Daffodil International University

Inventory Management System

Course Project

Software Engineering Project 2: Web Programming
Course Code: SWE331

NAME : Md. Sajib Mulla

ID : 143-35-816

DEPARTMENT : Software Engineering

Faculty of Science and Information Technology

December 2017

Table of Contents

Chapter 1

Introduction

1. Introduction

1.1 About the System

- In this proposal, we introduce a software for a shop that Calculate, Store Data, Shop Product items, Customer Info, Lender Info Etc. We developed this software for easily and safely Store Data and calculate total shop calculation. This project is complete system that help a shopkeeper for daily activity.

1.2 Purpose

- Easily Know How much product have in a time
- Easily Calculate
- Save Customer Information
- Save lender information
- Save Daily Sale information
- Easily sale a product

1.3 Scope

This system mainly create for shopkeeper. Shopkeeper easily can save daily activity on this system, like how was today condition on shop, how much sell, customer information, lander information, how much product have now in shop etc.

1.4 Vision

1.5 Why this system is necessary?

This system is necessary to reduce a shop information memories. For more easy to sale product and include product to shop. It's necessary for easily control a shop.

1.6 Proposed Solution

Chapter 2

System Analysis

2. System Analysis

2.1 Use Case Model

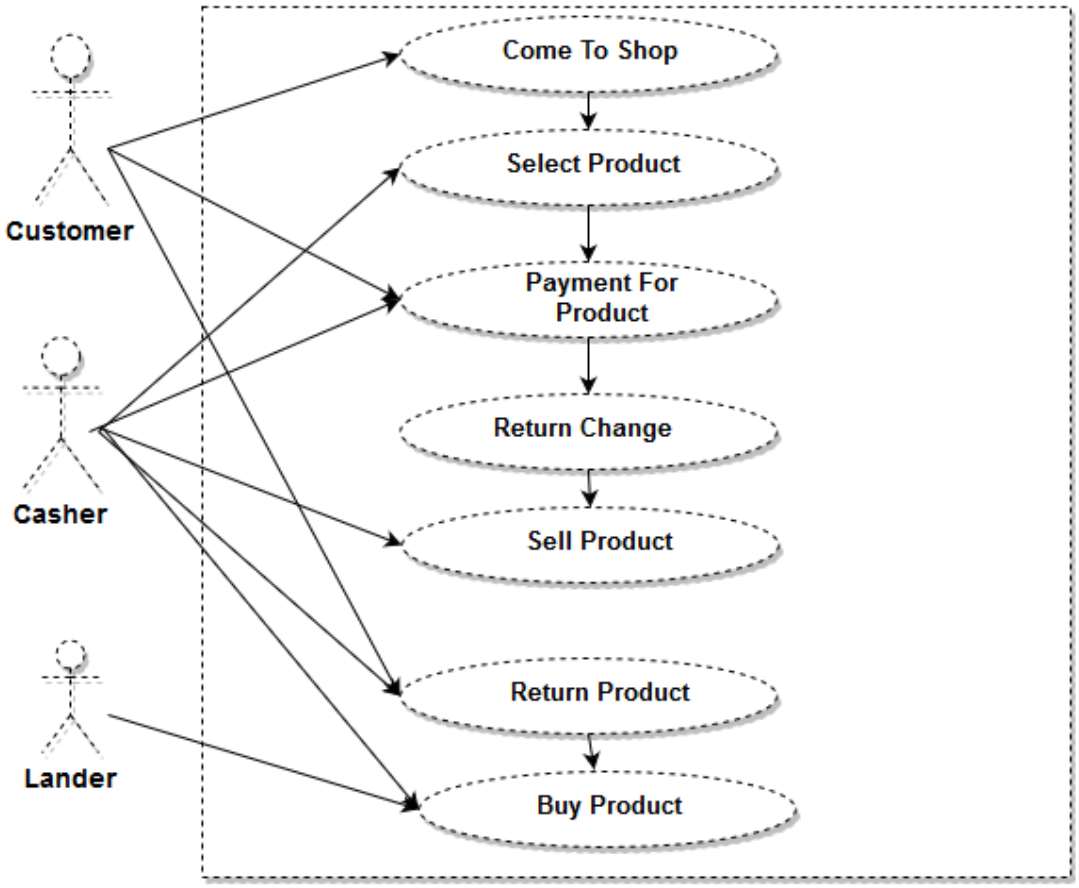


Fig 2.1:Use Case Diagram

2.2 Actor Goal List

2.2.1 Shopkeeper

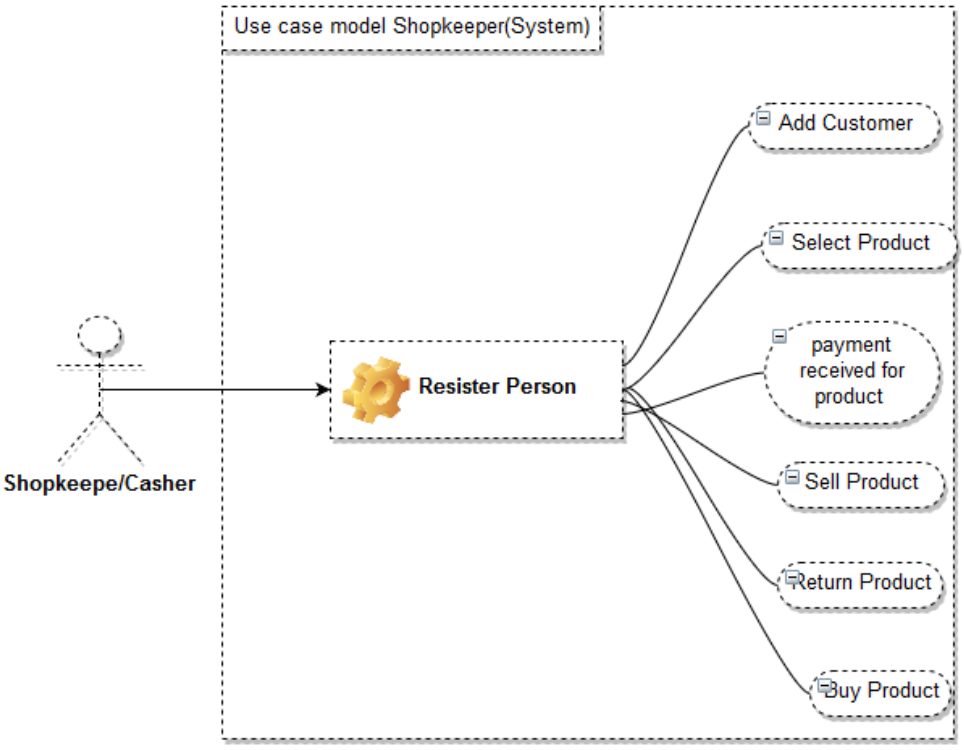
- Login to this system by using User_Name and Password.
- If customer come to buy product then shopkeeper cunt product which are selected by customer.
- Calculate the total product price and take payment for all product

- If customer back money return them
- If customer want to change or back product system can track it,
- Add product in shop which buy by Lander.

2.3 Use Case Model

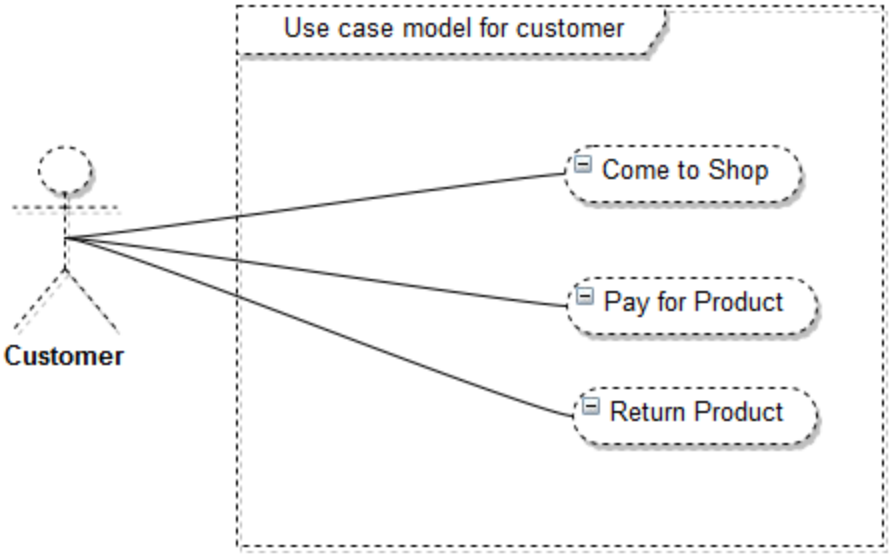
