



QA - Hobby/Interests Project (Soft Drinks)

Adil Akbarali



Introduction

Adil Akbarali (<https://github.com/adilakbarali>)

Design a full-stack application around a hobby/interest of choice.

Interest of choice: Soft Drinks!

Why? Who doesn't like soft drinks! It was also the first thing that came to mind, as I had one on my desk at the time



Presentation Structure

- Planning & Technologies
- Technologies Expansion
- Demo
- Summary
- Q&A



Planning

- Broke down the scope into multiple bullet points
- Create kanban board on Jira
- Created a risk assessment on Word (saved as PDF)
- Create and Initialise repository locally + on GitHub
- Validated results with a deliverables checklist for a MVP



Technologies Used: Jira

Jira was used to plan the entire project, and was chosen as it was the choice of Kanban during training.

Working with it has allowed me to learn:

- Importance of structuring tasks
- Importance of allocating time to tasks/estimating time for tasks

Improvements:

- Use child issues (Used in IMS but not here as I overlooked them)
- Potentially add more tasks/additional tasks outside MVP

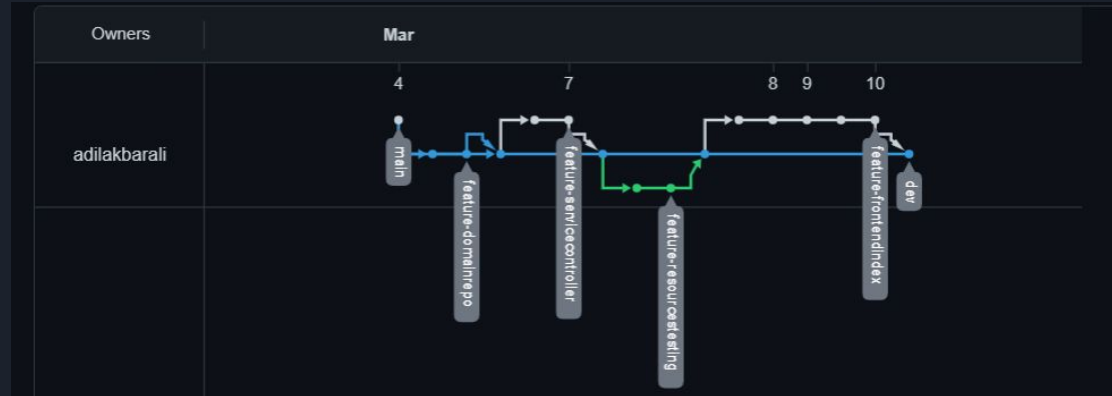
Jira: Epics



Technologies Used: Git

Git provides a method of local version control, however this can be made into a cloud version control through the use of GitHub.

Development was carried out through feature branches, where some slightly broad branch names were chosen to implement multiple quick classes.





Technologies Used: Java & MockMVC

Java was the core backend programming language in use, alongside the Sprint Boot framework to allow for mapping of URLs to methods (as known as an API).

MockMVC allowed for simple and effective testing of server-sided functionality within the application.

When used with each other, this allowed for a quick and easy way of testing the functionality of the API without using external applications or designing a front-end.



Technologies Used: HTML, CSS & JS

The core of the front-end development, these allowed for the creation of a webpage, formatting and styling, and functionality within the webpage and interactions with the backend.

This will be shown in the demo.

- Simple and effective layout
- Easy to use interface
- Fancier styling
- More options for the user (both back-end and front-end)

Now time for a
demonstration!





Summary

- Overall, project was deemed successful.
 - Everything worked as intended
 - Front-end came out looking better than expected
-
- Time management was not carried out well
 - Front-end despite looking good and simplistic, could have had a lot more work
 - More functionality on the back end so that there are more options to the user

Allocate more time to newer technologies so that I can adapt to their implementations. I have expanded my skills in designing web pages and utilising external libraries from this project, as well as gained a better understanding of APIs.



Thank You!

Any Questions?