

HOBBY WEB APPLICATION

BY SAYED SAKKAF

INTRODUCTION

- My name is Sayed Sakka and welcome to my Hobby Web Application project presentation!
- My initial approach for completing this application was to first thoroughly read through the project brief to understand what would be expected of in in terms of a Minimum Viable Product.
- Having been given the creative freedom to choose what to do in terms of creation of our project, I began to conceptualize ideas that I would enjoy working towards and completing for this project.

CONCEPT

- The approach I took to conceptualizing this project was to understand both where and what I enjoy doing. I have been in education for all my life, and I enjoy learning so I decided to make a student management system where course administrators can use CRUD (Create, Read, Update, Delete) functionality with the entities Student and Course.

SPRINT PLANNING (KANBAN BOARD)

- For planning, I utilized Jira's Kanban Board.
- This allowed me to add different user stories under different epics, shown below:

IN PROGRESS 2 OF 3 ISSUES

As a professor, I want to be able to enrol a student on to my course, so that I can teach it to that student.

ENROLMENT FUNCTIONS

☒ SMS-12

As a professor, I want to be able to remove a student from my course, so that when they finish my course they will no longer be enrolled to it.

ENROLMENT FUNCTIONS

☒ SMS-13

DONE 4 OF 15 ISSUES ✓

As a professor, I want to be able to add my course to the system, so then students can enrol onto it.

COURSE FUNCTIONS

☒ SMS-7 ✓

As a student, I want to be able to view all the courses in the system, so then I know what courses I can apply for.

COURSE FUNCTIONS

☒ SMS-8 ✓

As a professor, I want to be able to change the courses in the system, so then if my courses change, I can update them

COURSE FUNCTIONS

☒ SMS-9 ✓

As a professor, I want to be able to delete a course from the system, so that when I no longer teach that course, it can be removed from the system.

COURSE FUNCTIONS

☒ SMS-10 ✓

DONE 4 OF 15 ISSUES ✓ +

As an administrator, I want to be able to add student details to the system, so then I can enrol them on a course.

STUDENT FUNCTIONS

☒ SMS-3 ✓

As a professor, I want to be able to view all the students in the system, so then I can anticipate how many students I will be teaching in my course.

STUDENT FUNCTIONS

☒ SMS-4 ✓

As an administrator, I want to be able to change details on the system, so that student details are up to date and correct.

STUDENT FUNCTIONS

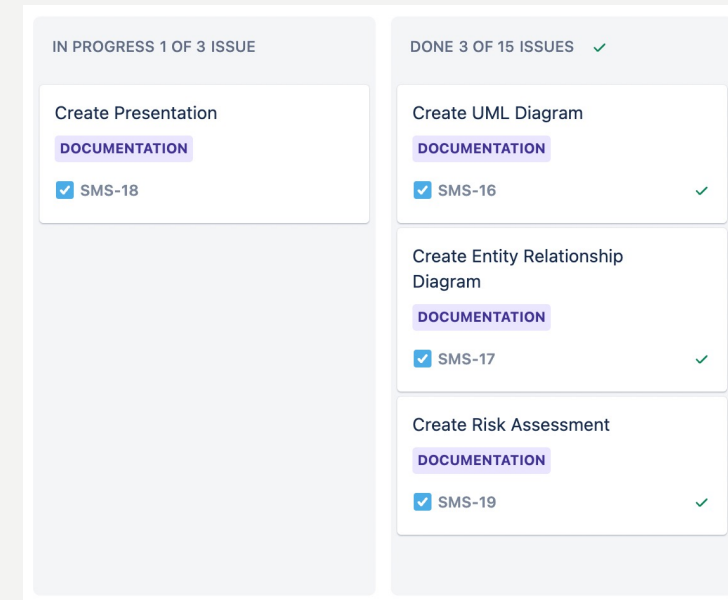
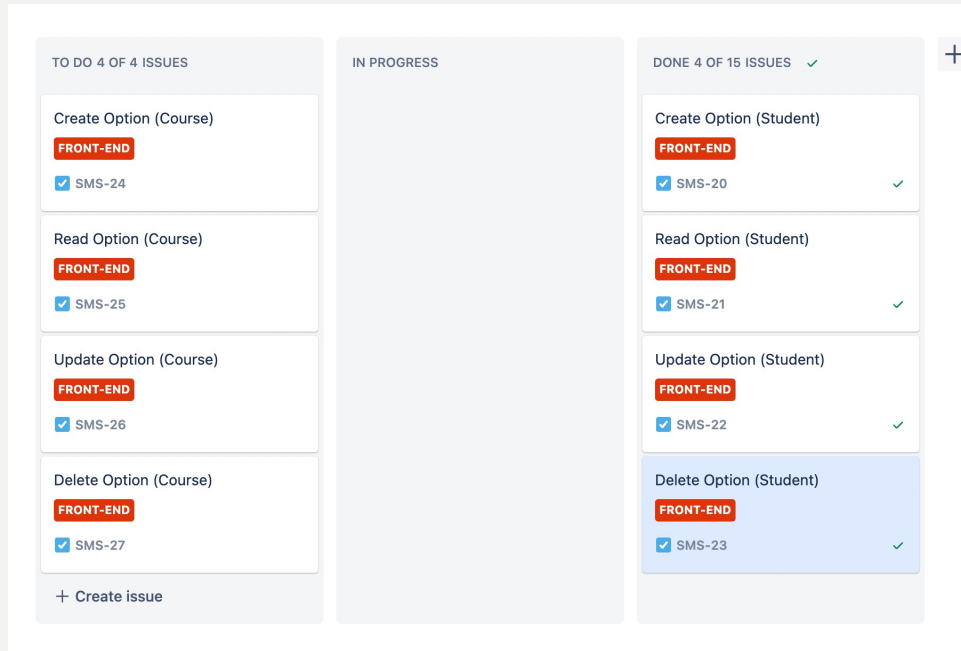
☒ SMS-5 ✓

