



IMS Project

Kieran Hanahoe

Introduction

Must have

- Add, view delete customers
- Add, view, delete items
- Add, view, delete orders
- Calculate cost of an order

Could have

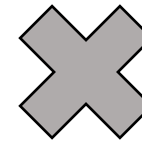
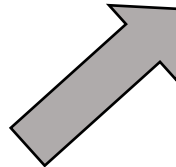
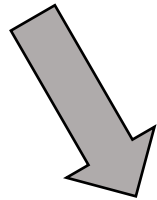
- View all orders for a specific customer
- View all orders containing a specific item
- View all customers who have ordered a specific item
- Select a database to use

Should have

- Update customer, item and order
- Add and remove items from an order

Won't have

- Graphical user interface
- Customize the database schema



Risk assessment

Risks	Risk statement	Response strategy	Objectives	Baseline			2020		
				Likelihood	Impact	Risk	Likelihood	Impact	Risk
Loss of data	Reduce risk	Use of version control so that any mistakes can be reverted, and work can be stored on a remote repository as well as locally.	Ensure that any single hardware failure, or loss of a service, doesn't cause a catastrophic loss of data.	1	5	3	1	3	2
Illness	Accept risk	Take precautions against being infected by disease. Aim to achieve MVP as soon as possible in case time is lost.	Minimize the chance disease affecting ability to deliver project.	2	1	1.5	3	4	3.5
Loss of cloud service	Reduce risk	Have the option to work with either a local MySQL server or a GCP instance.	Allow work on the project to continue even if GCP goes down.	2	4	3	2	1	1.5
Running out of time	Reduce risk	Ensure that I have something that works by the deadline.	Achieve minimum viable product as quickly as possible.	3	5	4	2	5	3.5

