

Inventory Management System - QA

Conor Kelly



Approach

Likelihood	Impact					
		Negligible 1	Minor 2	Moderate 3	Major 4	Catastrophic 5
	Almost certain 5	Moderate 5	High 10	Extreme 5	Extreme 20	Extreme 25
	Likely 4	Moderate 4	High 8	High 12	Extreme 16	Extreme 20
	Possible 3	Low 3	Moderate 6	High 9	High 12	Extreme 15
	Unlikely 2	Low 2	Moderate 4	Moderate 6	High 8	High 10
	Rare 1	Low 1	Low 2	Low 3	Moderate 4	Moderate 5

Risk	Likelihood	Impact	Risk Rating	Preventative Measures	Response
Run out of time	4	3	12	Complete tasks with higher prioritisation first and start tasks with higher story points earlier. Refer to kanban board	Complete the tasks for all user stories that you can with the time you have left. If you ran out of time due to personal or unforeseen circumstances, talk to project trainer
Loss of files	2	4	8	Push all files and commits to your Github repository regularly as well as always saving files	Use last saved peice of work or most recent work from Github to build your code/databases again
Poor quality code	3	2	6	Test code frequently and use best pratises learnt throughout the Java teaching	Resolve bugs and logical errors when they're found
Low productivity	5	2	10	Set goals during the project using the SMART technique	Take time away from the project to clear your mind and come back to it feeling fresh
Inaccurate estimations	4	4	16	Focus on the work that has immediate priority. Include techspikes - allocated time to research an unfamiliar part of the project	Work on proeject out of training hours. Talk to the project trainer and explain the circumstances
Poor risk management	2	2	4	Identify potential risks and the likelihood of them. Create risk mitigation plans and carefully monitor the risks	Create a risk assessment upon the first encounter with a risk in order to prevent any other risks from occuring or taking too much time out of the project

Backlog

CK
+
Epic ▾
Insights

▼ Backlog (23 issues)

174 0 0 Create sprint

IMS-2 As a developer I want to be able to create the schema for the databases to allow i...

CREATE DATABASES

5

TO DO ▾

IMS-32 As a developer I want to be able to create an ERD to map database relationships

CREATE DATABASES

3

TO DO ▾

IMS-5 As a client I want to be able to add a customer so that a record of t...

CREATE CUSTOMER FUNCTIONALI...

3

TO DO ▾

IMS-6 As a client I want to be able to view all customers to ensure all infor...

CREATE CUSTOMER FUNCTIONALI...

2

TO DO ▾

IMS-7 As a client I want to be able to update a customer so that all custo...

CREATE CUSTOMER FUNCTIONALI...

4

TO DO ▾

IMS-8 As a client I want to be able to delete a customer so there is no rec...

CREATE CUSTOMER FUNCTIONALI...

3

TO DO ▾

IMS-18 As a developer I want to be able to create a connection between ...

CREATE ITEM FUNCTIONALITY IN ...

15

TO DO ▾

IMS-9 As a client I want to be able to add an item to the system so that ne...

CREATE ITEM FUNCTIONALITY IN ...

3

TO DO ▾

IMS Sprint 2

Allow customer and item functionality

CK
+
Epic ▾
Full screen Snp
GROUP BY None ▾
Insights

TO DO 4 ISSUES

As a client I want to be able to add an item to the system so that new stock is available for purchase

CREATE ITEM FUNCTIONALITY IN ...

IMS-9

3

As a client I want to be able to view all items in the system so that a record of stock can be taken

CREATE ITEM FUNCTIONALITY IN ...

IMS-10

2

IN PROGRESS 1 ISSUE

As a developer I want to be able to create a connection between items in the system and database so item functionality can be carried out

CREATE ITEM FUNCTIONALITY IN ...

IMS-18

15

DONE 4 ISSUES ✓

As a client I want to be able to view all customers to ensure all information is correct

CREATE CUSTOMER FUNCTIONALI...

IMS-6

2

As a client I want to be able to update a customer so that all customer information is up to date

CREATE CUSTOMER FUNCTIONALI...

