



R.M.D. ENGINEERING COLLEGE

(An Autonomous Institution)

R.S.M Nagar, Kavaraipettai, Gummidipoondi Taluk, Thiruvallur District, Tamil Nadu- 601206

Affiliated to Anna University, Chennai / Approved by AICTE, New Delhi/Accredited by NAAC

An ISO 9001:2015 Certified Institution / All the Eligible UG Programs are accredited by NBA, New Delhi



DEPARTMENT OF INFORMATION TECHNOLOGY

21IT413 INTERNSHIP

POINT OF SALE TERMINAL FOR GROCERY STORE

111521203046 - SRIMATHI R

111521203050 - SWATHI V

111521203056 - VALAR SELVI M K

21IT401 SOFTWARE ENGINEERING LABORATORY**OBJECTIVES:**

- To understand the software engineering methodologies for project development.
- To gain knowledge about open source tools for Computer Aided Software Engineering.
- To develop an efficient software using case tools.

SOFTWARE REQUIRED:

Open source Tools: Star UML / UML Graph / Top cased

Prepare the following documents for each experiment and develop the software using software engineering methodology.

1. Problem Analysis and Project Planning -Thorough study of the problem – Identify Project scope, Objectives and Infrastructure.

2. Software Requirement Analysis - Describe the individual Phases/modules of the project and Identify deliverables.

3. Data Modelling - Use work products – data dictionary, use case diagrams and activity diagrams, build and test class diagrams, sequence diagrams and add interface to classdiagrams.

4. Software Development and Debugging – implement the design by coding

5. Software Testing - Prepare test plan, perform validation testing, coverage analysis, memory leaks, develop test case hierarchy, Site check and site monitor.

INDEX

SL.NO	NAME OF THE EXPERIMENT
	POINT OF SALE - TERMINAL FOR GROCERY STORE
1	Problem Analysis 1(a)Problem Statement 1(b)Project Planning
2	Software Requirement Analysis
3	Modeling 3(a)Design 3(b)Data Dictionary
4	Implementation
5	Testing - Test Cases
6	Documentation

Ex. No 1(a)**PROBLEM ANALYSIS****Problem Statement**

Point Of Sales software is the operating system you use to manage physical stores and sell in person. It's what store staff use to find products, add them to a customer's order, and accept payments. POS software also has helpful tools like sales reporting, inventory management, and integrated loyalty programs.

It needs to,

1. Generate Sales Receipt
2. Add item
3. Calculate Cost based on purchased quantity
4. Calculate GST on Final bit
5. Generation of Bill Receipt
6. The final bill has shop name w address in the header, and customer details w points earned in the footer.

Analysis

POS should allow the user to add new items and customers. The system has two categories of users: Administrator, Employee.

The administrator should be able to:

- Add new item details
- Modify item details
- Delete item
- Add new customer details
- Modify customer details
- Delete customer
- Search for item and customer details
- Generate reports, :
 1. Customer information
 2. Stock
 3. Cost
 4. GST
 5. Overall Sales Report

Employees should be able to:

- Search for an item
- Add an item
- Add customer information
- Stock
- Calculate Cost
- Calculate GST
- Shop name with address in bill
- Overall Sales Receipt.

Feasibility study

Technical feasibility

The Point Of Sale System (POS) runs with a minimum system resources:

- Python
- MySQL

Above said system resources are available as open source. Hence it is feasible to develop POS in this environment.

Operational feasibility

As the system has Python-Tkinter-based GUI no special skill set is required for working with the system, hence it is operationally feasible.

Economic feasibility

As the POS requires minimum system resources, hence it is economically feasible.

Ex. No 1(b)**PROJECT PLANNING****1. Overview**

Online Point Of Sale System is to handle the entire activity of a store. The system keeps track of all the information about the stocks and their complete details. The system contains a database where all the information will be stored. The system is user-friendly.

