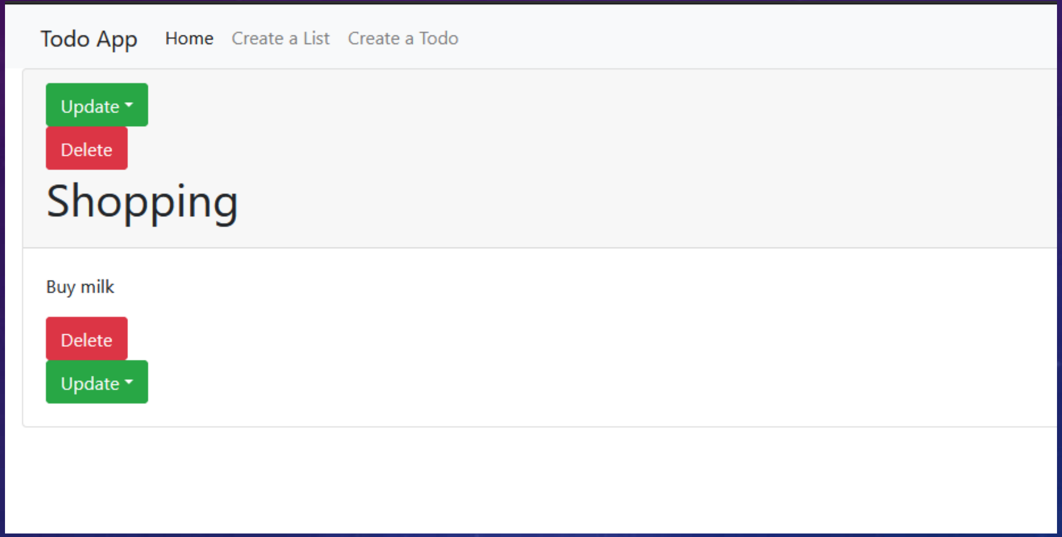
TODO APP



Shopping

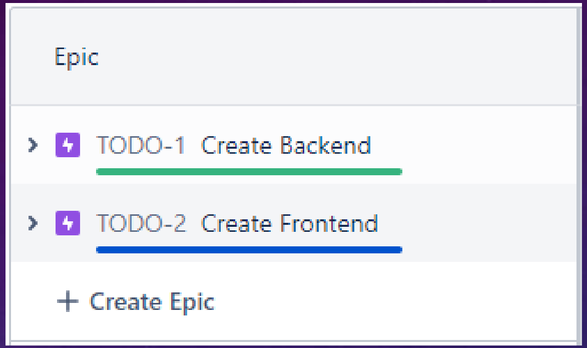
Buy milk

CONCEPT

- I thought that the best way to approach this was to start on the backend first, as it is the most important part in my opinion.
- I also thought that making the frontend look pretty wouldn't be wise as it would leave me less time to make everything functional, so design was the bottom of the priority list.

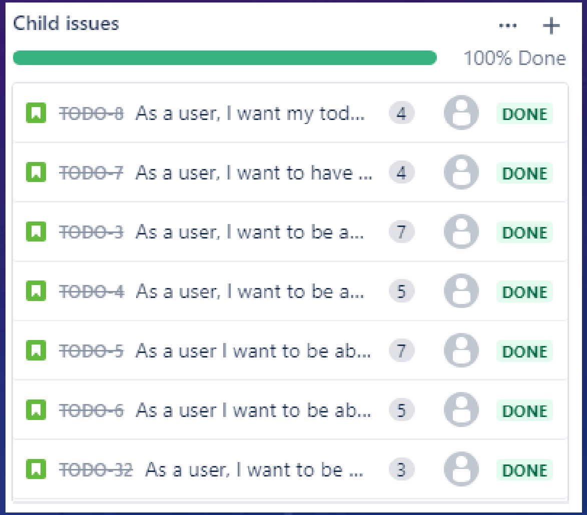
SPRINT PLAN

- The Backend implementation
- The Frontend



Epic

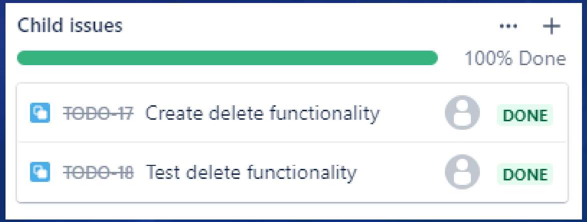
- TODO-1 Create Backend
- TODO-2 Create Frontend



