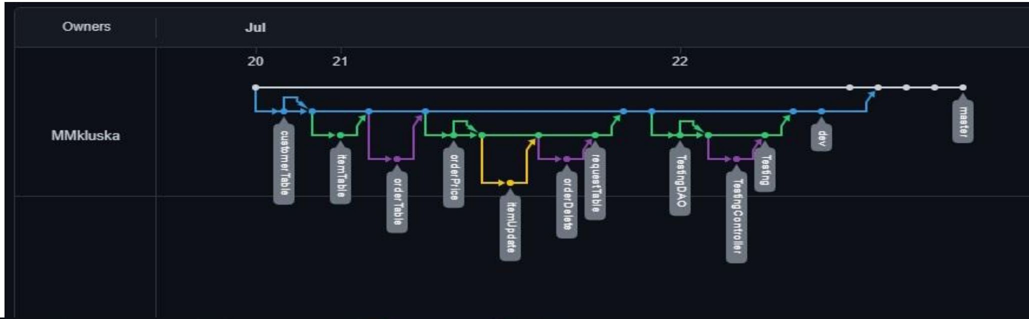
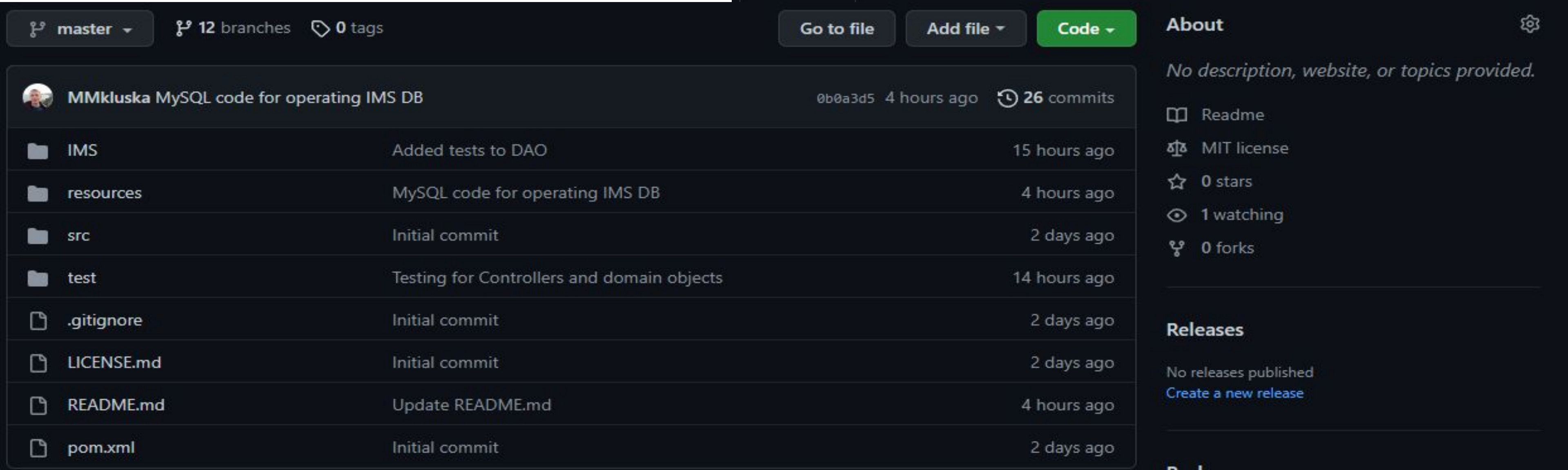
INVENTORY MANAGEMENT SYSTEM

By Mateusz Kluska

Git Homepage

- github.com/MMkluska/IMS
- This is my main repository
- I made 27 Commits following MVP
- To right my network
- To bottom my homepage





MMkluska MySQL code for operating IMS DB 0b0a3d5 4 hours ago 26 commits

IMS	Added tests to DAO	15 hours ago
resources	MySQL code for operating IMS DB	4 hours ago
src	Initial commit	2 days ago
test	Testing for Controllers and domain objects	14 hours ago
.gitignore	Initial commit	2 days ago
LICENSE.md	Initial commit	2 days ago
README.md	Update README.md	4 hours ago
pom.xml	Initial commit	2 days ago

About

No description, website, or topics provided.

Readme

MIT license

0 stars

1 watching

0 forks

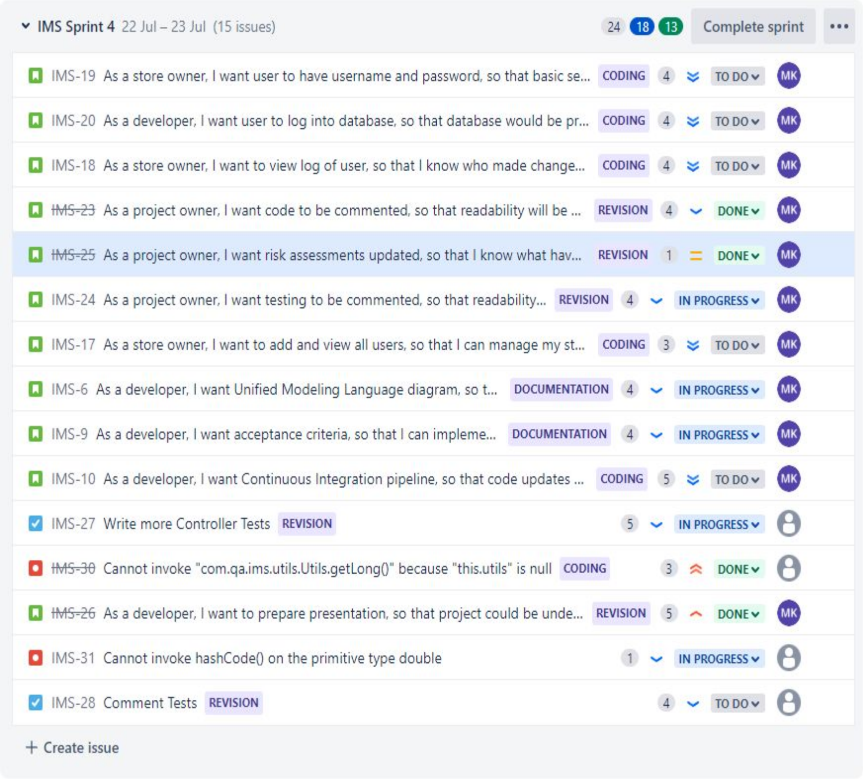
Releases

No releases published

Create a new release

Jira - Agile Project Management

- Link in README.md
- I made 3 Epics :
- Documentation - User Stories related to preparing documents for project
- Coding - User Stories related to coding
- Revision - User Stories related to Testing and logging
- Initially 3 Sprint with around 30 Story Points each
- Kanban board with To do, In progress, Done
- Following MoSCoW for User Stories
- Due to unfinished work I started 4th Sprint
- I added bugs and tasks I encounter during project

