DEPARTMENT OF NETWORKING AND COMMUNICATIONS

FACULTY OF ENGINEERING & TECHNOLOGY

MINI PROJECT

SUBJECT CODE: 18CSC202J

SUBJECT TITLE: OBJECT ORIENTED DESIGN AND PROGRAMMING

PROJECT TITLE

TEAM MEMBER'S (NAME AND REG.NO)

Anamika Jain

RA2111030010098

Sharanya Ghosh

RA2111030010097

SRM University, SRM Nagar, Kattankulathur-603203
Kancheepuram District, Tamil Nadu

Rubrics

Experiment Component	Max. Marks	Grading Rubrics		
Documentation/ Procedure	10	UML Diagrams are well documented. The documentation supporting all functional requirement and non-functional requirement(10 Marks)	Missing two or more required functional requirement .The documentation work not up to the mark. (5 Mark)	
Concept	5	Completeness of concept, consistent variable naming and relationship in static view. (5 Marks)	Completeness of concept, inconsistent variable naming and relationship in static view. (3 Marks)	Incomplete static view. (1 Mark)
Usage of Symbols	3	Precise usage of symbols in dynamic view. (3 Marks)	Improper usage of Symbol's. (2 Marks)	Symbol's misplaced in diagram. (1 Mark)
Diagrams	4	Completion of all 8 UML Diagrams using Visual Paradigm Tool. (4 Marks)	Construction of UML Diagrams using other tools. (2 Marks)	Construction of few diagrams'. (1 Mark)
Viva and Innovative Idea	3	Oral Viva and Innovative approach. (3 Marks)	Oral Viva and partial idea. (2 Marks)	Oral Viva not fulfilled. (1 Mark)
TOTAL	25			

INDEX

- Package diagram
- Component diagram
- Deployment diagram

4. Conclusion
5. References

1. PROBLEM DESCRIPTION

This software project is a functionally enhanced version of a conventional billing system. This technology is designed to quickly process data and generate bills for customers. An efficient front end created in Asp.net and a SQL database make up the billing system. A sizable collection of product name, price, and other product-specific information can be found in the billing database. When a product is billed, the database is searched for it, and the price is added to the bill based on the quantity of the product. The system also includes discounts on numerous products, enabling the product to be billed at a lower price. The supermarket billing system was developed to assist supermarkets with bill calculation, bill display, and customer service. This software solution includes a useful and simple GUI to assist the employee in easily calculating bills and provide great customer support.

2. SOFTWARE REQUIREMENTS SPECIFICATION

2.1 INTRODUCTION

2.1.1 Purpose

The purpose of this application is to make the billing process in the supermarkets easier and accurate. By the help of product data base the bill can be generated easily including the discounts and taxes.

2.1.2 Indented Audience and Reading Suggestion

This document should be read by developers, users, project managers and teasers. The developers should read every section to ensure that there is an understanding of the project.

2.1.3 Project Scope

The Supermarket Billing System is an application that can be accessed by anyone. This application will generate bills. It provides complete information about the discounts and taxes and products.

2.2 OVERALL DESCRIPTION

2.2.1 Product Perspective

The proposed Supermarket Billing system provides an easy way out for the long billing process, reducing the time of writing every product's detail and calculating the amount to be paid.

2.2.2 Product Function

The Supermarket Billing System provides the information of the products available in the store with the discount (if any) further making the list of products and the total amount to be paid including the taxes.

It will also control and manage and update the customer database.

2.2.3 User Classes and Characteristics

We have levels of users:

Biller Module:

- Login/Logout
- Remove Product
- Add Product
- Payment

Administrator Module:

- Update Product Database

2.2.4 Operating Environment

Manual Use Only

2.2.5 Design and Implementation Constraints

Any update regarding the product and customer database is to be recorded and updated with correct and new information.

2.2.6 User documentation

There will be a basic tutorial document

2.3 EXTERNAL INTERFACE REQUIREMENTS

2.3.1 User Interface

The software provides good graphical interface for the user. Any administrator can operate on the system, should prompt user interface performing the required task.

