Inventory Management System

By Adam Hackworth

Introduction

Started off with documentation:

- Kanban Board using Jira Software
- Risk Assessment document drawn up
- ERD and UML diagrams where also drawn up

Starting the IMS

- Built a MySQL database using MySQL Workbench which I copied into the SQL Schema on the IMS.
- Started by building an Item class, Data access object and controller.
- Then moved onto making the orders entity in the IMS.

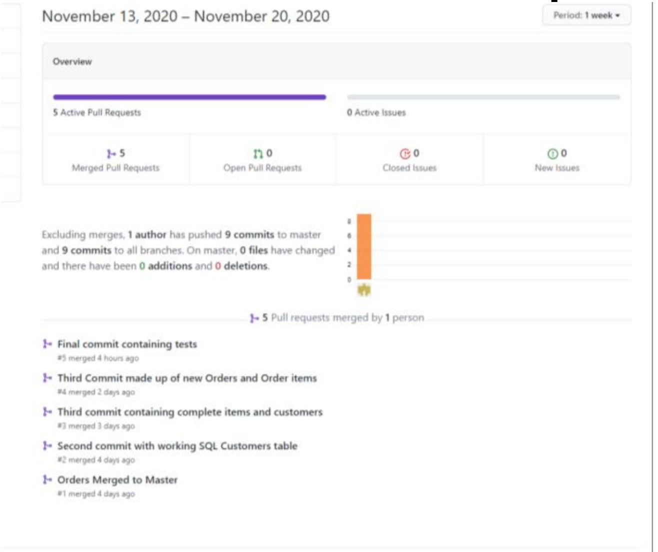
Consultant Journey

Over the past few weeks I have learnt a variety of new software:

- Java
- Git
- Jira
- Eclipse
- Maven
- Command Line

Continuous Integration

- Continuous Integration was performed using Git Bash software and Git Hub as a online repository to store completed code from eclipse.
- Using the branch feature on Git Bash I was able to upload various parts of the code I had been working on from a given day while still keep a safe working backup on the GitHub which allowed me to restart from a previous known working IMS.



November 13, 2020 – November 20, 2020

Overview

5 Active Pull Requests

0 Active Issues

5 Merged Pull Requests

0 Open Pull Requests

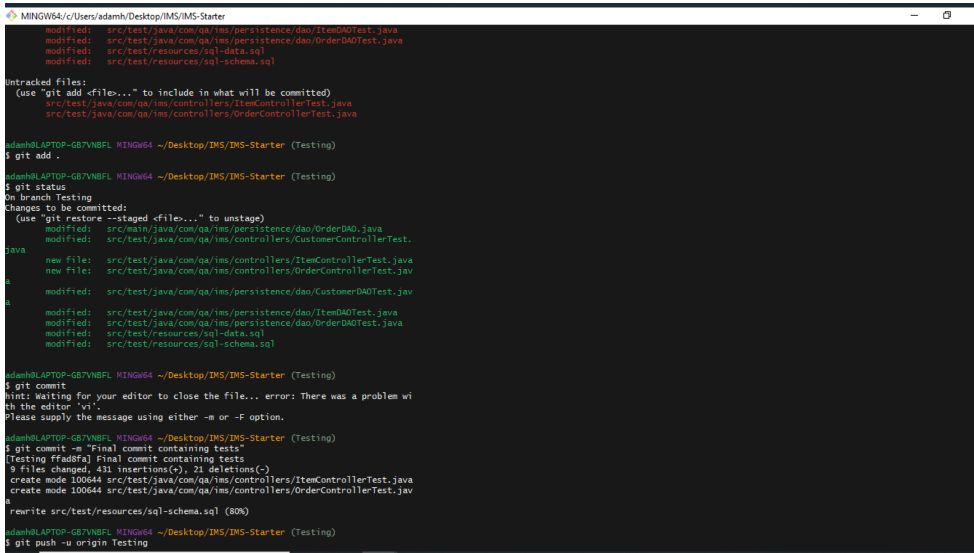
0 Closed Issues

0 New Issues

Excluding merges, **1 author** has pushed **9 commits** to master and **9 commits** to all branches. On master, **0 files** have changed and there have been 0 **additions** and 0 **deletions**.