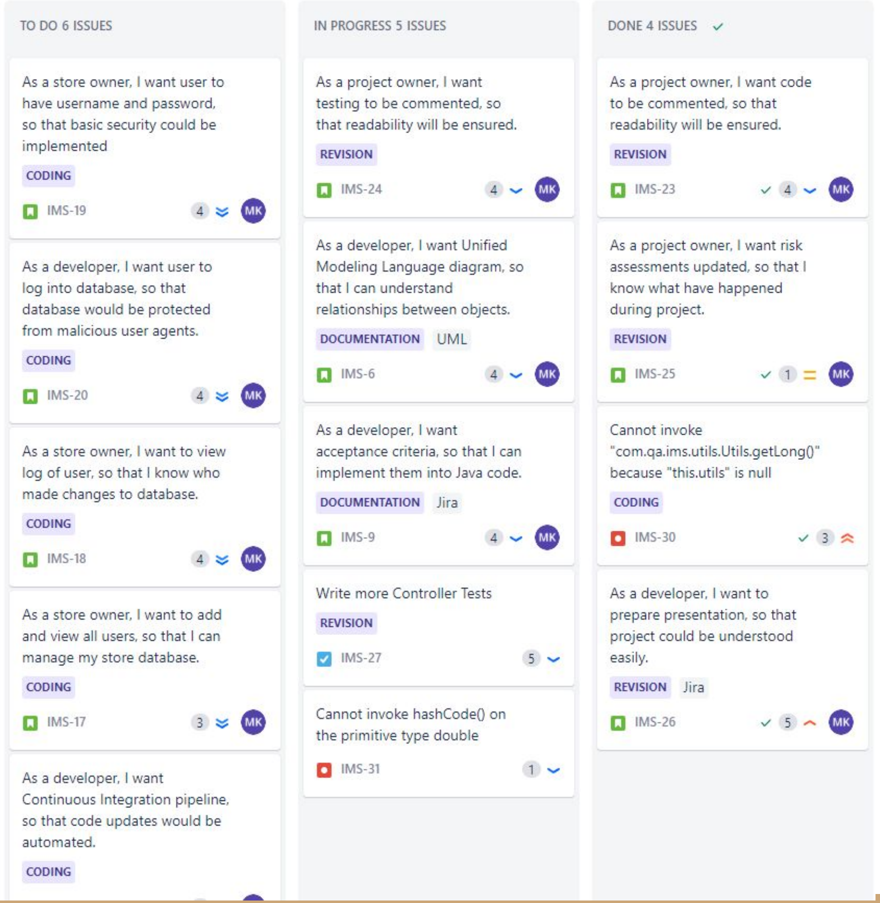


IMS Sprint 4 22 Jul – 23 Jul (15 issues) 24 18 13 Complete sprint •••

IMS-19	As a store owner, I want user to have username and password, so that basic se...	CODING	4	TO DO	MK
IMS-20	As a developer, I want user to log into database, so that database would be pr...	CODING	4	TO DO	MK
IMS-18	As a store owner, I want to view log of user, so that I know who made change...	CODING	4	TO DO	MK
~~IMS-23~~	As a project owner, I want code to be commented, so that readability will be ...	REVISION	4	DONE	MK
~~IMS-25~~	As a project owner, I want risk assessments updated, so that I know what hav...	REVISION	1	DONE	MK
IMS-24	As a project owner, I want testing to be commented, so that readability...	REVISION	4	IN PROGRESS	MK
IMS-17	As a store owner, I want to add and view all users, so that I can manage my st...	CODING	3	TO DO	MK
IMS-6	As a developer, I want Unified Modeling Language diagram, so t...	DOCUMENTATION	4	IN PROGRESS	MK
IMS-9	As a developer, I want acceptance criteria, so that I can impleme...	DOCUMENTATION	4	IN PROGRESS	MK
IMS-10	As a developer, I want Continuous Integration pipeline, so that code updates ...	CODING	5	TO DO	MK
IMS-27	Write more Controller Tests	REVISION	5	IN PROGRESS	
~~IMS-30~~	Cannot invoke "com.qa.ims.utils.Utils.getLong()" because "this.utils" is null	CODING	3	DONE	
~~IMS-26~~	As a developer, I want to prepare presentation, so that project could be unde...	REVISION	5	DONE	MK
IMS-31	Cannot invoke hashCode() on the primitive type double		1	IN PROGRESS	
IMS-28	Comment Tests	REVISION	4	TO DO	

+ Create issue

Jira - Agile Project Management



TO DO 6 ISSUES

As a store owner, I want user to have username and password, so that basic security could be implemented
CODING
IMS-19 4 MK

As a developer, I want user to log into database, so that database would be protected from malicious user agents.
CODING
IMS-20 4 MK

As a store owner, I want to view log of user, so that I know who made changes to database.
CODING
IMS-18 4 MK

As a store owner, I want to add and view all users, so that I can manage my store database.
CODING
IMS-17 3 MK

As a developer, I want Continuous Integration pipeline, so that code updates would be automated.
CODING

IN PROGRESS 5 ISSUES

As a project owner, I want testing to be commented, so that readability will be ensured.
REVISION
IMS-24 4 MK

As a developer, I want Unified Modeling Language diagram, so that I can understand relationships between objects.
DOCUMENTATION UML
IMS-6 4 MK

As a developer, I want acceptance criteria, so that I can implement them into Java code.
DOCUMENTATION Jira
IMS-9 4 MK

Write more Controller Tests
REVISION
IMS-27 5

Cannot invoke hashCode() on the primitive type double
IMS-31 1

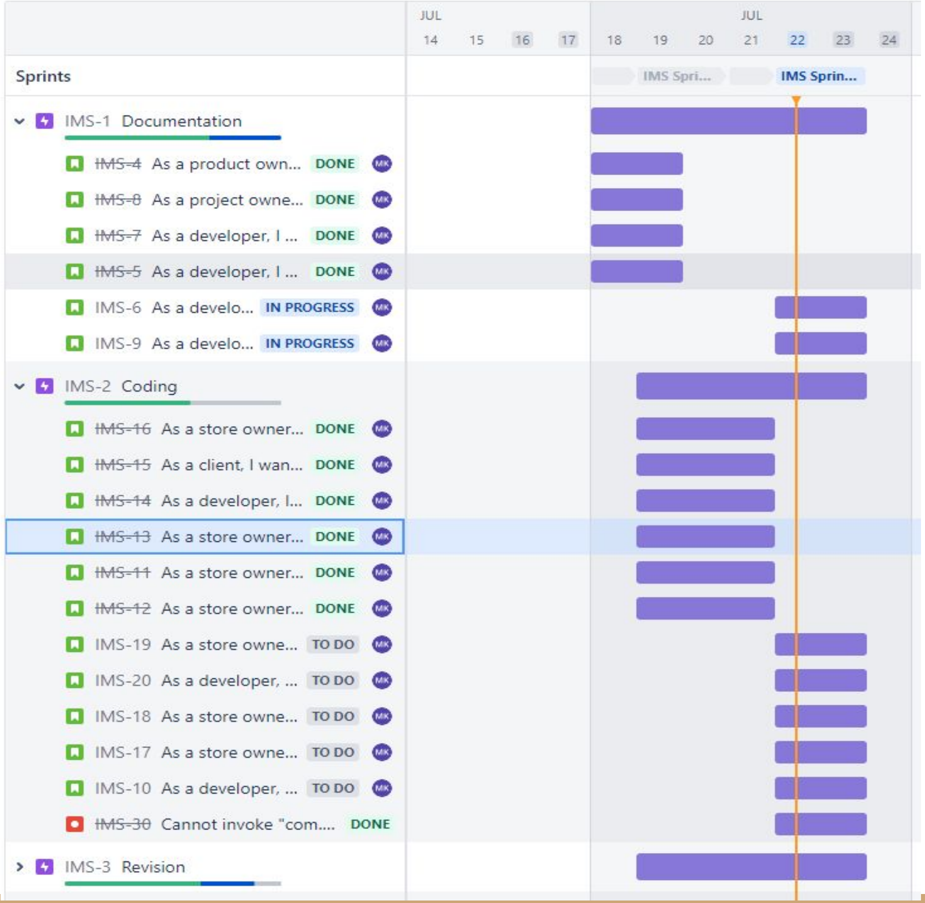
DONE 4 ISSUES ✓

As a project owner, I want code to be commented, so that readability will be ensured.
REVISION
IMS-23 ✓ 4 MK

As a project owner, I want risk assessments updated, so that I know what have happened during project.
REVISION
IMS-25 ✓ 1 = MK

Cannot invoke "com.qa.ims.utils.Utils.getLong()" because "this.utils" is null
CODING
IMS-30 ✓ 3

