

A dark blue vertical bar runs down the left side of the page. A blue arrow points to the right from this bar, containing the date.

11/29/2021

Software Engineering

Virtual Market Application

Requirement Analysis Report

Several thin, curved lines in shades of blue and grey sweep upwards from the bottom left corner of the page.

Group Members

- Michael Derece Kebede
- Morteza Yosefy
- Kibru Joba Kuture
- Atiqurahman Mayar

Table of Contents

Introduction	1
Identification of Viewpoints	2
Principal Viewpoints of the System	2
Interactor Viewpoints	2
Indirect Viewpoints	2
Domain Viewpoints Requirements	2
Description of each Viewpoint	2
Viewpoint Hierarchy Diagram	4
Requirements Definition	5
Definition of requirements of each viewpoints	5
Interactor Viewpoints Requirements	5
Indirect Viewpoints Requirements	6
Domain Viewpoints Requirements	6
Requirements Classification (Considering Functionality)	7
Functional Requirements	7
General System Requirement	7
Customer related requirements	8
Store-manager related requirements	8
System administrator related requirements	8
Non functional requirements	9
Domain Requirements	9
Requirements Classification (considering lifetime)	10
Volatile Requirements	10
Enduring Requirements	11
Requirements Prioritization and Negotiation	12
Prioritization	12
Negotiation	13
Fully Dressed Use Cases of Main Scenarios	14
UC1	14
UC2	15
UC3	16
UC4	17
UC5	18

UC6.....	19
UC7.....	20
UC8.....	21
UC9.....	22
Domain Model as a UML Diagram	24
Conclusion	25

Introduction

We are asked to develop an Online Virtual Market Application (VMA) that lets users shop for required products and have these products delivered to their addresses. Some of the capabilities of this application are expected but not limited to:

VMA offers various product categories such as “Fruits & Vegetables”, “Dairy Products”, “Cleaning Products”, “Electronics” and so on.

Some of the tasks that Customers can do are given below:

- Search and filter products
- Add desired products into their shopping cart
- Benefit from coupons and discounts
- Specify some of the products as their favorites,
- Set and update their personal information such as address details and payment information
- Place their order, etc.

Some of the tasks that Market/Store Managers can do are given below:

- Set/update a physical store location
- Define/update new products
- Keep track of the stock
- Define/update coupons and discounts
- Fulfill orders
- Get statistics & reports about orders, etc.

The developed VMA will be accessible from different platforms such as web browsers, smartphones/tablets etc. (It's responsive)

The use case to be implemented is placing an order by the customer, it is assumed that the use cases that the store manager must perform before this use case are already completed.

Example: products are already defined

Identification of Viewpoints

Principal Viewpoints of the System

Interactor Viewpoints

- Customer
- Market/ Store Manager(s)
- System Administrator

Indirect Viewpoints

- Courier/ Cargo Companies
- Payment System
- Stock Market

Domain Viewpoints Requirements

- UI Standards
- Ministry of Commerce and Finance

Description of each Viewpoint

Interactor Viewpoints

- **Customer** – The user role which can search and buy the products.
- **Market/ Store Manager** – The user role which sells, define, update, and fulfill the orders.
- **System Administrator** – The user role, which have access to all the system and maintain the server.