Consultant journey

Some familiarity

Version control : Git

Source code
management: GitHub

Database: MySQL

No familiarity

Programming language : Java

Build tool: Maven

Unit testing: JUnit and Mockito

Kanban board: Jira

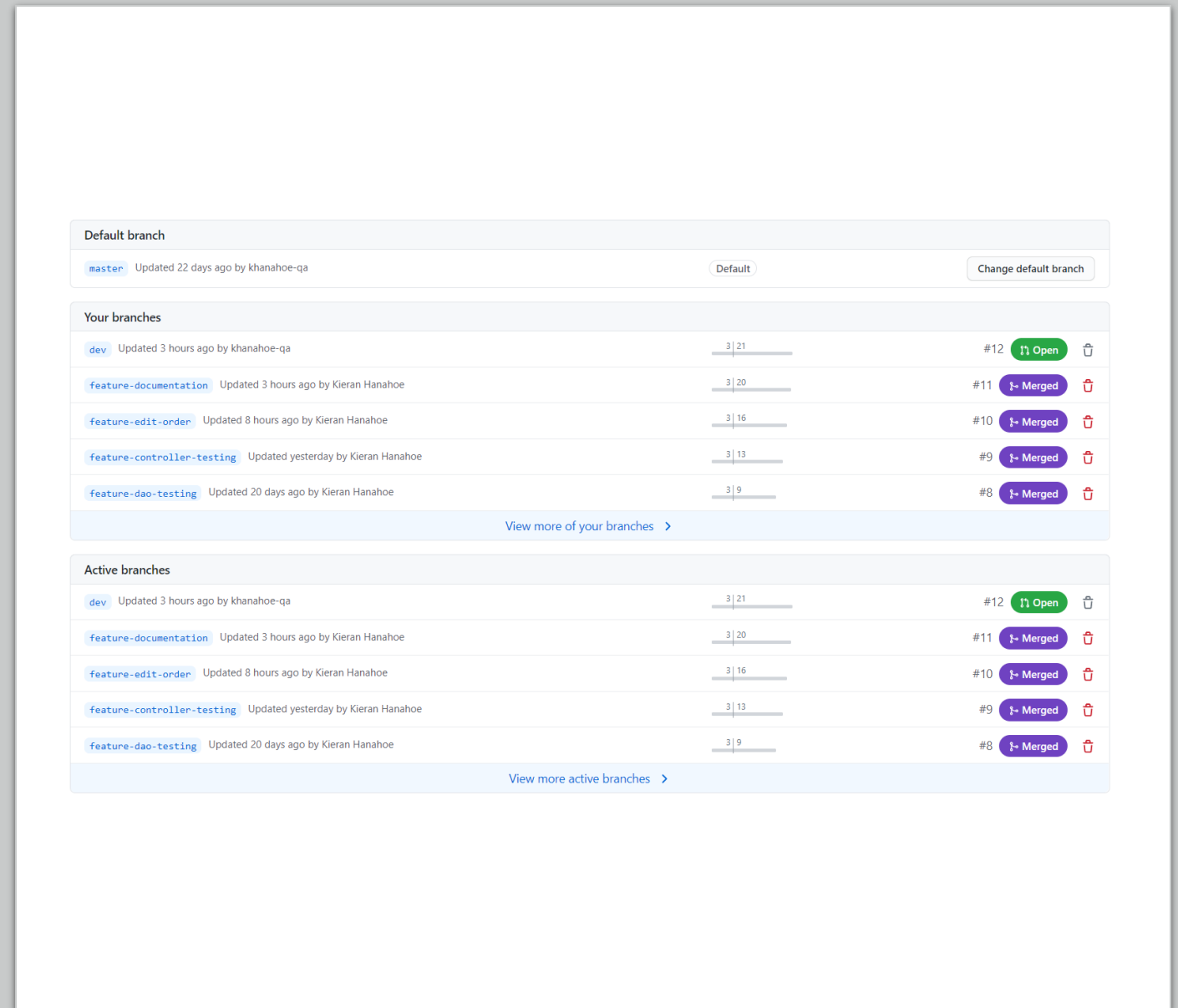
Jira

- User stories initially
- Added tasks later to cover tests etc.
- Didn't use story points

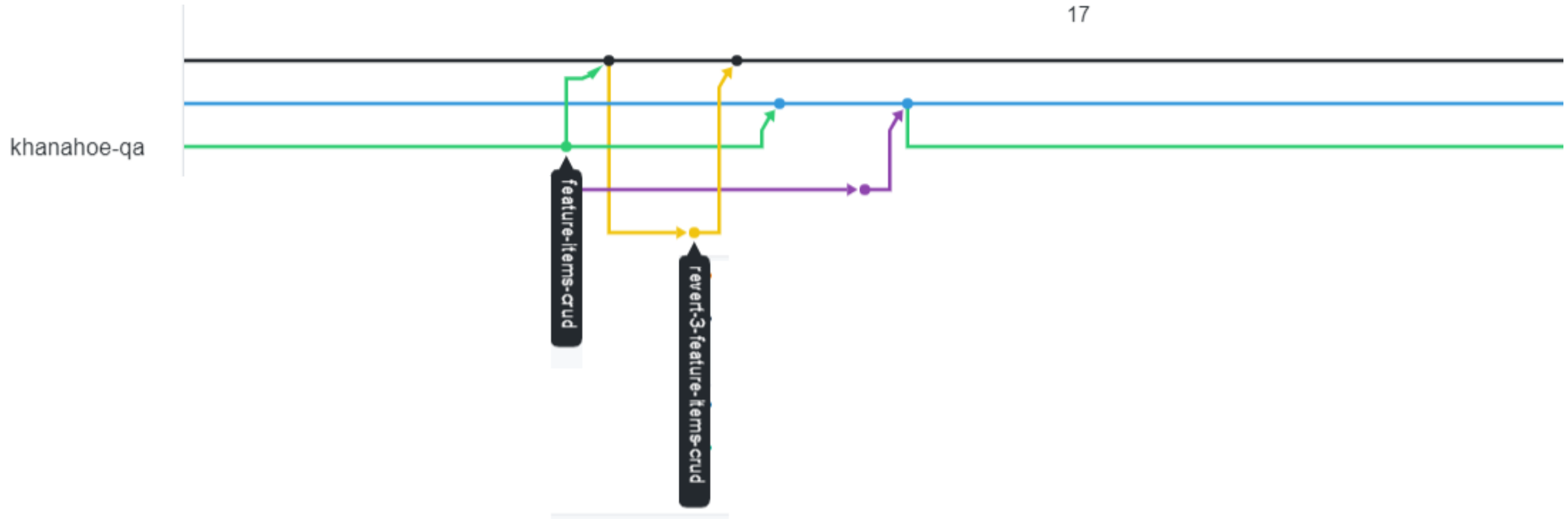
Epic	DEC	JAN '21
<div>Customer records</div> <div><div>As a user I want to be able to create ...</div><div>As a user I want to be able to delete ...</div><div>As a user, I want to be able to updat...</div><div>As a user I want to view all customer...</div></div>	<div></div> <div>TO DO</div> <div>TO DO</div> <div>TO DO</div> <div>TO DO</div>	<div></div>
<div>Orders</div> <div><div>As a user I want to be able to delete ...</div><div>As a user I want to be able to calcula...</div><div>As a user I want to be able to add an...</div><div>As a user I want to be able to delete ...</div><div>As a user I want to be able to view al...</div><div>As a user I want to be able to create ...</div></div>	<div></div> <div>TO DO</div> <div>TO DO</div> <div>TO DO</div> <div>TO DO</div> <div>TO DO</div>	<div></div>
<div>Item records</div> <div><div>As a user I want to be able to delete ...</div><div>As a user I want to be able to update...</div><div>As a user I want to be able to view al...</div><div>As a user I want to be able to add an...</div><div>As a user I want to be able to see all ...</div></div>	<div></div> <div></div> <div></div> <div></div> <div></div>	<div></div> <div>TO DO</div> <div>TO DO</div> <div>TO DO</div> <div>TO DO</div>
+ Create Epic		