As a developer, I want to prepare presentation, so that project could be understood easily.
REVISION Jira
IMS-26 ✓ 5 MK



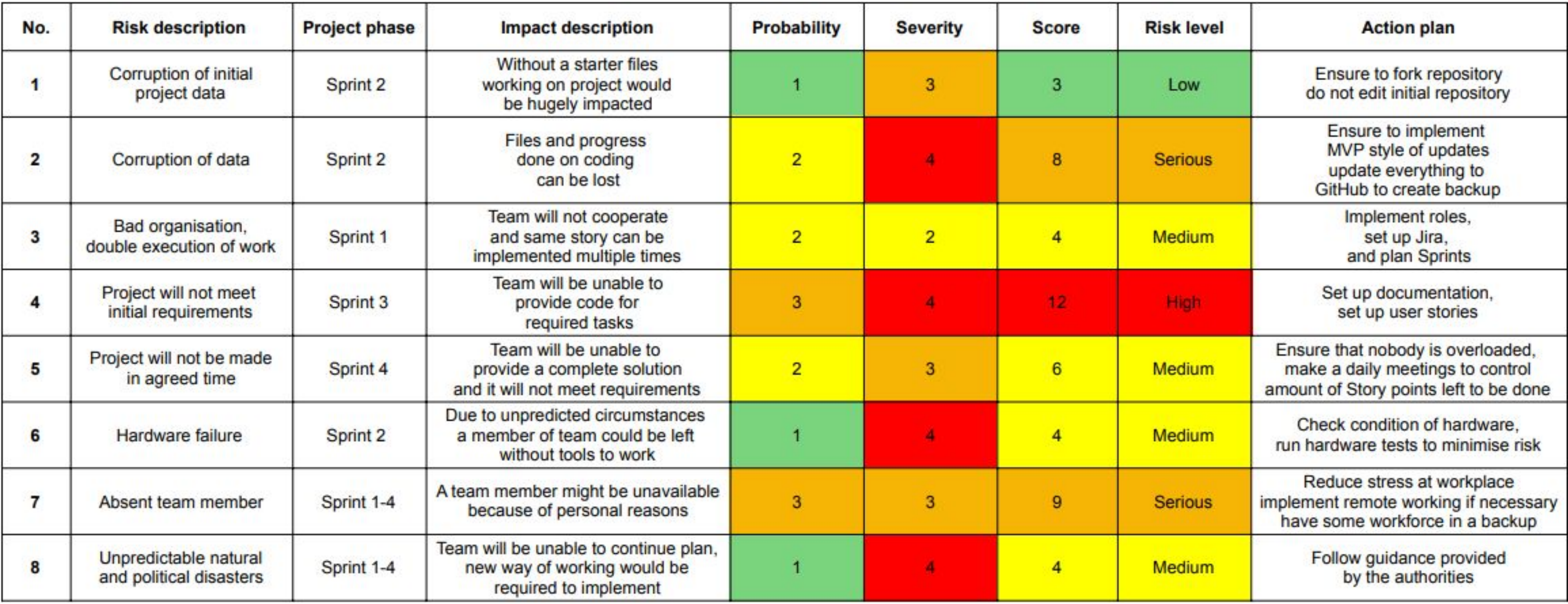
Sprints

IMS-1 Documentation
~~IMS-4~~ As a product own... DONE MK
~~IMS-8~~ As a project owne... DONE MK
~~IMS-7~~ As a developer, I ... DONE MK
~~IMS-5~~ As a developer, I ... DONE MK
IMS-6 As a develo... IN PROGRESS MK
IMS-9 As a develo... IN PROGRESS MK

IMS-2 Coding
~~IMS-16~~ As a store owner... DONE MK
~~IMS-15~~ As a client, I wan... DONE MK
~~IMS-14~~ As a developer, I... DONE MK
~~IMS-13~~ As a store owner... DONE MK
~~IMS-11~~ As a store owner... DONE MK
~~IMS-12~~ As a store owner... DONE MK
IMS-19 As a store owne... TO DO MK
IMS-20 As a developer, ... TO DO MK
IMS-18 As a store owne... TO DO MK
IMS-17 As a store owne... TO DO MK
IMS-10 As a developer, ... TO DO MK
~~IMS-30~~ Cannot invoke "com.... DONE

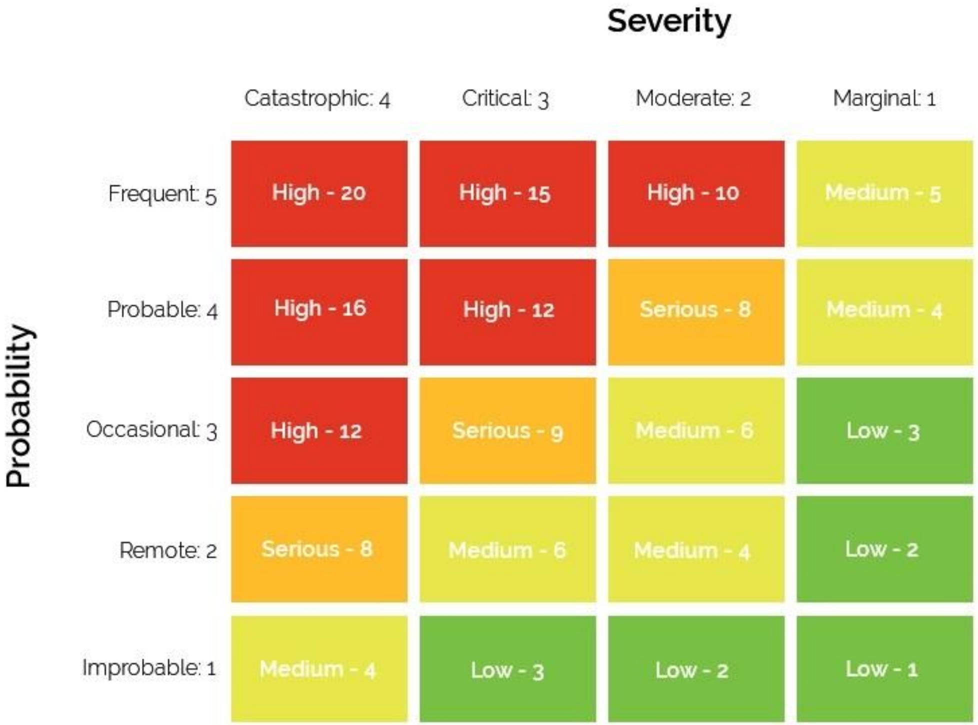
IMS-3 Revision

Risk Assessments Matrix



No.	Risk description	Project phase	Impact description	Probability	Severity	Score	Risk level	Action plan
1	Corruption of initial project data	Sprint 2	Without a starter files working on project would be hugely impacted	1	3	3	Low	Ensure to fork repository do not edit initial repository
2	Corruption of data	Sprint 2	Files and progress done on coding can be lost	2	4	8	Serious	Ensure to implement MVP style of updates update everything to GitHub to create backup
3	Bad organisation, double execution of work	Sprint 1	Team will not cooperate and same story can be implemented multiple times	2	2	4	Medium	Implement roles, set up Jira, and plan Sprints
4	Project will not meet initial requirements	Sprint 3	Team will be unable to provide code for required tasks	3	4	12	High	Set up documentation, set up user stories
5	Project will not be made in agreed time	Sprint 4	Team will be unable to provide a complete solution and it will not meet requirements	2	3	6	Medium	Ensure that nobody is overloaded, make a daily meetings to control amount of Story points left to be done
6	Hardware failure	Sprint 2	Due to unpredicted circumstances a member of team could be left without tools to work	1	4	4	Medium	Check condition of hardware, run hardware tests to minimise risk
7	Absent team member	Sprint 1-4	A team member might be unavailable because of personal reasons	3	3	9	Serious	Reduce stress at workplace implement remote working if necessary have some workforce in a backup
8	Unpredictable natural and political disasters	Sprint 1-4	Team will be unable to continue plan, new way of working would be required to implement	1	4	4	Medium	Follow guidance provided by the authorities

