) {
        Integer percent ;
        try
        {
            percent = Integer.parseInt(percentage);
        }
        catch (NumberFormatException e)
        {
            return "invalid input";
        }
        if (percent > 100) {...}
        else if (percent < 50 ) {
            return "5";

        } else if (percent < 63 ) {
            return "4";

        }else if (percent < 75) {
            return "3";

        }else if (percent < 88) {
            return "2";
        }
        else {
            return "1";
        }
    }
}
```

Commit file.

Create Class for Tests:

```
class MarkControllerTest {  
    @Test  
    void returnMark() {  
        // arrange (z.B. einen Instanz der Klasse erstellen)  
        var controller = new MarkController();  
        // act (Methode mit Testdaten ausführen)  
        var eins : String = controller.returnMark( percentage: "89");  
        var zwei : String = controller.returnMark( percentage: "76");  
        var drei : String = controller.returnMark( percentage: "65");  
        var vier : String = controller.returnMark( percentage: "52");  
        var fünf : String = controller.returnMark( percentage: "3");  
        // assert (Vergleichen ob Ergebnis mit Erwartung übereinstimmt)  
        assertEquals( expected: "1", eins);  
        assertEquals( expected: "2", zwei);  
        assertEquals( expected: "3", drei);  
        assertEquals( expected: "4", vier);  
        assertEquals( expected: "5", fünf);  
    }  
  
