IMS Project

(Stephen) William Goddard

Introduction: The Approach

How I initially approached the project:

- Understanding the specification
- Using the deliverables checklist
- Ordering the tasks
- Planning the user stories
- Understanding the project template code

All of this was done before starting any work on the project itself

Consultant Journey

Technologies learned for this project:

- Git and GitHub
- Java
- JUnit and Mockito
- MySQL
- Jira Kanban Boards

Planning - User Stories

All things related to the customer
`Customer`
IM-1

All things related to the item
`Item`
IM-2

All things related to the order
`Order`
IM-3

As a user, I want to add a customer to the system so that I can store any necessary data about a customer.
`Customer`
IM-5

As a user, I want to view all customers in the system so that I can see who my customers are.
`Customer`
IM-6

As a user, I want to update a customer in the system so that I can make sure the customer's details are up-to-date.
`Customer`
IM-7

As a user, I want to delete a customer in the system so that I can clear the system of data about customers which I no longer need.
`Customer`
IM-8

As a user, I want to add an item to the system so that I can store data about what items are in the inventory.
`Item`
IM-9

As a user, I want to view all items in the system so that I can see data about the items in the inventory.
`Item`
IM-10

As a user, I want to update an item in the system so that I can keep up-to-date data about the items in the inventory.
`Item`
IM-11

As a user, I want to delete an item in the system so that I can remove data about items which I no longer need.
`Item`
IM-12

As a user, I want to create an order in the system so that I can store necessary data about orders.
`Order`
IM-13

As a user, I want to view all orders in the system so that I can see data about orders.
`Order`
IM-14

As a user, I want to delete an order in the system so that I can remove data about orders which I no longer need.
`Order`
IM-15

As a user, I want to add an item to an order so that when a customer orders an item, it is stored in an order for them.
`Order`
IM-16

As a user, I want to calculate a cost for an order so that I know how much the customer should be charged for an order
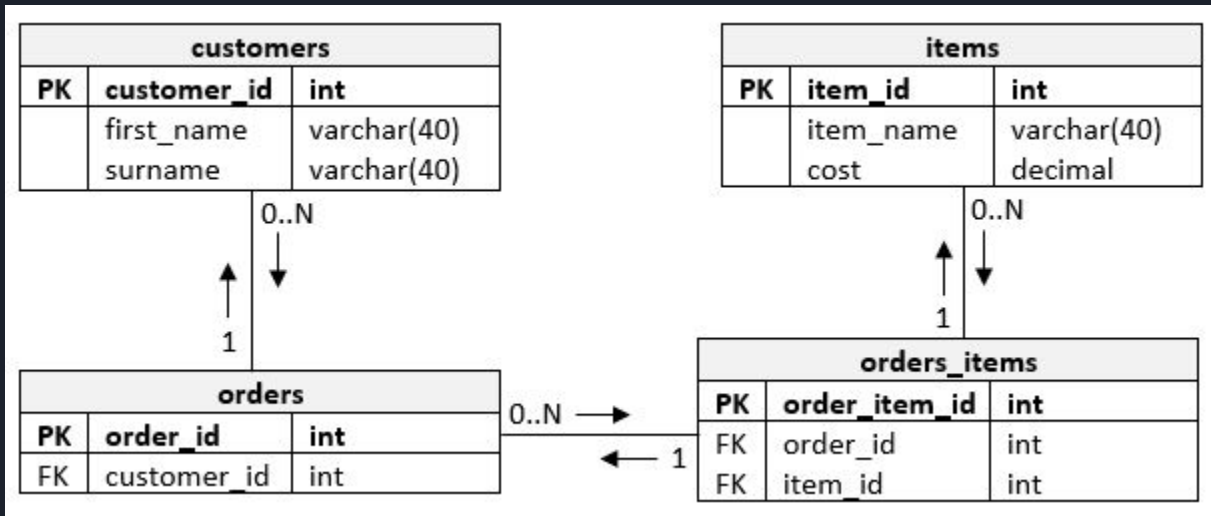`Order`
IM-17

