INVENTORY MANAGEMENT SYSTEM PROJECT

Sam McIvor

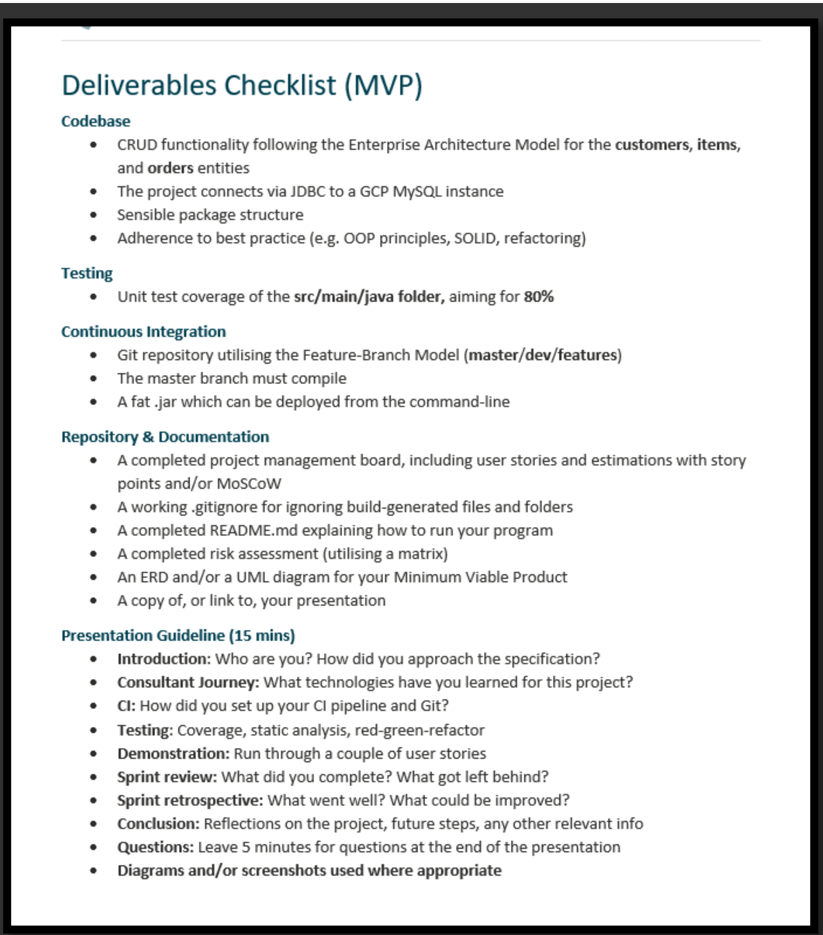
INTRODUCTION

Approaching the Specification:

- Looked through the requirements
- Studied the code in the IMS
- Planned out tasks, matching them up with User-Stories on my JIRA

Started with creating Items, due to its similarity to Customers

Then planned out how to structure Orders & Order-Items



Deliverables Checklist (MVP)

Codebase

- CRUD functionality following the Enterprise Architecture Model for the **customers**, **items**, and **orders** entities
- The project connects via JDBC to a GCP MySQL instance
- Sensible package structure
- Adherence to best practice (e.g. OOP principles, SOLID, refactoring)

Testing

- Unit test coverage of the **src/main/java folder,** aiming for **80%**

Continuous Integration

- Git repository utilising the Feature-Branch Model (**master/dev/features**)
- The master branch must compile
- A fat .jar which can be deployed from the command-line