2.3.2 Hardware Interface

Operating System: Windows

Hard Disk: 2TB

RAM: 16GB

Processor: Intel Core i5 1100H

2.3.3 Communication Interface

Window

2.4 SYSTEM FEATURES

2.4.1 Portability

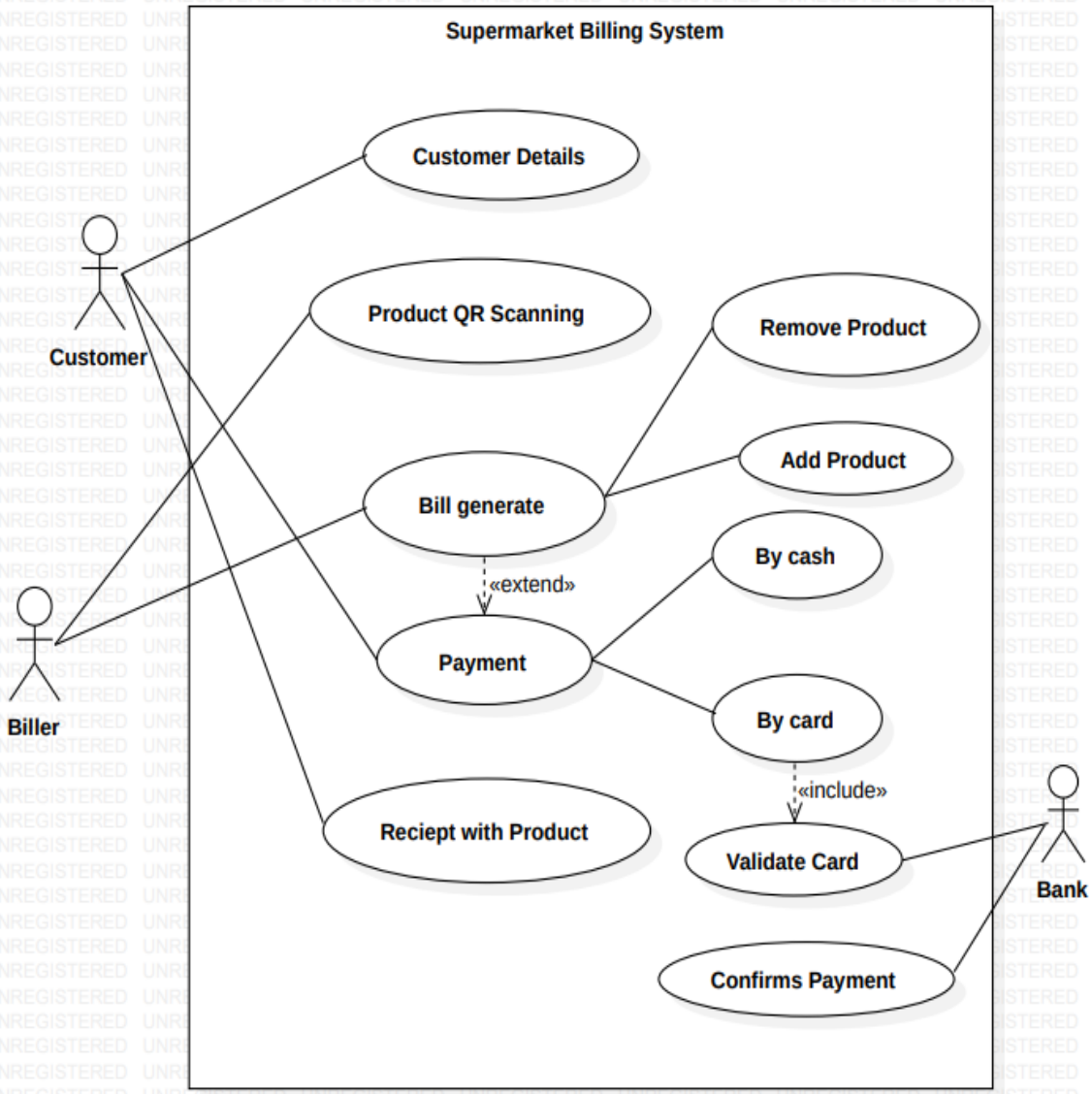
The system is developed for secured purposes, so it can’t be portable.

3. DIAGRAMS

The unified visual representation of the UML (Unified Modelling Language) system is shown in a UML diagram, which is meant to help developers or business owners understand, examine, and implement their system's structure and behaviour.

