UNIVERSITY OF DAR ES SALAAM

COLLEGE OF INFORMATION AND COMMUNICATION TECHNOLOGIES

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CS332 OBJECT-ORIENTED PROGRAM DESIGN AND ANALYSIS

PROJECT TITLE: STOCK CONTROL SYSTEM

GROUP No. 01

Name	Reg No.	Programme
Helela Robinson	2015-04-02575	BsCS
Rwegasira, Lucy M.	2015-04-02460	CEIT
Kamando, Frank G.	2015-04-02451	CEIT
Mbele, Amina J.	2015-04-07033	CEIT
Christopher, Emmanuel	2015-04-07051	BsCS
Temu, Prosper W.	2015-04-02455	CEIT
Mugoro, Claire F.	2015-04-07049	BsCS
Jonas, Fred E.	2015-04-07050	BsCS
Mbuba, Harun T.	2015-04-02574	BsCS
Faustine, Walter V.	2015-04-08507	BsCS

STOCK CONTROL SYSTEM

1. PROJECT OVERVIEW AND OBJECTIVE

1.1. OVERVIEW

Stock control is the process of ensuring that appropriate amount of stock is maintained by the business, so as to be able to meet customer demands without delay while keeping the overall cost to a minimum. Stock control process involves having the right level of stock to satisfy customer needs and identify excess and old stock.

During stock control it is not advised to run stock too low but also having too much stock can cost a lot of money. The estimated cost of holding stock which includes storage cost, insurance, keeping accurate records and controlling to avoid theft are estimated to be ten to thirty percent of the overall stock value.

Currently management of stock in the business is done through manual method. The use of paper-based system to record all information associated with stock always lead to the wastage of time, excessive inventory in stock and unable to move it quickly as well as misplacement of information during the process of receiving and delivering stock product.

1.2. MAIN OBJECTIVE

The main objective of the project is to develop a stock control system which will keep record of all products in formation and give notification.

1.3. GOALS

The goals of the stock control system are: -

i. To make sure that customers always have access to products when they need.
ii. To avoid excess inventory by balancing the fine line between too much and too little.
iii. To move goods efficiently by quickly receiving and storing products as they come in and when they go out.
iv. To determine current stock level and value of stock by tracking the individual items
v. Looking at sales records to find which items are highly sold, slow moving and which are seasonal items.
vi. To maximize profit margins.

2. SYSTEM FUNCTIONS

2.1. MAIN PROCESSES

i. Management of product
   a) Input of the purchased products
   b) Categorize products

ii. Inventory control
   a) Categorize products
   b) Evaluate inventory flow
   c) Provide notification
   d) Updating the inventory information

iii. Management of product dispatch
   a) Record moving out of products
   b) Update product information

2.2. FUNCTIONAL REQUIREMENTS

Ref#		Functions	Category
R.1		**Management of purchased products**	
	R.1.1	The system should allow categorization of products	Evident
	R.1.2	The system should allow registering of product based on their categories.	Evident
	R.1.3	The system should calculate the total number products	Hidden
	R.1.4	The system should update the inventory information	Hidden
R.2		**Management of product dispatch**	
	R.2.1	The system should allow recording of the moving out of products	Evident
	R.2.2	The system should update the inventory information	Hidden
	R.2.3	The system should provide the list of the product available.	
R.3		**Inventory control**	
	R.3.1	The system should allow setting of parameters	Evident
	R.3.2	The system should track the rate of output of products	Hidden
	R.3.3	The system should provide notification about product status	Evident
R.4		**Management of User**	
	R.4.1	The system should allow registering of user	Evident
	R.4.2	The system should allow adding of user roles	Evident
	R.4.3	The system should update user information	Hidden

2.3. NON-FUNCTIONAL REQUIREMENT

Attributes	Constraints
Usability	The system will be easy to use, learn and adapt. Users will become skillful while using it.
Correctness	The system must guarantee the correct result for the correct input data to obtained required results.
Security	System will provide security and protection of information through the following: - a. The system will sign-off user automatically when the system is idle for more than fifteen minutes. b. In order to use the system, system will authenticate user by asking them to enter credentials i.e. username/email and password c. All emails registered to the system will be verified in order to prevent user to enter unregistered email that will help password recovery process whenever is forgotten.
Scalability	The system will be able to allow scalability by allowing the integration with other technologies and new inputs depending on the existing potential technology in stock control.
Reliability	System will guarantee no inconvenience, unnecessary waiting time and be able to recover to its stable state after experiencing downtime.
Maintainability	The system should be easy to maintain to avoid unnecessary maintenance costs.

