

# IMS Project

Ali Khattab – 21SepSoftware

# Introduction

- Who am I?
- Brief: Design and build an Inventory Management System (IMS) for a store
- How did I approach the specification?

# Consultant Journey

- Technologies learned for this project:



# Jira

Jira Software

Your work

Projects

Filters





Dashboards

People

Apps

Create

Q Search



IMS-Project

Software project

Roadmap

Backlog

Board

Code

Project pages

Add shortcut

Project settings

You're in a team-managed project


Learn more

Projects / IMS-Project

Backlog

Q

AK



Epic

Insights

Epic

Issues without epic

Implement items into IMS

+ Create Epic

IMS Sprint 1

Add dates

(4 issues)

000Start sprint


...

IMS-1

Add an item in the system

IMPLEMENT ITEMS INTO IMS

TO DO




IMS-2

View all items in the system

IMPLEMENT ITEMS INTO IMS

TO DO




IMS-3

Update an item in the system

IMPLEMENT ITEMS INTO IMS

TO DO




IMS-4

Delete an item in the system

IMPLEMENT ITEMS INTO IMS

TO DO



+ Create issue

Backlog


(6 issues)

000Create sprint

IMS-5

Create an order in the system


TO DO



IMS-6

View all orders in the system


TO DO



IMS-7

Delete an order in the system


TO DO



IMS-8

Add an item TO an order


TO DO



IMS-9

Calculate a cost for an order


TO DO



IMS-10

Delete an item IN an order

TO DO




+ Create issue

# Continuous Integration

- Version Control System – Git
- Cloud-based hosting service – GitHub
- Smart commits from GitHub linking to Jira

# Jira Sprint

 Jira Software

Your work ▾

Projects ▾


Filters ▾


Dashboards ▾


People ▾


Apps ▾


Create


 **IMS-Project**  
Software project


 Roadmap


 Backlog


 **Board**

 Code

 Deployments

 Project pages

 Add shortcut


 Project settings

Does your team need more from Jira? [Get a free trial of our Star](#)

Projects / IMS-Project

**IMS Sprint 2**

AK




Epic ▾

TO DO 2 ISSUES

Update an order in the order\_items table


IMPLEMENT ORDERED ITEMS INT...

 IMS-15 

3

Delete an entry from the order\_items table

IMPLEMENT ORDERED ITEMS INT...


 IMS-16 

3

IN PROGRESS 1 ISSUE

Read from the order\_items table

IMPLEMENT ORDERED ITEMS INT...


 IMS-14 

2


DONE 1 ISSUE ✓

Create a list of entries for items that have been ordered

IMPLEMENT ORDERED ITEMS INT...