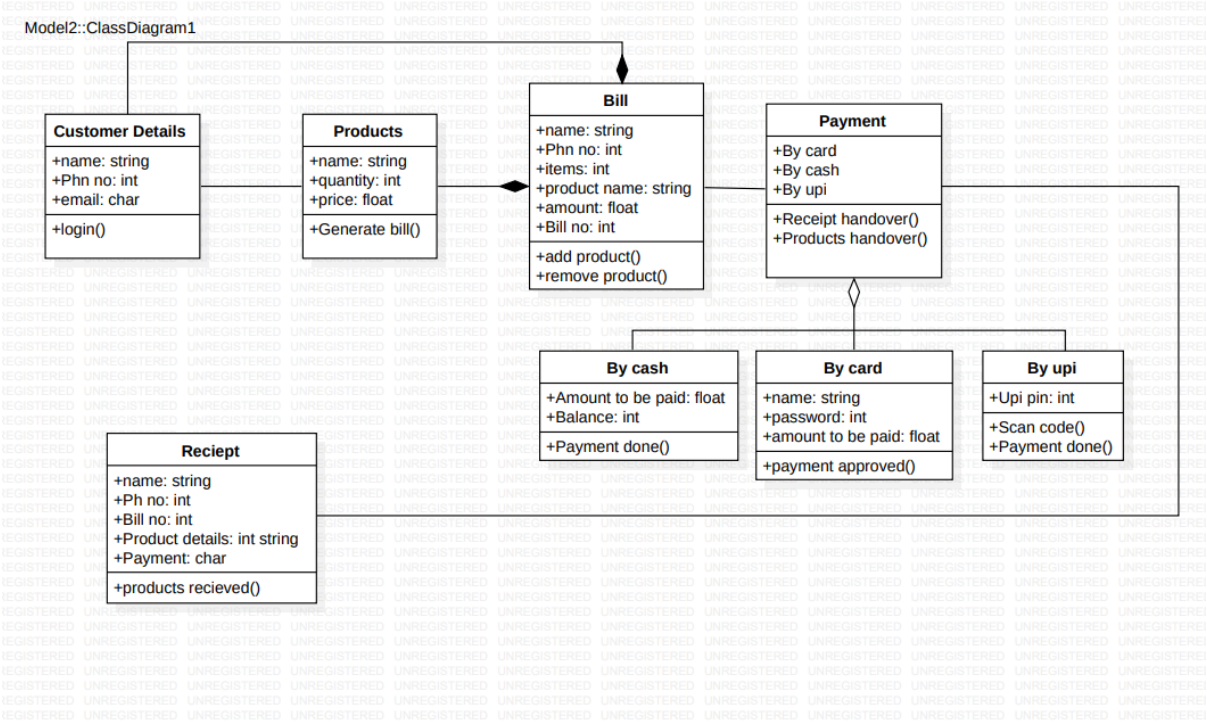
USECASE DIAGRAM

The main representation of system/software requirements for an unfinished new software programme is a UML use case diagram. Use cases describe the desired behaviour (what), not the precise means of achieving it.



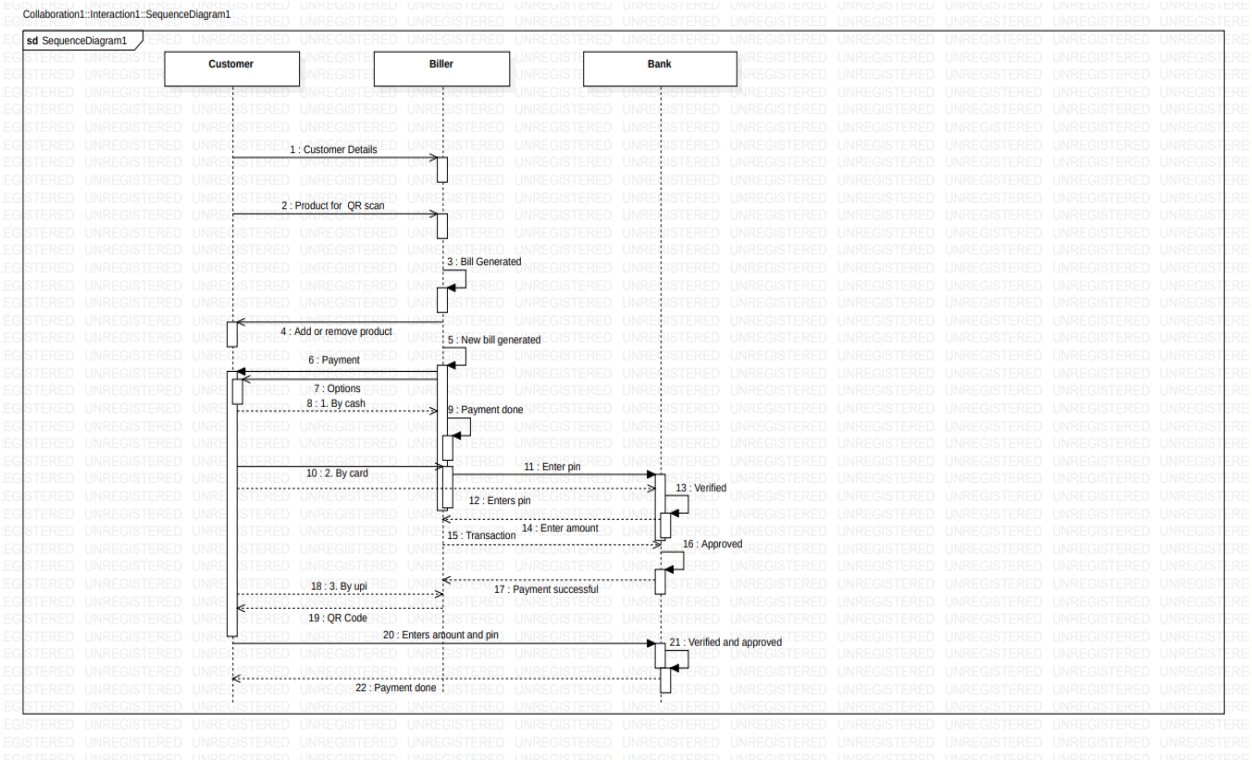
CLASS DIAGRAM

A class diagram is a specific kind of diagram and a component of the unified modelling language (UML), which specifies and presents the overview and structure of a system in terms of classes, characteristics, and methods, as well as the connections between various classes.