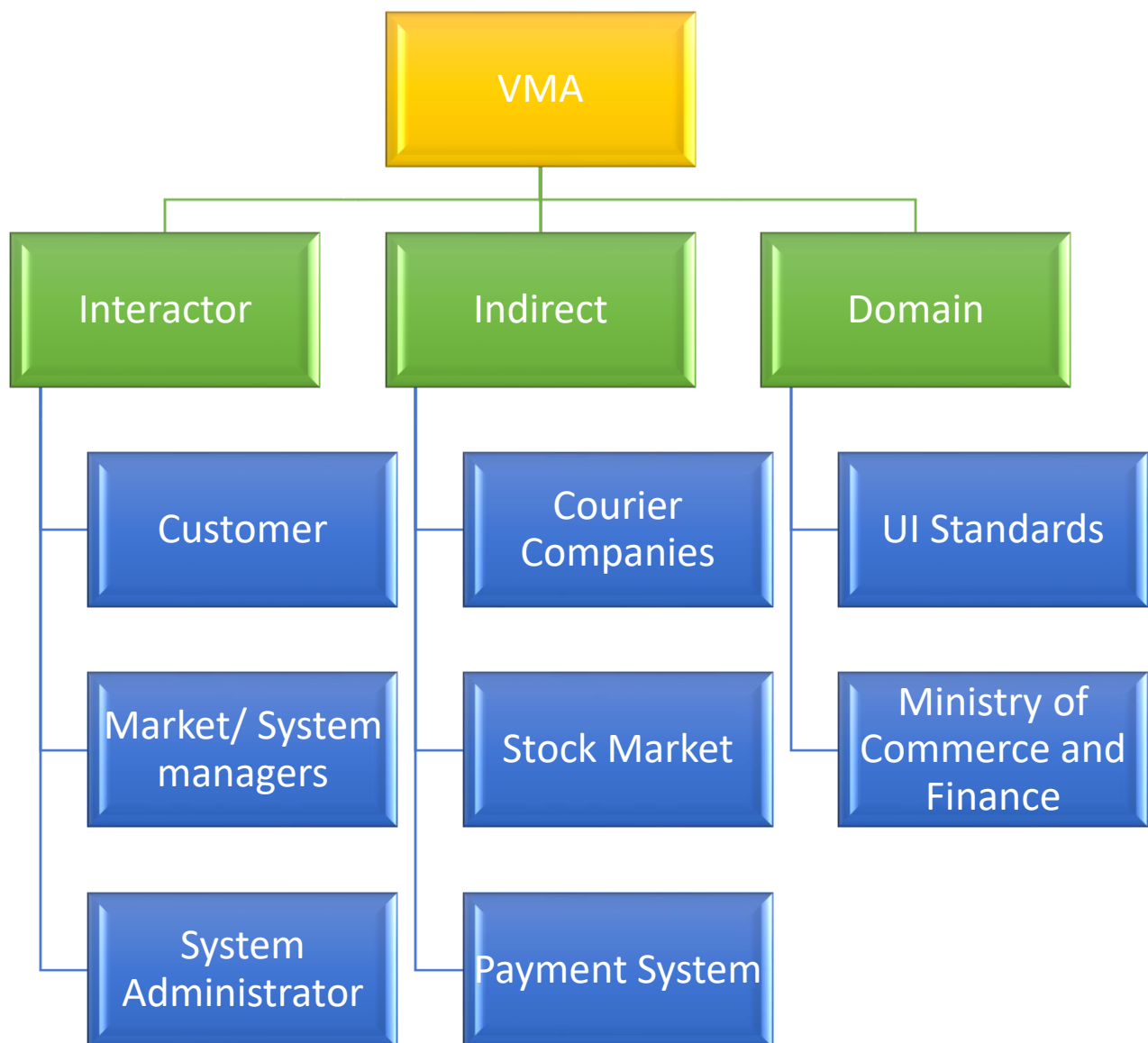
Indirect Viewpoints

- **Courier/ Cargo Companies**
 - Deal with the shipment of orders.
 - May demand access to customer's details such as their name, address, invoice...
- **Payment System**
 - Covers the payment process.
 - Control the transactions
 - Can accept/reject the order depending on customer's balance or coupon.
- **Stock Market**
 - The increase/ decrease in the number of stocks can affect the system

Domain Viewpoints Requirements

- **UI Standards**
 - Interface rules that abstract the user from complex code
- **Ministry of Commerce and Finance**
 - Rules that contain all the restrictions for each product (Customs affairs, tax acts...)

Viewpoint Hierarchy Diagram



Requirements Definition

Definition of requirements of each viewpoints

Interactor Viewpoints Requirements

1. Customer

- A customer wants to be able to log into the system and view and edit their personal information such as address details and payment information.
- A customer wants to be able to search and filter the products
- A customer wants to be able to add desired products into their shipping cart.
- A customer wants to be able to check details about his product delivery.
- A customer wants to be notified about / able to use discounts and coupons

2. Market / store Manager

- A store Manager wants to be able to log in and see information about himself his Physical store location and his products.
- A store Manager wants to be able to define, update new products.
- A store Manager wants to keep track of the stocks
- A store Manager wants to be able to define/update coupons and discounts.
- A store Manager wants to be able to fulfil orders.
- A store Manager wants to be able to get statistics & reports about orders.