2. Goals and Scope

Goal : To automate the Point Of Sale System with the following functional goals

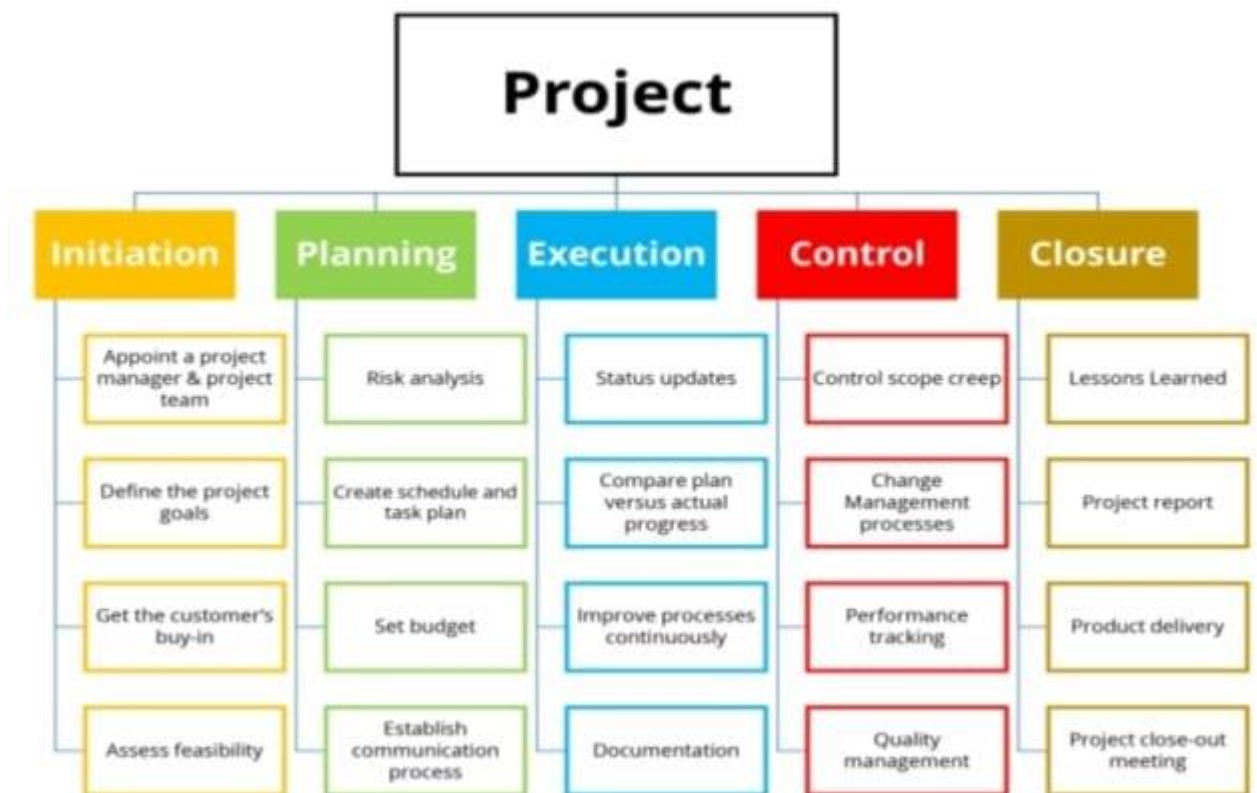
- i. Login
- ii. Adding Item/Customer Details.
- iii. Updating Item/Customer Details
- iv. Deleting Item/Customer Details
- v. Generating Reports
- vi. Viewing/Searching for Details.

Scope : The system accepts the General Sales Transactions of ordering item and return for the Customers. Different areas where we can use these applications are:

1. Any retail store can make use of it for managing their store.
2. The system would provide basic set of features to add/update Customer, add/update items, and manage check in specifications for the systems based on the client's statement of need.