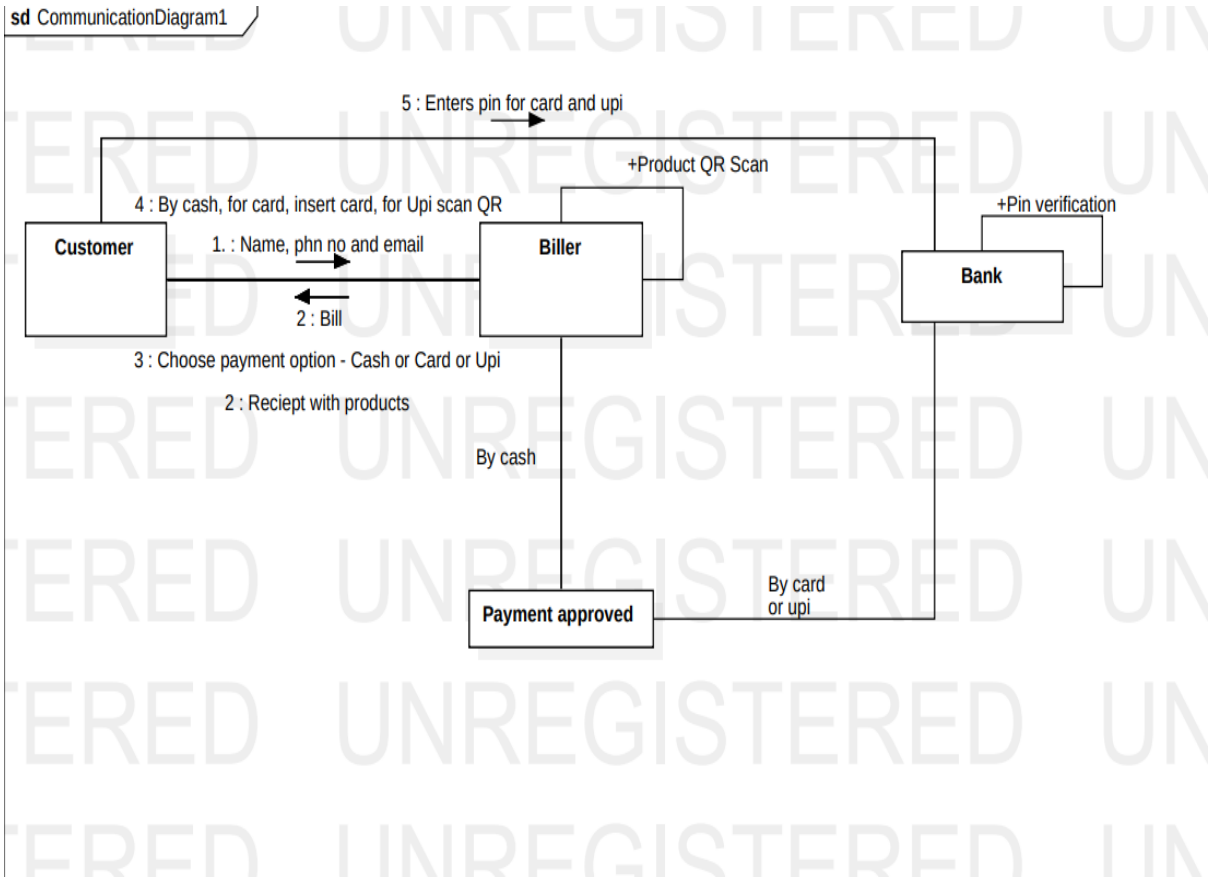
SEQUENCE DIAGRAM

Sequence Diagrams are interaction diagrams that describe the steps used to complete an operation. They depict how items interact within the framework of a cooperation. By using the vertical axis of the diagram to represent time and the messages that are transmitted and when, sequence diagrams, which have a time focus, can visually depict the order of an interaction.