As a professor, I want to be able to remove a student from the enrolment system, so then when they leave, they won't be enrolled on to a course.

STUDENT FUNCTIONS

☒ SMS-6 ✓

SPRINT PLANNING (KANBAN BOARD)



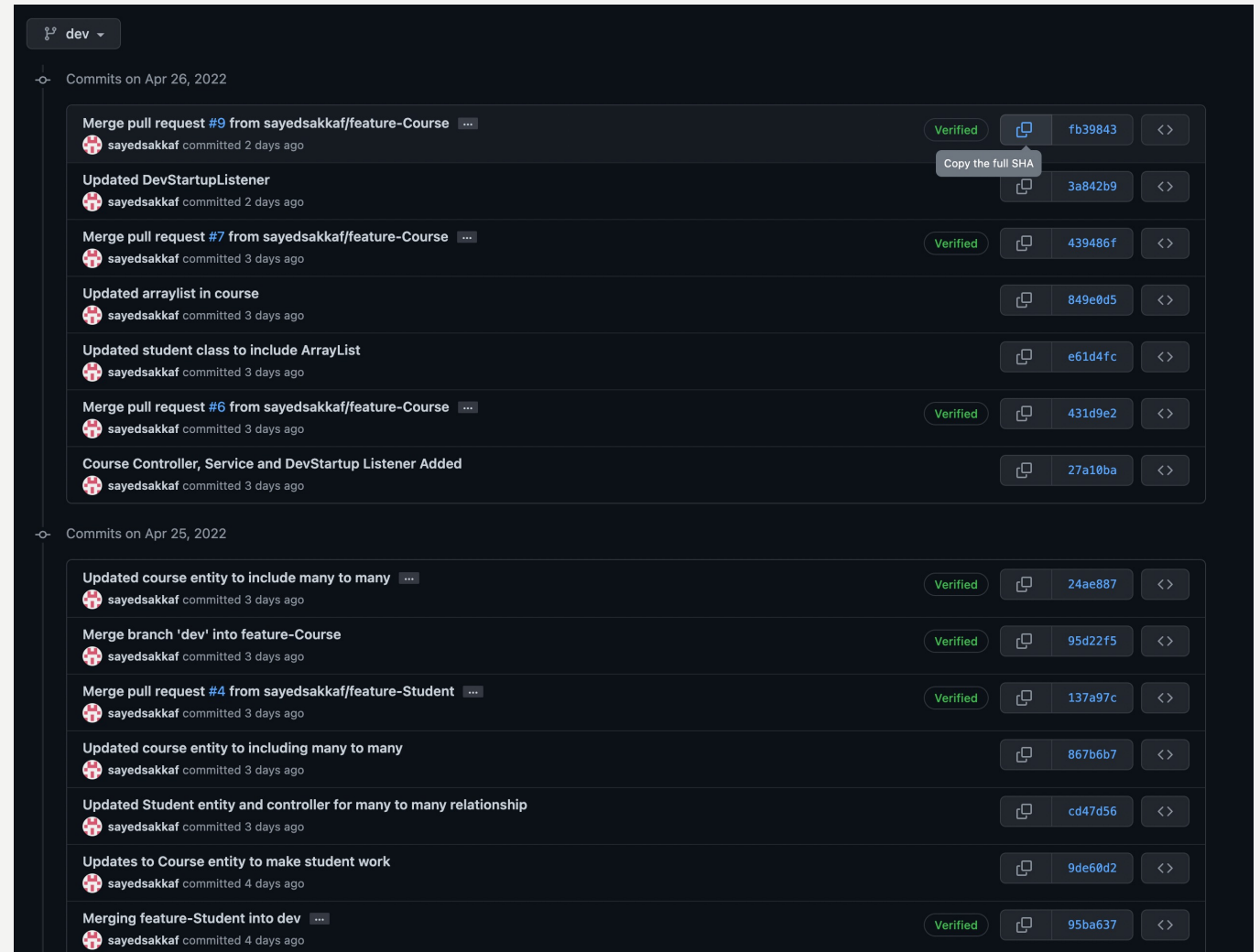
CONSULTANT JOURNEY

What have I learned so far for this project?

- **Technological constraints:**
 - Version Control: Git
 - Source Code Management: GitHub
 - Kanban Board: Jira
 - Database Management System: MySQL
 - Back-end: Java
 - API Development: Spring
 - Front-end: HTML, CSS, JavaScript
 - Build Tool: Maven
 - Unit Testing: Junit, Mockito

CONTINUOUS INTEGRATION

- For version control, I initialized the two folders I would be using for my front-end and back-end Git repositories. I then created a dev branch, and then incorporating the feature-Branch model to add new features and keep my integration clean. After having completed work on a specific feature-Branch, I would the merge the branch with dev, branching of into a new feature-Branch for a new feature.



TESTING

- Having attempted testing, I managed to achieve 63% coverage in src/main/java, I will go over this further in the demonstration

| | | | | | | |
|---|-----------------------------|------------------------|--------|-------|-----|-------|
| ▼ | 📁 student_management_system | <div><div></div></div> | 84.3 % | 2,390 | 445 | 2,835 |
| > | 📁 src/main/java | <div><div></div></div> | 64.0 % | 770 | 433 | 1,203 |
| > | 📁 src/test/java | <div><div></div></div> | 99.3 % | 1,620 | 12 | 1,632 |

DEMONSTRATION OF BACK-END AND FRONT-END

SPRINT REVIEW

- When reviewing my sprints, I found myself constraint for time as I often found myself during the week trying to do a lot of things at once.
- My final deliverable had only 15 of the 22 issues completed.
- I constantly found myself reviewing the course material whilst working on the project which I felt really set me back in terms of timing.
- I didn't have enough time to implement my course or student_course functions.
- Overall, however, I was proud at the fact that despite my complete lack of experience prior to the academy, I was able to build a back-end and a front-end in the span of a week.

SPRINT RETROSPECTIVE

- Looking back at the project, I see many areas for improvement.
- Understanding my own abilities and working towards improve my comprehension of technical software.
- Proper time management, utilizing to-do lists, setting time frames via Jira and overall working on my ability to allocating enough time for tasks which may seem a bit easier.
- This would be evident with the fact that I spent too much time on my back-end, that my lack of experience with dealing with front-end software resulting in me not fully completing everything I wanted to complete (Course functions and Student_course functions).

CONCLUSION

- To conclude, I created a back-end which linked to a front end, and this application adhered to the CRUD functionality.
- I have learned how to create front-end projects using HTML, CSS and JavaScript and will be investing more time in creating my own project during my free time.
- I also learned however, that I still have more to learn when it comes to testing and Java overall so that will be something I will keep in mind.