Child issues

100% Done

~~TODO-8~~	As a user, I want my tod...	4	DONE
~~TODO-7~~	As a user, I want to have ...	4	DONE
~~TODO-3~~	As a user, I want to be a...	7	DONE
~~TODO-4~~	As a user, I want to be a...	5	DONE
~~TODO-5~~	As a user I want to be ab...	7	DONE
~~TODO-6~~	As a user I want to be ab...	5	DONE
~~TODO-32~~	As a user, I want to be ...	3	DONE



Child issues

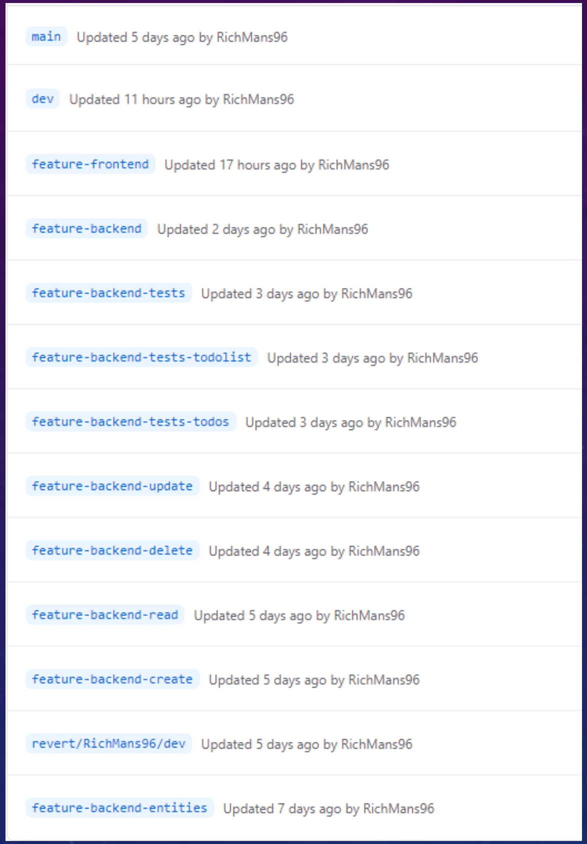
100% Done

~~TODO-17~~	Create delete functionality	DONE
~~TODO-18~~	Test delete functionality	DONE

TECHNOLOGIES LEARNED FOR THIS PROJECT

- Spring Boot – A Development framework
- Selenium – automated testing frameword
- SonarQube- static code analysis tool
- Bootstrap – CSS framework

