

# **QA - Project Feb 2022**

## Inventory Management System

---

Asad Bajwa

# Specifications of Project

**The goal and main objective of this project is to create an Inventory Management System (IMS) that a user can interact with via a Command Line Interface (CLI)**

The project will aim to achieve this by carrying out the following:

- Adding a Customer, Item and Order to the system
- Viewing all Customers, Items and Orders on the system
- Update Items, Customers and Orders on the system

# Meeting Standards and Requirements

- User stories were created along with tasks to carry out these scenarios using Jira
- Relational database on MySQL must be created for Customers, Items and Orders
- Unit testing was to be and should have been conducted \*
- Risk assessment to gain an overall scope of what issues could occur and how best to overcome them
- Using GitHub as Version Control System

\* = testing made me cry ;(

Epic		
▼	IMS-15 <u>Create code for Java</u>	
	IMS-19 Go back through notes to understand	DONE
▼	IMS-16 <u>Create IMS</u>	
	IMS-17 Create Code in Java	DONE
	IMS-18 Specify code for an IMS	DONE
▼	IMS-20 <u>SQL Database migrate together</u>	
	IMS-21 As a user I need to add edit or delete or update an item	DONE
	IMS-22 Link together with Java	DONE
▼	IMS-23 <u>Make a Readme file</u>	
	IMS-24 As a user I need to understand how the application works	DONE
▼	IMS-25 <u>ERD diagram</u>	
	IMS-26 Understand the concept of ERDs	DONE
	IMS-27 Create presentation for QA	DONE
▼	IMS-28 <u>Create presentation for QA</u>	
	IMS-29 As a student I need to present this project to my trainer so that I can pass and move further in course	DONE

## JIRA Planning

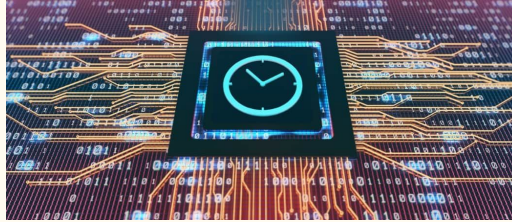
**Jira was used for planning - A total of 9 epics were used**

Epic		
IMS-23	Make a Readme file	
IMS-24	As a user I need to understand how the application works	DONE
IMS-25	ERD diagram	
IMS-26	Understand the concept of ERDs	DONE
IMS-27	Create presentation for QA	DONE
IMS-28	Create presentation for QA	
IMS-29	As a student I need to present this project to my trainer so that I can pass and move further in course	DONE
IMS-30	Create database for Customers	
IMS-31	As a user I need a database which can store customer information	DONE
IMS-32	Create a database for Items	
IMS-33	Make a database that can store Item details	DONE
IMS-34	Create database for orders	
IMS-35	Make a database that will store order details	DONE
IMS-36	Create Testing methods	
IMS-37	As a developer I need to run testing on my program to ensure my work meets the industry standards	IN PROGRESS
IMS-38	Unfortunately could not complete - testing was not a strong point for me - need to revisit and liaise with trainers and do out of hours learning. Lack of un...	IN PROGRESS

## JIRA continued

**Bugs were highlighted in Red**....of course it was testing....which made me cry.....again.

# Demo



# Time

eclipse-workspace - AB-IMS/src/main/java/com/qa/ims/Runner.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

- AB-IMS [AB-IMS master]
- com.qa.ims
  - src/main/java
    - IMS.java
    - package-info.java
    - Runner.java
  - com.qa.ims.controller
    - Action.java
    - CrudController.java
    - CustomerController.java
    - ItemController.java
    - OrderController.java
    - package-info.java
  - com.qa.ims.exceptions
  - com.qa.ims.persistence
    - com.qa.ims.persistence.dao
      - com.qa.ims.persistence.domain
        - Customer.java
        - Domain.java
        - ItemModel.java
        - OrderModel.java
        - package-info.java
  - com.qa.ims.utils
  - src/main/resources
  - src/test/java
    - com.qa.ims.controllers
    - com.qa.ims.persistence.dao
    - com.qa.ims.persistence.domain
  - src/test/resources
  - JRE System Library [JavaSE-1.8]
  - Maven Dependencies
  - src
    - main
    - test
  - target
  - entries.log
  - LICENSE.md
  - pom.xml
  - README.md
  - HelloWorld
  - testing

Runner.java

```
1 package com.qa.ims;
2
3 import org.apache.logging.log4j.LogManager;
4
5 public class Runner {
6
7     public static final Logger LOGGER = LogManager.getLogger();
8
9     public static void main(String[] args) {
10         IMS ims = new IMS();
11         ims.runSystem();
12         LOGGER.info("SO LONG!");
13     }
14 }
15
16
17
18
19
```

Outline

- com.qa.ims
  - Runner
    - LOGGER: Logger
    - main(String[]): void

Problems | Javadoc | Declaration | Console | Coverage

Runner [Java Application] C:\Users\d41786.g2\psoaf\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86\_64.17.0.2.v20220201-1208\jre\bin\javaw.exe (21 Feb 2022, 10:49:27)