Version control

- Git: feature-branch model
- Working on dev branch
- Had trouble when I tried to merge into master.



Merged feature into master by mistake



Testing

- JUnit and Mockito
- Test coverage 89.8%
- No coverage of Utils because Mockito can't mock scanners

CustomerDAOFail (7 Jan 2021 15:34:40)

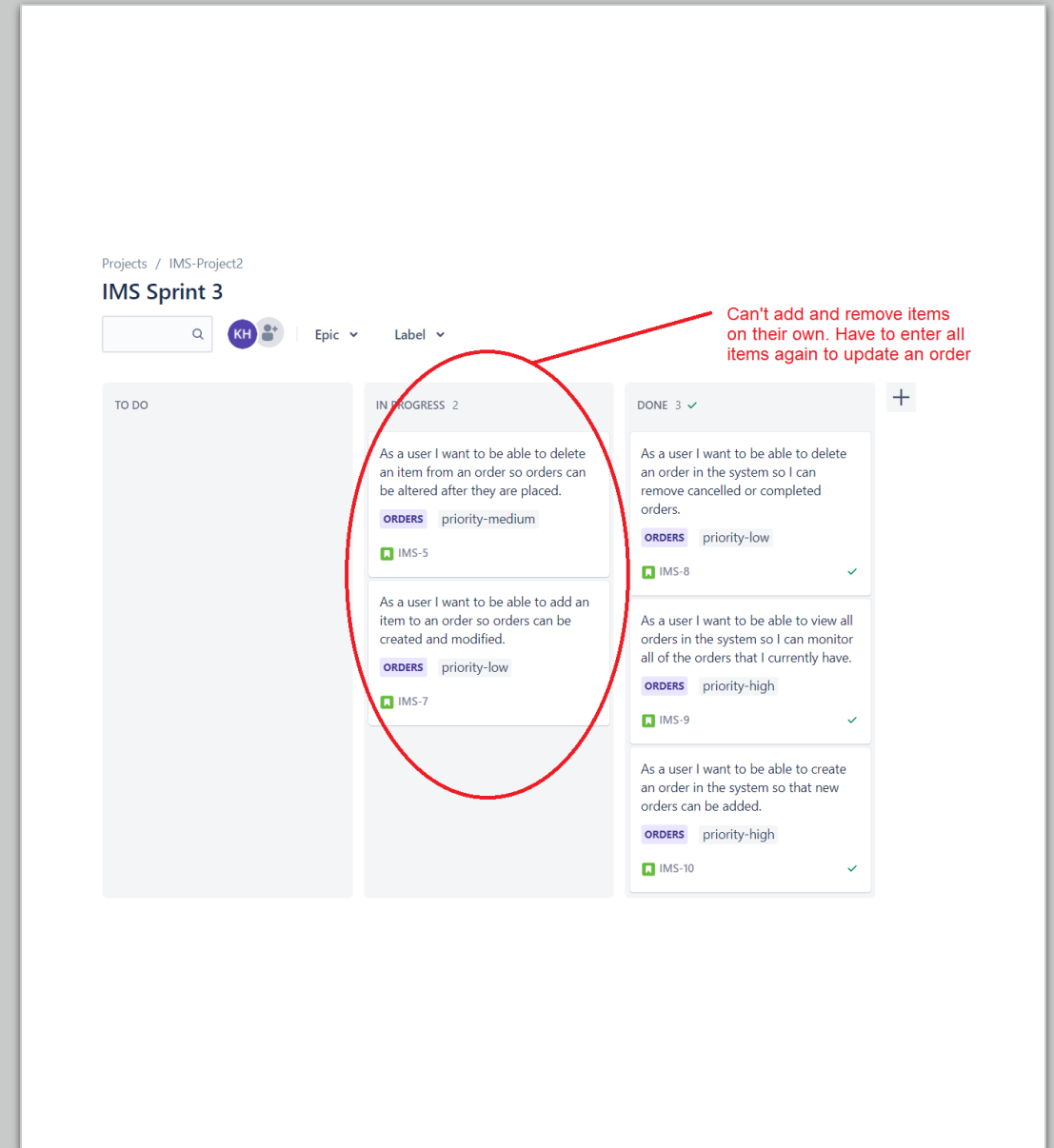
Element	Coverage	Covered Instructions	Missed Instructions	Total Instructions
IMS-Project	94.9 %	5,708	305	6,013
src/main/java	89.8 %	2,673	305	2,978
com.qa.ims	66.5 %	123	62	185
IMS.java	72.8 %	123	46	169
Runner.java	0.0 %	0	16	16
com.qa.ims.controller	99.5 %	800	4	804
Action.java	100.0 %	119	0	119
CustomerController.java	100.0 %	110	0	110
ItemController.java	100.0 %	131	0	131
OrderController.java	100.0 %	347	0	347
OrderSelection.java	95.9 %	93	4	97
com.qa.ims.persistence.dao	93.8 %	1,020	67	1,087
CustomerDAO.java	100.0 %	253	0	253
ItemDAO.java	100.0 %	270	0	270
OrderDAO.java	88.1 %	497	67	564
com.qa.ims.persistence.domain	89.5 %	577	68	645
Customer.java	83.1 %	108	22	130
Domain.java	100.0 %	105	0	105
Item.java	88.6 %	187	24	211
Order.java	88.9 %	177	22	199
com.qa.ims.utils	59.5 %	153	104	257
DBUtils.java	92.0 %	150	13	163
Utils.java	3.2 %	3	91	94
src/test/java	100.0 %	3,035	0	3,035



Demonstration

Sprint review

- Most sprints went smoothly
- Sprint 3 had some issues
- I didn't have time to add any could have



Sprint retrospective



Most of my sprints were effective



Could have planned in more detail



Didn't make use of story points



Left order update until the last minute by mistake

Conclusion

Successes

- Achieved minimum viable product
- Gained a lot of confidence using SQL, Java, Maven, and testing

Areas to improve

- Ideally would have gone slightly beyond MVP
- Could have planned in more detail
- Still having some trouble with version control

Next steps

- Add could haves e.g. show all customers who ordered a specific item
- Add GUI



Questions?