3. ACTORS AND USE CASES

3.1. Identified actors

S/N	Actor	Description
1	System Administrator	This is the person who monitors the system functionality, he/she ensures that the system works correctly to provide the required functionality.
2	Store keeper	This is the one who is in charge of all the products in the business
3	Business owner	This is the one who owns everything in the business he/she supplies the necessary business requirements

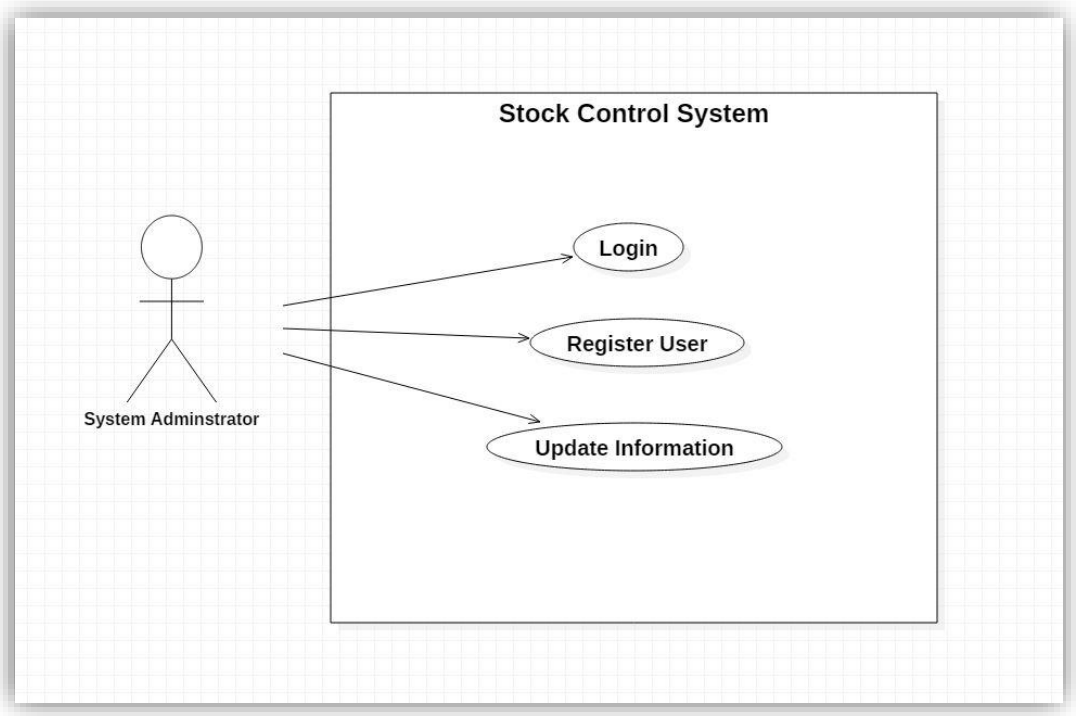
3.2. Identified use cases

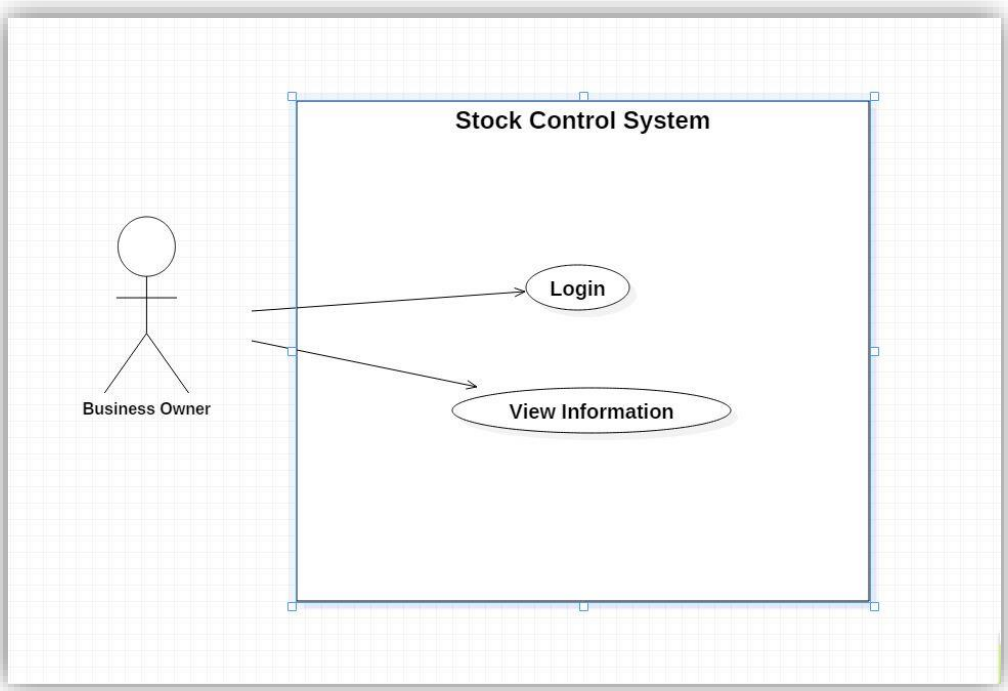
S/N	Use Cases	Description
1	Login	Business Owner, Store keeper and System Administrator Login in the stock control System.
2	Register Users	System Administrator register Business Owner and Store keeper in the stock control System.
3	Update User Information	System Administrator update all information's of Business Owner and Store keeper in the stock control System.