3. Schedule and Budget

Work Breakdown Structure



Schedule and Milestones

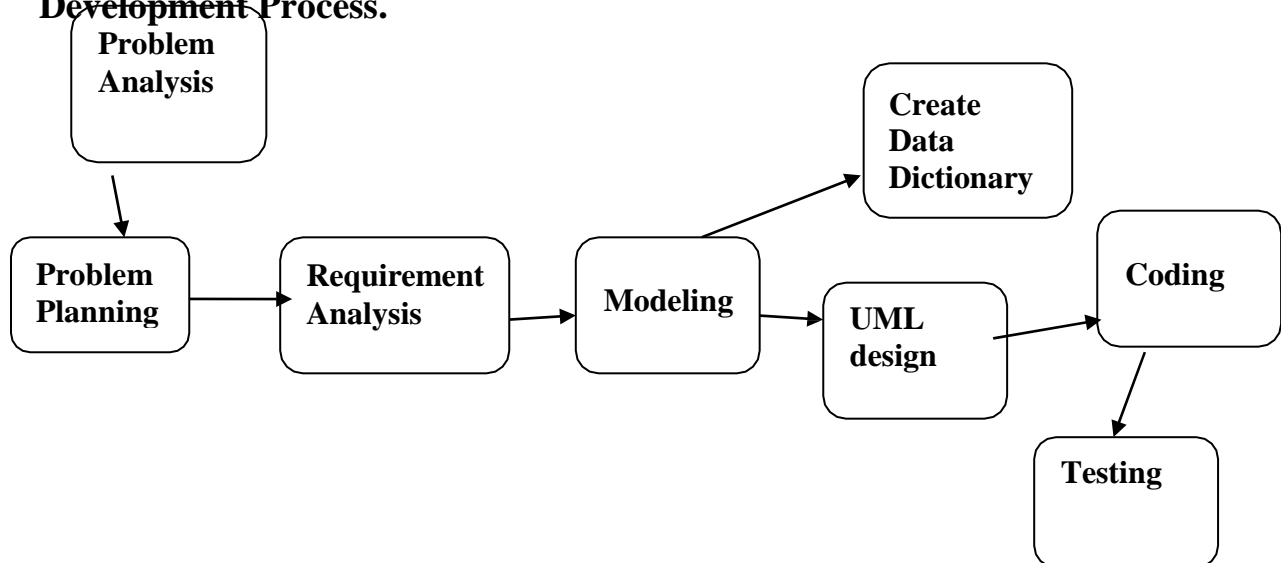
Milestones	Description	Milestone Criteria	Planned week
M0	Problem Analysis		1 st week
		Problem statement, Analysis, Feasibility Study	
M1	Project Planning		2 nd week
		Scope and concept described	
M2	Requirement Analysis		2 nd and 3 rd week
		Draft SRS, Design Specification, Test Plan, Requirement Analysis (Final)	
M3	Study of UML Notations		3 rd week
		Architecture reviewed and stable	
M4	Modeling		4 th week
		Software Design, Data Dictionary	

Milestones	Description	Milestone Criteria	Planned week
M5	Implementation		5 th week
		Coding of functionality, Debugging, System Test Plan.	
M6	Testing		6 th week
		Database & System Integration, Acceptance Testing	
M7	Documentation		7 th week
		User Manual	

Budget

Category	Budget for Period in kUS\$					
	M0-M1	M1-M2	M2-M3	M3-M4	M4-M5	M5-M6
Human Resources (internal)						
Human Resources (external)						
Purchases (COTS)						
Equipment						
Premises						
Tools						
Travel costs						
Training						
Review activities						
Other						
Total	1	1	2	5	2	1
Total cumulated	1	2	4	9	11	12

Development Process.



Risk Management

Unexpected Holidays, Non-availability of computer resources, and Absence of Human Resources are the identified risks for not meeting the deadlines. Additional efforts need to put in by the human resources to complete the work within the deadline by the way of working after working hours.

Delivery Plan

Ident.	Deliverable	Planned Date	Receiver
D1	Analysis and Feasibility Report	1 st week	Client
D2	Project Plan	2 nd week	Client
D3	SRS	3 rd week	Client
D4	Design	4 th week	Client
D5	Test Plan	5 th week	Client
D6	Code	6 th week	Client
D7	Test Report	6 th week	Client

Ex. No.2 SOFTWARE REQUIREMENT ANALYSIS**Software Requirement Specification (SRS)****1. Introduction**

Point Of Sales software for Grocery Store is the operating system you use to manage physical stores and sell in person. It's what store staff use to find products, add them to a customer's order, and accept payments. POS software also has helpful tools like sales reporting, inventory management, and integrated loyalty programs. It needs to maintain the record of items and retrieve the details of items available in the store. Should be able to perform basic operations in a store like sales receipt, add a item, calculate cost based on purchased quantity calculate GST on final bit, generation of bill receipt and payment and the final bill has shop name w address in the header, and customer details w points earned in the footer. The report generation facility should allow generating various reports.

This project is specifically designed for the use of staffs and admins in retail store which allows them to accept payments from customers and keep track of sales.

1.2 Purpose of the requirements document

The purpose of this document is to analyze and elaborate on the high-level needs and features of the POS System. It focuses on the capabilities and facilities provided by a POS system. The main objective of this document is to illustrate the requirements of the Point Of Sale System. This document gives the detailed description of both functional and non-functional requirements proposed by the clients.

