Inventory Management System Project

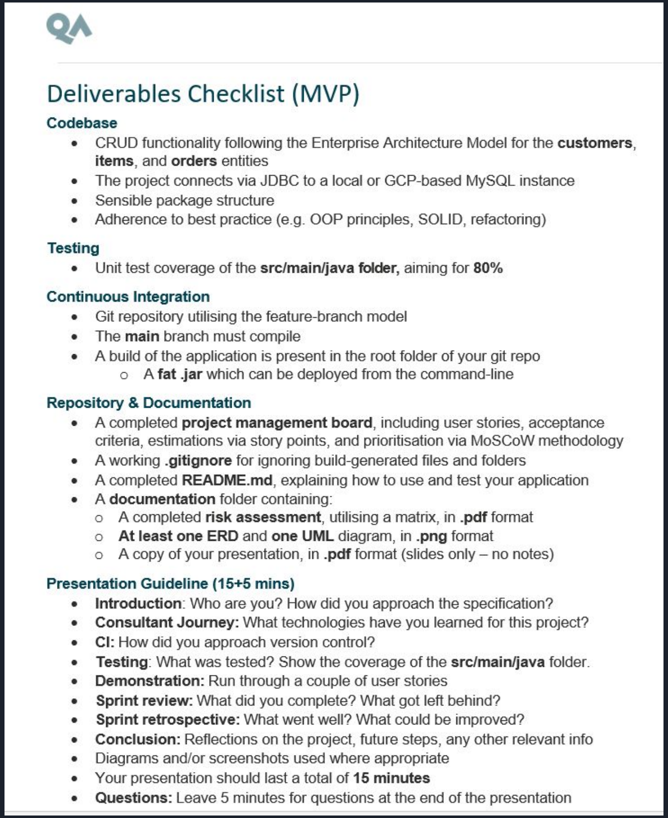
By Farzan Chowdhury

Introduction

Approaching the Specification:

- First, I Forked the repository QA provided us and imported it into Eclipse so I can study the code.
- After studying the code, I identified the client requirements
- Next, I created a Kanban Board on Jira.
- I planned out tasks and matched them up with user-stories (created child issues and blockers on my user-stories).
- Furthermore, I made an ERD to represent the database which I later created on MySQL.
- After creating the database, I began coding on Eclipse.

I started with coding the domain, DAO and controller for "Items", due to it having a similarity with "Customers".



QA

Deliverables Checklist (MVP)

Codebase
- CRUD functionality following the Enterprise Architecture Model for the **customers**, **items**, and **orders** entities
- The project connects via JDBC to a local or GCP-based MySQL instance
- Sensible package structure
- Adherence to best practice (e.g. OOP principles, SOLID, refactoring)

Testing
- Unit test coverage of the **src/main/java folder,** aiming for **80%**

Continuous Integration
- Git repository utilising the feature-branch model
- The **main** branch must compile
- A build of the application is present in the root folder of your git repo
  - A **fat .jar** which can be deployed from the command-line

Repository & Documentation
- A completed **project management board**, including user stories, acceptance criteria, estimations via story points, and prioritisation via MoSCoW methodology
- A working **.gitignore** for ignoring build-generated files and folders
- A completed **README.md**, explaining how to use and test your application
- A **documentation** folder containing:
  - A completed **risk assessment**, utilising a matrix, in **.pdf** format
  - **At least one ERD** and **one UML** diagram, in **.png** format
  - A copy of your presentation, in **.pdf** format (slides only – no notes)

Presentation Guideline (15+5 mins)
- **Introduction**: Who are you? How did you approach the specification?
- **Consultant Journey:** What technologies have you learned for this project?
- **CI:** How did you approach version control?
- **Testing**: What was tested? Show the coverage of the **src/main/java** folder.
- **Demonstration:** Run through a couple of user stories
- **Sprint review:** What did you complete? What got left behind?
- **Sprint retrospective:** What went well? What could be improved?
- **Conclusion:** Reflections on the project, future steps, any other relevant info
- Diagrams and/or screenshots used where appropriate
- Your presentation should last a total of **15 minutes**
- **Questions:** Leave 5 minutes for questions at the end of the presentation

Consultant Journey

Learned Technologies:

Git/GitHub

Jira

MySQL

Java

Maven

JUnit

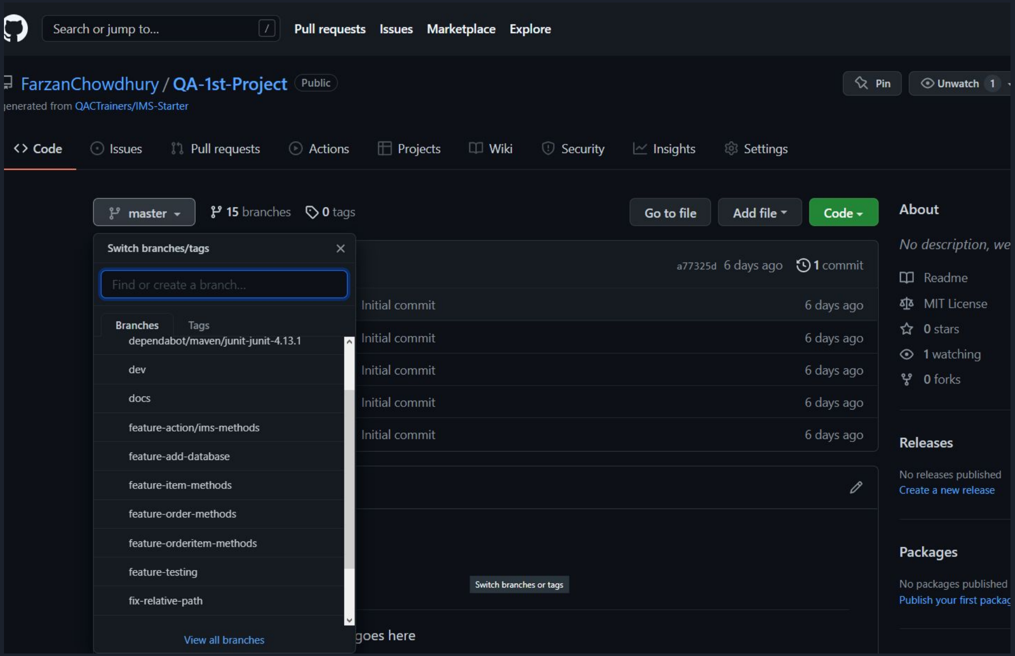
CI

Approaching Version Control:

I made multiple feature branches .

Constantly pushed any completed files up so I could keep all my work in my repository.

I could git pull from the feature branch if any work was lost.

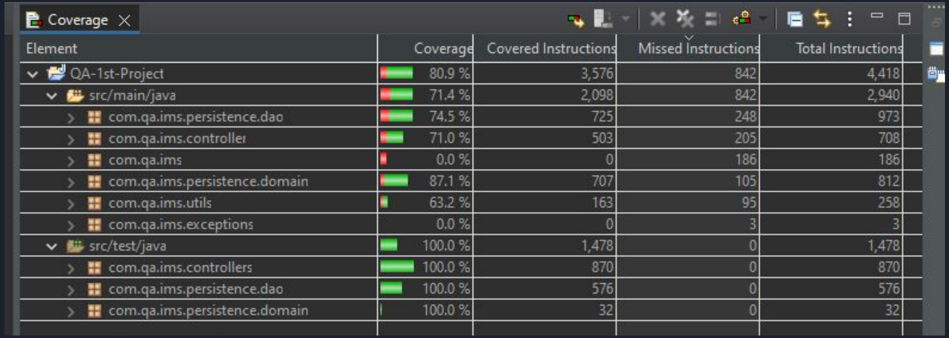


Testing

For testing, I used JUnit and Mockito.

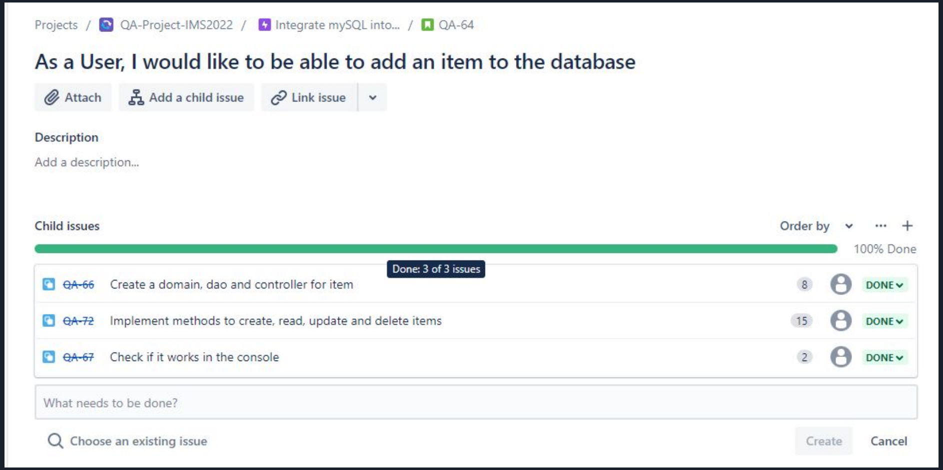
I achieved a test coverage of approximately 80.9%.