Risk Assessments Matrix



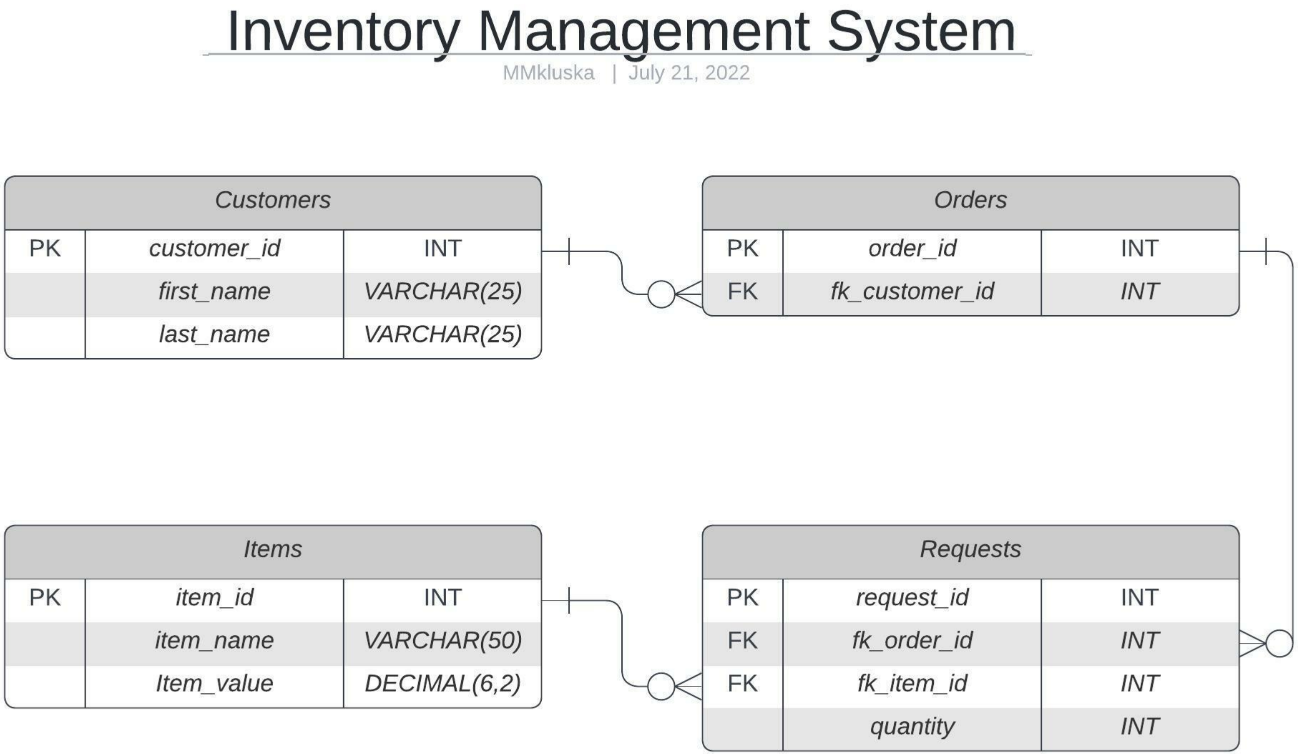
Severity

Probability

	Catastrophic: 4	Critical: 3	Moderate: 2	Marginal: 1
Frequent: 5	High - 20	High - 15	High - 10	Medium - 5
Probable: 4	High - 16	High - 12	Serious - 8	Medium - 4
Occasional: 3	High - 12	Serious - 9	Medium - 6	Low - 3
Remote: 2	Serious - 8	Medium - 6	Medium - 4	Low - 2
Improbable: 1	Medium - 4	Low - 3	Low - 2	Low - 1

Entity Relationship diagram

- Primary Keys as PK
- Foreign Keys as FK
- Following naming convention
- Requests table to avoid Many to many relationship
- Customer may have 0 to many Orders
- Order may have 0 to many Requests
- Item may have 0 to many Requests



Inventory Management System

MMkluska | July 21, 2022

Customers		
PK	*customer_id*	INT
	first_name	*VARCHAR(25)*
	last_name	*VARCHAR(25)*

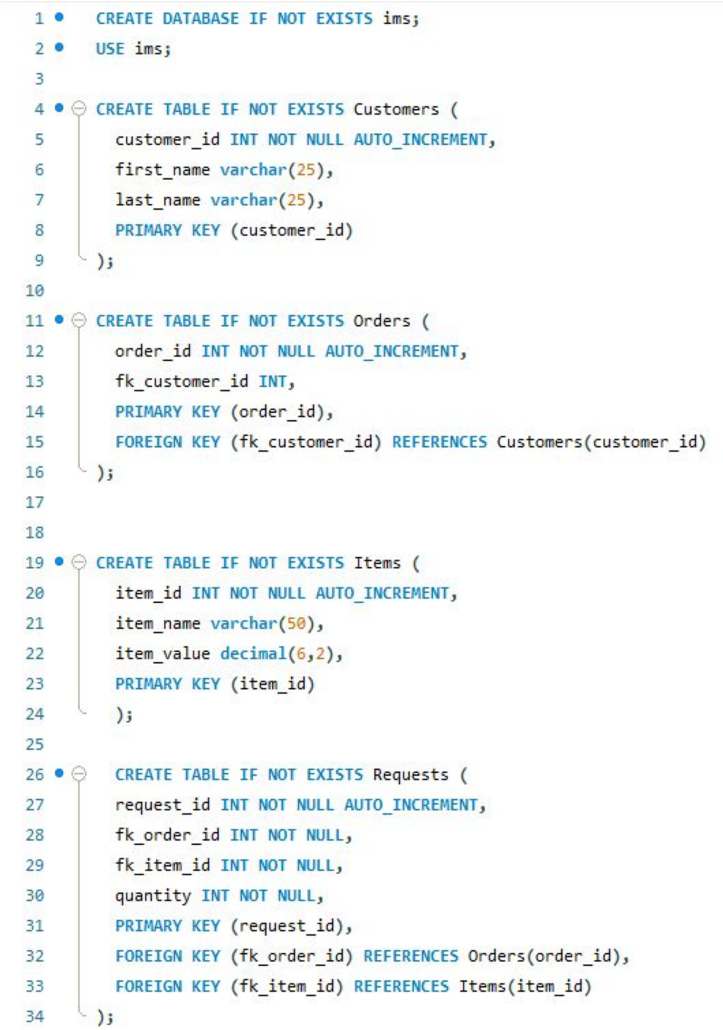
Orders		
PK	*order_id*	INT
FK	*fk_customer_id*	*INT*

Items		
PK	*item_id*	INT
	item_name	*VARCHAR(50)*
	Item_value	*DECIMAL(6,2)*

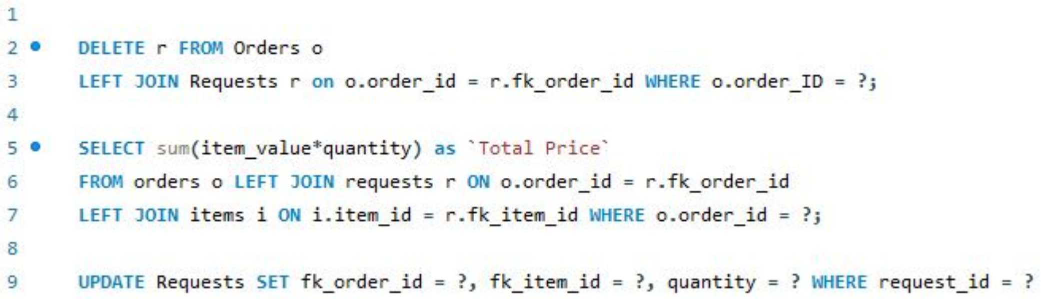
Requests		
PK	*request_id*	INT
FK	*fk_order_id*	*INT*
FK	*fk_item_id*	*INT*
	quantity	*INT*