	Add Product	Store keeper add products in the stock control System.
	Update Products	Store keeper update products in the stock control System.
	View Product Information	Store keeper and Business owner view product information's in the stock control System.

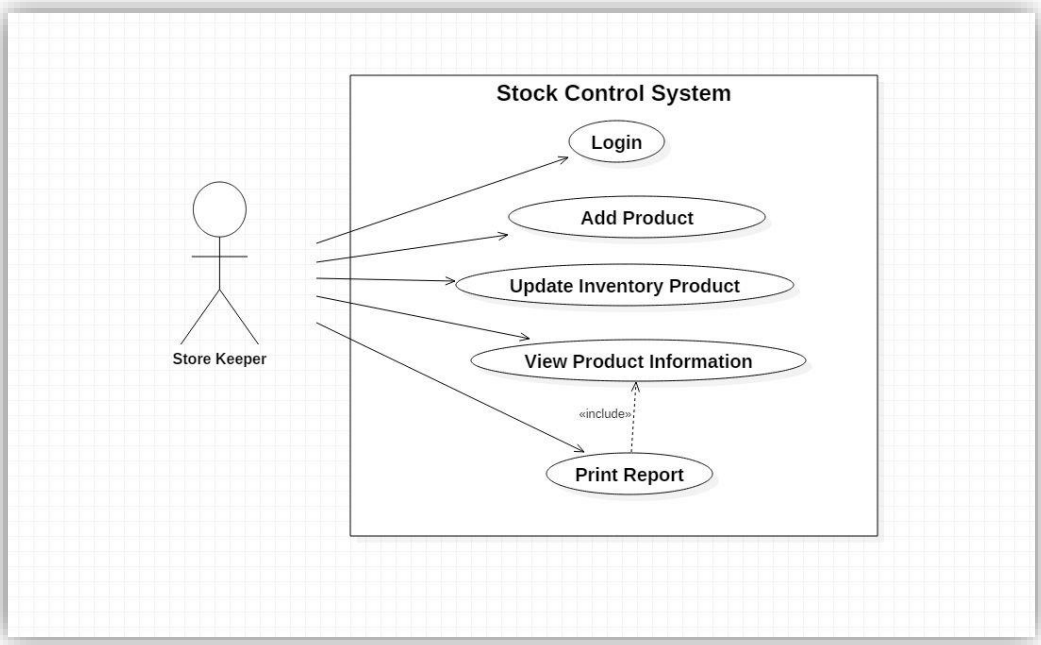
3.3. Mapping actors and use cases

Actor	Description	Use Cases
System Administrator	This is the person who monitors the system functionality, he/she ensures that the system works correctly to provide the required functionality.	a. Login. b. Register users. c. Update user information. d. Remove users.
Store keeper	This is the one who is in charge of all the products in the business	a. Login. b. Add inventory product. c. Update inventory product. d. Remove inventory product. e. Print report. f. View product information.
Business owner	This is the one who owns everything in the business he/she supplies the necessary business requirements	a. Login b. View information