3. System Administrator

- The system administrator wants to be able to login and make changes to the system.
- The system administrator wants to be able to view and manage users.

Indirect Viewpoints Requirements

1. Courier companies

- Courier companies want to be able to Deal with the shipment of orders by demanding customer's details such a name, address, and the invoice.

2. Payment System

- Actors behind the Payment system want to be able to deal with the payment of orders.
- Actors behind the Payment system want to be able to confirm the payment by sending confirmation email / SMS to customers.

3. Stock market

- The stock market wants to be able to determine increment and decrement of the stocks.

Domain Viewpoints Requirements

1. UI Standards

- It limits what can be added to the appearance of the system.

2. Ministry of commerce

- Rules that contain all restrictions for a product (customs affairs, tax percentage etc...).

Requirements Classification (Considering Functionality)

Functional Requirements

General System Requirement

1.1 The system should provide a login UI that only an authorized user can access.

1.2 The system should provide high uptime speed

I. Image size: reduce or resize image size

II. Caching

1.3 The system should provide mobile first /responsive UI.

1.4 The system should provide personalization.

I. Successful e-commerce sites allow for a high level of personalization, without the customer even realizing they are being served up dynamic content.

1.5 Accessibility should be a standard

I. Choosing the right CMS (Content Management System) that logically organizes content and headings, provides a place for alt text images, labels fields appropriately, gives links descriptions and allows full access by screen readers and keyboard navigation are key.

1.6 The system should assure that a customer's data is safe

I. Remove customer fear with evidence that their information is safe and secure.

1.7 The system should provide buyer review page.

I. Customers want to hear from real people to see if the products fit their desires and expectations.

1.8 The system should be capable of promoting new products/campaigns

I. Posting new promotions to widgets and web banners across your site helps customers see and share the latest special offers.

1.9 The system should have a well designed shopping cart

1.10 The system should have an easy checkout

1.11 The system should provide product details

1.12 The system should be built with the capability of Search and Filter.

1.13 The system must have a payment system.

1.14 The legal and policy structure.

Customer related requirements

1.15 Customers can log into their profiles with login credentials.

1.16 Customers can change password.

1.17 Customers can change their personal information and modify their profiles.

1.18 Customers can place an order/ orders.

1.19 Customers can cancel their order and have right to get their money back.

1.20 Customers can talk to customer-help-agent online and get help.

Store-manager related requirements

1.21 Store managers can log into their profiles with login credentials.

1.22 Store managers can change password.

1.23 Store managers can change their personal information and modify their profiles.

1.24 Store managers can store products, put price tag , and describe products in the system.

1.25 Store managers can put the physical location of stores.

1.26 Store managers can put, delete, update products and their price tag.

1.27 Store managers can access customers information like names and their addresses.

1.28 Store managers are capable of contacting carrier companies.

System administrator related requirements

1.29 System admin can create and delete a user account.

1.30 System admin can access all the system information.

1.31 The system admin can view and access all the accounts of the users.

1.32 The system admin can hide any functionality and features from both users.

1.33 The system administrator can edit, change, insert and delete the information.

Non functional requirements

- 2.1 The software must support the use of multiple users at the same time.
- 2.2 System should be efficient, consistent, secure, reliable and robust.
- 2.3 System should be easy to use (user graphic interface should have high usability).
- 2.4 The system should support all device types.
- 2.5 Response time of the system shall not exceed 2.5 seconds.
- 2.6 The system should keep personal information from any threats.
- 2.7 The users ID must be unique, and users' passwords should be stored with the SHA 256 encryption algorithm.
- 2.8 There should only be three account roles (Customer, Store manager and System Admin).

Domain Requirements

- 3.1 The system should comply with General Data Protection Regulation.
- 3.2 The System should follow copyright rules.
- 3.3 The system should comply with the rules put forward by ministry of commerce.