This document defines and describes the operations, interfaces, performance, and quality assurance requirements of the POS System. The document also describes the nonfunctional requirements such as the user interfaces. It also describes the design constraints that are to be considered when the system is to be designed, and other factors necessary to provide a complete and comprehensive description of the requirements for the software.

1.3 Scope of the product

The Software Requirements Specification captures all the requirements in a single document. The Point Of Sale System that is to be developed provides the Administrator and Employees, the product information, and many other facilities.

The system accepts the payments and transactions. The system would provide a basic set of features to add/update product details, add/update customer details, and manage product specifications based on the customer statement of need.

1.4 Definitions, acronyms and abbreviations

SQL- Structured Query Language

GUI- Graphical User Interface

POS – Point Of Sale System

1.5 References

- (i) Roger s Pressman ,”Software Engineering -a practitioner's approach” 7th edition
- (ii) Ian sommerville , “software engineering “ 9th edition

1.6. Overview of the remainder of the document

The SRS will provide a detailed description of the Point Of Sale System. This document will provide the outline of the requirements, and overview of the characteristics and constraints of the system.

Section 2 of this document provides the General description such as Product perspective, Product functions and the characteristics of the user's of this product. Section 3 describes the Specific requirements which cover the functional, non-functional and interface requirements. This is obviously the most substantial part of the document but because of the wide variability in organizational practice, it is not appropriate to define a standard structure for this section. The requirements may document external interfaces, describe system functionality and performance, specify logical database requirements, design constraints, emergent system properties, and quality characteristics.

2. General description

Product perspective

The Point Of Sale System is a package to be used by retailers to improve the efficiency of Administrators and Employees. The system provides a product catalog and information about the quantity available and helps the customer decide on buying the product. The Admin/Employee can keep the product catalog updated all the time so that the customers get the updated information all the time.

Product functions

The Point Of Sale System provides real-time information about the products available in the Store and customer information. The Product functions are more or less the same as described in the product perspective. The functions of the system include the system providing different types of services based on the type of users [Administrator/Employee].

1. Administrator should be able to add new Item/Customer details, Modify Item/Customer details, Delete Item/Customer details, Search for item and customer details, and Generate various reports such as
 - Customer information
 - Stock
 - Cost
 - GST
 - Overall Sales Receipt
2. Employees should be able to Search for a product and view the details of the product for sale.
3. Customer should be able to Search for a product and add it to cart.

User characteristics

The users of the system are administrators and employees, The administrator maintains the system. The members are assumed to have basic knowledge of computers. The administrators of the system to have more knowledge of the internals of the system and is able to rectify the small problems that may arise due to disk crashes, power failures, and other catastrophes to maintain the system. The proper user interface, user's manual, online help, and guide to installing and maintaining the system must be sufficient to educate the users on how to use the system without any problems.

General constraints

- The information of all the items/products for selling must be stored in a database that is accessible by the Point Of Sale System.
- The Point Of Sale System is connected to the server computer and is running all 24 hours a day.

Assumptions and dependencies

- The users have sufficient knowledge of computers.
- The users know the English language, as the user interface will be provided in English
- The product can access the customer database

3. Specifications

Functional Requirements

This section describes in detail all the functional requirements.

(It shows what the system can do)

1. Adding Item/Customer Details.
2. Updating Item/Customer Details
3. Deleting Item/Customer Details
4. Generating Reports
5. Viewing/Searching for Details.

Non- Functional Requirements

Usability

- The system is user-friendly and self-explanatory.

Reliability

The system has to be very reliable due to the importance of data and the damages incorrect or incomplete data can do.

Availability

The system is available 100% for the user and is used 24 hrs a day and 365 days a year.

The system shall be operational 24 hours a day and 7 days a week.

Mean Time Between Failures (MTBF)

The system will be developed in such a way that it *may* fail once in a year.

Mean Time to Repair (MTTR)

Even if the system fails, the system will be recovered back up within an hour or less.

Accuracy

The accuracy of the system is limited by the accuracy of the speed at which the employees of the store and admin of the store use the system.

Maximum Bugs or Defect Rate

Not specified.

Access Reliability

The system shall provide 100% access reliability.

Performance**Response Time**

The system shall respond to the user in not less than two seconds from the time of the request submittal. The system shall be allowed to take more time when doing large processing jobs.

The requirements may document external interfaces, describe system functionality and performance, specify logical database requirements, design constraints, emergent system properties and quality characteristics.