CI - GIT



main Updated 5 days ago by RichMans96

dev Updated 11 hours ago by RichMans96

feature-frontend Updated 17 hours ago by RichMans96

feature-backend Updated 2 days ago by RichMans96

feature-backend-tests Updated 3 days ago by RichMans96

feature-backend-tests-todolist Updated 3 days ago by RichMans96

feature-backend-tests-todos Updated 3 days ago by RichMans96

feature-backend-update Updated 4 days ago by RichMans96

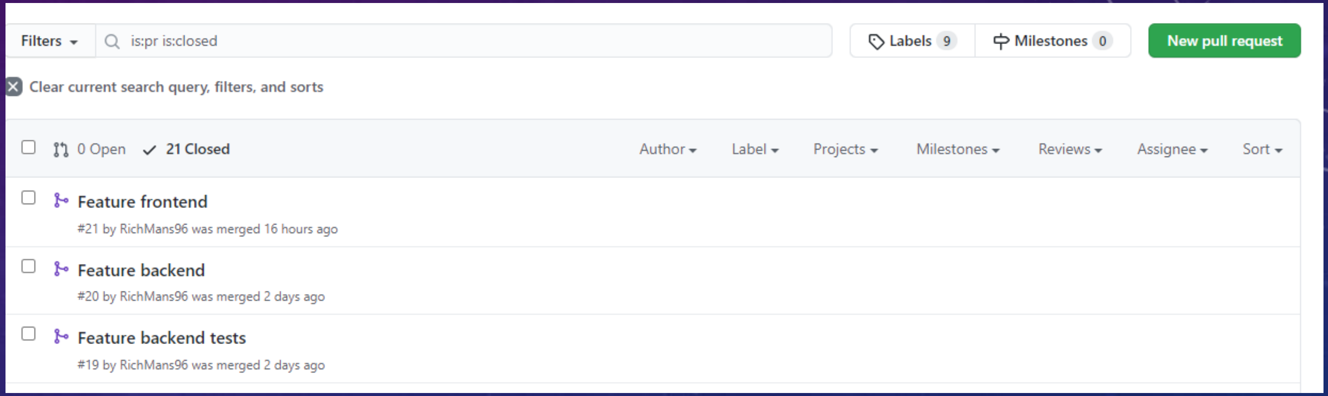
feature-backend-delete Updated 4 days ago by RichMans96

feature-backend-read Updated 5 days ago by RichMans96

feature-backend-create Updated 5 days ago by RichMans96

revert/RichMans96/dev Updated 5 days ago by RichMans96

feature-backend-entities Updated 7 days ago by RichMans96



Filters ▾ is:pr is:closed Labels 9 Milestones 0 New pull request

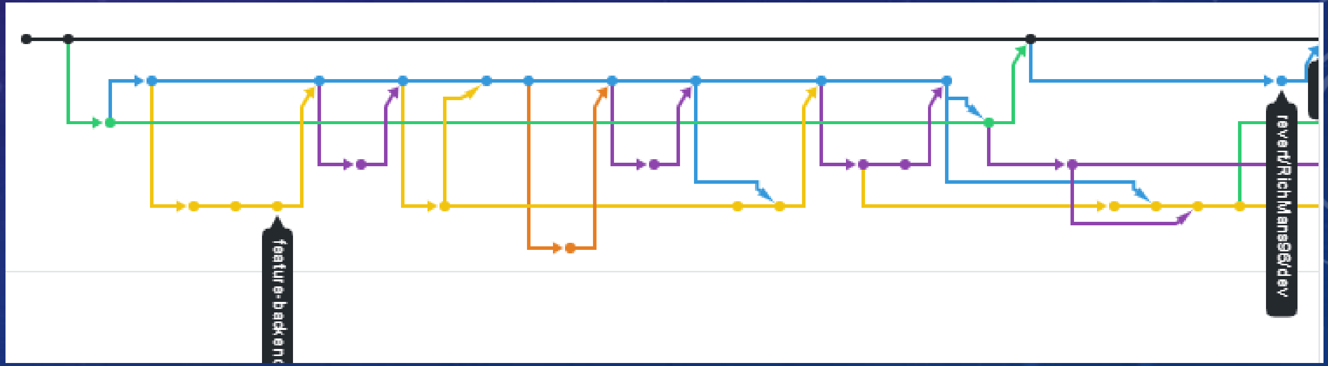
Clear current search query, filters, and sorts

□ 0 Open ✓ 21 Closed Author ▾ Label ▾ Projects ▾ Milestones ▾ Reviews ▾ Assignee ▾ Sort ▾