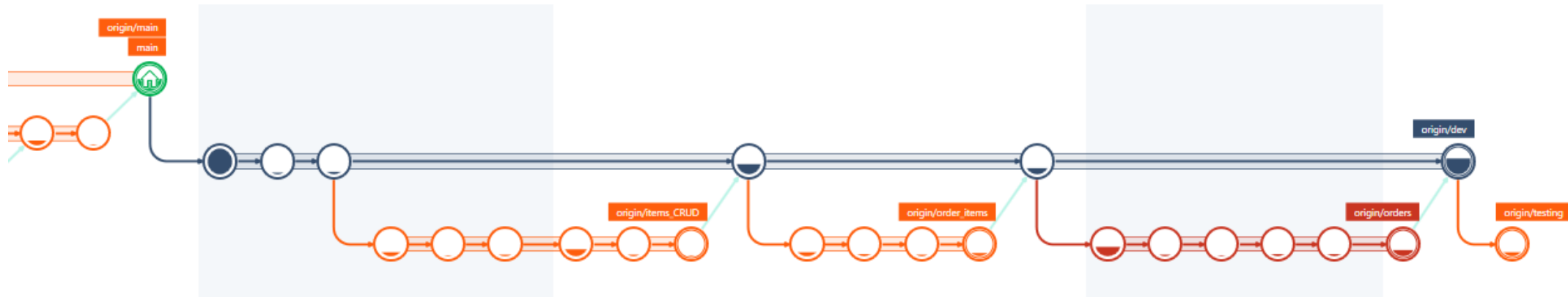
 IMS-13 

8



# GMaster

- Feature Branch Model



# Testing



```
@Test
public void testRead() {

    final Long ID = 1L;
    assertEquals(new Order(ID, 1L, 1d), orderDAO.read(ID));

}

@Test
public void TestReadAll() {

    List<Order> expected = new ArrayList<>();
    expected.add(new Order(1L, 1L, 1D));
    assertEquals(expected, orderDAO.readAll());

}
```

```
@RunWith(MockitoJUnitRunner.class)
public class ItemControllerTest {

    @Mock
    private Utils utils;

    @Mock
    private ItemDAO dao;

    @InjectMocks
    private ItemController controller;

    @Test
    public void testCreate() {
        final String ITEM_NAME = "Banana";
        final Double ITEM_VALUE = 30d;
        final Item created = new Item(ITEM_NAME, ITEM_VALUE);

        Mockito.when(utils.getString()).thenReturn(ITEM_NAME);
        Mockito.when(utils.getDouble()).thenReturn(ITEM_VALUE);
        Mockito.when(dao.create(created)).thenReturn(created);

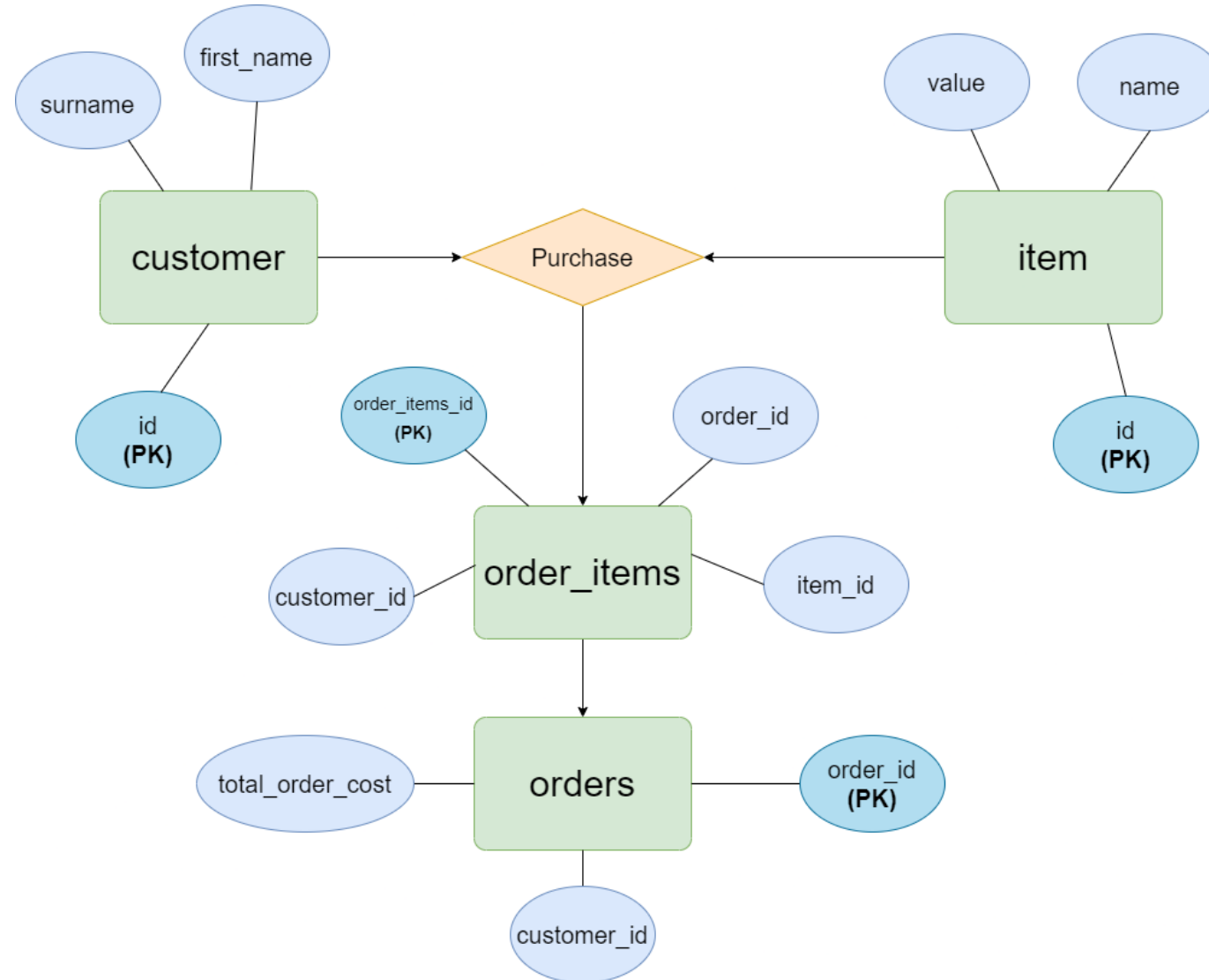
        assertEquals(created, controller.create());
    }
}
```