Requirements Classification (considering lifetime)

Requirements evolution during the re-engineering process and after a system has gone into service is inevitable. Developing software requirements focuses attention on software capabilities, business objectives and other business systems. As the requirements definitions is developed, you normally develop a better understanding of users' needs. This feeds information back to the user who may then propose a change to the requirements. Furthermore, it may take several years to specify and develop a large system. Over that time, the system's environment and the business objectives change, and the requirements evolve to reflect this.

From an evolution perspective, requirements fall into two classes:

Volatile Requirements

These are requirements that are likely to change during the system development process or after the system has been become operational. For example, in a virtual market application, data, cookie and privacy requirements (because they are subject to national and international law which are always evolving).

- I. Most non-functional requirements are likely to change and can be considered as volatile requirements.
- II. Users' passwords should be stored with the SHA-2 encryption algorithm (encryption algorithm may change).
- III. User Interface is likely to change with the system administrator choosing more modern UIs to keep customers engaged.
- IV. Implementation requirements (organization might demand to change development process and development environment).
- V. Data usage, cookie and privacy agreements are likely to change with evolving national and international laws.
- VI. The maximum response time for any user request (it may need to be faster)

Enduring Requirements

These are relatively stable requirements that derive from the core activity of the organization, and which relate directly to the domain of the system. For example, in a virtual market application, there will always be requirements concerned with system administrators, sellers, payment solutions, couriers and customers.

Core functional requirements are unlikely to change and can be considered as enduring requirements.

- a. System administrators can manage customers, sellers, payment systems, couriers, and all related data. Managing may include on-boarding to system, management of data, off-boarding from system, defining relationship between on-boarders, managing interfaces, and setting quality standards.
- b. The system permissions for each user should be different.
- c. The system must have a memory capacity and bandwidth that supports all registered users.
- d. The system should be open 24/7.
- e. The system software must use a secure algorithm to ensure safety.
- f. Sellers can fulfill orders, set/update a physical store location, define/update new products, keep track of the stock, define/update coupons and discounts and get statistics & reports about orders.
- g. The system permissions for each user should be different.
- h. Customers can search and filter products, add desired products into their shopping cart, benefit from coupons and discounts, specify some of the products as their favorites, set and update their personal information such as address details and payment information and place their order.
- i. Couriers can receive information about orders that they can fulfill and gain access to customers' addresses.
- j. Payment solutions can get information regarding the sellers, customers, and system administrator's bank/payment details.

Requirements Prioritization and Negotiation

Prioritization

1. Functional Requirement
2. Non-Functional Requirement
3. Domain Requirement

Importance Level	Requirement Number
High	1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.8, 1.12, 1.13, 1.16, 1.17, 1.19,1.22, 1.25, 1.26, 1.27,1.29, 1.30, 1.33, 1.34, 1.35, 1.36, 1.37, 1.38, 1.39, 2.1, 2.2, 2.4, 2.5, 2.6, 2.7, 2.9, 2.10, 2.11, 2.12, 2.13, 2.17, 2.18, 2.20, 2.21, 3.1, 3.2, 3.3
Medium	1.7, 1.9, 1.10, 1.20, 1.28
Low	1.31, 2.3, 2.8, 2.15 1.14, 1.15, 1.21, 1.23, 1.24, 1.32, 2.14, 2.16, 2.19

Negotiation

Conflicts

- When system goes down, it shall be recovered in 5 minutes (conflict with 2.5).

Resolve as:

- The solution for this problem; we can consider a backup server. In case of a system failure, the backup server will be automatically activated, and user will be able to use it after few seconds.

Fully Dressed Use Cases of Main Scenarios

UC1

Use Case Name: view products

Scope: Virtual Market application (search Bar)

Level: User goal

Primarily Actor: Customer

Stakeholder Interests: Customer wants to search, filter products

Precondition:

Customer doesn't need to be registered or logged in as products can be viewed by any customer

postcondition:

customer finds details regarding his desired products.

Main Scenario:

1. The customer starts for looking a product.
2. Customer enters the search product parameters and request a search product
3. The system checks the product through product category in database.
4. The system sets / displays the product with details (including price / shipment date).
5. The customer selects the product.
6. The system displays add to card option.
7. The customer either adds or terminates the search.