SQL Queries

- Just a sample of code:
- To the Left code for creating a Database
- To the bottom example queries from DAO



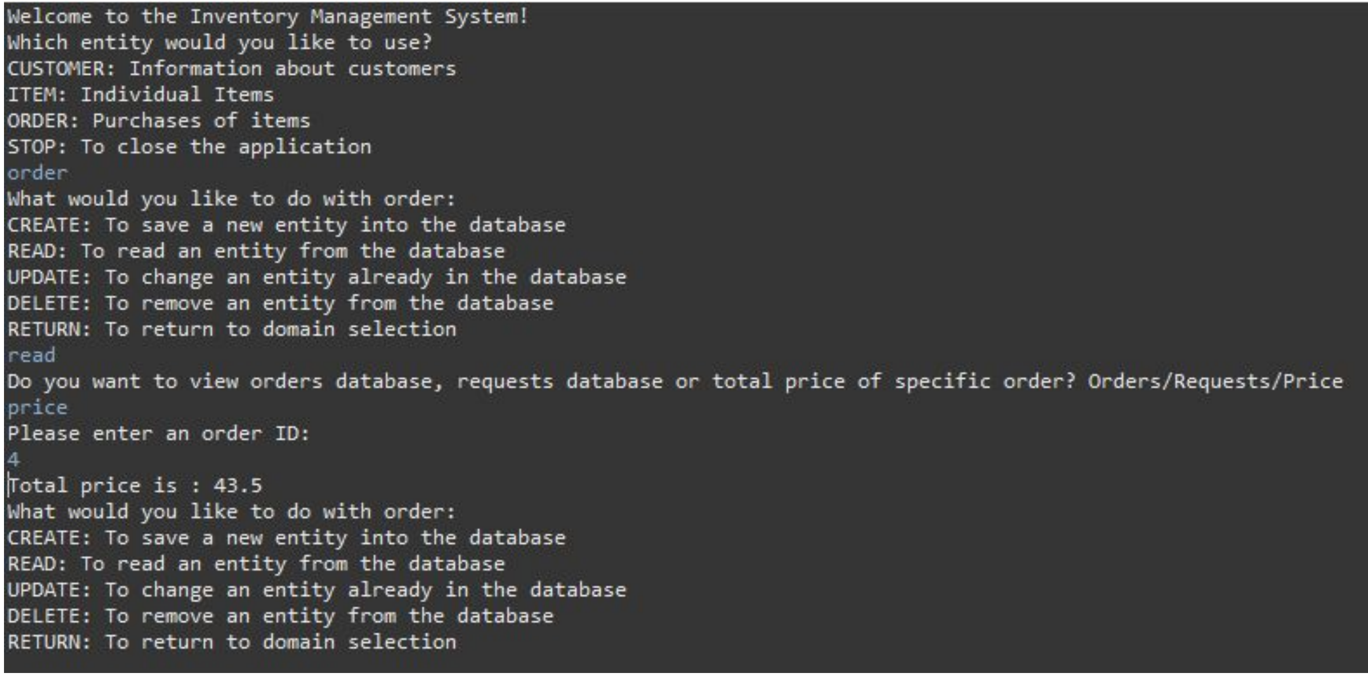
```sql
CREATE DATABASE IF NOT EXISTS ims;
USE ims;

CREATE TABLE IF NOT EXISTS Customers (
  customer_id INT NOT NULL AUTO_INCREMENT,
  first_name varchar(25),
  last_name varchar(25),
  PRIMARY KEY (customer_id)
);

CREATE TABLE IF NOT EXISTS Orders (
  order_id INT NOT NULL AUTO_INCREMENT,
  fk_customer_id INT,
  PRIMARY KEY (order_id),
  FOREIGN KEY (fk_customer_id) REFERENCES Customers(customer_id)
);


CREATE TABLE IF NOT EXISTS Items (
  item_id INT NOT NULL AUTO_INCREMENT,
  item_name varchar(50),
  item_value decimal(6,2),
  PRIMARY KEY (item_id)
  );

  CREATE TABLE IF NOT EXISTS Requests (
  request_id INT NOT NULL AUTO_INCREMENT,
  fk_order_id INT NOT NULL,
  fk_item_id INT NOT NULL,
  quantity INT NOT NULL,
  PRIMARY KEY (request_id),
  FOREIGN KEY (fk_order_id) REFERENCES Orders(order_id),
  FOREIGN KEY (fk_item_id) REFERENCES Items(item_id)
);
```



```sql

DELETE r FROM Orders o
LEFT JOIN Requests r on o.order_id = r.fk_order_id WHERE o.order_ID = ?;

SELECT sum(item_value*quantity) as `Total Price`
FROM orders o LEFT JOIN requests r ON o.order_id = r.fk_order_id
LEFT JOIN items i ON i.item_id = r.fk_item_id WHERE o.order_id = ?;

UPDATE Requests SET fk_order_id = ?, fk_item_id = ?, quantity = ? WHERE request_id = ?
```

Project demo

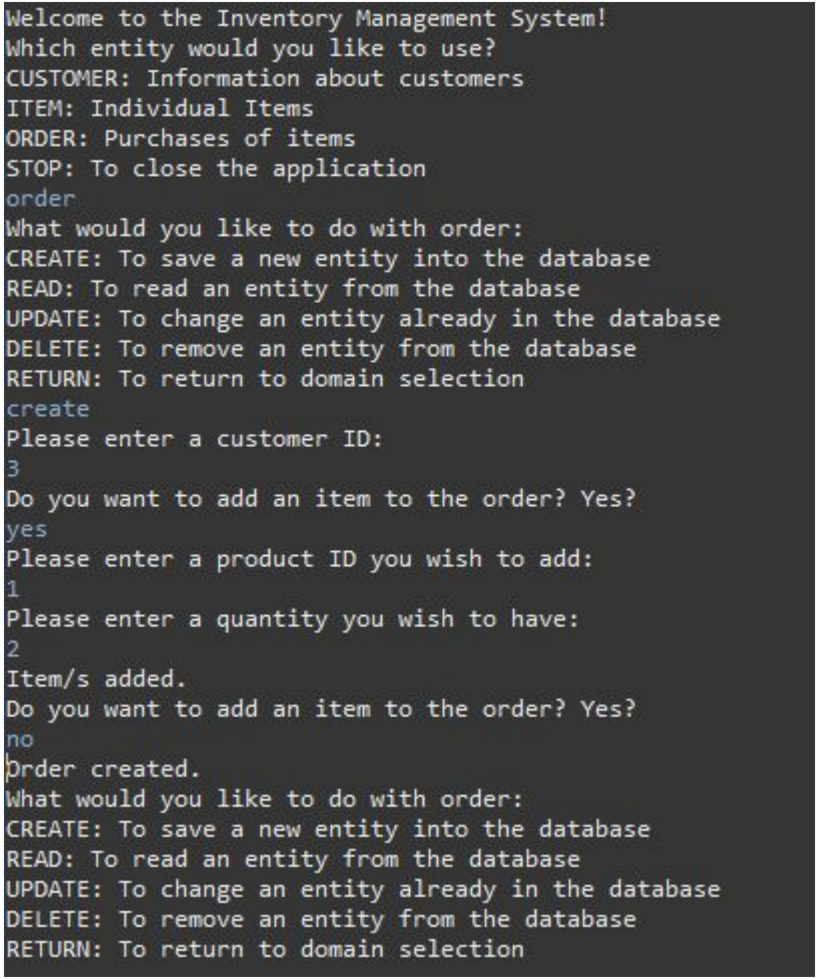
- 1. Getting total price of an order
- 2. Type order
- 3. Type read
- 4. Type price
- 5. Type order ID
- 6. You will receive total price



```
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
order
What would you like to do with order:
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
Do you want to view orders database, requests database or total price of specific order? Orders/Requests/Price
price
Please enter an order ID:
4
Total price is : 43.5
What would you like to do with order:
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
```