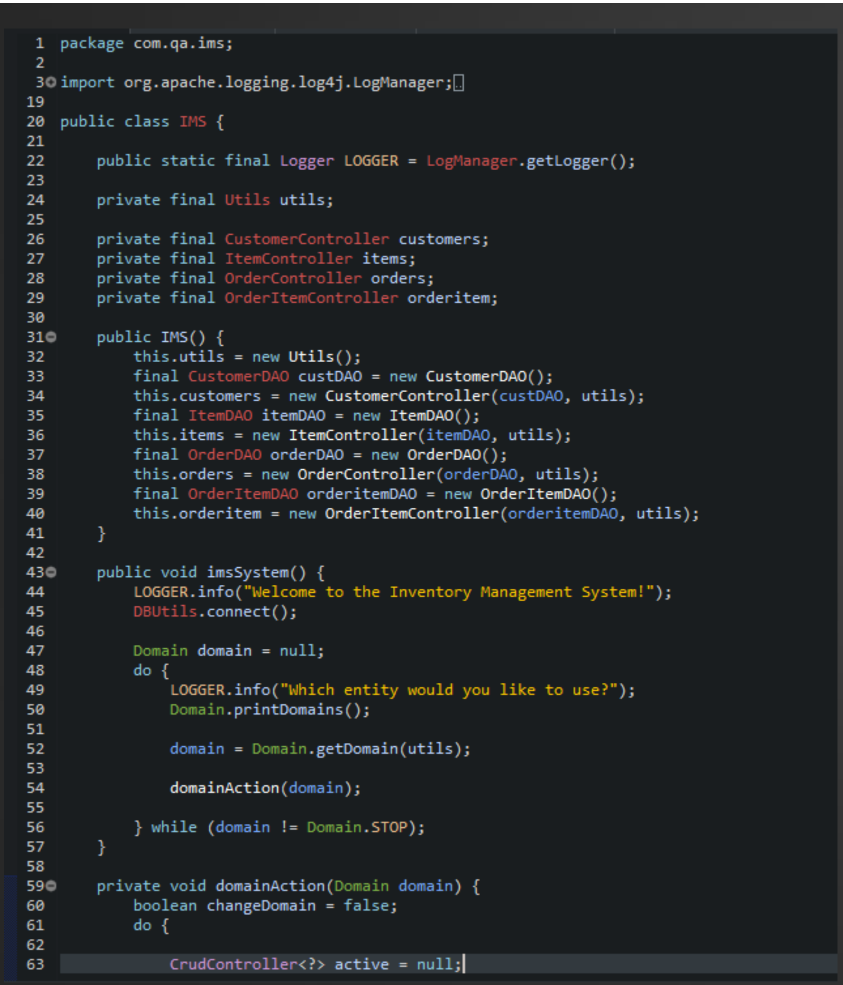
Repository & Documentation

- A completed project management board, including user stories and estimations with story points and/or MoSCoW
- A working .gitignore for ignoring build-generated files and folders
- A completed README.md explaining how to run your program
- A completed risk assessment (utilising a matrix)
- An ERD and/or a UML diagram for your Minimum Viable Product
- A copy of, or link to, your presentation

Presentation Guideline (15 mins)

- **Introduction:** Who are you? How did you approach the specification?
- **Consultant Journey:** What technologies have you learned for this project?
- **CI:** How did you set up your CI pipeline and Git?
- **Testing:** Coverage, static analysis, red-green-refactor
- **Demonstration:** Run through a couple of user stories
- **Sprint review:** What did you complete? What got left behind?
- **Sprint retrospective:** What went well? What could be improved?
- **Conclusion:** Reflections on the project, future steps, any other relevant info
- **Questions:** Leave 5 minutes for questions at the end of the presentation
- **Diagrams and/or screenshots used where appropriate**