IMS-7

1

Roadmap

CK
+
Status category ▾
Give feedback
Share
Export

Epic	FEB T 10	F 11	S 12	S 13	M 14	T 15	W 16	FEB T 17	F 18	S 19	S 20	M 21	T 22	W 23	FEB T 24	F 25	S 26	S 27	M 28
IMS-1 Create databases																			
IMS-4 Create customer functionality in t...																			
IMS-19 Create item functionality in the ...																			
IMS-20 Create order functionality in the...																			
IMS-28 Testing functionality																			
IMS-22 Deliverables for project submiss...																			

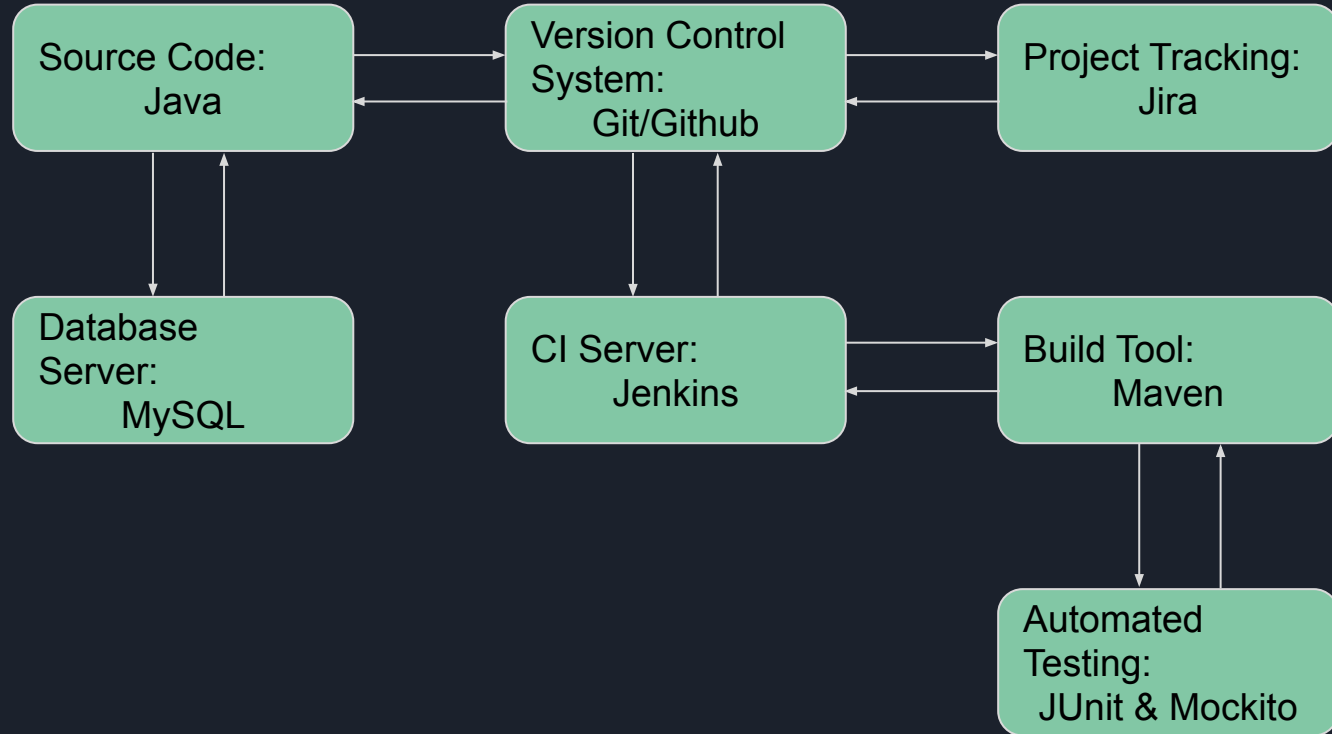
Project Planning - Jira



Consulting Journey

- 01 Agile - Jira (Scrum, Kanban)
- 02 Source Control - Git, Github
- 03 Relational Databases - MySQL
- 04 Programming Language - Java
- 05 IDE - Eclipse
- 06 Build Tool - Maven
- 07 Testing - JUnit, Mockito

