CASE STUDY DOCUMENTATION ON Amazon Clone

Peerzada Mutien Ahmad

Analyst, Capgemini

Employee ID: 46209103

Index:

- Synopsis
- Requirements
- Software Requirements
- Hardware Requirements
- Stories
- Microservice Architecture Diagram
- Use Case Diagram
- Class Diagram
- Test Coverage

Amazon Clone

1. Synopsis:

Amazon clone is a simple application which demonstrates an E-commerce website using the Java Spring Boot. Ecommerce is the activity of buying or selling of products on online services or over the Internet. There are two actors in this application one is customer and the other is Amazon. Customer can browse various types of products, customer can add the products to cart and then do payment.

2. Requirements:

- Customer can add its information
- Customer can browse all products
- Customer can add products to cart
- Customer can delete the product data from cart.

3. Software Requirements:

1. JDK-11
2. MongoDB
3. Junit 5
4. Mockito
5. Spring Suite Tool IDE
6. Web Server
7. Postman
8. Git
9. Jacoco

4. Minimum Hardware Requirements:

CPU: 2Ghz Processor

RAM: 2GB

HDD: 10GB

5. Stories:

1. The Application will have 6 microservices:

a) Eureka Service
b) API Gateway Service
c) Login Service
d) Customer Service
e) Product Service
f) Cart Service.
g) Payment Service.

2. The Customer service is responsible for storing all information about the customer. Such as Name, Address, Customer Id, Email id. It will contain the user data.

3. The Product service is responsible for storing all information about Product. Such as Product name, Product id, product rating, price. It will contain the product data.

4. The Cart Service is responsible for simple mapping between the Customer id and product id by which we get to know which customer is putting which product into the cart.

5. The Payment Service is done through Razorpay API which is responsible for placing order and giving order related information such as order id , amount paid , receipt id . It will contains order and payment related data.

6. End points that will be created in our API are:

Customer Service:

1. Add Customer Data
2. Retrieve All Customer Data
3. Find Customer by its ID

Product Service:

1. Add Product Data
2. Retrieve all Product Data
3. Retrieve Details for particular product based on Id

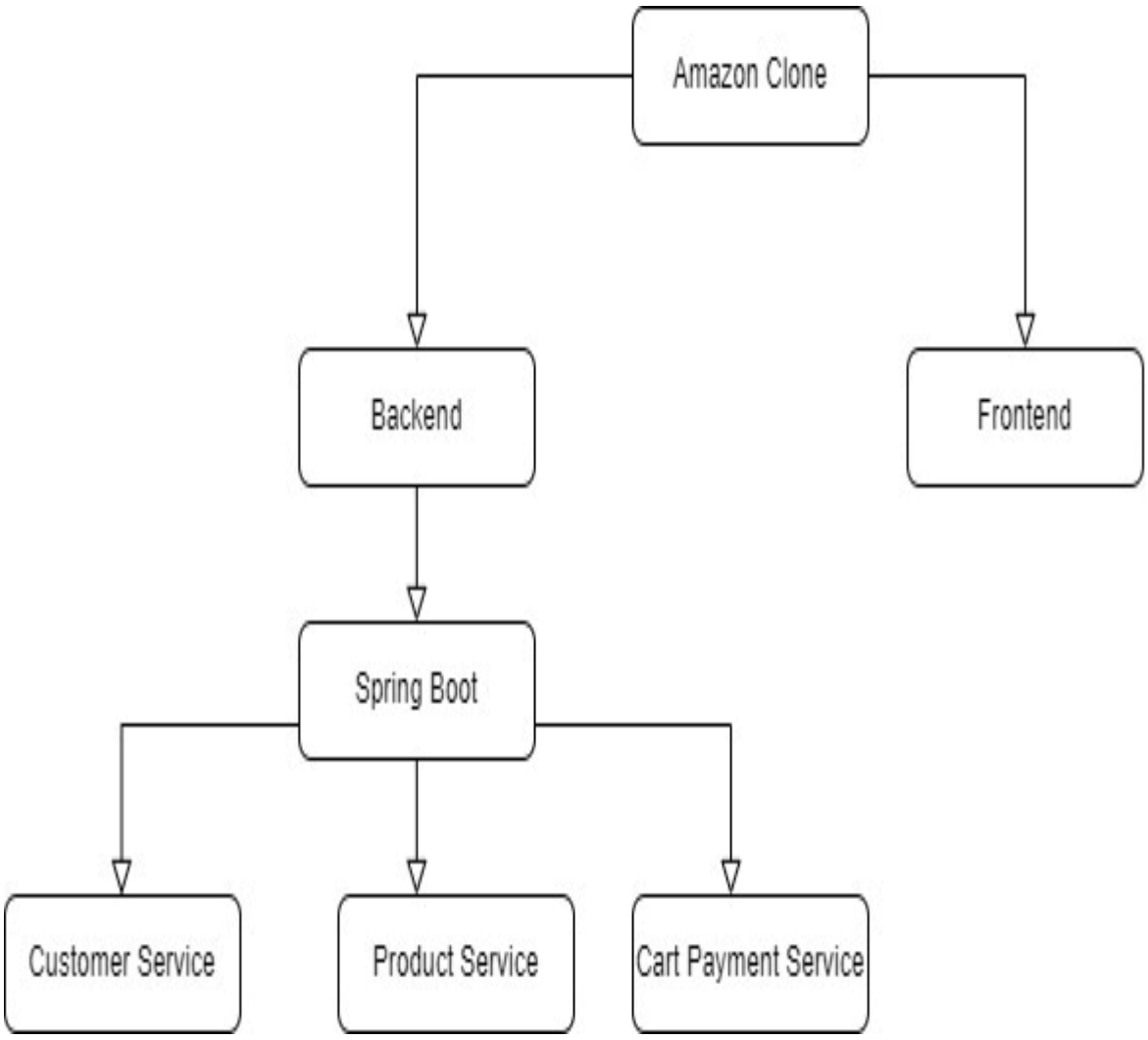
Cart Service:

1. Adding items into cart
2. Deleting Items from cart.
3. Retrieve products that customer has put in a cart

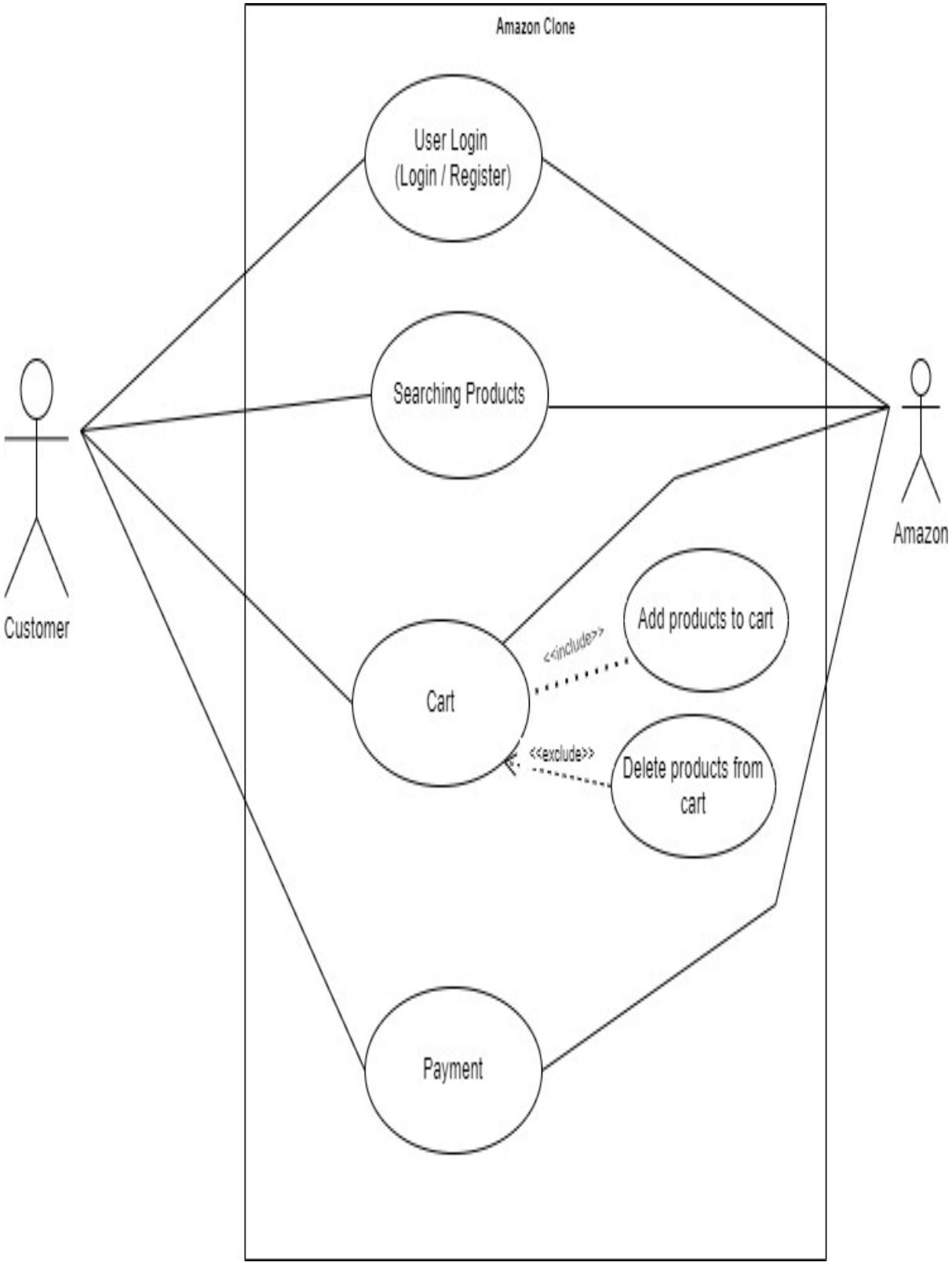
Payment Service:

1. Place the order

6. Microservice Architecture Diagram:

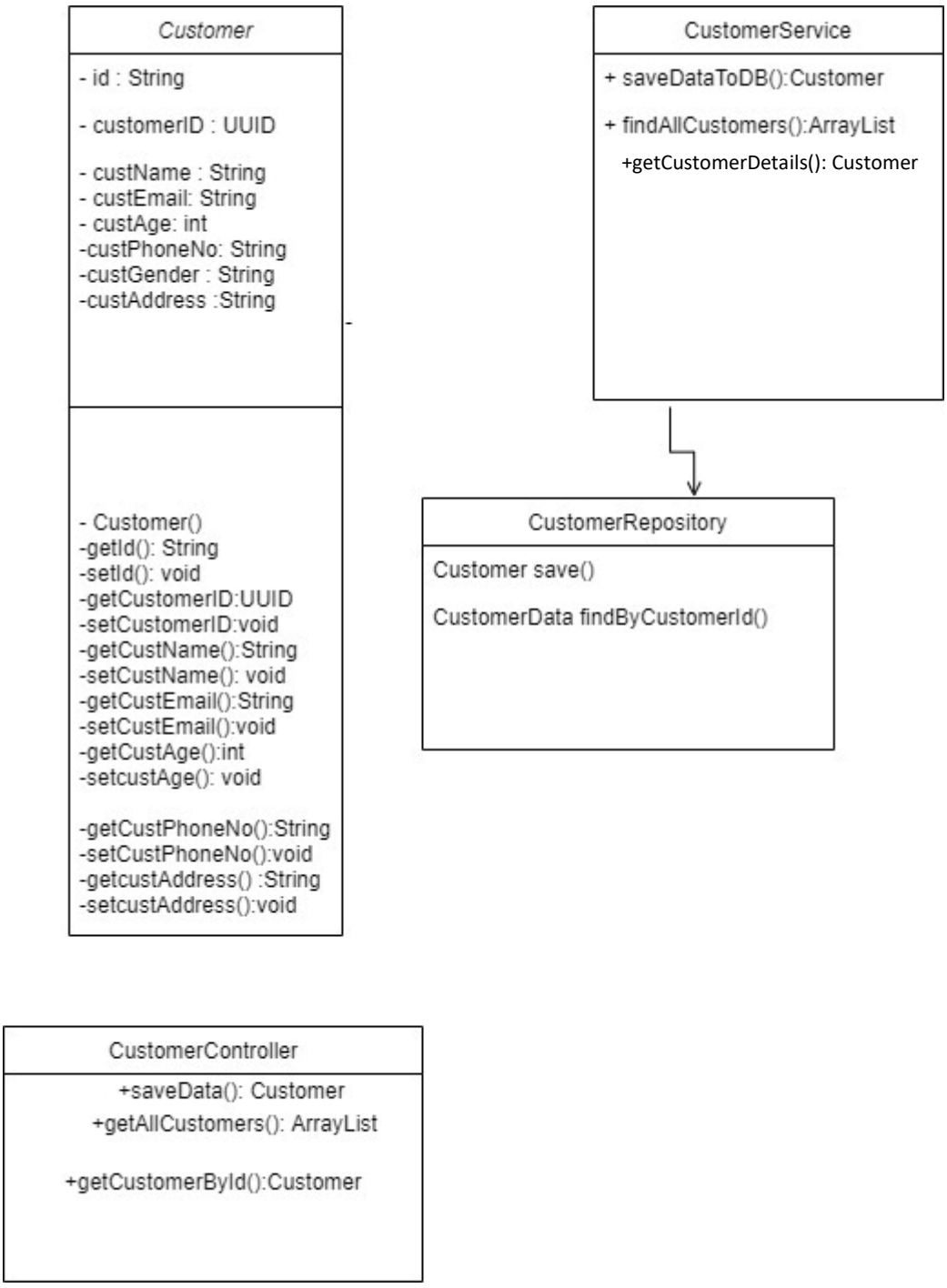


7. Use Case Diagram:

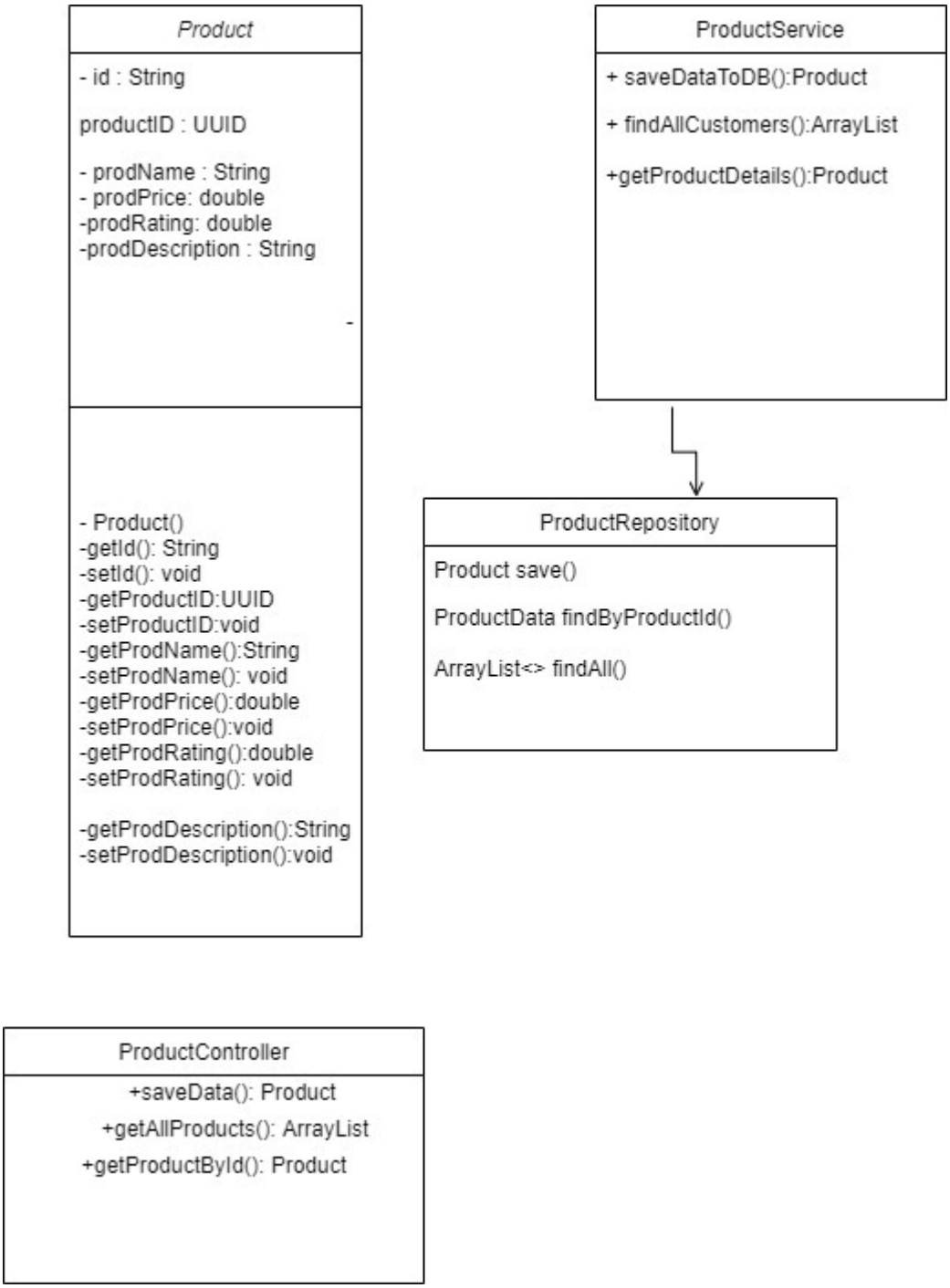


8. Class Diagram:

Customer Service:



Product Service



9. Test Coverage:

Customer Service: (70.4%)