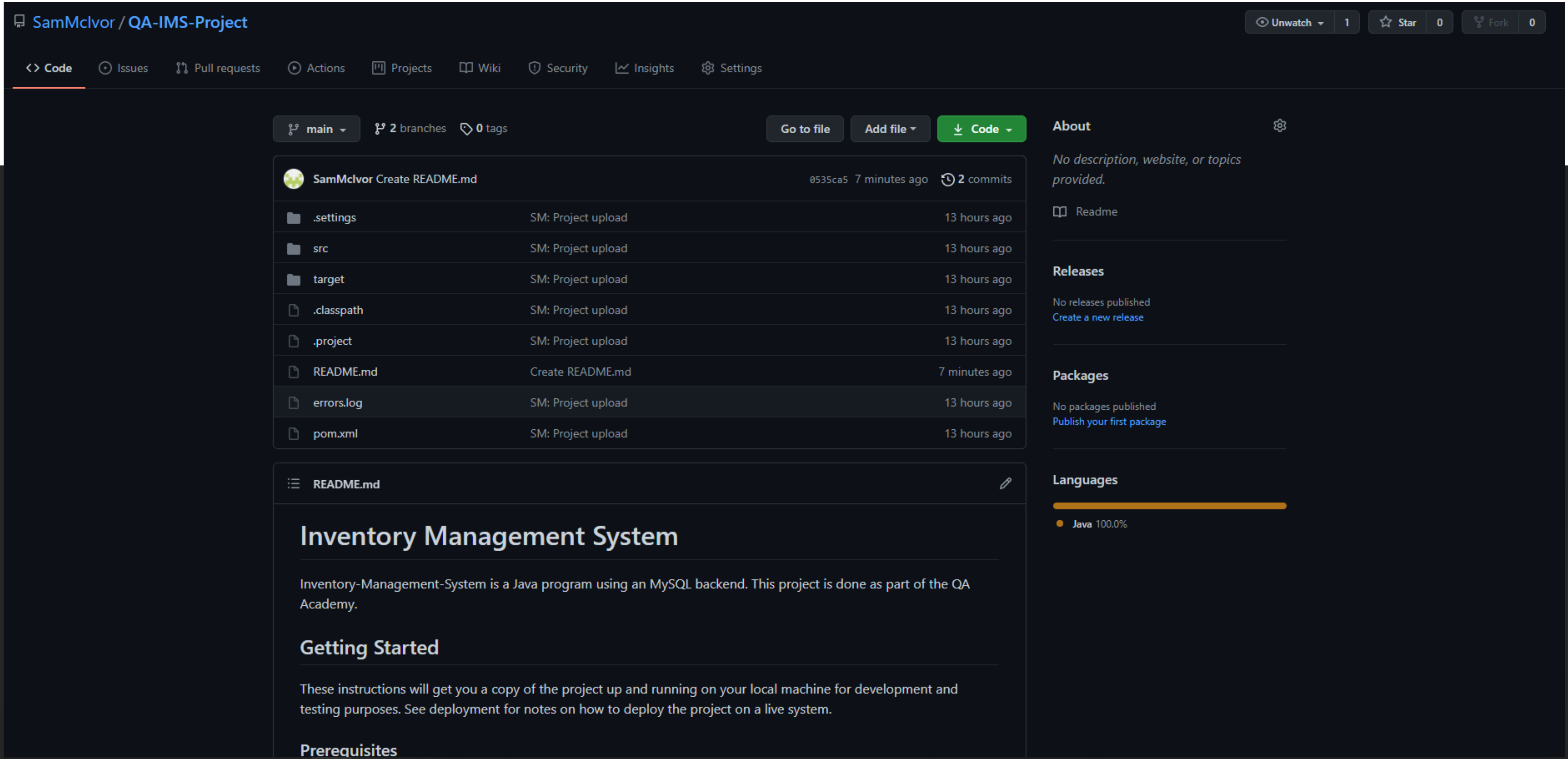
CONSULTANT JOURNEY



```java
package com.qa.ims;

import org.apache.logging.log4j.LogManager;

public class IMS {

    public static final Logger LOGGER = LogManager.getLogger();

    private final Utils utils;

    private final CustomerController customers;
    private final ItemController items;
    private final OrderController orders;
    private final OrderItemController orderitem;

    public IMS() {
        this.utils = new Utils();
        final CustomerDAO custDAO = new CustomerDAO();
        this.customers = new CustomerController(custDAO, utils);
        final ItemDAO itemDAO = new ItemDAO();
        this.items = new ItemController(itemDAO, utils);
        final OrderDAO orderDAO = new OrderDAO();
        this.orders = new OrderController(orderDAO, utils);
        final OrderItemDAO orderitemDAO = new OrderItemDAO();
        this.orderitem = new OrderItemController(orderitemDAO, utils);
    }

    public void imsSystem() {
        LOGGER.info("Welcome to the Inventory Management System!");
        DBUtils.connect();

        Domain domain = null;
        do {
            LOGGER.info("Which entity would you like to use?");
            Domain.printDomains();

            domain = Domain.getDomain(utils);

            domainAction(domain);

        } while (domain != Domain.STOP);
    }

    private void domainAction(Domain domain) {
        boolean changeDomain = false;
        do {

            CrudController<?> active = null;
```

Learned Technologies:

- Java
- Maven
- Junit Testing
- Jira
- Repository Management



No description, website, or topics provided.