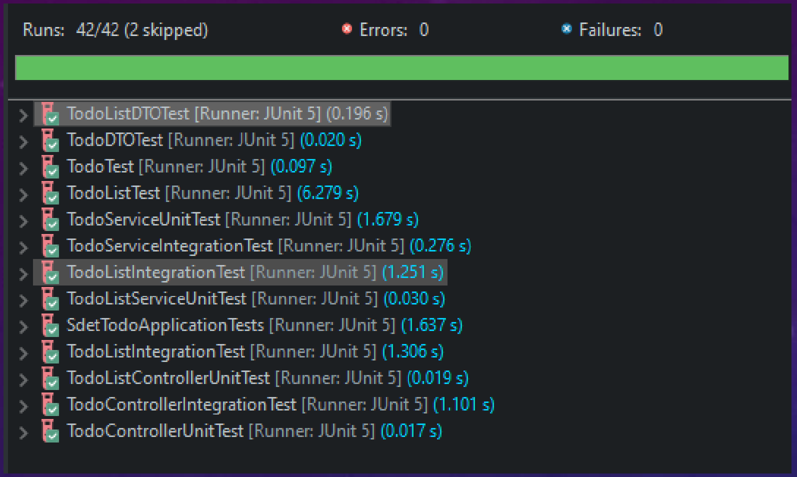
□ Feature frontend
#21 by RichMans96 was merged 16 hours ago

□ Feature backend
#20 by RichMans96 was merged 2 days ago

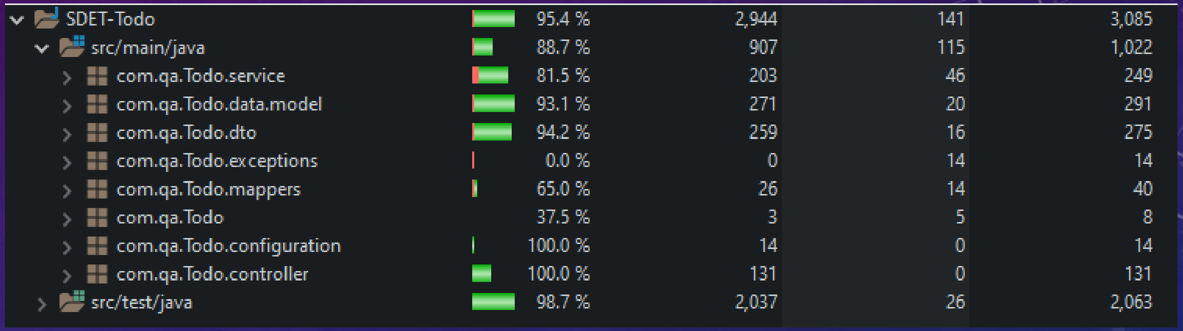
□ Feature backend tests
#19 by RichMans96 was merged 2 days ago