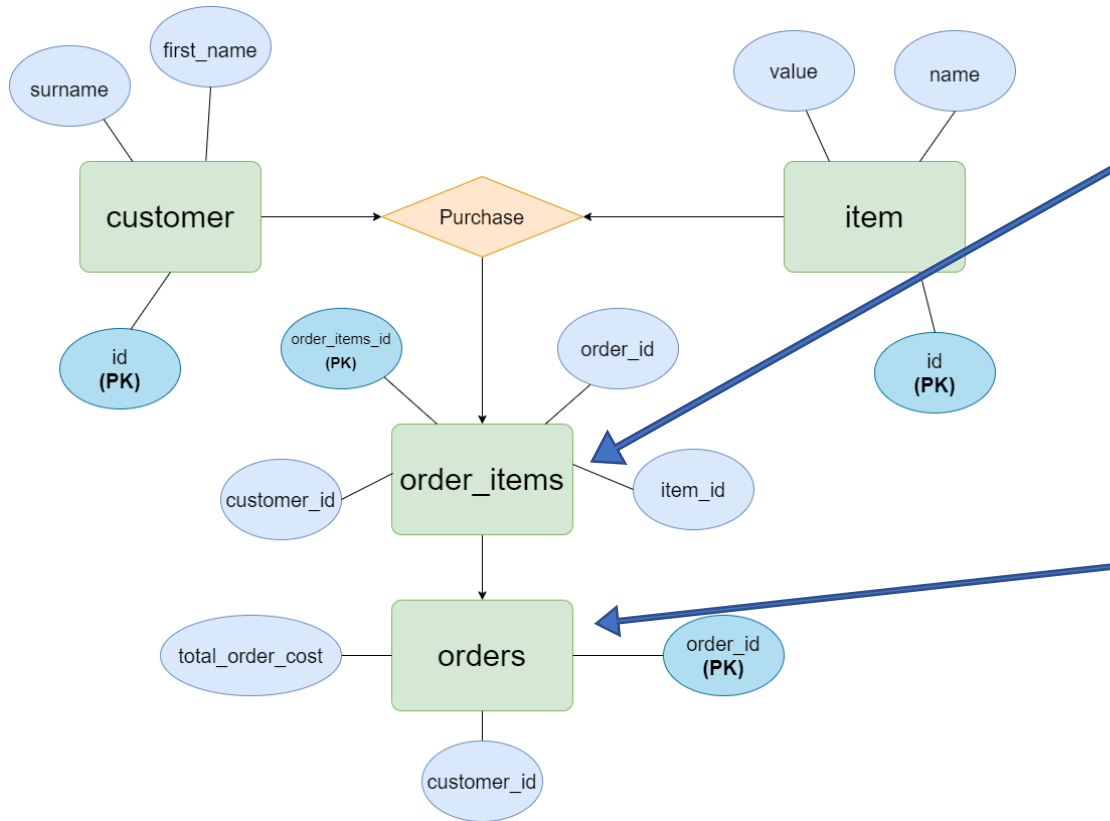
# Testing Coverage

▼	src/main/java		67.9 %
>	com.qa.ims.persistence.dao		76.6 %
▼	com.qa.ims.persistence.domain		61.5 %
>	Domain.java		0.0 %
>	OrderItem.java		61.8 %
>	Order.java		78.0 %
>	Customer.java		79.3 %
>	Item.java		79.7 %
>	com.qa.ims		0.0 %