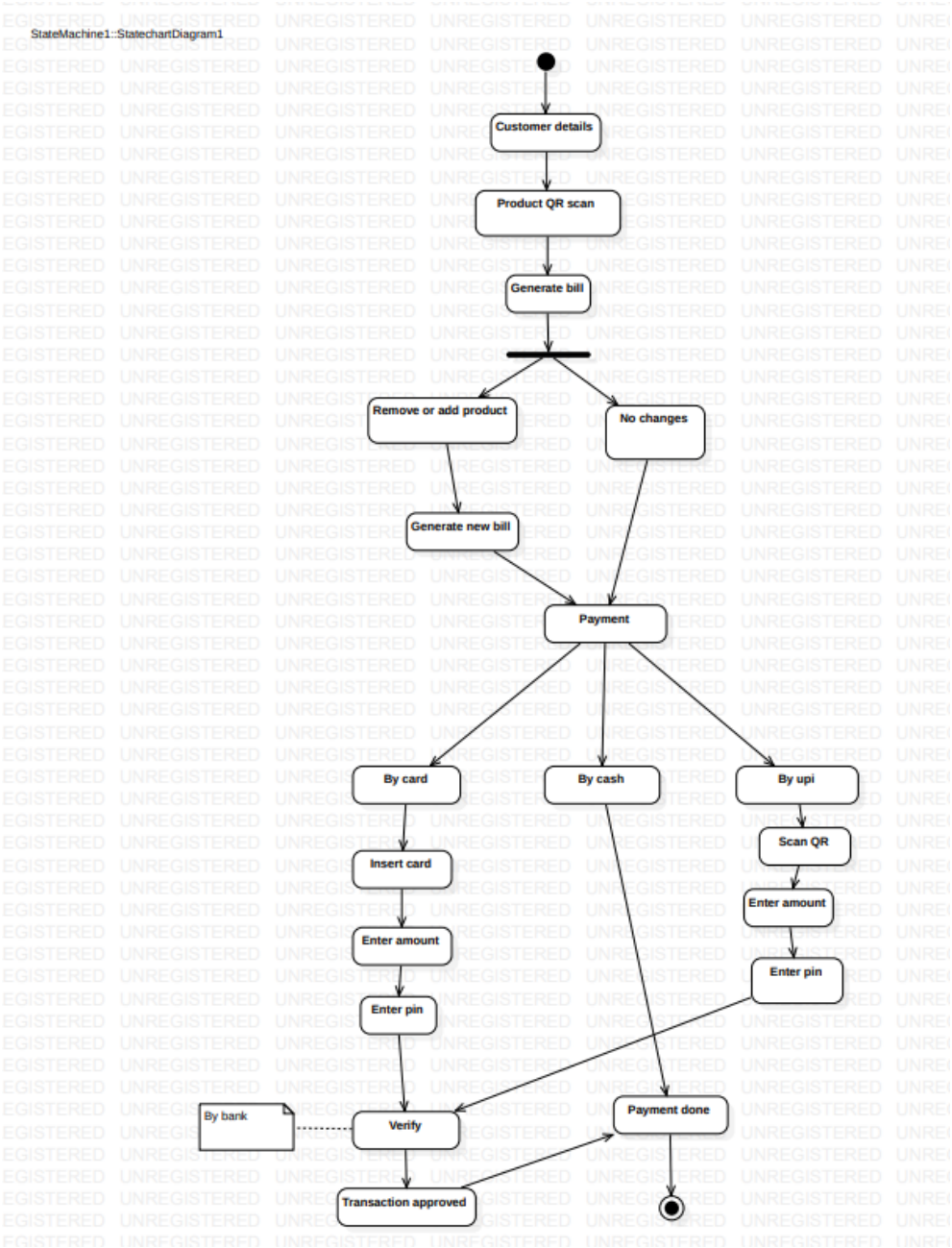
COMMUNICATION DIAGRAM

A type of behavioural diagram that depicts the interactions between objects in a piece of software or a system is the UML Communication Diagram, formerly known as the Collaboration Diagram. This kind of diagram places emphasis on the communications between items.