3.4. Use case diagram




Stock Control System
Business Owner
Login
View Information


Stock Control System
Store Keeper
Login
Add Product
Update Inventory Product
View Product Information
«include»
Print Report

3.5. Use case description

Use Case:	Register product
Actors:	Storekeeper
Short Description:	A use case allows a storekeeper to record a new product to the stock
Pre-Condition:	A product to be recorded is present and its properties are described well
Post- condition:	Product will be added to the stock and stock level will increment
Main Flow:	1. Storekeeper browse a system, and a list of products will display with their properties. 2. Storekeeper record a product in the system according to its category and specification. 3. A system will increment a stock level.
Alternative Flow:	If the product does not being recorded the system will prompt to re-record again.
Exception Flow:	If the product to be recorded is present exception handling.

Use Case:	Product dispatch
Actors:	Storekeeper
Short Description:	A use case allows a storekeeper to dispatch products quantity from the stock
Pre-Condition:	A product to be dispatched is present and its properties are described well
Post- condition:	Product will be dispatched to the stock and product level will be decremented
Main Flow:	1. Storekeeper browse a system, and a list of products will display with their properties. 2. Storekeeper search for the product to be dispatched in the system according to its category and specification. 3. Storekeeper will dispatch the quantity required by decrementing current value. 4. System will decrease the product quantity in the stock.

Alternative Flow:	If the product is not being recorded/known by the system, will prompt to register that product.
Exception Flow:	If the storekeeper wants to dispatch more than the current value present.

Use case:	Update product
Actors:	Storekeeper
Short descriptions:	This use case allow storekeeper to keep update of product in and out the stock
Pre-condition:	Transaction must occur in a stock (either addition of the product or the remove of the product)
Post-conditions:	A number of product balance has calculated and system update product information.
Main flow	1. Storekeeper invokes a page for registering products and a system displays a page for entering a product information. 2. Storekeeper enter type and amount of product involved in the transaction and submits. 3. System tracks the information of the specified product type and perform calculation depending on the transaction type and amount of the product. 4. System validate the new amount of the product and update the product information. 5. System send a notification that the product information has updated successfully
Alternative flow(s):	In item (3), if the product information doesn't exist in the system, system will (i) notifies user that the product information does not exist and (ii) then system prompt user to register a new product
Exception flow(s):	In item (4) if the new amount is less than zero the system displays a message 'invalid amount', amount cannot be less than zero.