# Entity Relationship Diagram



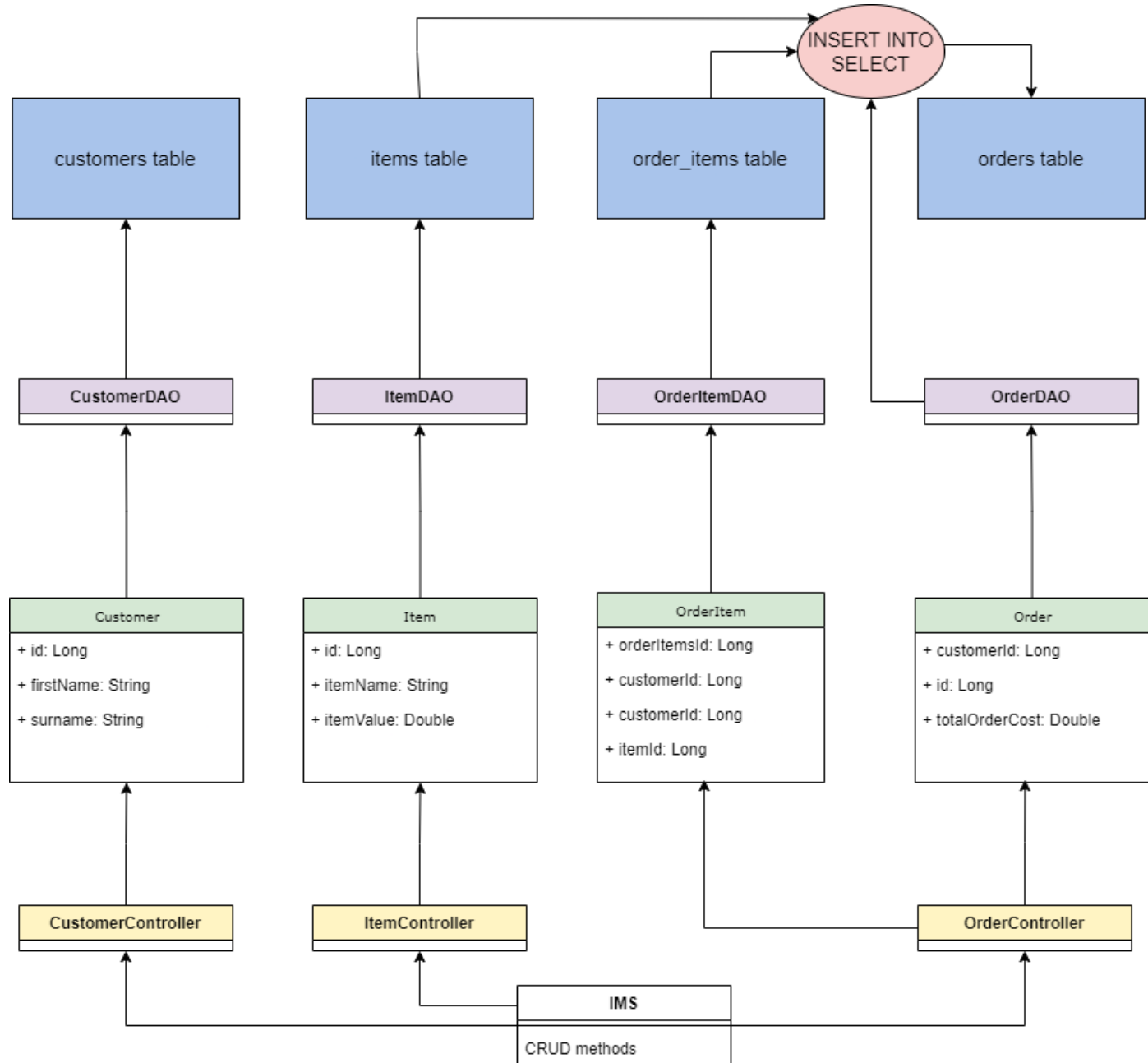
# Entity Relationship diagram



order_item_id	order_id	customer_id	item_id
1	1	9	3
2	1	9	4
3	1	9	4
4	1	9	6
5	2	4	8
6	2	4	4

order_id	customer_id	total_cost
1	9	17.0
2	4	12.0

# UML Diagram



# Demonstrations

# Sprint Review

- What did I complete
  - MVP of IMS including:
    - MySQL database
    - Full Java Backend application with CRUD functionality
- What got left behind
  - Testing above 80%

# Sprint retrospective

- What went well?
  - Managed to get a working solution
  - Planned my project around diagrams to help simplify the development
  - Using feature branch models within GitHub
- What could be improved?
  - Write tests as part of the feature branch as development goes along
  - More functionality in the console
  - More comments in source code to make things clearer

# Conclusion

- Questions?
- GitHub: <https://github.com/aliktb/IMS-Starter>
- Jira: <https://emaplejiralc1.atlassian.net/jira/software/projects/IMS/code>