2.4 Use Case Description (Brief)

2.4.1 Shopkeeper/Casher



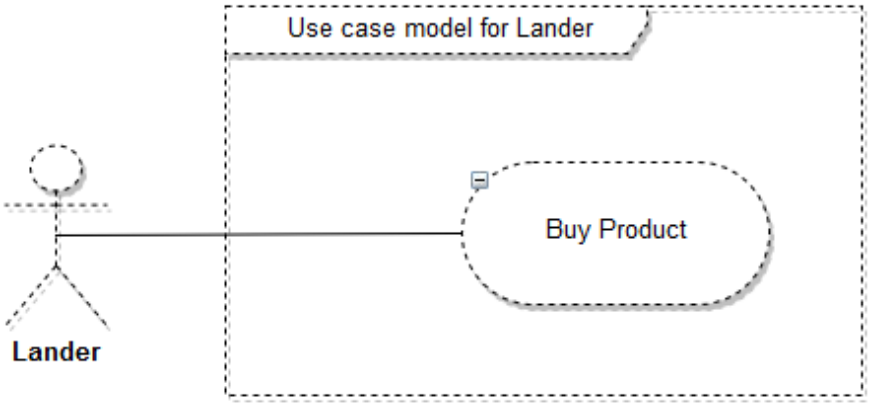
- In this phase who wants to use this system they should be resister by authority.
- Casher can add customer
- Select product for counting bill
- Take payment from customer
- Sell product which selected by customer
- Return product if customer want to back product
- Buy product for shop increase .

2.4.2 Customer



- Customer have to come shop if they wants to buy somethings
- Then pay for selected product
- If product have any problem customer can change or return money.

2.4.3 Lander



- Lander mainly he/she was the owner of this system. Who buy product for this shop.

2.5 Use Case Description (Detailed)

2.5.1 Register Shopkeeper

Use Case ID	1	
Name	Shopkeeper	
Primary Actor	Casher	
Goal	Sale product and also add customer information.	
Precondition	After Successfully join the system they fully control the system	
Post Condition	Shopkeeper can change Customer information and shop inventory.	
Main Success Scenario	**Actor**	**System**
	1. After login the system shopkeeper can add customer 2. Calculate bill 3. Make recite for product 4. Able to see all customer 5. Able to buy product for shop and add to shop	1.1) System return successful message to user 2.1) System return message when new product add
Failure Scenario	1) Don't able to login. 2) Don't able to add customer. 3) Connection problem to database 4) Don't save new product to database	

2.5.2 Customer

Use Case ID	2	
Name	Customer	
Primary Actor	Customer	
Goal	Select product to buy.	
Precondition	Customer have to come shop to buy product. Pay for selected product	
Post Condition	After buying product take recite for total information	
Main Success Scenario	**Actor**	**System**
	1. When come to shop customer can see which product available now 2. Can return product 3. Invoice copy	1.1) System show product now available 2.1)After complete buy system give a invoice copy .