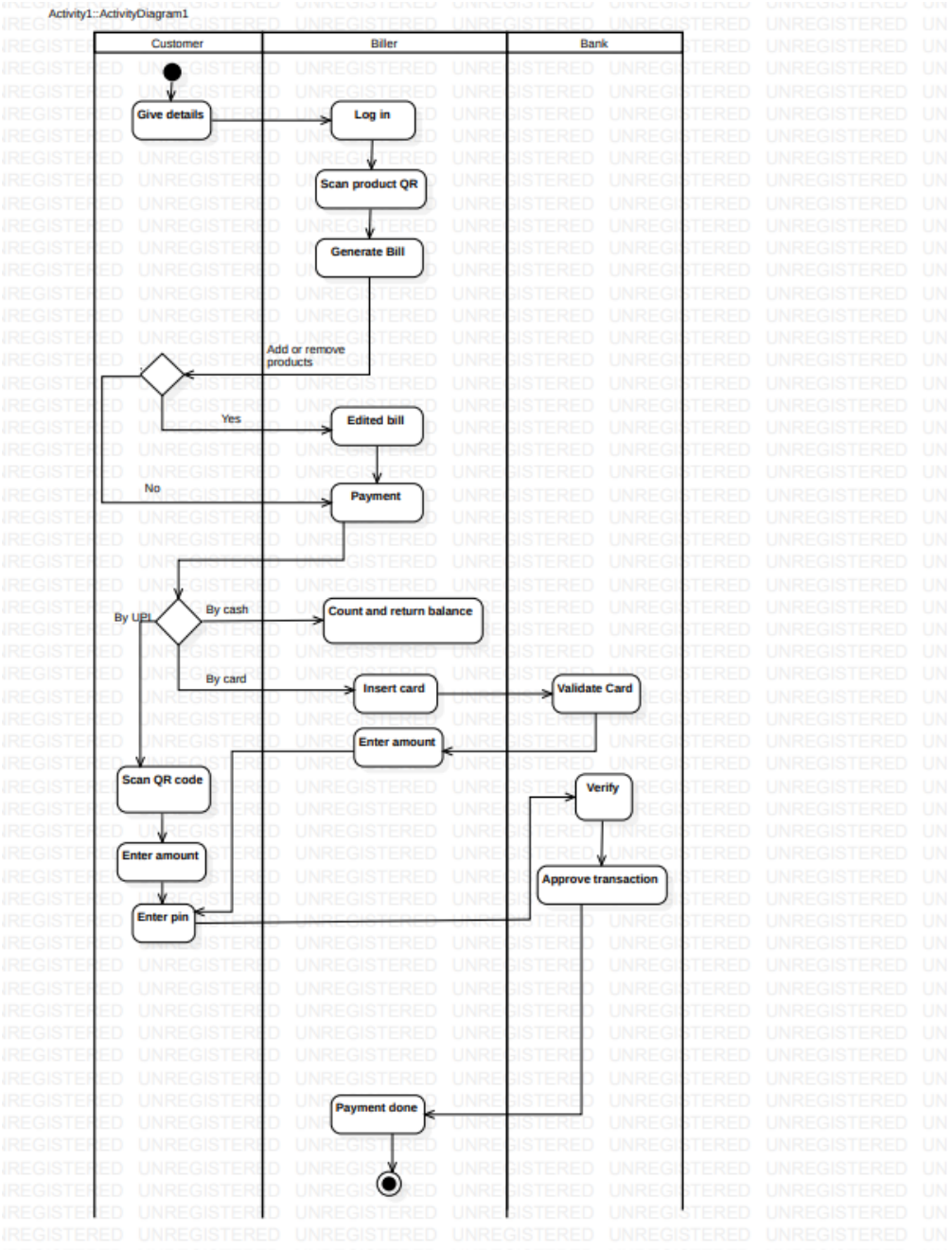
STATE CHART DIAGRAM

One of the five UML diagrams used to depict a system's dynamic nature is the state-chart diagram. Throughout an object's existence, they define several states, and these states are altered by events. The reactive systems can be modelled with state-chart diagrams. A system that reacts to internal or external events is known as a reactive system.