Project demo

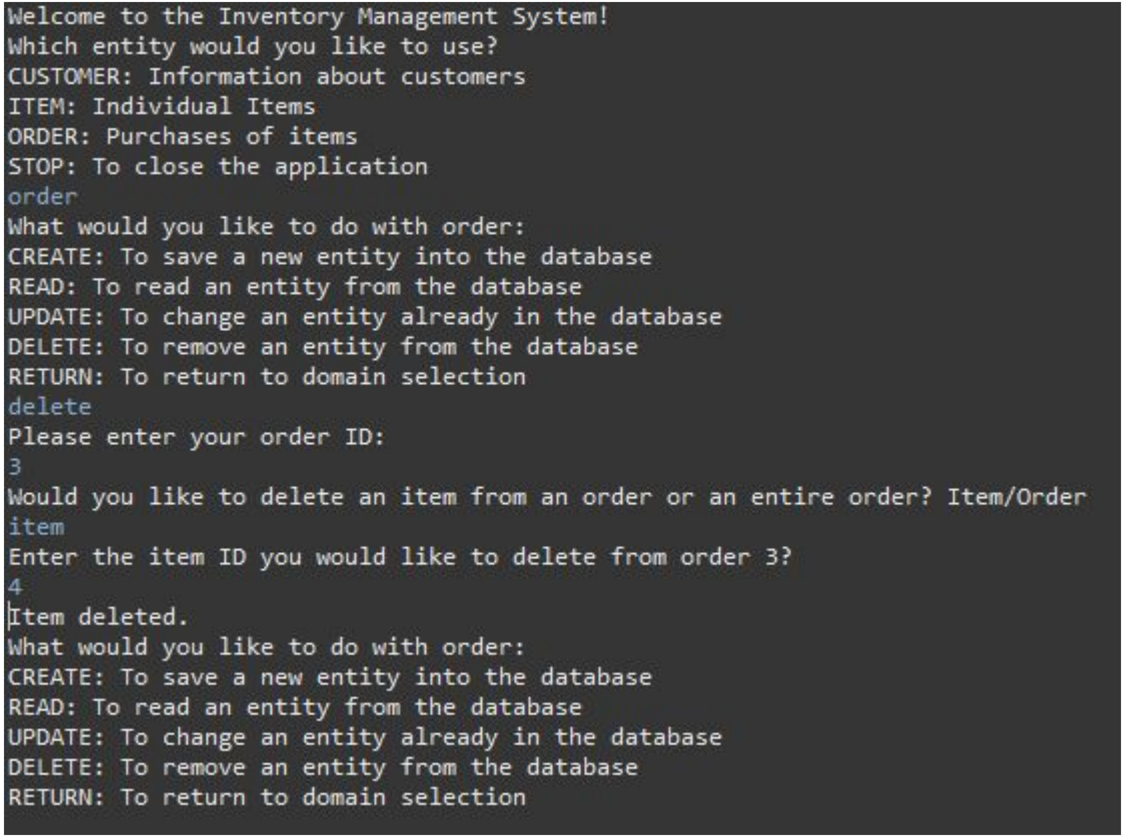
- 1. Creating an new order
- 2. Type order
- 3. Type create
- 4. Type customer ID
- 5. Chose if you want to add item
- 6. If yes, type yes
- 7. Type item ID
- 8. Type quantity
- 9. Chose if you want to add another item
- 10. If yes, repeat from 6



```
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
order
What would you like to do with order:
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
create
Please enter a customer ID:
3
Do you want to add an item to the order? Yes?
yes
Please enter a product ID you wish to add:
1
Please enter a quantity you wish to have:
2
Item/s added.
Do you want to add an item to the order? Yes?
no
Order created.
What would you like to do with order:
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
```

Project demo

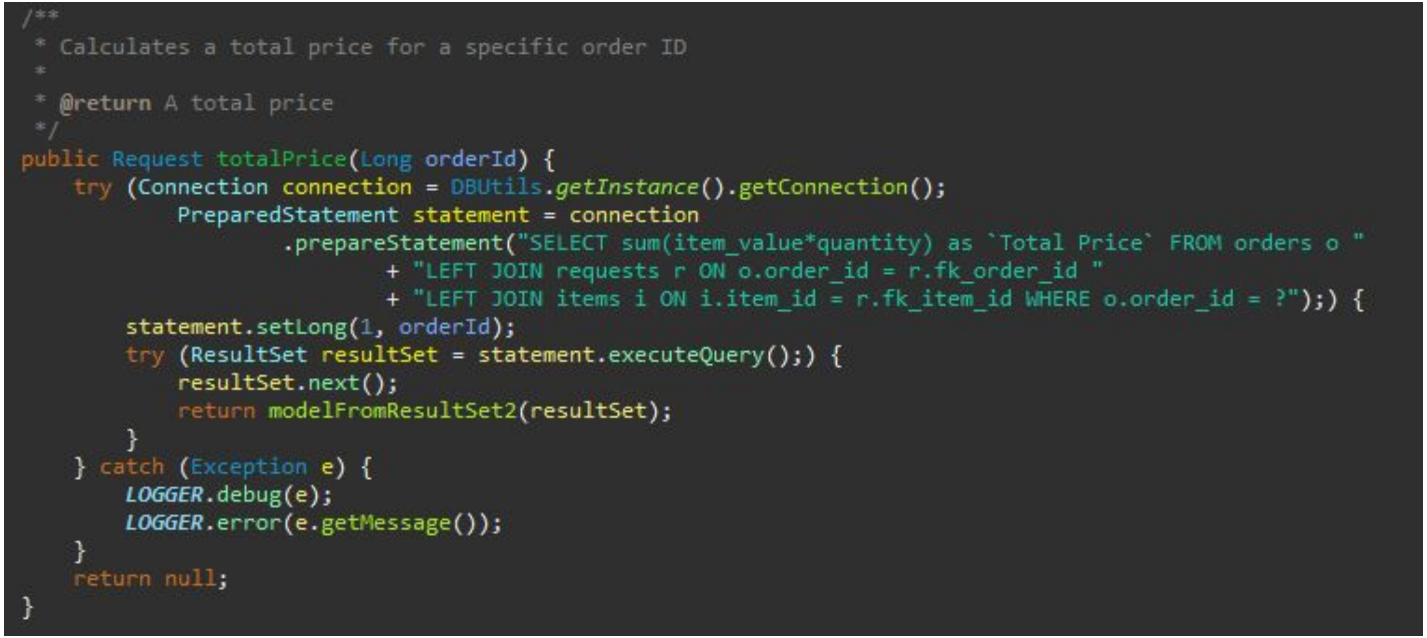
- 1. Deleting an item from order
- 2. Type order
- 3. Type delete
- 4. Type order ID
- 5. Type item
- 6. Type item ID



```
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
order
What would you like to do with order:
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
delete
Please enter your order ID:
3
Would you like to delete an item from an order or an entire order? Item/Order
item
Enter the item ID you would like to delete from order 3?
4
Item deleted.
What would you like to do with order:
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
```