CI Pipeline



Approach to version control

master/main -> dev > features

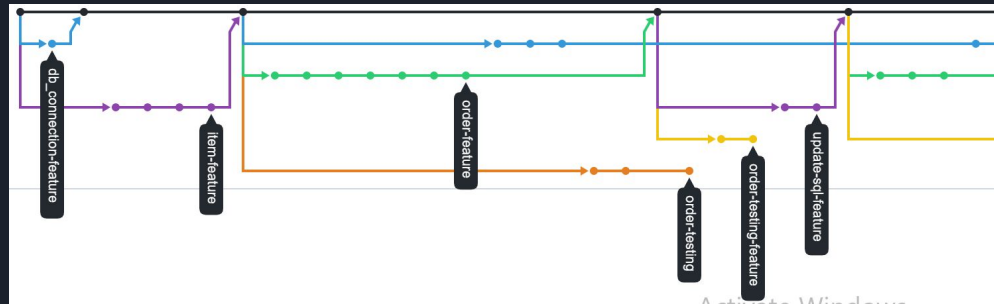
Separate feature branches used for all groupable software implementations/changes

- Order-feature branch used to create and make changes to Order.java, OrderDAO.java and OrderController.java

Once each task is completed, changes are committed pushed to the local repository

Feature branches get merged with the dev branch

When 100% confident your code is complete, the dev branch gets merged to the master/main branch





Testing





What was tested?

Customer, Item & Order:

- Domain: equals method
- Controllers: CRUD functionality with user input
- Data Access Objects: Methods interaction with the database

Finished after 10.003 seconds

Runs: 34/34 Errors: 0 Failures: 13

> com.qa.ims.controllers.CustomerControllerTest [Runner: JUnit 4] (1.619 s)
> com.qa.ims.persistence.domain.ItemTest [Runner: JUnit 4] (0.257 s)
> com.qa.ims.persistence.domain.CustomerTest [Runner: JUnit 4] (0.019 s)
> com.qa.ims.controllers.ItemControllerTest [Runner: JUnit 4] (0.045 s)
> com.qa.ims.persistence.dao.CustomerDAOTest [Runner: JUnit 4] (2.420 s)
▼ com.qa.ims.persistence.dao.ItemDAOTest [Runner: JUnit 4] (2.401 s)
 testReadLatest (0.417 s)
 testRead (0.415 s)
 testReadAll (0.516 s)
 testCreate (0.318 s)
 testDelete (0.353 s)
 testUpdate (0.381 s)
> com.qa.ims.persistence.domain.OrderTest [Runner: JUnit 4] (0.089 s)
> com.qa.ims.controllers.OrderControllerTest [Runner: JUnit 4] (0.091 s)
▼ com.qa.ims.persistence.dao.OrderDAOTest [Runner: JUnit 4] (2.957 s)
 testReadLatest (0.364 s)
 delete (0.427 s)
 testRead (0.475 s)
 testReadAll (0.441 s)
 testCreate (0.395 s)
 deleteItem (0.456 s)
 testUpdate (0.399 s)

Element

Element	Coverage
▼ ims	72.4 %
▼ src/main/java	65.6 %
▼ com.qa.ims.persistence.dao	64.5 %
> OrderDAO.java	44.1 %
> CustomerDAO.java	75.1 %
> ItemDAO.java	79.4 %
> com.qa.ims	0.0 %
▼ com.qa.ims.persistence.domain	77.0 %
> Domain.java	0.0 %
> Order.java	88.6 %
> Item.java	90.6 %
> Customer.java	100.0 %
▼ com.qa.ims.controller	74.4 %
> Action.java	0.0 %
> CustomerController.java	100.0 %
> ItemController.java	100.0 %
> OrderController.java	100.0 %
> com.qa.ims.utils	68.5 %
> com.qa.ims.exceptions	0.0 %
> src/test/java	87.4 %



DEMO

Sprint Review

COMPLETE

INCOMPLETE

CRUD functionality for:

- Items
- Orders

Calculate the cost of an order

Testing (with failures)

Adding and deleting items from an order

▼ Backlog (1 issue)

0 10 0 Create sprint

IMS-16 As a client I want to be able to add and remove an item... CREATE ORDER FUNCTIONALITY I... 10 IN PROGRESS



Sprint Retrospective

WHAT WENT WELL

Project planning

Database handling

Creating CRUD functionalities for items and orders

IMPROVEMENTS

Improve java knowledge all around, especially on ArrayLists and how to manipulate them in methods

Testing

Time management

Debugging

User interface



CONCLUSION