Exceptions:

*At any time, system fails:

-The system restarts again.

Ex-1 customer types of invalid product name:

- System issues an error message.

Ex-2 the Item isn't available due to shortage.

- System issues and message to customer and asks to notify if the item is soon available.
- customer starts from step 2
- customer terminates the process.

UC2

Use Case Name: customer Registration

Scope: Virtual Market application (sign up bar)

Level: User goal

Primarily Actor: Customer

Stakeholder Interests:

Customer: wants to have an easy access, rapid and more personal experience of shopping. In this regard the customer would like to:

- set, update their personal information such as address, payment information
- have more user-friendly experience of the Market such as benefiting from coupons and discounts.
- specify some of products as their favorite.

System Admin: wants to track the customer search, order record for better experience and service.

Precondition:

Customer needs to have:

- an active email
- Phone number
- Address

postcondition:

Customer becomes a registered user.

Main Scenario:

1. The customer wants to sign up.
2. The System provides the GUI interface for the customer to sign up.
3. The customer enters name, surname, email, and password for sign up process.
4. The System cross check the information entered by the customer
5. The system verifies the customer sign up
6. The customer enters his bank account, delivery address to be saved by the system
7. The system offers a coupon and several discounts as the first-customer policy.
8. The customer either wishes to continue using the service by adding products into shopping cart and make a purchased or end the process

Exceptions:

*At any time, system fails:

-The system restarts again resumes from the last selected segment.

Ex-1 customer enters invalid email:

- System issues an error regarding the mail.
- System goes to phase 2

Ex-2 email is already registered in the system.

- The system offers a sign in operation to the customer
- Customer goes to phase 2 for a new Sign- up

UC3

Use Case Name: make a purchase

Scope: Virtual Market application

Level: User goal

Primarily Actor: Customer

Stakeholder Interests:

Customer: wants to buy, order products

Precondition:

Customer should be registered

postcondition:

Customer orders, buy his desired product.

Main Scenario:

1. The customer wants to order a product.
2. The customer signs into the system.
3. Customer chooses his product from favorite list or search for the new product.
4. The System opens add to cart option for the customer.
5. The customer proceeds to order the product.
6. the system proceeds to checkout
7. The system shows the courier option to the customer.
8. The system provides a Payment System GUI to confirm the order.

9. The system confirms the Payment and issues an order number for tracking the order.
10. The system also issues an invoice to customer's email.
11. Customer either start a new process or end the process.

Exceptions:

*At any time, system fails:

- The system restarts again resumes from the last selected segment.

Ex-1 customer enters invalid Username:

- System issues an error regarding the Username.
- System goes to phase 2

Ex-3 the product is out of stock

- The system offers a new product from another market
- Customer ends the process.

Ex-8 the payment is denied

- The system shows a message regarding the denial.
- The system goes to phase 6
- Customer ends the process.

UC4

Use case name: Delivery Processing

Scope: VMA and Courier

Level: User goal

Primary actor: Courier, store manager

Stakeholder interests: store manager wants to deliver the item to customer

Courier wants to deliver customer's order to customer's address

Precondition: Customer must buy item(s) form VMA. Courier must be on-boarded to the VMA system.

Post-condition: Customer receives order.

Main Scenario:

1. Customer makes an order on the VMA
2. System issues delivery request by giving the market manager the order number, customer ID and address
3. Seller validates order
4. Seller contacts courier service for delivery of order
5. Seller provides customer ID and address to a courier service
6. Courier service picks order from seller
7. Courier service delivers order to user

Exception:

*At any time, system fails:

-The system resumes from beginning

Ex-1: If the customer provides inaccurate address, the courier service may fail to deliver the order

Ex-7: If the customer is not present at the delivery address when the courier is trying to deliver the order, the order may be sent back to the seller

UC5

Use case name: Payment Processing

Scope: VMA and Payment Processor

Level: User goal

Primary actor: Customer

Stakeholder interests: Customer wants to pay for items in baskets

Precondition:

Customer must have valid bank/finance details.

Seller must have valid bank/finance details.

VMA must have valid bank/finance details.