Failure Scenario	1) Don't able connect database 2) Don't show available product. 3) system loss previous record

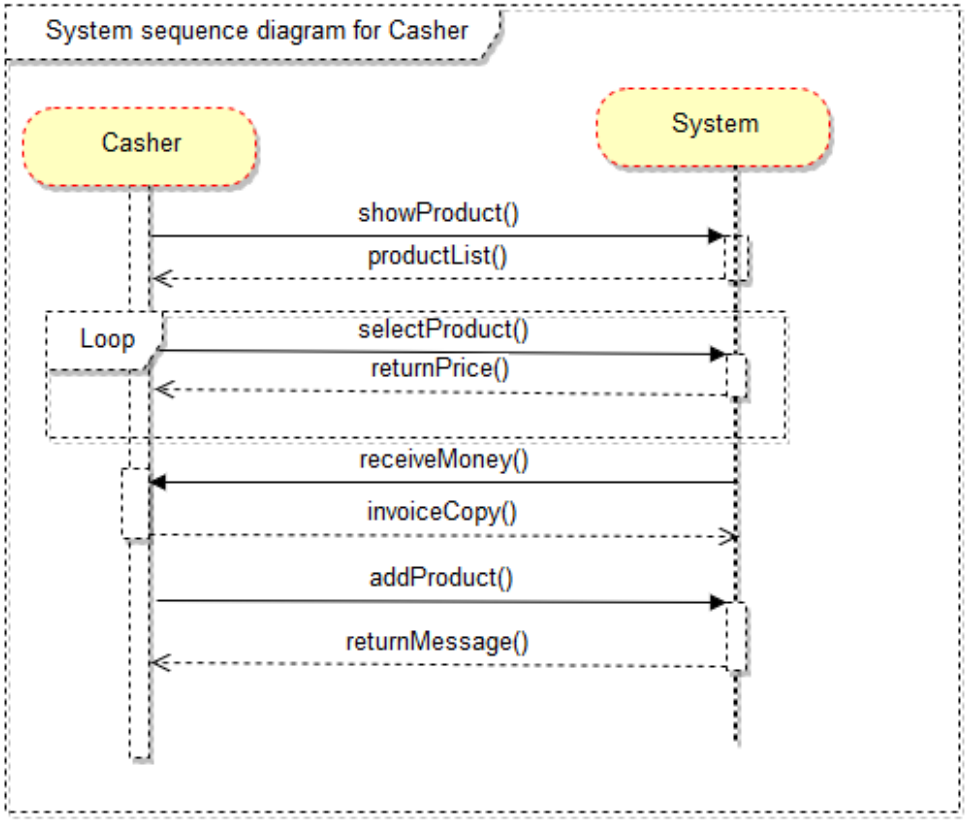
2.5.3 Lander

Use Case ID	3
Name	Lander
Primary Actor	Lander
Goal	To add product and monitor the shop.
Precondition	Have to buy product to add shop
Post Condition	Take a confirmation message that product add successfully
Main Success Scenario	1. Product add system 2. System give information that which product below now
Failure Scenario	1. Can't add product to customer 2. Add product system can't add product.