Code best practice

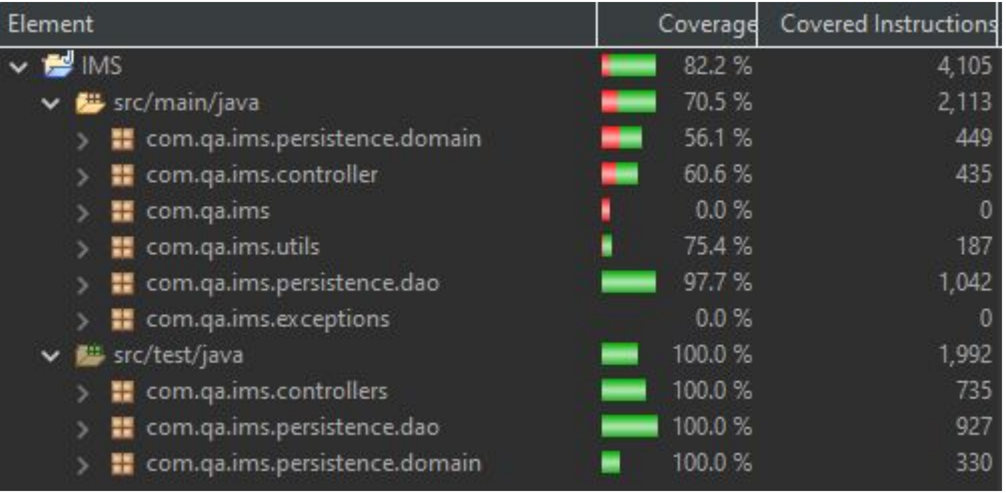
- When typing I stuck to naming convention
- I named functions based on what they do
- I commented all implemented methods
- I stuck to SOLID principals



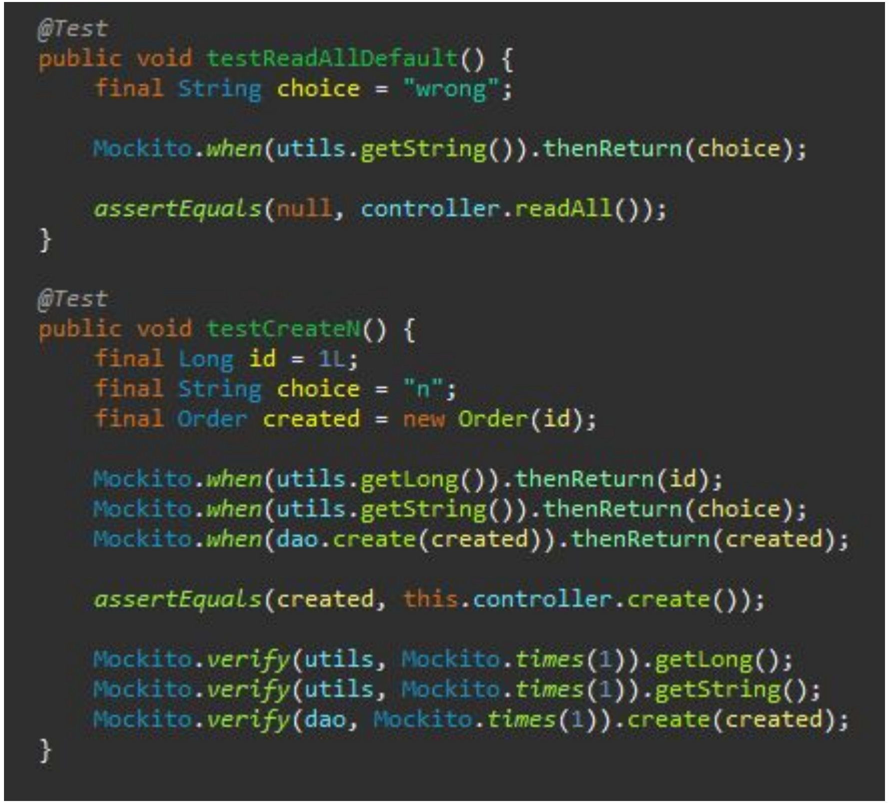
```
/**
 * Calculates a total price for a specific order ID
 *
 * @return A total price
 */
public Request totalPrice(Long orderId) {
    try (Connection connection = DBUtils.getInstance().getConnection();
            PreparedStatement statement = connection
                    .prepareStatement("SELECT sum(item_value*quantity) as `Total Price` FROM orders o "
                            + "LEFT JOIN requests r ON o.order_id = r.fk_order_id "
                            + "LEFT JOIN items i ON i.item_id = r.fk_item_id WHERE o.order_id = ?");) {
        statement.setLong(1, orderId);
        try (ResultSet resultSet = statement.executeQuery();) {
            resultSet.next();
            return modelFromResultSet2(resultSet);
        }
    } catch (Exception e) {
        LOGGER.debug(e);
        LOGGER.error(e.getMessage());
    }
    return null;
}
```

Code Testing

- Code written to best practice standards
- Unit Testing implemented
- Done by JUnit4
- To right an example of test
- To bottom my test coverage