Postcondition: Customer successfully pays for item in basket. Seller and System administrator receive payment.

Main Scenario:

1. Customer adds item(s) to basket
2. Customer proceeds to checkout screen
3. Customer uses any coupon available for him.

4. Customer enters bank/finance details
5. Customer accepts agreements regarding sale
6. Customer makes payment
7. Payment processor accepts payment
8. Payment processor processes payment
9. Payment processor makes payment to seller and system administrators

Exception:

*At any time, system fails:

-The system resumes from beginning

Ex-4 The customer may not have enough money in their bank account and thus the system processor will reject the payment.

- System jumps to step 2

EX-1: Shortage of product in a warehouse.:

- the system notifies the customer regarding the shortage of product.

UC6

Use Case Name: Return Item

Scope: Virtual Market application (Return section)

Level: User goal

Primarily Actor: Customer

Stakeholder Interests:

Customer wants to return the product(s) within the specific dates and demands for a change or refunds.

Cargo firms are responsible to deliver the product(s) back to the vendors.

Precondition: Customer must be registered and have an order number

Postcondition: the items are received by the store manager. Customers are paid back.

Main Scenario:

1. Customer receives the product(s).
2. Customer wants to return the product.
3. Customer makes a return request and fill the regarding enquiry.

4. System provides the efficient instructions.
5. Customer receives a code to show the responsible delivery staff.
6. Customer hand over the products to the cargo firm.
7. the System provides a track record for the shipment
8. the system notify customer when the delivery is done.
9. System will provide the following option after the delivery process:
 - a. Change the product
 - b. Place a new order
 - c. Receives their money back
10. Customer ends the process.

Exceptions:

*At any time, system fails:

-The system resumes from last selected segment (Return last user activity).

Ex-2 due date for returning item has passed

- The process ends.

Ex-9 invalid product

- Customer has violated the purchased item rules.
- The process ends.

UC7

Use case name: Flash / discount

Scope: virtual market application (content page)

Primary actor: store manager

Stakeholder interests: a store manager sets flash sale products, system admin displays, and customer buys products.

Precondition: all customers can take advantage of flash sale

Post condition: a customer gets a product with discount price.

Main scenario:

1. A store manager wants to apply discount on some products
2. The system displays the products store manager wants to apply discounts.
3. The system sets the price and the flash details per store managers request.
4. The system notify the store manager regarding the latest change.

5. The system sending promotion message through email to the customers

Exception:

*At any time, system fails:

-The system resumes from last selected segment (Return last user activity).

UC8

Use case name: notification

Scope : virtual application market (pop up window)

Level: user goal

Primary user: system admin

Stakeholder interests: a customer or store manager gets notified.

Precondition: In order to get a notification a customer or a store manager should be registered in the system.

Post condition: system notifies customers and store managers.

Main scenario:

1. a customer buys a products and store manager gets notified
2. a store manager hands over the products to a carrier firm and the customer gets notified to track the product.
3. before delivering the product to the customer , the carrier firm notifies the customer.
4. a customer is notified to change his/her password if a suspicious behaviour happens in the system.
5. a customer is notified of a product in his/her basket if the product remains in the basket for long and system detects price tag change.
6. if store manager lowers price tag of product during flash sale event, a customer gets notified of to take advantage of discount.

Exception:

*At any time, system fails:

-The system resumes from last selected segment (Return last user activity).

UC9

Use Case Name: store / manager Registration

Scope: Virtual Market application (sign up bar)

Level: User goal

Primarily Actor: store / market manager

Stakeholder Interests:

Store manager:

wants to set /update a physical store.

Define/ update new products.

Keep track of stocks

Wants to get statistics and report about orders.

System Admin: wants to track the customer search, order record for better experience and service.

Precondition:

Store manager needs to have:

- an active email
- Phone number
- Store Address
- Legal authorization to launch a market

postcondition:

user becomes a store manager.