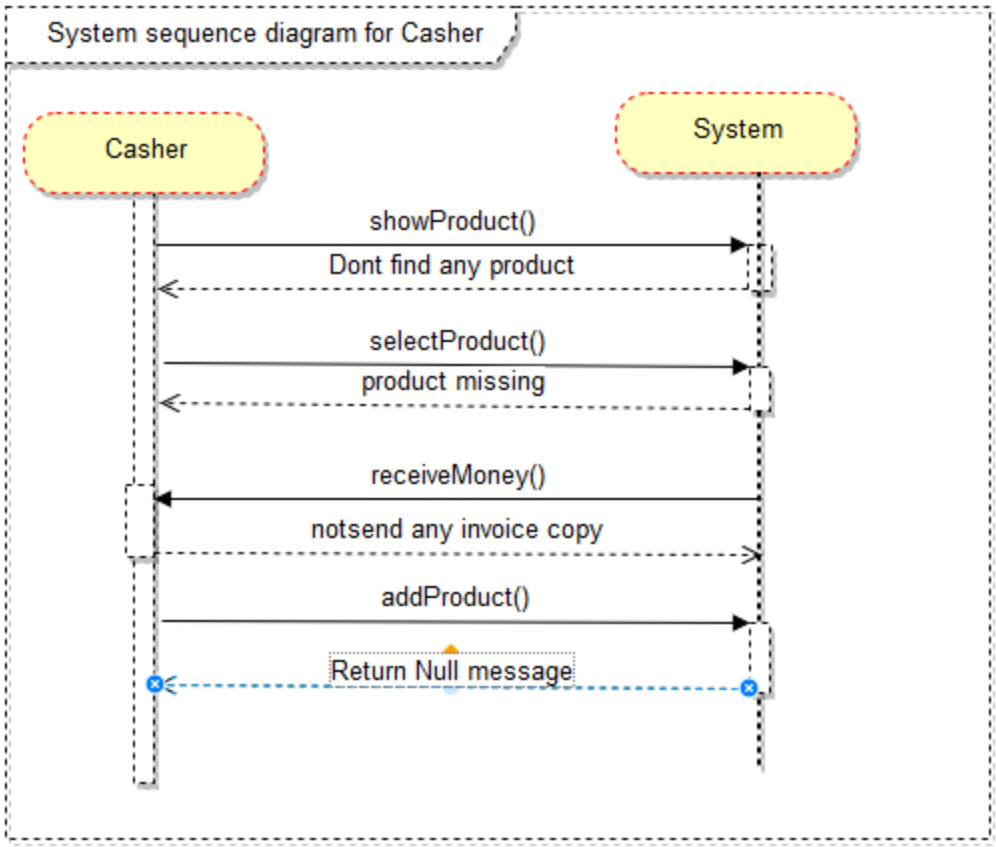
2.6 System Sequence Diagrams

2.6.1 Register Shopkeeper

2.6.1.1 Success scenario

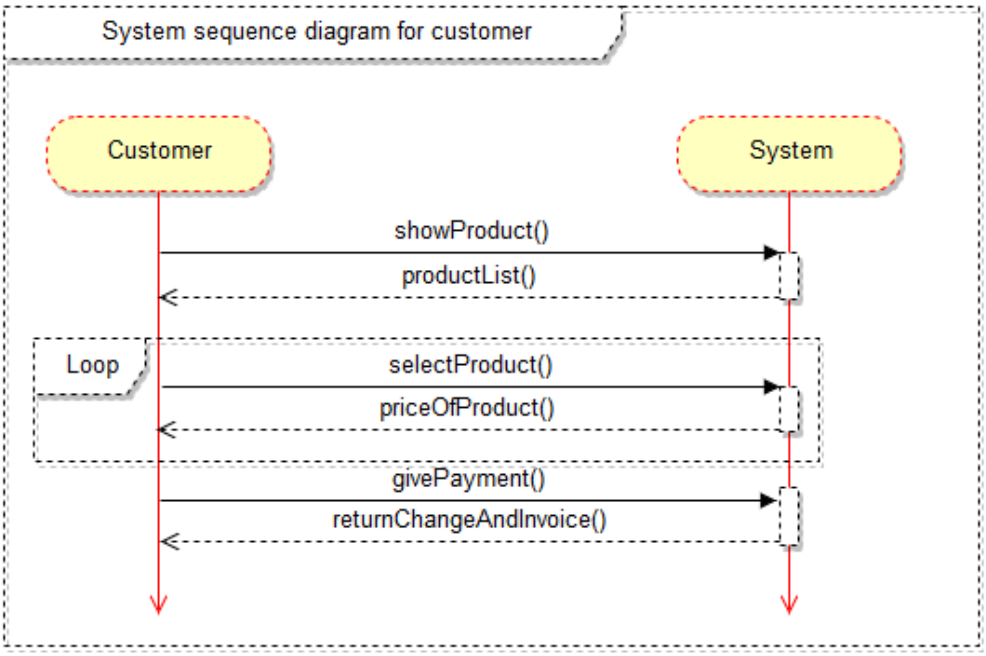


2.6.1.2 Failure Scenario

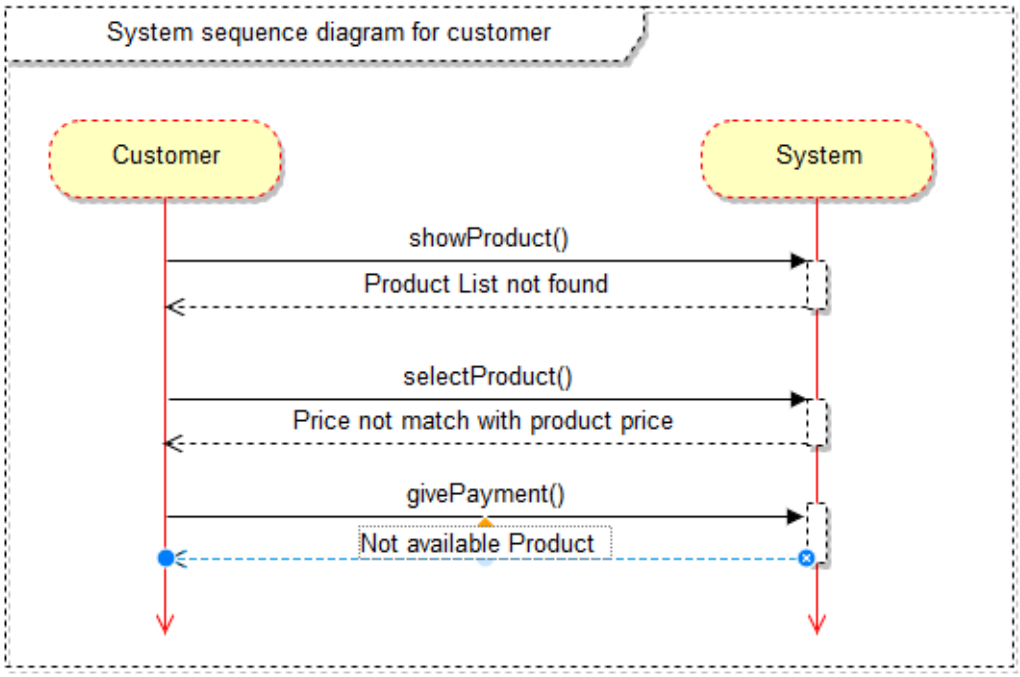


2.6.2 Castomer

2.6.2.1 Success scenario