Welcome to the Inventory Management System!  
which entity would you like to use?  
CUSTOMER: Information about customers  
ITEM: Individual Items  
ORDER: Purchases of items  
STOP: To Close the application

Writable Smart Insert 19: 1: 310

# Command Line User Interface



The screenshot shows the Eclipse IDE's console window. The title bar indicates the runner is a Java Application. The console output displays a welcome message and a list of available entities for the Inventory Management System.

```
Runner [Java Application] C:\Users\44786\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_17.0.2.v20220201-1208\jre\bin\javaw.exe (21 Feb 2022, 10:49:27)
Welcome to the Inventory Management System!
Which entity would you like to use?
CUSTOMER: Information about customers
ITEM: Individual Items
ORDER: Purchases of items
STOP: To close the application
```

**Initial message greeting and asking which entity would like to be performed**

---

# CLI continued

The screenshot shows a database management interface. On the left, there's a sidebar with 'Administration' and 'Schemas' tabs. Under 'Schemas', the 'customers' table is selected. Below this, the table's columns are listed: 'id' (int AI PK), 'first\_name' (varchar(40)), and 'surname' (varchar(40)). The main area displays a 'Result Grid' with a table of customer data. Below the grid, there's an 'Output' section showing an 'Action Output' log with a table of actions performed, including 'use ims' and 'SELECT \* FROM customers LIMIT 0, 1000'.

id	first_name	surname
1	John	Smith
2	Mo	Salah
4	Barry	Hearn
5	Best	Trainerever
NULL	NULL	NULL

#	Time	Action
26	10:48:31	use ims
27	10:48:31	SELECT * FROM customers LIMIT 0, 1000
28	10:48:41	use ims
29	10:48:41	SELECT * FROM item LIMIT 0, 1000
30	12:17:47	use ims
31	12:17:47	SELECT * FROM customers LIMIT 0, 1000

Further examples - and you make a feature too!  
(id5)

The screenshot shows a Java application console window titled 'Runner [Java Application]'. The application is an 'Inventory Management System' CLI. It prompts the user to 'Welcome to the Inventory Management System!' and 'Which entity would you like to use?'. The user enters 'customer'. The application then prompts 'What would you like to do with customer:'. The user enters 'read'. The application then displays the details for the customer with id=1: 'id:1 first name:john surname:smith'. The user then enters 'id:2 first name:Mo surname:Salah'. The application then displays the details for the customer with id=2: 'id:2 first name:Mo surname:Salah'. The user then enters 'id:4 first name:Barry surname:Hearn'. The application then displays the details for the customer with id=4: 'id:4 first name:Barry surname:Hearn'. The user then enters 'id:5 first name:Best surname:Trainerever'. The application then displays the details for the customer with id=5: 'id:5 first name:Best surname:Trainerever'. The user then enters 'What would you like to do with customer:'. The application then prompts 'CREATE: To save a new entity into the database', 'READ: To read an entity from the database', 'UPDATE: To change an entity already in the database', 'DELETE: To remove an entity from the database', and 'RETURN: To return to domain selection'.

```
Runner [Java Application] C:\Users\44786\p2\pool\plugins\org.eclipse.justj.open
Welcome to the Inventory Management System!
Which entity would you like to use?
CUSTOMER: Information about customers
ITEM: Individual Items
ORDER: Purchases of items
STOP: To close the application
customer
What would you like to do with customer:
CREATE: To save a new entity into the database
READ: To read an entity from the database
UPDATE: To change an entity already in the database
DELETE: To remove an entity from the database
RETURN: To return to domain selection
read
id:1 first name:john surname:smith
id:2 first name:Mo surname:Salah
id:4 first name:Barry surname:Hearn
id:5 first name:Best surname:Trainerever
What would you like to do with customer:
CREATE: To save a new entity into the database
READ: To read an entity from the database
UPDATE: To change an entity already in the database
DELETE: To remove an entity from the database
RETURN: To return to domain selection
```





# Testing

Total coverage across tests was 36.4%

The screenshot shows an IDE with the following components:

- Editor:** Displays `Runner.java` with the following code:

```
1 package com.qa.ims;
2
3 import org.apache.logging.log4j.LogManager;
4
5 public class Runner {
6
7     public static final Logger LOGGER = LogManager.getLogger();
8
9     public static void main(String[] args) {
10         IMS ims = new IMS();
11         ims.imsSystem();
12         LOGGER.info("SO LONG!");
13     }
14 }
```
- Outline:** Shows the project structure with `com.qa.ims` and `Runner` class.
- Problems:** Shows a single problem: `java (21 Feb 2022 12:38:57)`.
- Coverage:** A table showing the coverage results for the `AB-IMS` element.

Element	Coverage	Covered Instructio...	Missed Instructions	Total Instructions
> AB-IMS	36.4 %	886	1,548	2,434

# Summary

To summarise, I learnt that each detail must be followed very carefully and every single aspect of the project must be covered thoroughly.

Things that I can improve on are my time management and definitely understanding Testing more, I will be going back through videos out of class time to ensure I get up to speed.

I feel I done well is get to grips with my coding skills, simplify and manage tasks well and hopefully I have presented well to you today.

Thank you for taking the time to listen to my presentation, please feel free to ask any questions.