No releases published
Create a new release

No packages published
Publish your first package

SamMcIvor Create README.md 0535ca5 7 minutes ago 2 commits

.settings	SM: Project upload	13 hours ago
src	SM: Project upload	13 hours ago
target	SM: Project upload	13 hours ago
.classpath	SM: Project upload	13 hours ago
.project	SM: Project upload	13 hours ago
README.md	Create README.md	7 minutes ago
errors.log	SM: Project upload	13 hours ago
pom.xml	SM: Project upload	13 hours ago

README.md

Inventory Management System

Inventory-Management-System is a Java program using an MySQL backend. This project is done as part of the QA Academy.

Getting Started

These instructions will get you a copy of the project up and running on your local machine for development and testing purposes. See deployment for notes on how to deploy the project on a live system.

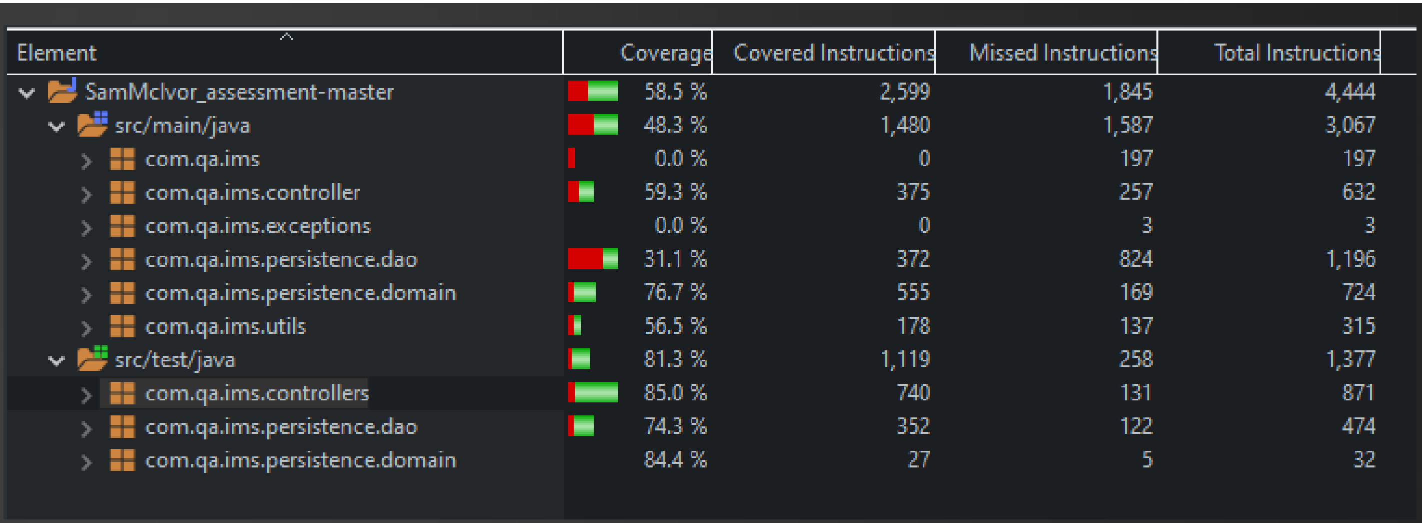
Prerequisites

TESTING

- Achieved approximately 59% testing coverage

- Primarily using JUnit and Mockito

- Went through re-writing code to better fit tests as well as changing tests to fit changes to the IMS