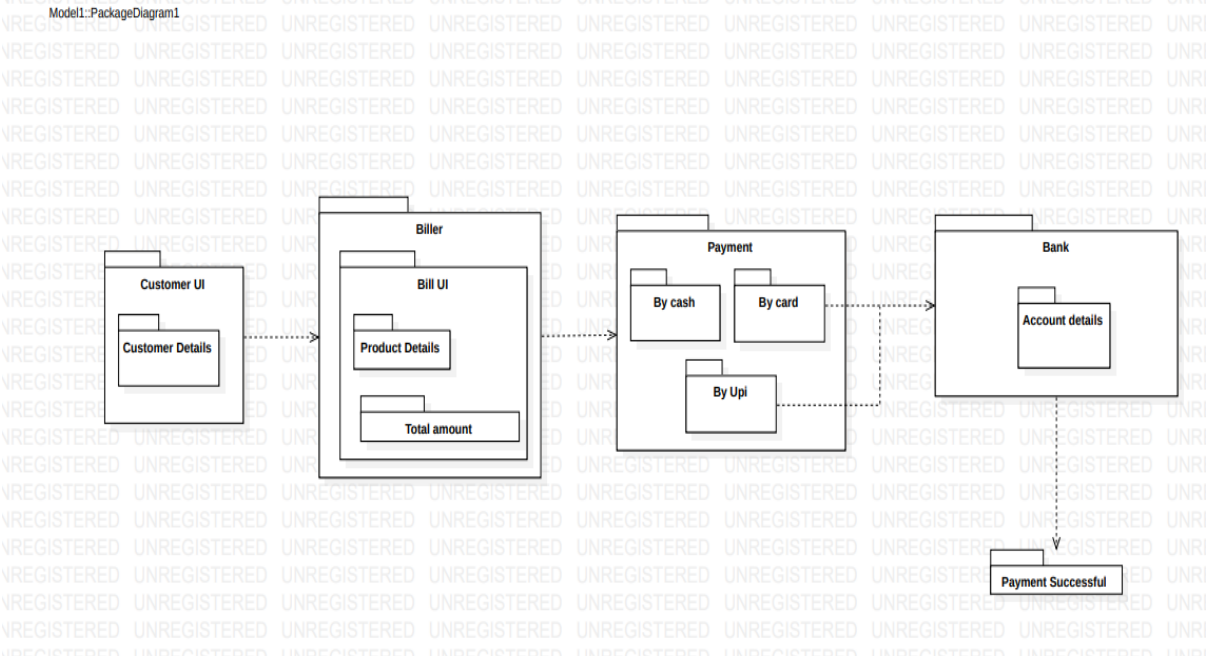
ACTIVITY DIAGRAM

Similar to a flowchart or data flow diagram, an activity diagram visually displays a series of actions or the flow of control in a system. A common tool in business process modelling is the activity diagram. Additionally, they can outline the procedures in a use case diagram.



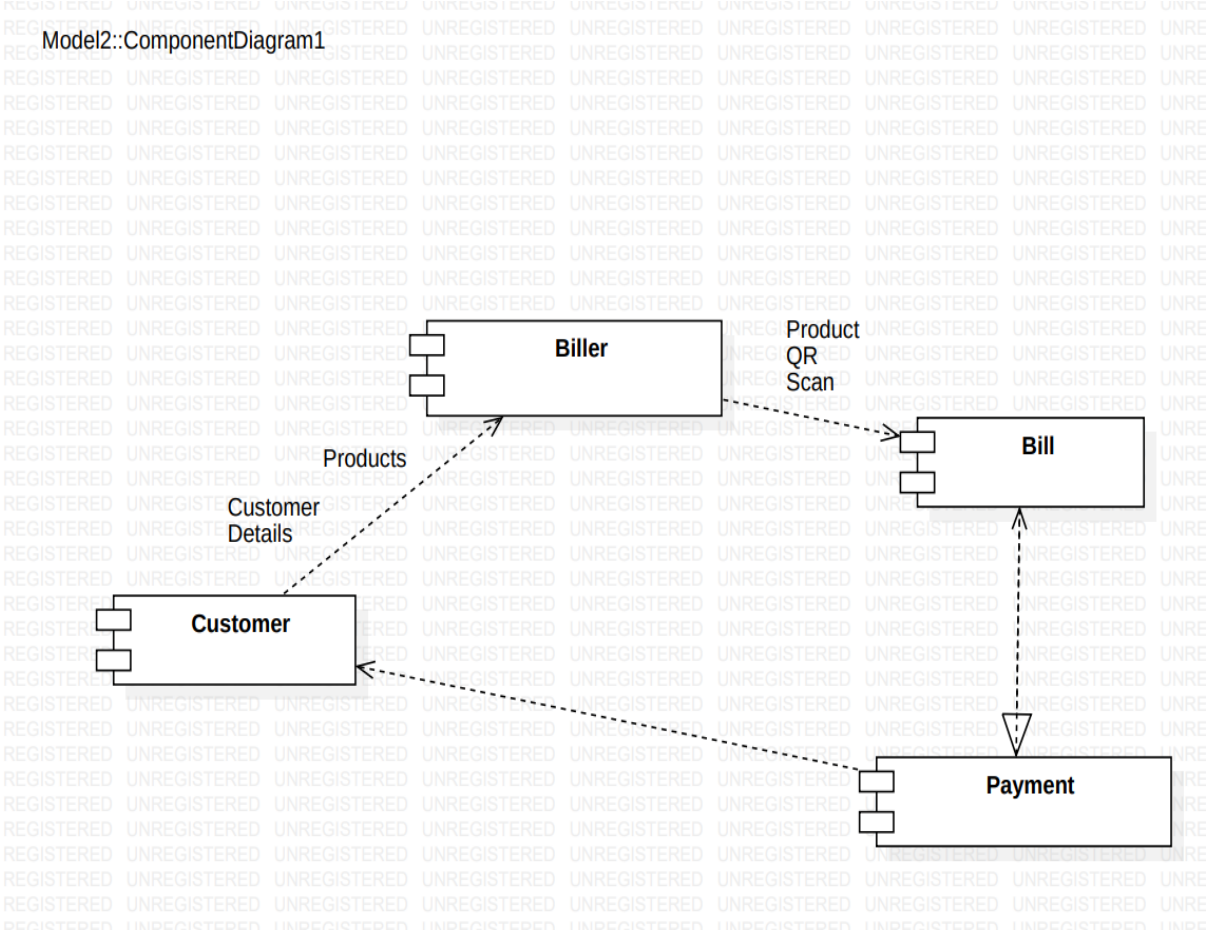
PACKAGE DIAGRAM

Package diagrams are structural diagrams that are frequently used to group classes into packages and simplify complex class diagrams. An assortment of related UML elements, such as diagrams, documents, classes, and event packages, make up a package.