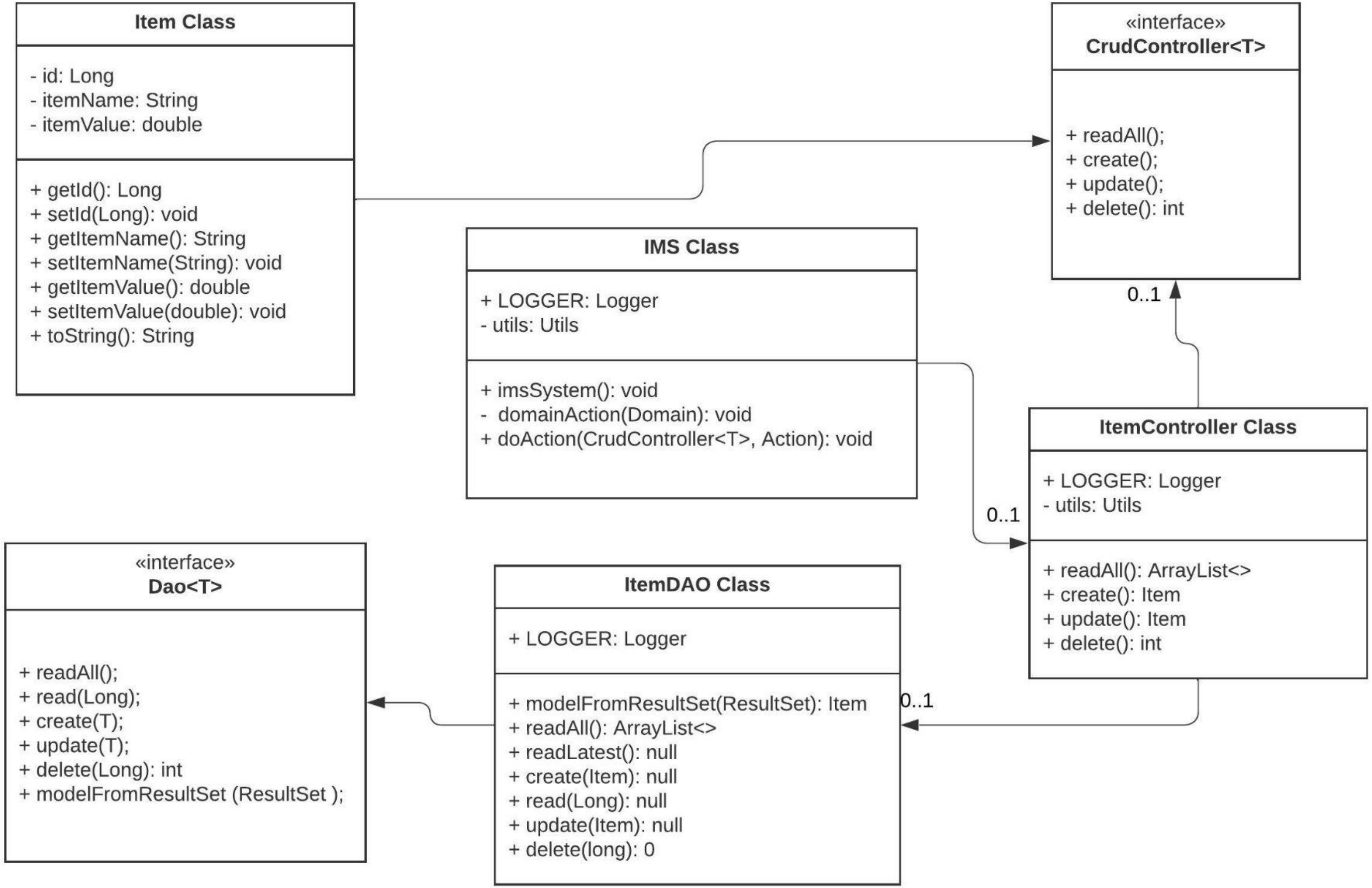
Element	Coverage	Covered Instructions
IMS	82.2 %	4,105
src/main/java	70.5 %	2,113
com.qa.ims.persistence.domain	56.1 %	449
com.qa.ims.controller	60.6 %	435
com.qa.ims	0.0 %	0
com.qa.ims.utils	75.4 %	187
com.qa.ims.persistence.dao	97.7 %	1,042
com.qa.ims.exceptions	0.0 %	0
src/test/java	100.0 %	1,992
com.qa.ims.controllers	100.0 %	735
com.qa.ims.persistence.dao	100.0 %	927
com.qa.ims.persistence.domain	100.0 %	330



```java
@Test
public void testReadAllDefault() {
    final String choice = "wrong";

    Mockito.when(utils.getString()).thenReturn(choice);

    assertEquals(null, controller.readAll());
}

@Test
public void testCreateN() {
    final Long id = 1L;
    final String choice = "n";
    final Order created = new Order(id);

    Mockito.when(utils.getLong()).thenReturn(id);
    Mockito.when(utils.getString()).thenReturn(choice);
    Mockito.when(dao.create(created)).thenReturn(created);

    assertEquals(created, this.controller.create());

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

Simplified UML for Items

MMkluska | July 22, 2022



Item Class

- id: Long
- itemName: String
- itemValue: double

+ getId(): Long
+ setId(Long): void
+ getItemName(): String
+ setItemName(String): void
+ getItemValue(): double
+ setItemValue(double): void
+ toString(): String

«interface»
CrudController<T>

+ readAll();
+ create();
+ update();
+ delete(): int

IMS Class

+ LOGGER: Logger
- utils: Utils

+ imsSystem(): void
- domainAction(Domain): void
+ doAction(CrudController<T>, Action): void

ItemController Class

+ LOGGER: Logger
- utils: Utils

+ readAll(): ArrayList<>
+ create(): Item
+ update(): Item
+ delete(): int

«interface»
Dao<T>

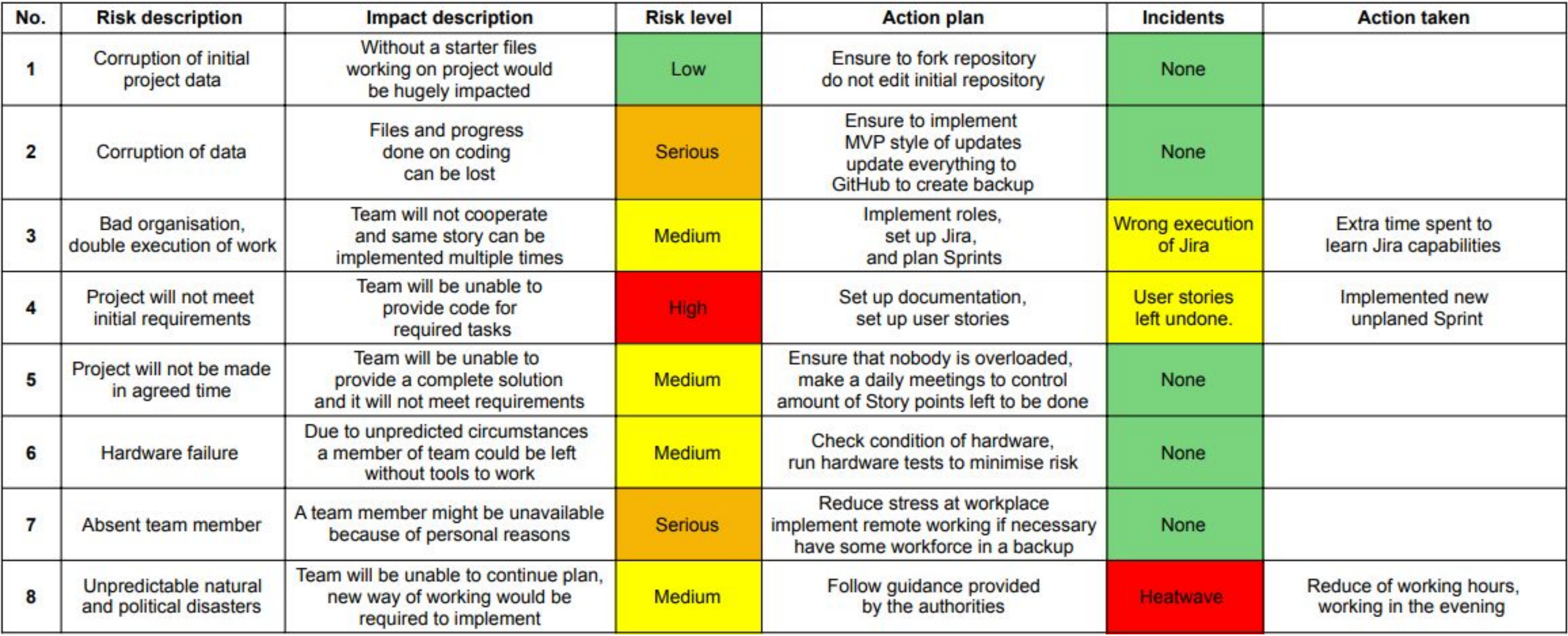
+ readAll();
+ read(Long);
+ create(T);
+ update(T);
+ delete(Long): int
+ modelFromResultSet (ResultSet);

ItemDAO Class

+ LOGGER: Logger

+ modelFromResultSet(ResultSet): Item
+ readAll(): ArrayList<>
+ readLatest(): null
+ create(Item): null
+ read(Long): null
+ update(Item): null
+ delete(long): 0

Risk Assessments Review



No.	Risk description	Impact description	Risk level	Action plan	Incidents	Action taken
1	Corruption of initial project data	Without a starter files working on project would be hugely impacted	Low	Ensure to fork repository do not edit initial repository	None	
2	Corruption of data	Files and progress done on coding can be lost	Serious	Ensure to implement MVP style of updates update everything to GitHub to create backup	None	
3	Bad organisation, double execution of work	Team will not cooperate and same story can be implemented multiple times	Medium	Implement roles, set up Jira, and plan Sprints	Wrong execution of Jira	Extra time spent to learn Jira capabilities
4	Project will not meet initial requirements	Team will be unable to provide code for required tasks	High	Set up documentation, set up user stories	User stories left undone.	Implemented new unplaned Sprint
5	Project will not be made in agreed time	Team will be unable to provide a complete solution and it will not meet requirements	Medium	Ensure that nobody is overloaded, make a daily meetings to control amount of Story points left to be done	None	
6	Hardware failure	Due to unpredicted circumstances a member of team could be left without tools to work	Medium	Check condition of hardware, run hardware tests to minimise risk	None	
7	Absent team member	A team member might be unavailable because of personal reasons	Serious	Reduce stress at workplace implement remote working if necessary have some workforce in a backup	None	
8	Unpredictable natural and political disasters	Team will be unable to continue plan, new way of working would be required to implement	Medium	Follow guidance provided by the authorities	Heatwave	Reduce of working hours, working in the evening

Summary

- I finished basic requirements
- I learned a lot about OOP
- I am better in time management
- I learned Jira and Java from more practical use

- I would like to finish some of extension tasks
- I would like to automate my tests in the future
- I would like to implement a simple CI Pipeline
- I would like to format more human friendly console output