5 Pull requests merged by 1 person

- Final commit containing tests
  #5 merged 4 hours ago
- Third Commit made up of new Orders and Order items
  #4 merged 2 days ago
- Third commit containing complete items and customers
  #3 merged 3 days ago
- Second commit with working SQL Customers table
  #2 merged 4 days ago
- Orders Merged to Master
  #1 merged 4 days ago

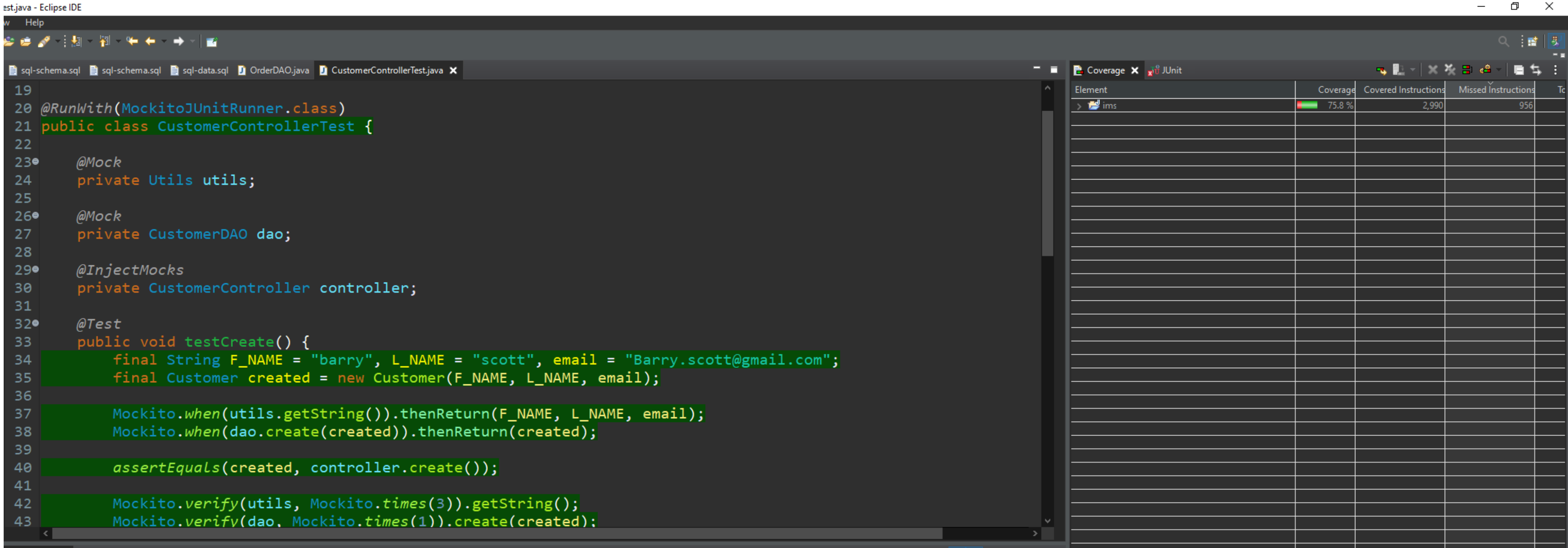


MINGW64:/c/Users/adamh/Desktop/IMS/IMS-Starter

```
        modified:   src/test/java/com/qa/ims/persistence/dao/ItemDAOTest.java
        modified:   src/test/java/com/qa/ims/persistence/dao/OrderDAOTest.java
        modified:   src/test/resources/sql-data.sql
        modified:   src/test/resources/sql-schema.sql

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        src/test/java/com/qa/ims/controllers/ItemControllerTest.java
        src/test/java/com/qa/ims/controllers/OrderControllerTest.java


adamh@LAPTOP-GB7VNBFL MINGW64 ~/Desktop/IMS/IMS-Starter (Testing)
$ git add .

adamh@LAPTOP-GB7VNBFL MINGW64 ~/Desktop/IMS/IMS-Starter (Testing)
$ git status
On branch Testing
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   src/main/java/com/qa/ims/persistence/dao/OrderDAO.java
        modified:   src/test/java/com/qa/ims/controllers/CustomerControllerTest.
java
        new file:   src/test/java/com/qa/ims/controllers/ItemControllerTest.java