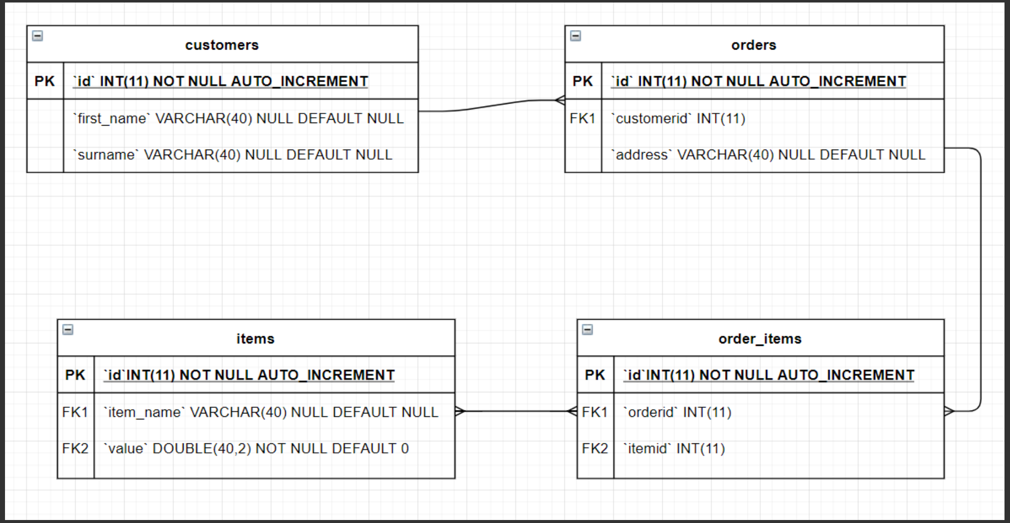
Element	Coverage	Covered Instructions	Missed Instructions	Total Instructions
SamMcIvor_assessment-master	58.5 %	2,599	1,845	4,444
src/main/java	48.3 %	1,480	1,587	3,067
com.qa.ims	0.0 %	0	197	197
com.qa.ims.controller	59.3 %	375	257	632
com.qa.ims.exceptions	0.0 %	0	3	3
com.qa.ims.persistence.dao	31.1 %	372	824	1,196
com.qa.ims.persistence.domain	76.7 %	555	169	724
com.qa.ims.utils	56.5 %	178	137	315
src/test/java	81.3 %	1,119	258	1,377
com.qa.ims.controllers	85.0 %	740	131	871
com.qa.ims.persistence.dao	74.3 %	352	122	474
com.qa.ims.persistence.domain	84.4 %	27	5	32

OPERATIONAL STRUCTURE

IMS Operations

- Customers
  - Create
  - Read
  - Update
  - Delete
- Items
  - Create
  - Read
  - Update
  - Delete
- Order
  - Create
  - Read
  - Delete
- Edit_Order (order_items)
  - Add_item
  - Order_cost
  - Delete_item
  - Return (not shown in ui)

ERD



	customers
PK	`id` INT(11) NOT NULL AUTO_INCREMENT
	`first_name` VARCHAR(40) NULL DEFAULT NULL
	`surname` VARCHAR(40) NULL DEFAULT NULL

	orders
PK	`id` INT(11) NOT NULL AUTO_INCREMENT
FK1	`customerid` INT(11)
	`address` VARCHAR(40) NULL DEFAULT NULL

	items
PK	`id`INT(11) NOT NULL AUTO_INCREMENT
FK1	`item_name` VARCHAR(40) NULL DEFAULT NULL
FK2	`value` DOUBLE(40,2) NOT NULL DEFAULT 0

	order_items
PK	`id`INT(11) NOT NULL AUTO_INCREMENT
FK1	`orderid` INT(11)
FK2	`itemid` INT(11)

DEMONSTRATION

SPRINT REVIEW

Completed

- All necessary tasks completed
  - Includes all Customer / Item operations and their user stories
- Majority Order Operations
  - Create, Read, Update, Delete
  - Add item, Calculate cost, Delete item

Returned to Backlog

- Full coverage testing

SPRINT RETROSPECTIVE

Accomplishments

- Created a working application with all necessary functionalities
- Focussed on the most important tasks and moved out to the smaller tasks.

Areas for Improvement

- Better use of Jira
- More time focussed on testing
- Better time management

CONCLUDING REMARKS

Conclusions

- Picked up a lot of useful skills and techniques
- Need to work on my management skills
  - Project management
  - Time management
- Would try to aim for more test coverage

If I were to continue this project…

- Clean up the UI, adding an easier system to input options, using integers for example.
- Add a recycle bin functionality to be able to retrieve accidentally deleted information.

ANY QUESTIONS?