TESTING



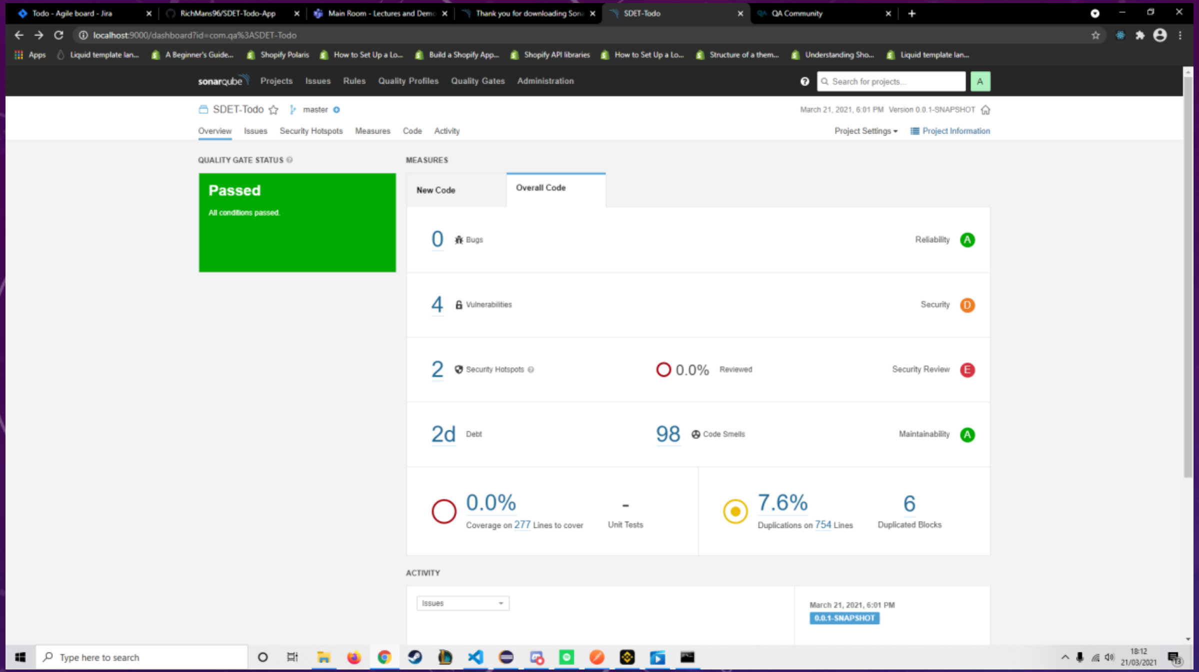
```
Runs:  42/42 (2 skipped)          Errors:  0          Failures:  0

> TodoListDTOTest [Runner: JUnit 5] (0.196 s)
> TodoDTOTest [Runner: JUnit 5] (0.020 s)
> TodoTest [Runner: JUnit 5] (0.097 s)
> TodoListTest [Runner: JUnit 5] (6.279 s)
> TodoServiceUnitTest [Runner: JUnit 5] (1.679 s)
> TodoServiceIntegrationTest [Runner: JUnit 5] (0.276 s)
> TodoListIntegrationTest [Runner: JUnit 5] (1.251 s)
> TodoListServiceUnitTest [Runner: JUnit 5] (0.030 s)
> SdetTodoApplicationTests [Runner: JUnit 5] (1.637 s)
> TodoListIntegrationTest [Runner: JUnit 5] (1.306 s)
> TodoListControllerUnitTest [Runner: JUnit 5] (0.019 s)
> TodoControllerIntegrationTest [Runner: JUnit 5] (1.101 s)
> TodoControllerUnitTest [Runner: JUnit 5] (0.017 s)
```

