IMS Project

Sam Costello

20DecSoftware1

Introduction

- Name : Sam Costello

- Approach : I wanted a fully functioning database that met the spec first, then I could look at different functions and implementations

- I looked at the possibility:

- Admin/viewer pages

- Basket like function

Consultant Journey

- Version Control – Git and Git Hub
- Programme language – Java and MySQL
- Databases – GCP
- Unit testing – JUnit
- JIRA – Kanban board

Continuous Integration

5-6 commits a day

Merge Conflicts

Feature Branch Model

Maven Build Tool

JUnit

Testing

- Using JUnit and Mockito – Domains, Controllers, Services and Dao were all tested

- JUnit – testing individual methods

```java
@Before
public void setUp() {
    customer = new Customer(1L, "Chris", "Perrins");
    other = new Customer(1L, "Chris", "Perrins");
}

@Test
public void checkEqualityBetweenDifferentObjects() {
//    assertTrue(customer.equals(other));
    assertEquals(customer, other);
}

@Test
public void otherSurnameDifferent() {
    other.setSurname("thompson");
//    assertFalse(customer.equals(other));
    assertNotEquals(customer, other);
}
```

∨ 📁 sam-ims	86.4 %	5,752	906	6,658
> 📁 src/main/java	73.2 %	2,399	879	3,278
> 📁 src/test/java	99.2 %	3,353	27	3,380

Testing

- Mockito – Mocks objects in my controller and services tests

```java
@Test
public void createTest() {
    String itemName = "carrot cake";
    Double price = 5.00;
    Integer stock = 50;
    Mockito.doReturn(itemName, price.toString(), stock.toString()).when(itemController).getInput();
    Items item = new Items(itemName, price, stock);
    Items savedItem = new Items(1L, "carrot cake", 5.00, 50);
    Mockito.when(itemServices.create(item)).thenReturn(savedItem);
    assertEquals(savedItem, itemController.create());

}
```
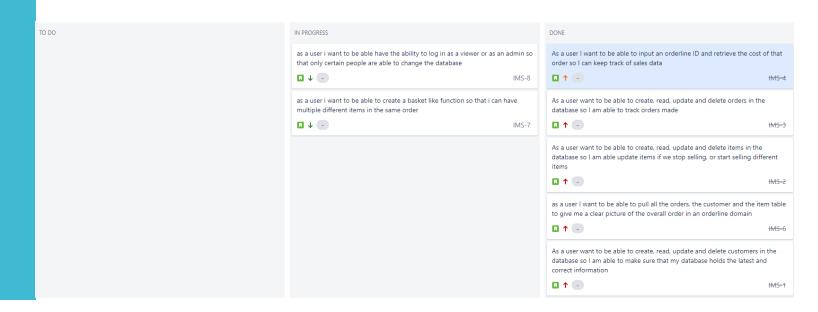
Sprint review

- What was completed?

I managed to complete the main aspects of the design as listed on the MVP

Prioritisation symbolises MoSCoW

- What got left behind?

Log in for admin and viewing were partially complete

TO DO

IN PROGRESS

as a user i want to be able have the ability to log in as a viewer or as an admin so that only certain people are able to change the database

IMS-8

as a user i want to be able to create a basket like function so that i can have multiple different items in the same order

IMS-7

DONE

As a user I want to be able to input an orderline ID and retrieve the cost of that order so I can keep track of sales data

IMS-4

As a user want to be able to create, read, update and delete orders in the database so I am able to track orders made

IMS-3

As a user want to be able to create, read, update and delete items in the database so I am able update items if we stop selling, or start selling different items

IMS-2

as a user I want to be able to pull all the orders, the customer and the item table to give me a clear picture of the overall order in an orderline domain

IMS-6

As a user want to be able to create, read, update and delete customers in the database so I am able to make sure that my database holds the latest and correct information

IMS-1

Sprint Retrospective

- What went well?

Core concepts

Targets

- What could be improved?

Bulky code

Logging reflections

Demonstration

As a user want to be able to create, read, update and delete items in the database so I am able update items if we stop selling, or start selling different items

IMS-2

as a user I want to be able to pull all the orders, the customer and the item table to give me a clear picture of the overall order in an orderline domain

IMS-6

Conclusion

- I will continue to work on this project to hone the skills I have learned and the skills that I will learn in the future

- This project has allowed me to cement my knowledge - I was very worried as I have not come from a coding background that I would struggle with this, but I have found that implementing what I have learned into a practical environment has been very helpful for me

Questions