4. Conceptual Modelling

4.1. Concepts

a. Business owner

b. System Administrator

c. Store Keeper

d. Product

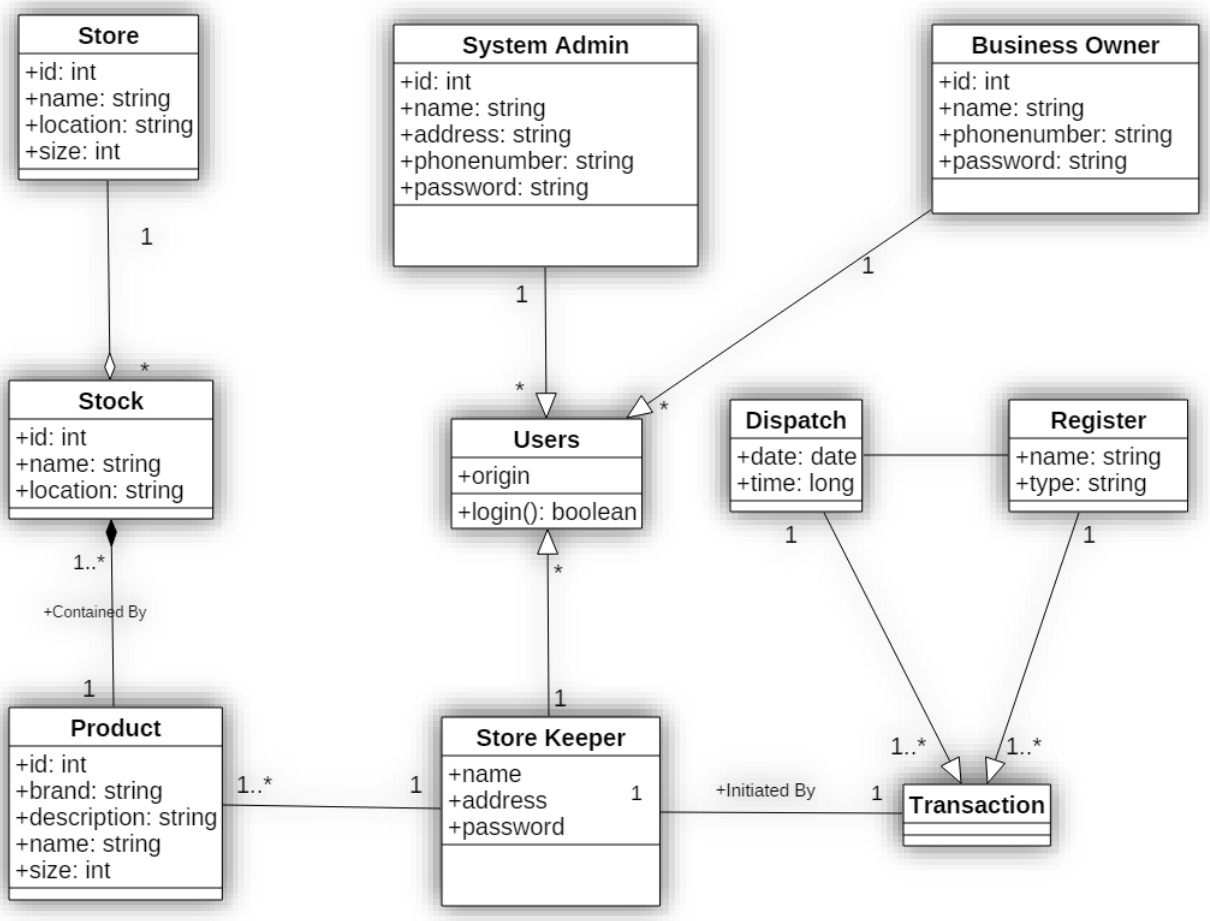
e. Product Category

f. Notification

g. Store

h. Stock

4.2. Concept Diagram



5. SYSTEM INPUT EVENT AND OPERATION

5.1. Storekeeper: Register Product

INPUT EVENT	OUTPUT EVENT
1. Select product category 2. Enter product details 3. Submit product details	4. Notify is the product is successfully inserted or not 5. Display entered product in the list.