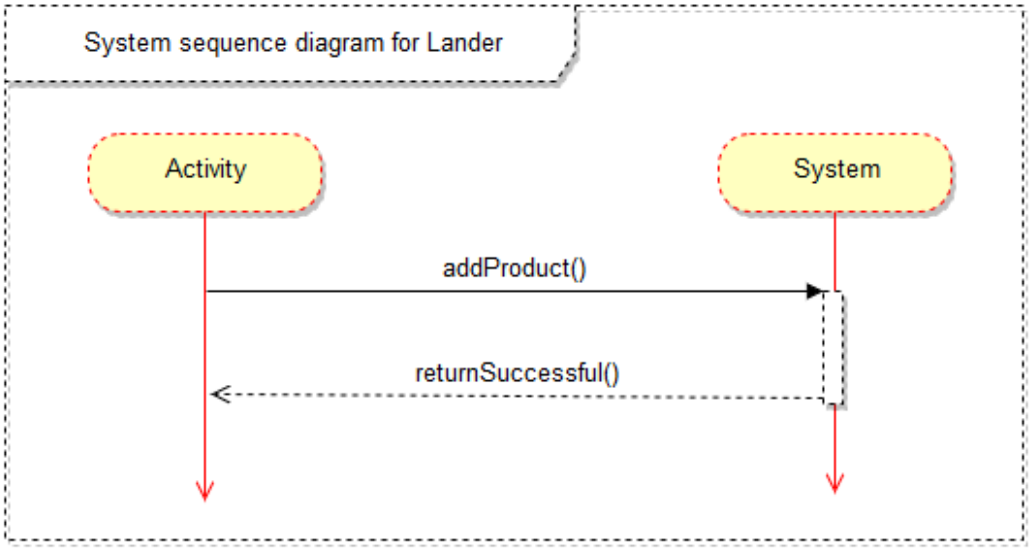


2.6.2.2 Failure scenario

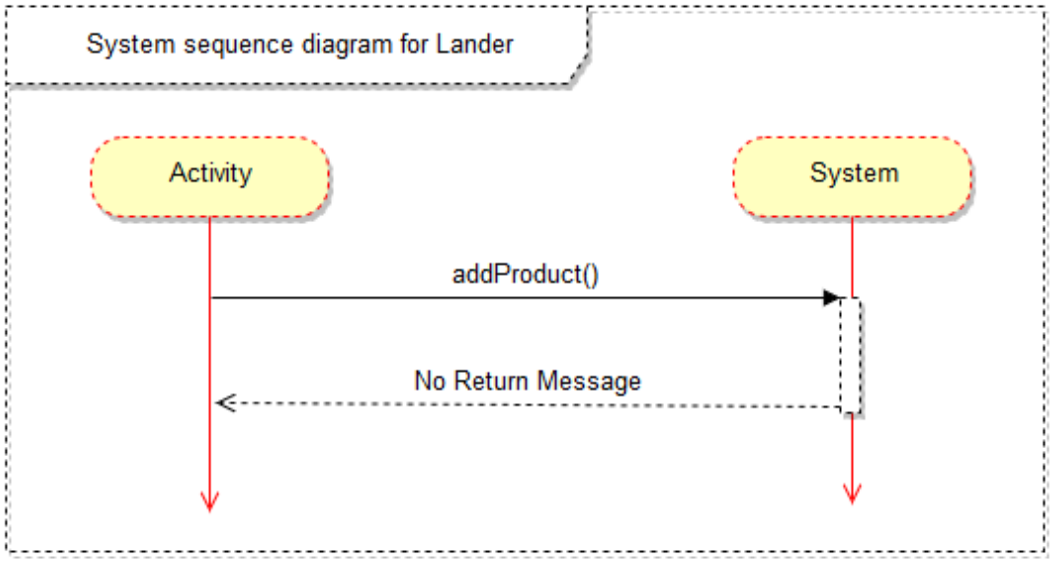


2.6.3 Lander

2.6.3.1 Success scenario



2.6.3.2 Failure scenario


System sequence diagram for Lander
Activity
System
addProduct()
No Return Message