        new file:   src/test/java/com/qa/ims/controllers/OrderControllerTest.jav
a
        modified:   src/test/java/com/qa/ims/persistence/dao/CustomerDAOTest.jav
a
        modified:   src/test/java/com/qa/ims/persistence/dao/ItemDAOTest.java
        modified:   src/test/java/com/qa/ims/persistence/dao/OrderDAOTest.java
        modified:   src/test/resources/sql-data.sql
        modified:   src/test/resources/sql-schema.sql


adamh@LAPTOP-GB7VNBFL MINGW64 ~/Desktop/IMS/IMS-Starter (Testing)
$ git commit
hint: Waiting for your editor to close the file... error: There was a problem wi
th the editor 'vi'.
Please supply the message using either -m or -F option.

adamh@LAPTOP-GB7VNBFL MINGW64 ~/Desktop/IMS/IMS-Starter (Testing)
$ git commit -m "Final commit containing tests"
[Testing ffad8fa] Final commit containing tests
 9 files changed, 431 insertions(+), 21 deletions(-)
 create mode 100644 src/test/java/com/qa/ims/controllers/ItemControllerTest.java
 create mode 100644 src/test/java/com/qa/ims/controllers/OrderControllerTest.jav
a
 rewrite src/test/resources/sql-schema.sql (80%)

adamh@LAPTOP-GB7VNBFL MINGW64 ~/Desktop/IMS/IMS-Starter (Testing)
$ git push -u origin Testing
```

Testing

- Testing was performed with JUnit and Mockito. These two systems allowed me to test the IMS. While also allowing me to see if any bugs where in the system.