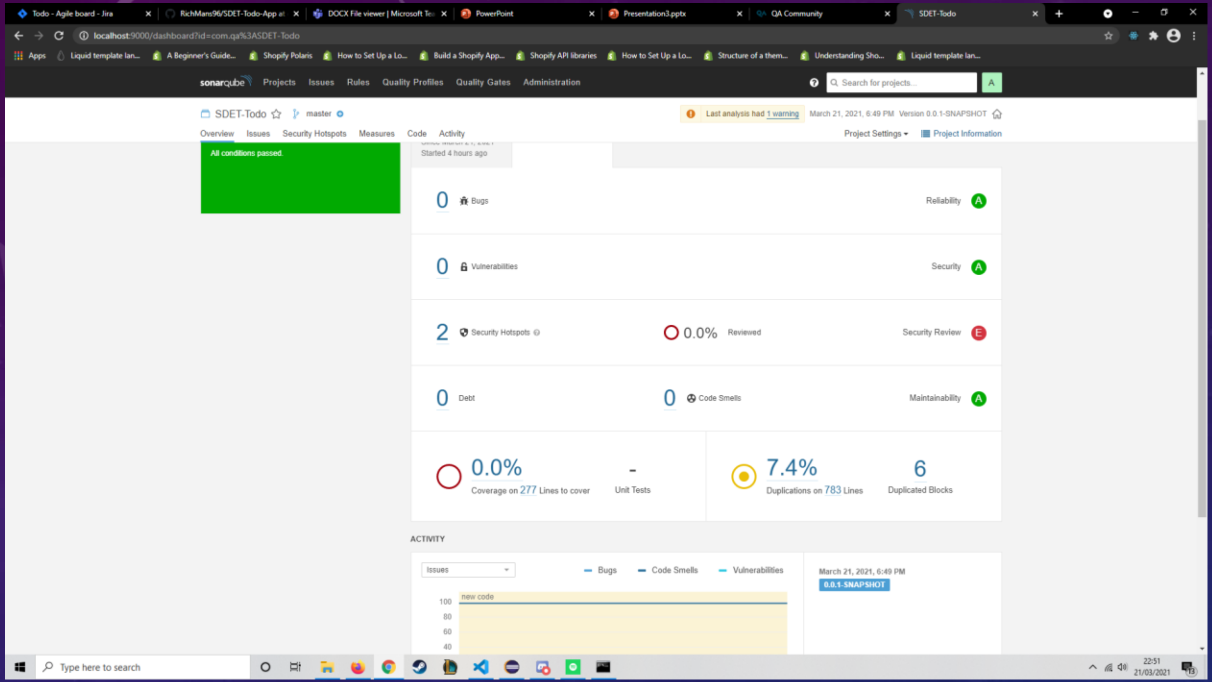


SDET-Todo	95.4 %	2,944	141	3,085
src/main/java	88.7 %	907	115	1,022
com.qa.Todo.service	81.5 %	203	46	249
com.qa.Todo.data.model	93.1 %	271	20	291
com.qa.Todo.dto	94.2 %	259	16	275
com.qa.Todo.exceptions	0.0 %	0	14	14
com.qa.Todo.mappers	65.0 %	26	14	40
com.qa.Todo	37.5 %	3	5	8
com.qa.Todo.configuration	100.0 %	14	0	14
com.qa.Todo.controller	100.0 %	131	0	131
src/test/java	98.7 %	2,037	26	2,063

I skipped 2 tests, 1 was due to an error with the Object mapper module, and 1 failure due to a test coming back with a null array, when it was expecting a todo in that array.

I have 88.7% test coverage. This could be improved by testing the exceptions. Also as I am skipping 2 tests that relate to the todolist, that is reducing coverage.





On the left you can see I had 98 code smells. Also, it couldn't pick up my code coverage, I spent a few hours trying to get that to work but to no avail.

A lot of these issues were down to me using public class for my tests, so I ignored those. I then had a view sys outs in my program from debugging so I removed those, there were also a view auto generated comments that I removed too. Sonarqube also didn't like the Autogenerated Equals methods so I just ignored those too.

DEMONSTRATION

ISSUES

- Eclipse deleting classpaths – towards the end of my project, my app could no longer run and I spent a few hours figuring out why. For some reason eclipse had removed all classpaths and I had to run mvn clean install to get everything back up and running again.
- CORS – on the frontend I encountered this error a few times. It's a bit ambiguous but in my case it was due to missing the @Crossorigin annotation in one of my controllers and errors in the url I was fetching.
- Sonarqube - Sonarqube wouldn't pick up my test coverage.
- Selenium! - I had a few issues with selenium, first being that the tests weren't being run from my main project, so I had to create a second project just for my selenium tests. Second problem with selenium is that it passes once but if tests are then run again it breaks my backend, meaning I have to restart the app every time.
- Bootstrap version 5 vs version 4 - I had an issue with dropdown menus, after spending a few hours trying to troubleshoot the issue I still couldn't get them to work, then I changed from Bootstrap 5 to 4 which fixed the issue.

SPRINT REVIEW

- I completed the basic MVP
- I didn't get to implement todo status.
- I didn't design the app as well as I could have.

REFLECTION

- I became a bit too fixated on trying to get my git network graph to look good, which wasted some of my time. Ultimately it really is that important if it looks nice.
- I will revisit this app and implement React for the frontend. I find React less fiddley than Bootstrap.
- I would like the revisit Selenium and SonarQube too as there are definitely gaps in my knowledge there that need to be seen to.
- I think backend implementation and unit/implementation tests are my strong points but as with anything they can still be worked upon.