As a user, I want to delete an item in an order so that an item can be removed from an order if a customer no longer wants to order that item
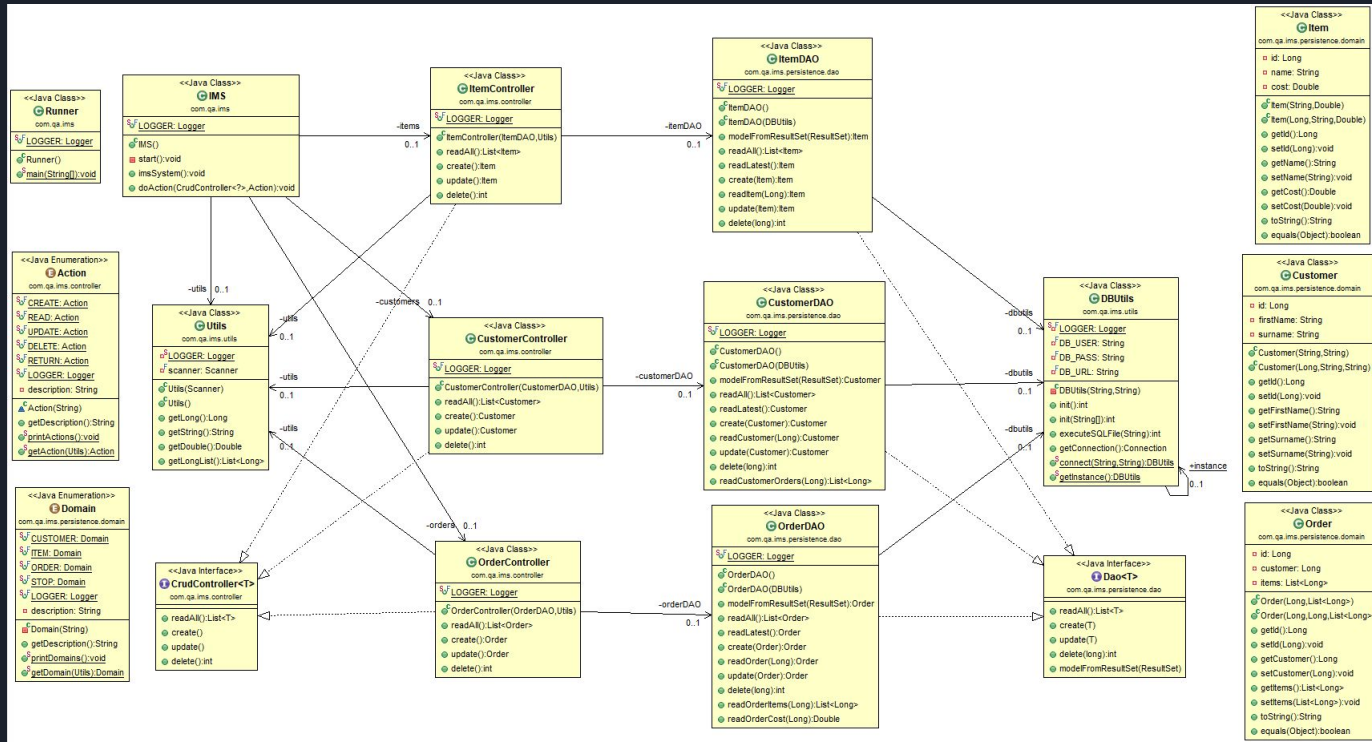`Order`
IM-18

Planning - ERD

Planning - UML

CI and Version Control

Version control was done using Git and GitHub.

The feature branch model was used.

Testing

Tests were used on most classes

- Customer, Item, Order
- CustomerDAO, ItemDAO, OrderDAO
- CustomerController, ItemController, OrderController
- Enums: Action, Domain

A mixture of unit and integration tests were used

Testing Coverage

| Element	Coverage		Covered Instructio...	Missed Instructions	Total Instructions
⌄ 📂 ims	▰▰▰	93.6 %	5,040	345	5,385
⌄ 🗁 src/main/java	▰▰	86.9 %	2,183	330	2,513
⌄ ⊞ com.qa.ims	▮	0.0 %	0	173	173
› 🗋 IMS.java	▰▰	0.0 %	0	157	157
› 🗋 Runner.java	▮	0.0 %	0	16	16
⌄ ⊞ com.qa.ims.utils	▰	58.2 %	160	115	275
› 🗋 Utils.java	▰	2.8 %	3	105	108
› 🗋 DBUtils.java	▰▰▰	94.0 %	157	10	167
⌄ ⊞ com.qa.ims.controller	▰▰	95.8 %	484	21	505
› 🗋 Action.java	▰▰	82.4 %	98	21	119
› 🗋 CustomerController.java	▰▰	100.0 %	110	0	110
› 🗋 ItemController.java	▰▰	100.0 %	110	0	110
› 🗋 OrderController.java	▰▰▰	100.0 %	166	0	166
⌄ ⊞ com.qa.ims.persistence.domain	▰	95.8 %	474	21	495
› 🗋 Domain.java	▰▰	80.0 %	84	21	105
› 🗋 Customer.java	▰▰	100.0 %	130	0	130
› 🗋 Item.java	▰▰	100.0 %	130	0	130
› 🗋 Order.java	▰▰	100.0 %	130	0	130
⌄ ⊞ com.qa.ims.persistence.dao	▰▰▰	100.0 %	1,065	0	1,065
› 🗋 CustomerDAO.java	▰▰	100.0 %	352	0	352
› 🗋 ItemDAO.java	▰▰	100.0 %	279	0	279
› 🗋 OrderDAO.java	▰▰	100.0 %	434	0	434
› 🗁 src/test/java	▰▰▰	99.5 %	2,857	15	2,872

Demonstration

Let's run through some user stories...

Sprint Review

Overall, all major objectives were met.

- Codebase is complete
- Tests are done, with good coverage
- Repository was used and well maintained
- Planning and documentation are complete

However, one objective was missed: Connecting to a GCP MySQL instance

With more time, or better time management, this objective could have been completed as well

Sprint Retrospective

I think the project went very well

- Most of the specification was met to a good standard
- I learned several new technologies

Improvements can still be made

- With new knowledge, I will have a better understanding of how to approach projects in the future
- Minor time management issues (UML diagram)

Conclusion

This project was an overall success, and the technologies and experience I learned from it will be very valuable moving forwards, both for my career and my personal projects.

The End