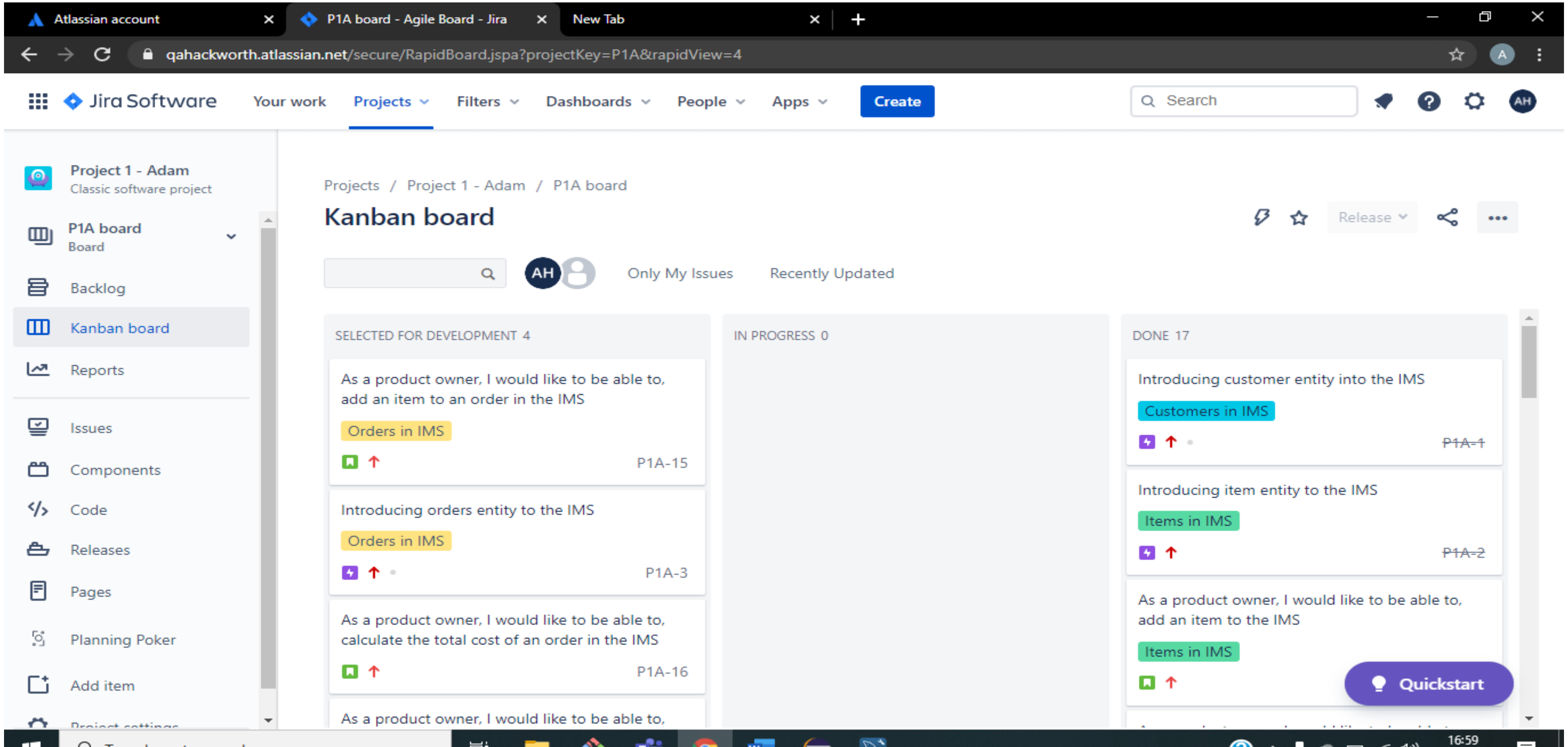
```java

@RunWith(MockitoJUnitRunner.class)
public class CustomerControllerTest {

    @Mock
    private Utils utils;

    @Mock
    private CustomerDAO dao;

    @InjectMocks
    private CustomerController controller;

    @Test
    public void testCreate() {
        final String F_NAME = "barry", L_NAME = "scott", email = "Barry.scott@gmail.com";
        final Customer created = new Customer(F_NAME, L_NAME, email);

        Mockito.when(utils.getString()).thenReturn(F_NAME, L_NAME, email);
        Mockito.when(dao.create(created)).thenReturn(created);

        assertEquals(created, controller.create());

        Mockito.verify(utils, Mockito.times(3)).getString();
        Mockito.verify(dao, Mockito.times(1)).create(created);
```

Element	Coverage	Covered Instructions	Missed Instructions	Tc
ims	75.8 %	2,990	956	

Demonstration

- Here is a demo of the system working.

Sprint Review



Kanban board

SELECTED FOR DEVELOPMENT 4

As a product owner, I would like to be able to, add an item to an order in the IMS
Orders in IMS
P1A-15

Introducing orders entity to the IMS
Orders in IMS
P1A-3

As a product owner, I would like to be able to, calculate the total cost of an order in the IMS
P1A-16