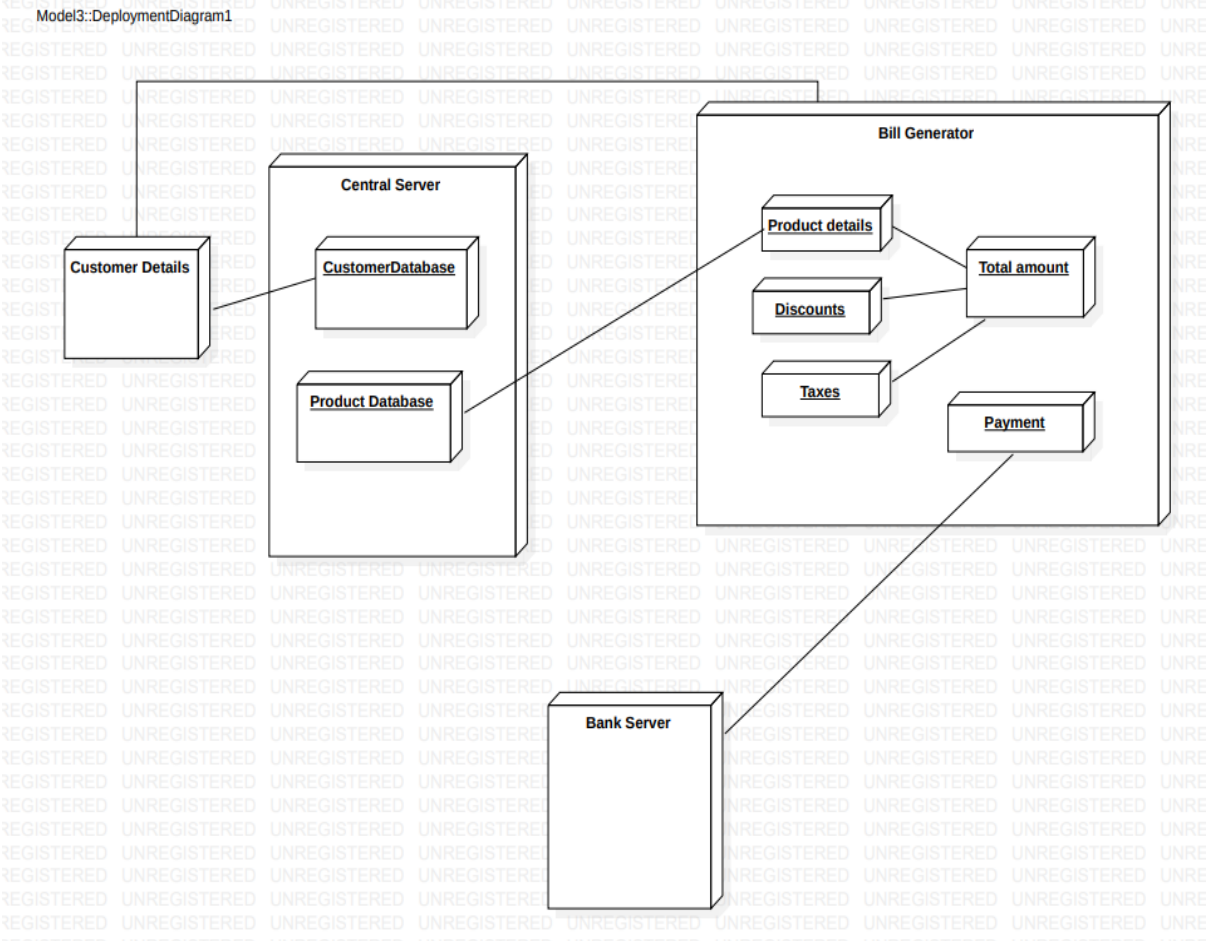
COMPONENT DIAGRAM

Component diagrams, which are basically class diagrams that concentrate on a system's components and are frequently used to depict the static implementation perspective of a system, are a subset of class diagrams.



DEPLOYMENT DIAGRAM

A deployment diagram is a visual representation of the configuration of the components located on run-time processing nodes. A type of structure diagram known as a deployment diagram is used to represent the physical components of an object-oriented system.



CONCLUSION

The different UML diagrams featured in this project serve as a basis for the implementation and realization of the supermarket billing system. The design of supermarket billing system is based on the key goal of simplifying the conventional billing system ensuring the accurate amount of the products.

REFERENCES

https://www.visual-paradigm.com
https://www.billingscript.com/supermarket-billing-system/
https://www.scribd.com/doc/125623529/supermarket-billing-system