2.7 Domain/Conceptual Model

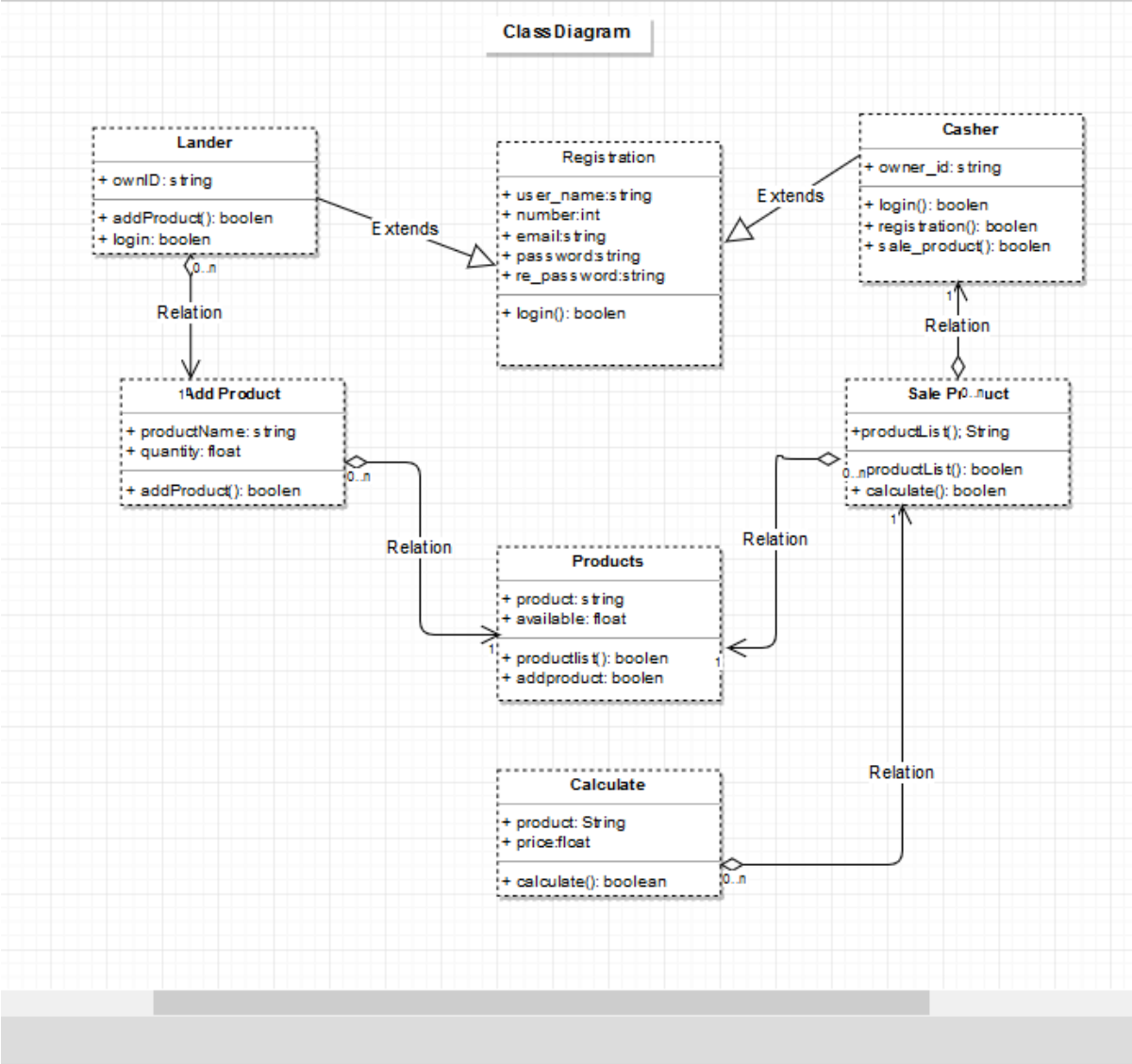
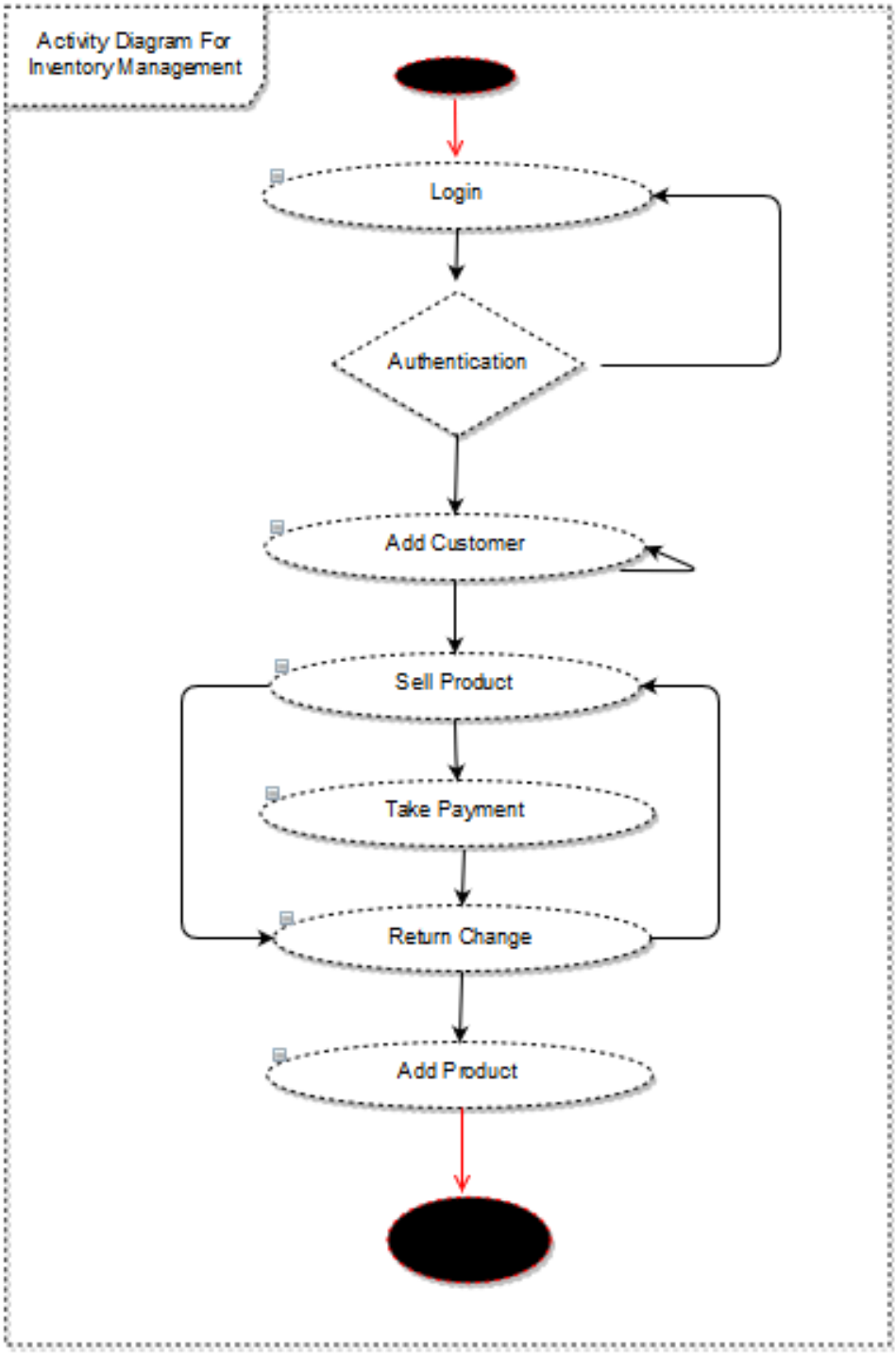