As a product owner, I would like to be able to,

IN PROGRESS 0

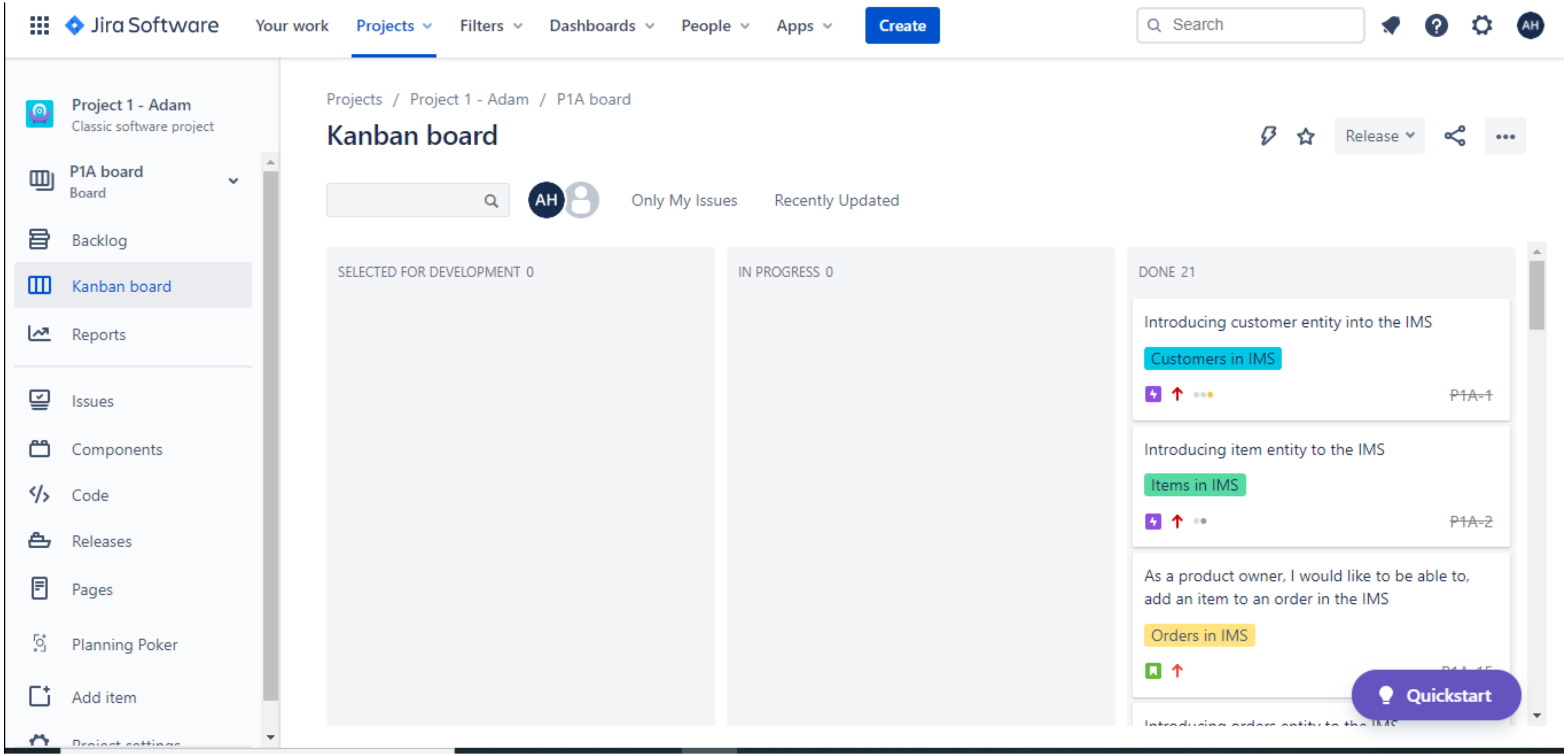
DONE 17

Introducing customer entity into the IMS
Customers in IMS
~~P1A-1~~

Introducing item entity to the IMS
Items in IMS
~~P1A-2~~

As a product owner, I would like to be able to, add an item to the IMS
Items in IMS

Sprint Retrospective



Kanban board

SELECTED FOR DEVELOPMENT 0

IN PROGRESS 0

DONE 21

Introducing customer entity into the IMS
Customers in IMS
~~P1A-1~~

Introducing item entity to the IMS
Items in IMS
~~P1A-2~~

As a product owner, I would like to be able to, add an item to an order in the IMS
Orders in IMS
P1A-15

Introducing orders entity to the IMS

To Conclude

- Enjoyed learning and using the various new software
- Struggled with parts but was able to get guidance
- Managed to successfully build a working IMS which was to the standard of the MVP

Any Questions…?