5.1.1 Register Product Tracer Diagram

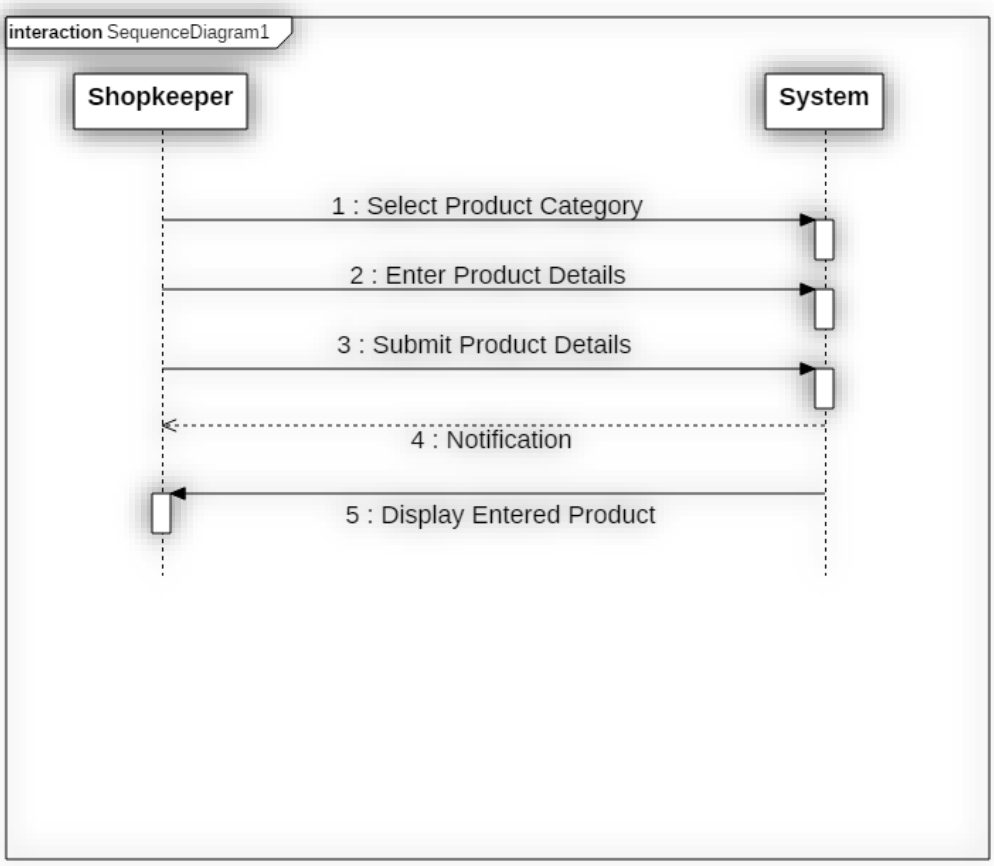


Figure 1: Register Product Tracer Diagram

5.1.2 Register Product Sequence Diagram

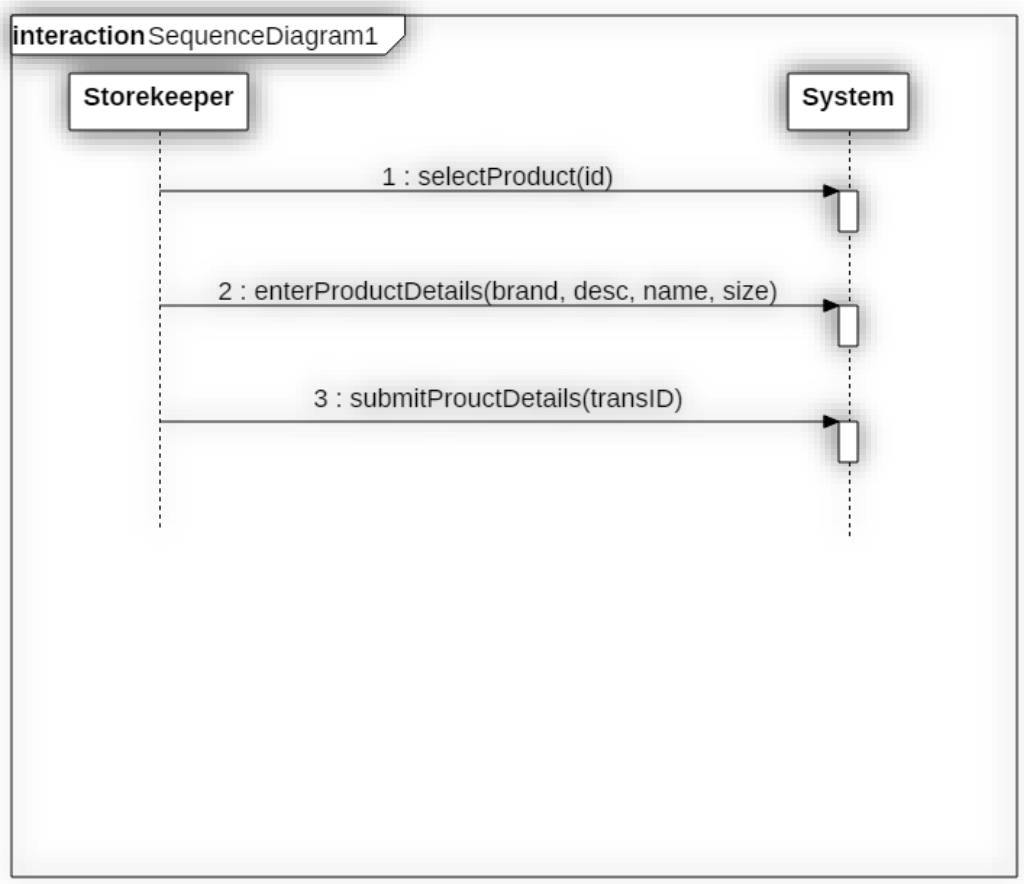


Figure 2: Register Product Sequence Diagram

5.2. Storekeeper: Product Dispatch

INPUT EVENT	OUTPUT EVENT
1.Storekeeper browse the system. 3.Select product to be dispatched. 5.Storekeeper enters amount of product required	2.Display list of products and its properties. 4. Display product information 6.System display the remaining quantity of the product