    @Test  
    void returnErrorMessageOnInvalidInput() {  
        // arrange (z.B. einen Instanz der Klasse erstellen)  
        var controller = new MarkController();  
        // act (Methode mit Testdaten ausführen)  
        var error : String = controller.returnMark( percentage: "cthulu");  
        // assert (Vergleichen ob Ergebnis mit Erwartung übereinstimmt)  
        assertEquals( expected: "invalid input", error);  
    }  
}
```

Commit file

Push to remote repository.

Create pull request.

A screenshot of a GitHub repository page for 'thorinaboenke / Boenke_BIC3_SLM_Exam'. The repository is public. The top navigation bar includes links for Code, Issues (4), Pull requests, Actions, Projects (1), Wiki, Security, Insights, and a gear icon. Below the navigation, a section titled 'Open a pull request' instructs users to create a new pull request by comparing changes across two branches. It also mentions the ability to compare across forks. A dropdown menu shows 'base: main' and 'compare: feat/implement-and-test-get...'. A green checkmark indicates 'Able to merge. These branches can be automatically merged.' The pull request details show a title 'Feat/implement and test get mark', a description mentioning issues #1 and #2, and a note about implementing api/mark route + tests. A 'Create pull request' button is at the bottom.

Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).

A screenshot of the GitHub pull request creation interface. The title is 'Feat/implement and test get mark'. The description area contains the text: 'Relates to issues #1 and #2' and 'Implements api/mark route + tests'. At the bottom, there is a note: 'Attach files by dragging & dropping, selecting or pasting them.' A 'Create pull request' button is visible.

Merge into main branch.

A screenshot of a GitHub merge confirmation message. It shows a commit being merged into the 'main' branch. The message says: 'Pull request successfully merged and closed. You're all set—the feat/implement... branch can be safely deleted.' A 'Delete branch' button is available.

Set issues auf done (analogous procedure during work on the other issues)

A screenshot of a GitHub project board. It has three columns: 'Todo' (2 items), 'In Progress' (0 items), and 'Done' (2 items). The 'Todo' column contains items: 'Boenke_BIC3_SLM_Exam #3' (description: 'As a developer I want to implement a continuous integration and continuous delivery pipeline (CI/CD) with Maven and Github Actions') and 'Boenke_BIC3_SLM_Exam #4' (description: 'As a developer i want to create a documentation for the whole project'). The 'Done' column contains items: 'Boenke_BIC3_SLM_Exam #1' (description: 'As a user I want to call the API with a percentage value and want to receive back the calculated grade.') and 'Boenke_BIC3_SLM_Exam #2' (description: 'As a developer I want to test an then implement the API path /api/mark').

Set Ticket #3 auf In Progress:

The screenshot shows a Jira backlog board with three columns: Todo, In Progress, and Done. The In Progress column has one item selected, which is highlighted with a blue border.

- Todo**: 1 item - Boenke_BIC3_SLM_Exam #4: As a developer i want to create a documentation for the whole project
- In Progress**: 1 item - Boenke_BIC3_SLM_Exam #3: As a developer I want to implement a continuous integration and continuous delivery pipeline (CI/CD) with Maven and Github Actions
- Done**: 2 items -
 - Boenke_BIC3_SLM_Exam #1: As a user I want to call the API with a percentage value and want to receive back the calculated grade.
 - Boenke_BIC3_SLM_Exam #2: As a developer I want to test an then implement the API path /api/mark

Create branch feat/ci-cd-pipeline

Set up Github actions with Java with Maven:

Get started with GitHub Actions

Build, test, and deploy your code. Make code reviews, branch management, and issue triaging work the way you want.

Skip this and [set up a workflow yourself →](#)

Search workflows

Suggested for this repository

The screenshot shows two suggested GitHub Actions workflows for a Java project:

- Publish Java Package with Maven** By GitHub Actions: Build a Java Package using Maven and publish to GitHub Packages. [Configure](#) Java
- Java with Maven** By GitHub Actions: Build and test a Java project with Apache Maven. [Configure](#) Java

thorinaboenke merged commit `ee12aee` into `main` now
1 check passed

[View details](#) [Revert](#)

Pull request successfully merged and closed
You're all set—the `feat/ci-cd-pipe...` branch can be safely deleted.

[Delete branch](#)

Actions run successfully:

The screenshot shows a GitHub Actions build log for pull request #6 from thorinaboenke/feat/ci-cd-pipeline. The build was successful and completed 1 minute ago in 26s. The log details the following steps:

- > Set up job (3s)
- > Run actions/checkout@v3 (1s)
- > Set up JDK 17 (1s)
- > Build with Maven (12s)
- > artifact (0s)
- > Upload a build Artifact (2s)
- > Post Set up JDK 17 (3s)
- > Post Run actions/checkout@v3 (0s)
- > Complete job (0s)

Summary: Summary, Jobs, Re-run all jobs, ...

← Java CI with Maven

Merge pull request #6 from thorinaboenke/feat/ci-cd-pipeline #2

Re-run all jobs, ...

Summary, Jobs, Re-run all jobs, ...

build succeeded 1 minute ago in 26s

Search logs, ...

Set up job 3s

Run actions/checkout@v3 1s

Set up JDK 17 1s

Build with Maven 12s

artifact 0s

Upload a build Artifact 2s

Post Set up JDK 17 3s

Post Run actions/checkout@v3 0s

Complete job 0s

Set Ticket #3 to Done

The Jira board displays three columns: Todo, In Progress, and Done.

- Todo (1):** Boenke_BIC3_SLM_Exam #4
As a developer i want to create a documentation for the whole project
- In Progress (0):** None
- Done (3):**
 - Boenke_BIC3_SLM_Exam #1
As a user I want to call the API with a percentage value and want to receive back the calculated grade.
 - Boenke_BIC3_SLM_Exam #2
As a developer I want to test an then implement the API path /api/mark
 - Boenke_BIC3_SLM_Exam #3
As a developer I want to implement a continuous integration and continuous delivery pipeline (CI/CD) with Maven and Github Actions

View 1, + New view, Filter by keyword or by field

Work on Documentation

View 1 + New view

Filter by keyword or by field

Todo 0	In Progress 1	Done 3
	<p>Boenke_BIC3_SLM_Exam #4 ... As a developer i want to create a documentation for the whole project</p>	<p>Boenke_BIC3_SLM_Exam #1 As a user I want to call the API with a percentage value and want to receive back the calculated grade.</p> <p>Boenke_BIC3_SLM_Exam #2 As a developer I want to test an then implement the API path /api/mark</p> <p>Boenke_BIC3_SLM_Exam #3 As a developer I want to implement a continous integration and continous delivery pipeline (CI/CD) with Maven and Github Actions</p>

Create branch feat/documentation
Complete Readme.md
git commit, git push
create pull request, merge

Project completed 😊 :

thorinaboenke merged commit d827001 into main now
1 check passed

[View details](#) [Revert](#)

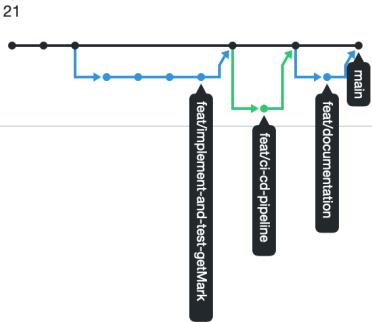
Pull request successfully merged and closed
You're all set—the feat/documentation... branch can be safely deleted.

[Delete branch](#)

View 1 + New view

Filter by keyword or by field

Todo 0	In Progress 0	Done 4
		<p>As a user I want to call the API with a percentage value and want to receive back the calculated grade.</p> <p>Boenke_BIC3_SLM_Exam #2 As a developer I want to test an then implement the API path /api/mark</p> <p>Boenke_BIC3_SLM_Exam #3 As a developer I want to implement a continous integration and continous delivery pipeline (CI/CD) with Maven and Github Actions</p> <p>Boenke_BIC3_SLM_Exam #4 ... As a developer i want to create a documentation for the whole project</p>



Branch History

main
-o Commits on Dec 21, 2022
Merge pull request #7 from thorinaboenke/feat/documentation ...
<thorinaboenke 3="" ago="" committed="" minutes="" th="" ✓<=""></thorinaboenke>
complete documentation
<thorinaboenke 5="" ago="" committed="" minutes="" th="" ✓<=""></thorinaboenke>
Merge pull request #6 from thorinaboenke/feat/ci-cd-pipeline ...
<thorinaboenke 12="" ago="" committed="" minutes="" th="" ✓<=""></thorinaboenke>
Create maven.yml
<thorinaboenke 14="" ago="" committed="" minutes="" th="" ✓<=""></thorinaboenke>
Merge pull request #5 from thorinaboenke/feat/implementation-and-test-getMark ...
<thorinaboenke 18="" ago="" committed="" minutes="" th="" ✓<=""></thorinaboenke>
implement test
<thorinaboenke 20="" ago="" committed="" minutes="" th="" ✓<=""></thorinaboenke>
implement test
<thorinaboenke 21="" ago="" committed="" minutes="" th="" ✓<=""></thorinaboenke>
implement rest route
<thorinaboenke 30="" ago="" committed="" minutes="" th="" ✓<=""></thorinaboenke>
implement rest route
<thorinaboenke 30="" ago="" committed="" minutes="" th="" ✓<=""></thorinaboenke>
Update README.md
<thorinaboenke 48="" ago="" committed="" minutes="" th="" ✓<=""></thorinaboenke>
Create README.md
<thorinaboenke 54="" ago="" committed="" minutes="" th="" ✓<=""></thorinaboenke>
Initial commit
<thorinaboenke 1="" ago="" committed="" hour="" th="" ✓<=""></thorinaboenke>

Commit History on main

Lastly: Add PDF Abgabe to repository as well (therefore not able to document this step).