Game Review

Abdul Salim

Introduction

My name is Abdul Salim, and I am currently a QA software consultant trainee.

I am currently learning to use a wide range of software applications such as Java, HTML, CSS and JavaScript for the first time.

I have just attempted my first full stack project, we'll see how I got on later on in the presentation.

The concepts were quite tricky to grasp, but i was continuously learning as the project went on.

Approach

To begin with I looked through the specification provided to identify what was begin asked of me.

I understood that the main chunk of the process would be used for the back end, making sure CRUD functionality was working properly for both entities and then ensuring connectivity to the frontend.
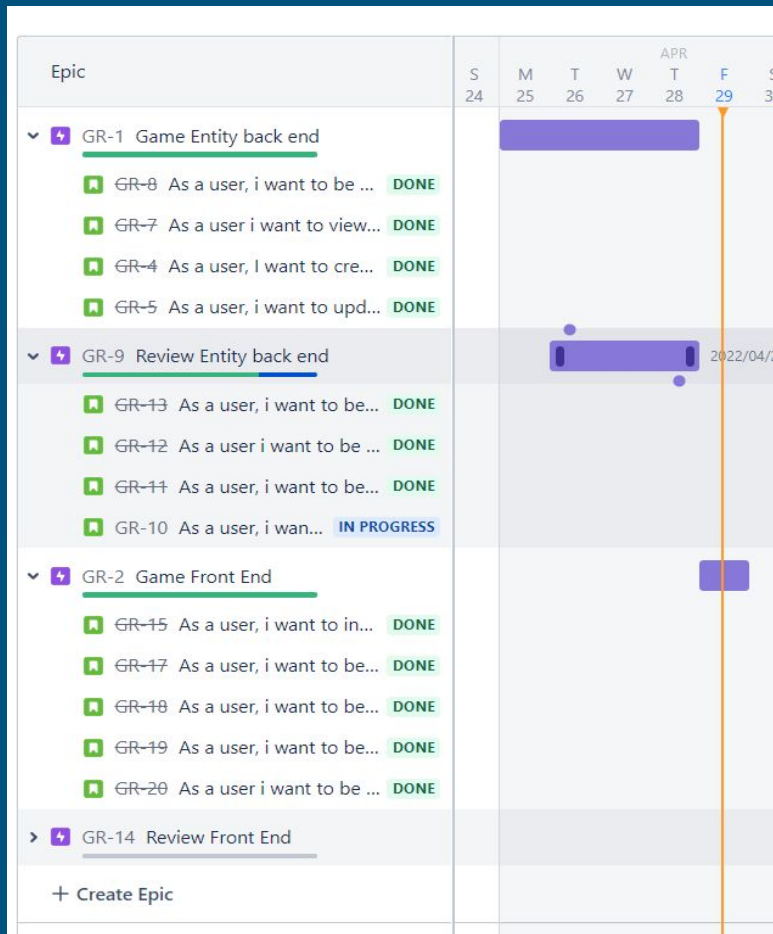
Sprint Plan

What needed to be included?

- A java backend developed with Spring Boot allowing for CRUD functionality and manipulating the SQL/H2 DB.
- A HTML/CSS/JS front-end developed that allows users to interact with games and their reviews.
- Mockito and Junit Unit and integration

What did I hope to achieve?

- A clean looking front end.

Using Jira Kanban board I :

Created 4 Epics, 2 targeting front end, and 2 for backend.

Broke down the specification into user stories.

Consultant Journey

Eclipse /w Spring suite - Used to develop back end with spring functionality

Jira - Used to plan the project

Git - Used to implement changes to my source code

VSCode - IDE used to develop front end

Postman - Used to interact with the api

MySql/H2 - Used to create a visual database

Github - Created a remote repository so i can access the project and keep track of changes I make from anywhere

Junit - Used to test my code to see if it does what's intended

Mockito - used to test dependencies

Maven - Used to package the code so it can be run without the source code, also allowed me to bring in dependencies

Continuous Integration

Version Control

Throughout the project, I would utilise branches on Github to track changes of my source code and make sure changes would not destroy any Initially when i created the repo, I created a develop branch. From here for any commit i needed to make i created feature branches. I would then merge it to the develop branch if the work was of satisfactory level. I will merge it back to the main branch, once the application is ready to run.