Figure 2.17 System Domain Model

2.8 Activity diagram

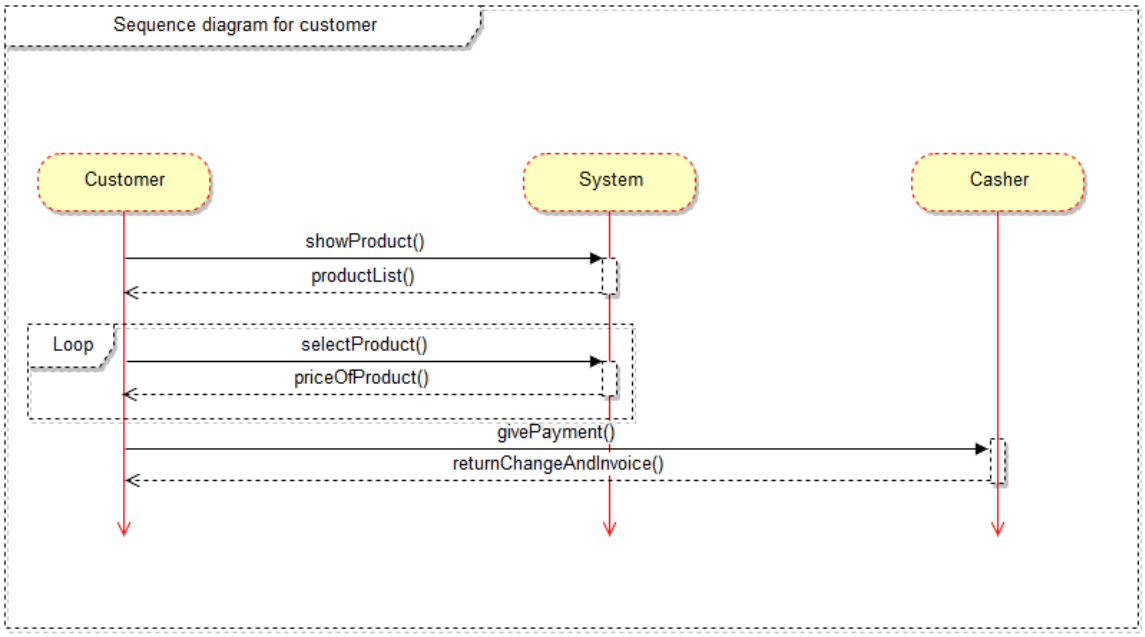


Chapter 3

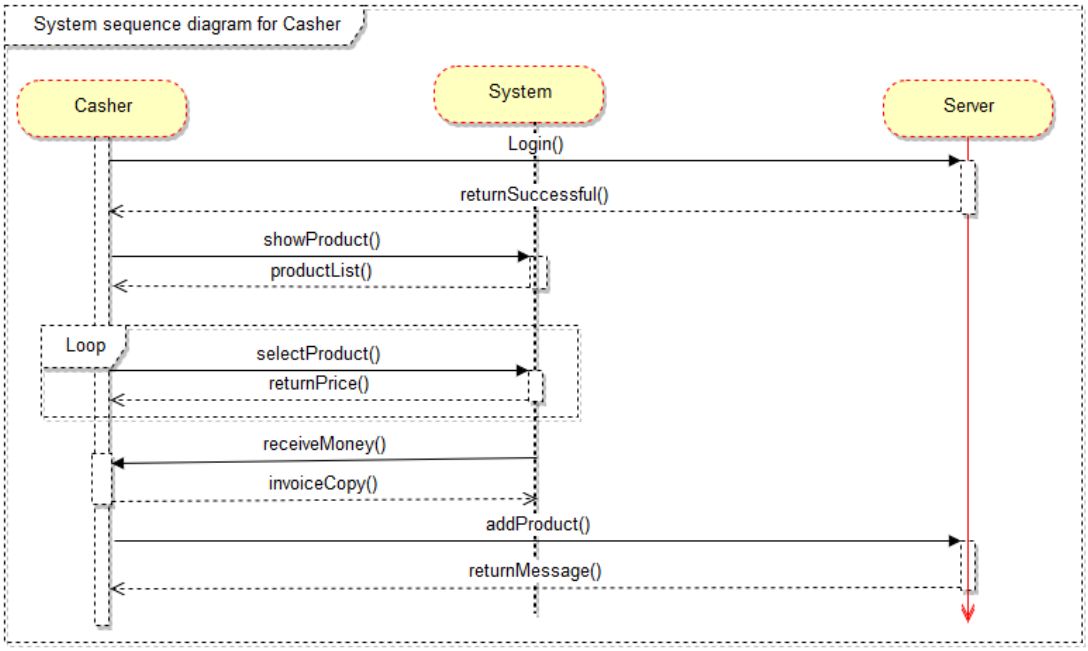
System Design

3.1) Sequence Diagram

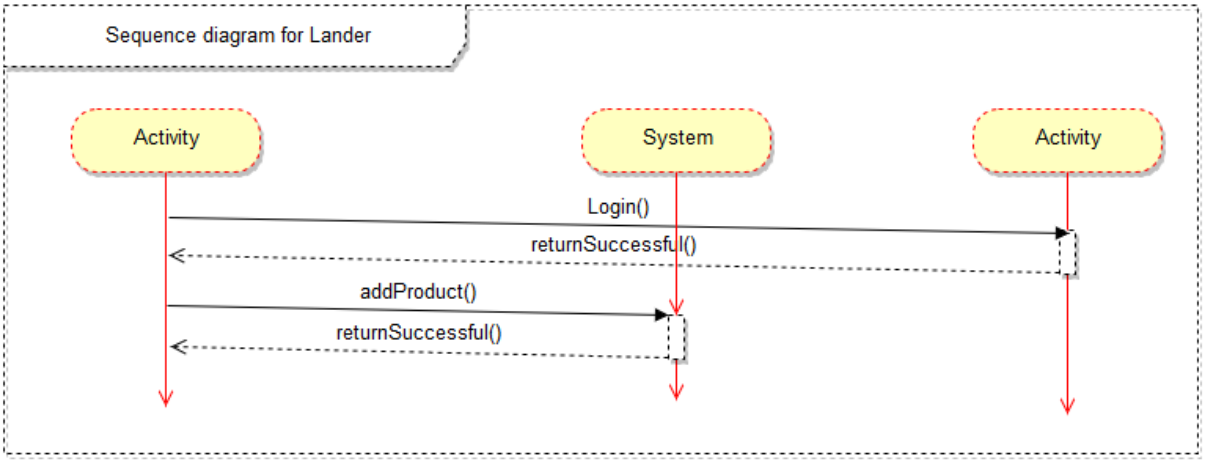
3.1.1) Shopkeeper:



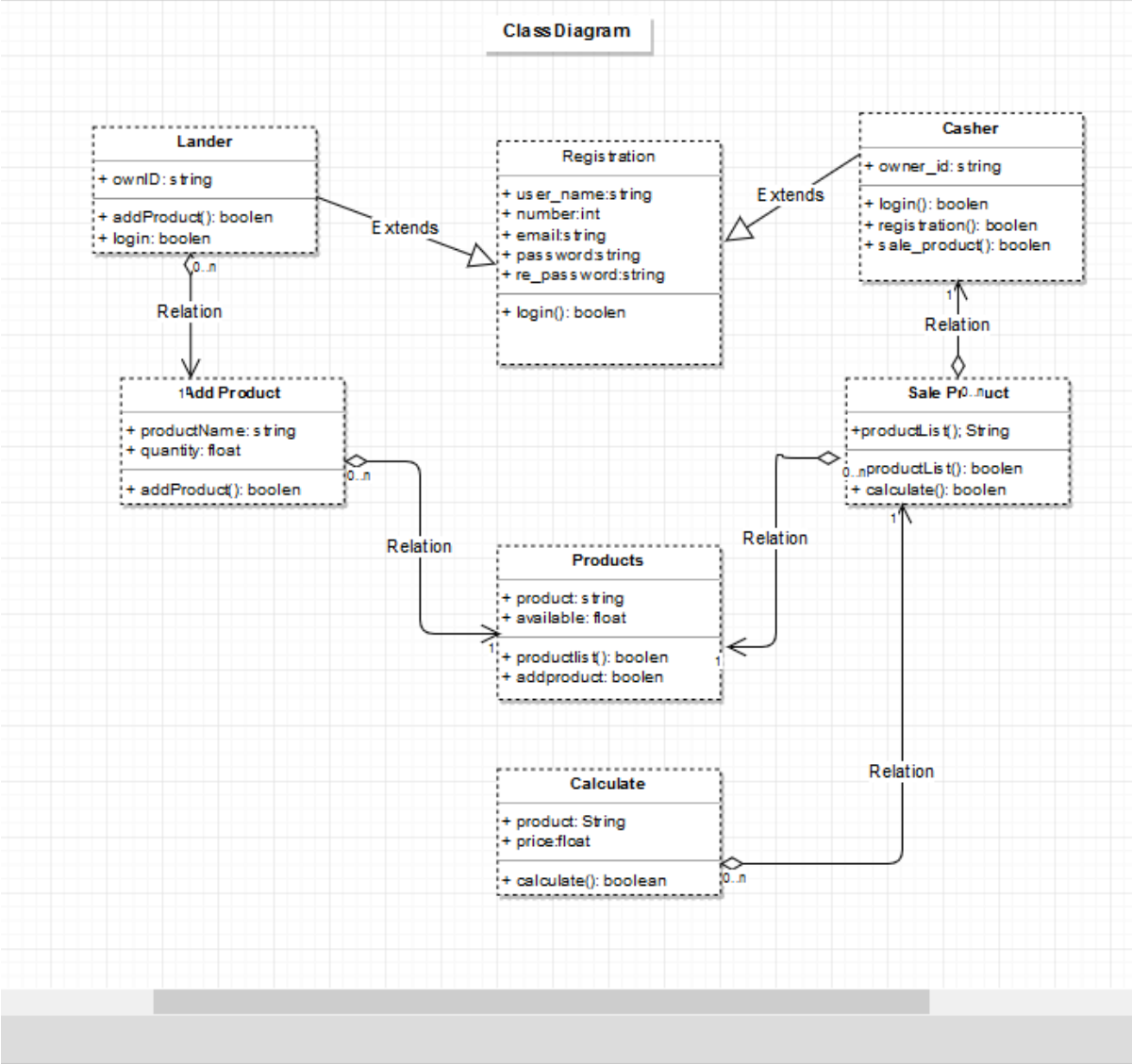
3.1.2) Casher:



3.1.3 Lander



3.2) Entity Relationship diagram:



Chapter 4

Implementation

3. Implementation

Implementation (software) perspective describes software implementations in a particular technology (such as C#). In the UP, Implementation means programming and building the system, not deploying it.

In the implementation phase, the developer builds the components either from scratch or by composition given the architecture document from the design phase and the requirement document from the analysis phase. The architecture document should give guidance. Sometimes, this guidance is found in the requirement document. The implementation phase deals with issues of quality, performance and debugging. The end deliverable of implementation phase is the product itself.

3.1 Tools &Technologies

Following are the tools and technologies used in development of this project:

Microsoft Visual Studio 2012

ASP.NET Framework

Microsoft SQL Server 2008

Telerik reports

Microsoft Visio

Microsoft Architect

HTML5, CSS, JavaScript, J-query, Twitter bootstrap

3.2 Project Link

****

https://github.com/Msajib/InventoryManagmentSystem

****

References:

https://makebills.pro/