Main Scenario:

1. The user wants to sign up as a store manager
2. The System provides the GUI interface for the user to sign up.
3. The store manager enters name, surname, email, and password for sign up process.
4. The System cross check the information entered by the customer
5. The system verifies the customer sign up
6. The system asks for documents regarding his legal authorizations for launching a market.
7. The store manager enters all the details and official documents including his bank account, store address.
8. The system accepts user as the new store manager and grant him an ID.
9. The store manager ends the process

Exceptions:

*At any time, system fails:

-The system restarts again resumes from the last selected segment.

Ex-1 store manager enters invalid details:

- System issues an error regarding the mail.
- System goes to phase 2

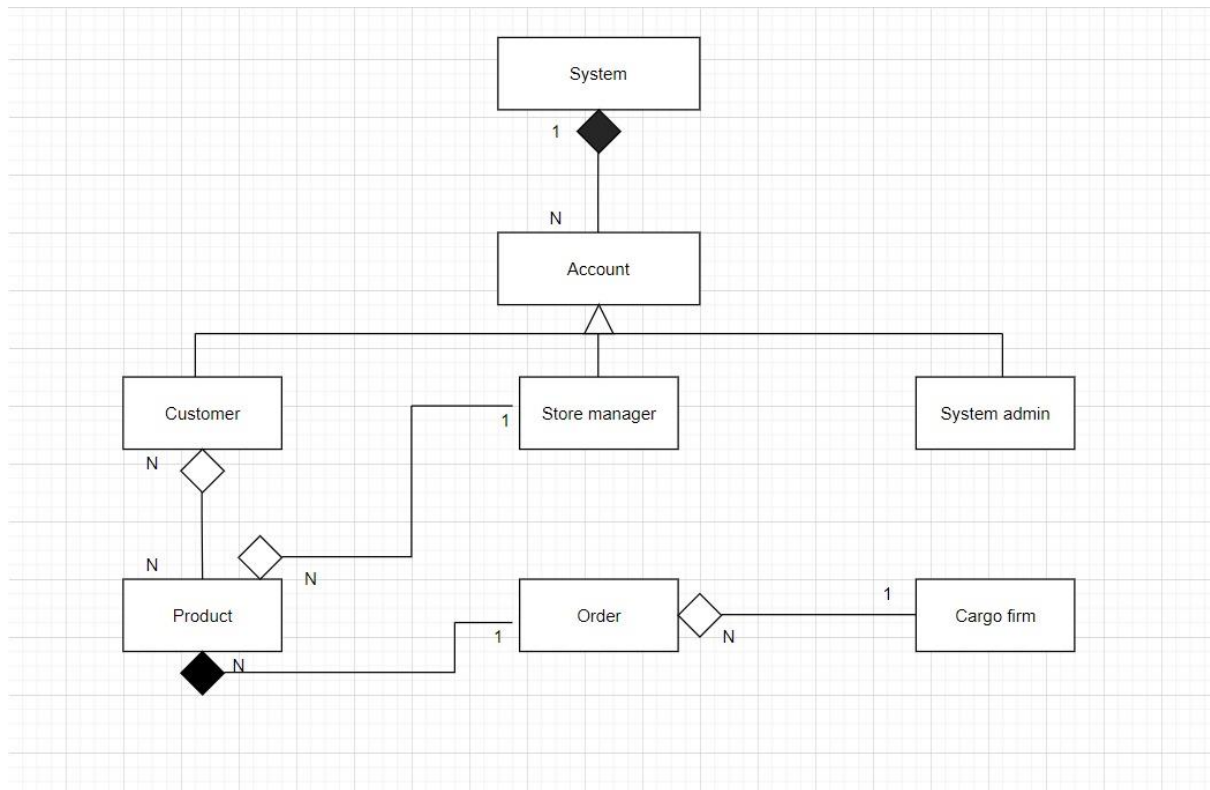
Ex-2 store manager is already registered in the system.

- The system offers a sign in operation to the store manager
- System jumps to phase 2

Ex-6 legal and paperwork problem:

- The process ends.

Domain Model as a UML Diagram



Conclusion

In this report we identified The Viewpoints and their Requirements. While preparing the report, functionality and lifetime are taken into consideration. And, we prepared fully dressed use cases (main use cases) which gives a useful idea of how the project process is going to be. At the end of this report work, we understood that requirement reports are helping to developers very well.

Teamwork: all contributions were equal and 20%.