ERD
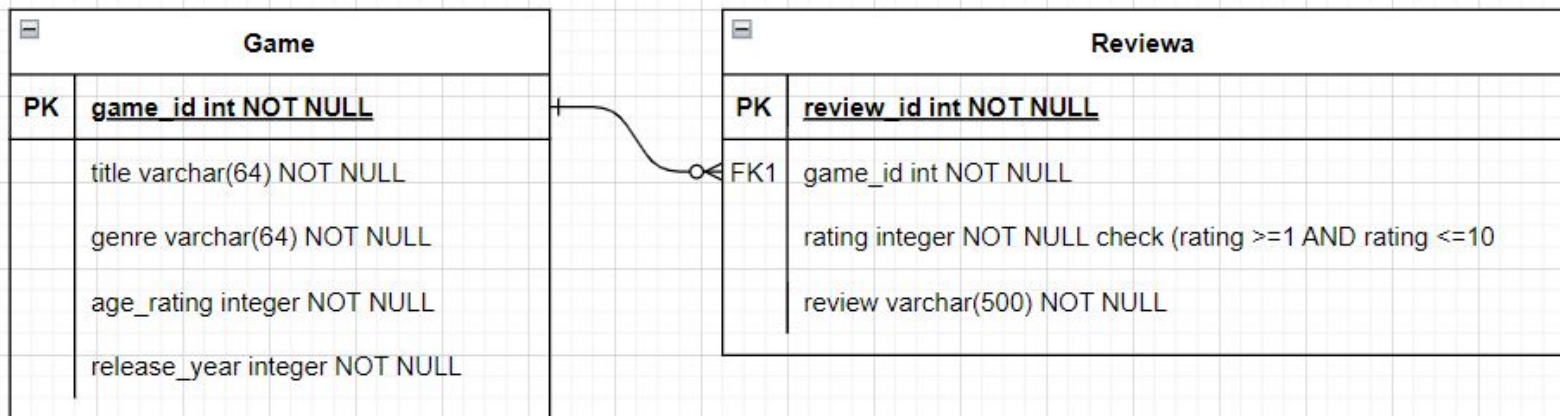
Testing

Element		Coverage	Covered Instru...	Missed Instruct...	Total Instructio...	
∨ 📁 Spring-Project-Game-Review		🟥 73.8 %	2,310	822	3,132	
> ⌗ src/main/java		🟥 51.5 %	860	810	1,670	
∨ 📁 src/test/java		🟩 99.2 %	1,450	12	1,462	
> ⌗ com.qa.GameReview.service		🟩 98.9 %	1,113	12	1,125	
> ⌗ com.qa.GameReview		100.0 %	4	0	4	
> ⌗ com.qa.GameReview.controller		100.0 %	333	0	333	

73% total coverage, only 51% for src/main/java
Missing a few methods on integration and unit tests

Demonstration!

Sprint review

➢ What Did I Complete?

Crud functionality for game entity was working in postman

Most of the CRUD functionality for review entity was working in postman

Able to get a working front end for the game entity

➢ What Got Left Behind?

Backend wasn't fully tested.

Was unable to implement crud functionality for review in the front end.

Design of the front end was not great.

Sprint Retrospective

➢ What went well?

Got very comfortable with my version control towards the end of the project

Building the back end classes was fine for the most part, although time consuming

➢ What could be improved?

I think my time management has to improve in future.

A better plan to begin with may combat that. Neglected front end trying to get back end to work.

I need to be able to use jira more effectively.

Conclusion

For my first full stack project, I am content with how the project went. However, improvements can most definitely be made and I am eager to improve.

I felt overwhelmed with the workload and started to feel burnt out towards the end of the project.

I do feel however, project based learning is the best way of cementing knowledge, and have gained a good amount of understanding due to this project.

Ultimately, I see the importance of continued self learning and practice in order to get comfortable with whichever software.

Any Questions?