5.2.1 Dispatched Product Tracer Diagram

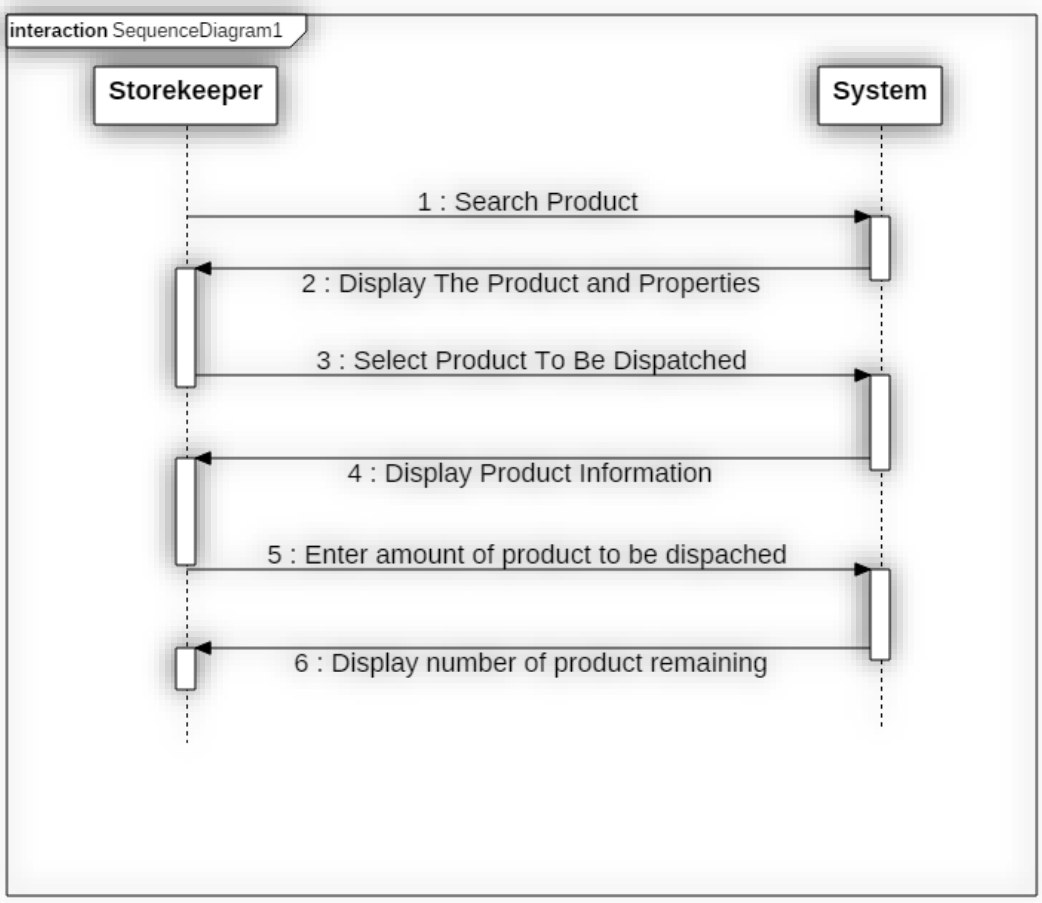


Figure 3: Register Product Tracer Diagram

5.2.2 Dispatched Product Sequence Diagram

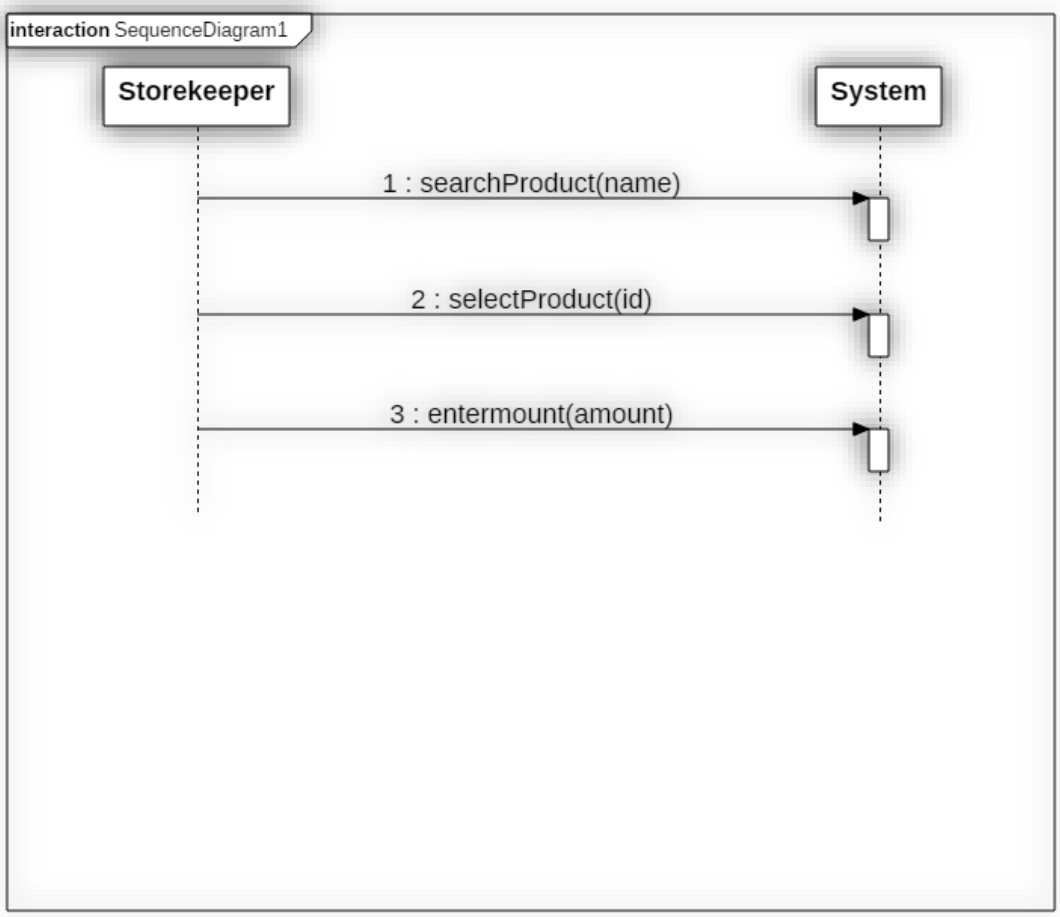


Figure 4: Register Product Sequence Diagram

6. CONTRACTS

Name	enterProductDetails(brand, desc, name, size)
Responsibility	Record product brand, description, name and size
Type	System
Cross reference	System function R.1.2 Usecase: Register Product.
Note	
Exceptions	If quantity not valid indicate and error
Output	
Pre-conditions	The product details and properties to be recorded must first be well described.
Post Condition	i. Product:name was set to name ii. If a new product. A new product was created iii. The product was associated with a stock

Name	amountDispatched(quantity)
Responsibility	Provide number of product to be dispached
Type	System
Cross reference	System function R.2.1 Usecase: Product Dispatch
Note	
Exceptions	If amount of product entered exceeds the current number of product indicate an error.
Output	
Pre-conditions	Product dispatched must be available
Post Condition	i. amountDispached:quantity is modified(decremented) ii. The product is associated with the store (The product amount is reduced)

Name	selectProductCategory(name)
Responsibility	To group the product into categories
Type	System
Cross reference	System function R.1.1 Usecase: Product Dispatch
Note	
Exceptions	If amount of product entered exceeds the current number of product indicate an error.
Output	
Pre-conditions	Presence of product with different categories must be available.
Post Condition	i. Product is associated with category

Name	searchProduct(name)
Responsibility	To get the list of product available
Type	System
Cross reference	System function R.2.3 Usecase: Product Dispatch
Note	
Exceptions	If item to be searched is not available in the system then show notification to notify user about occurred event.
Output	
Pre-conditions	Product to be searched must be first exist/registered to the system.
Post Condition	i. An association between store keeper and product is created. ii. A new instance product is created.