I tested all domains, DAOs and controllers.


Coverage

Element	Coverage	Covered Instructions	Missed Instructions	Total Instructions
QA-1st-Project	80.9 %	3,576	842	4,418
src/main/java	71.4 %	2,098	842	2,940
com.qa.ims.persistence.dao	74.5 %	725	248	973
com.qa.ims.controller	71.0 %	503	205	708
com.qa.ims	0.0 %	0	186	186
com.qa.ims.persistence.domain	87.1 %	707	105	812
com.qa.ims.utils	63.2 %	163	95	258
com.qa.ims.exceptions	0.0 %	0	3	3
src/test/java	100.0 %	1,478	0	1,478
com.qa.ims.controllers	100.0 %	870	0	870
com.qa.ims.persistence.dao	100.0 %	576	0	576
com.qa.ims.persistence.domain	100.0 %	32	0	32

Demonstration



Projects / QA-Project-IMS2022 / Integrate mySQL into... / QA-64

As a User, I would like to be able to add an item to the database

Attach | Add a child issue | Link issue

Description

Add a description...

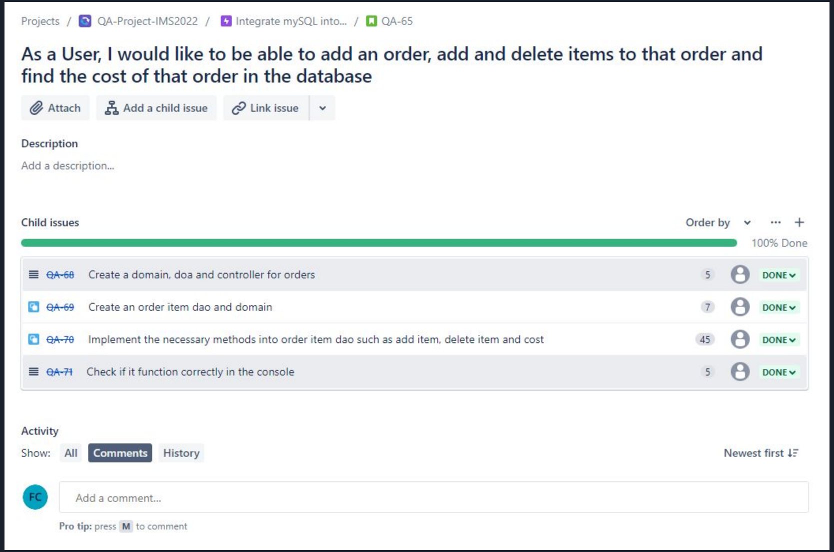
Child issues — Order by — 100% Done

Done: 3 of 3 issues

~~QA-66~~	Create a domain, dao and controller for item	8	DONE
~~QA-72~~	Implement methods to create, read, update and delete items	15	DONE
~~QA-67~~	Check if it works in the console	2	DONE

What needs to be done?

Choose an existing issue — Create — Cancel



Projects / QA-Project-IMS2022 / Integrate mySQL into... / QA-65

As a User, I would like to be able to add an order, add and delete items to that order and find the cost of that order in the database

Attach | Add a child issue | Link issue

Description

Add a description...

Child issues — Order by — 100% Done

~~QA-68~~	Create a domain, doa and controller for orders	5	DONE
~~QA-69~~	Create an order item dao and domain	7	DONE
~~QA-70~~	Implement the necessary methods into order item dao such as add item, delete item and cost	45	DONE
~~QA-71~~	Check if it function correctly in the console	5	DONE

Activity

Show: All | Comments | History — Newest first

FC — Add a comment...

Pro tip: press M to comment

Sprint Review

What I completed:

- All necessary tasks and user-stories completed on Jira.
- All necessary operations completed such as create,read,update and delete for items and orders.
- All necessary operations completed for orders such as add item, delete item and total cost.

What got left behind:

- Achieving higher testing coverage for the project
- Completing the extension task

Sprint Retrospective

What went well:

- Created a working application with all the required functionalities.
- Prioritised the all the important task first.

What could be improved:

- Better time management
- Updating my Kanban Board

Conclusion

Reflection on the project:

- I picked up lots of useful skills and techniques working through the project.
- I struggled on Java but this project helped grasp a better understanding of the Java functionalities.

Further steps:

- Further understand Java functionalities to help me code with less unnecessary codes.
- Rewrite any codes to better fit the tests for them.

Thank you for listening!

Any questions?