3.3 Hardware and software requirements

3.3.1 Hardware Interfaces

- ☐ Processor: Pentium or Higher.
- ☐ RAM: 312MB or Higher.

3.3.2. Software Interfaces

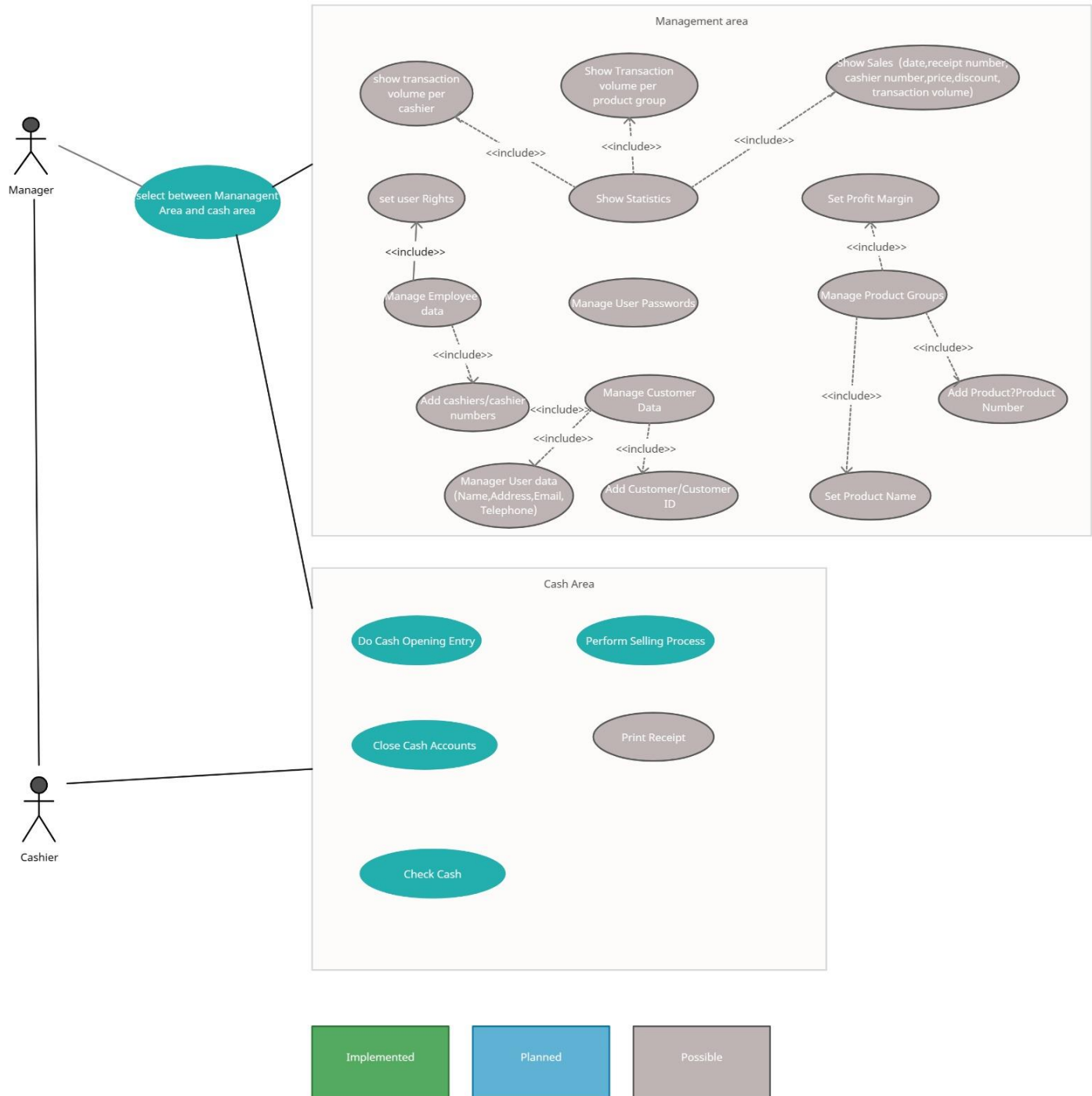
- ☐ Operating System: Unix, Linux, Mac, Windows etc..
- ☐ Development tool: Python: Tkinter.
- ☐ Data Base: MySQL
- ☐

3.4 External Interfaces

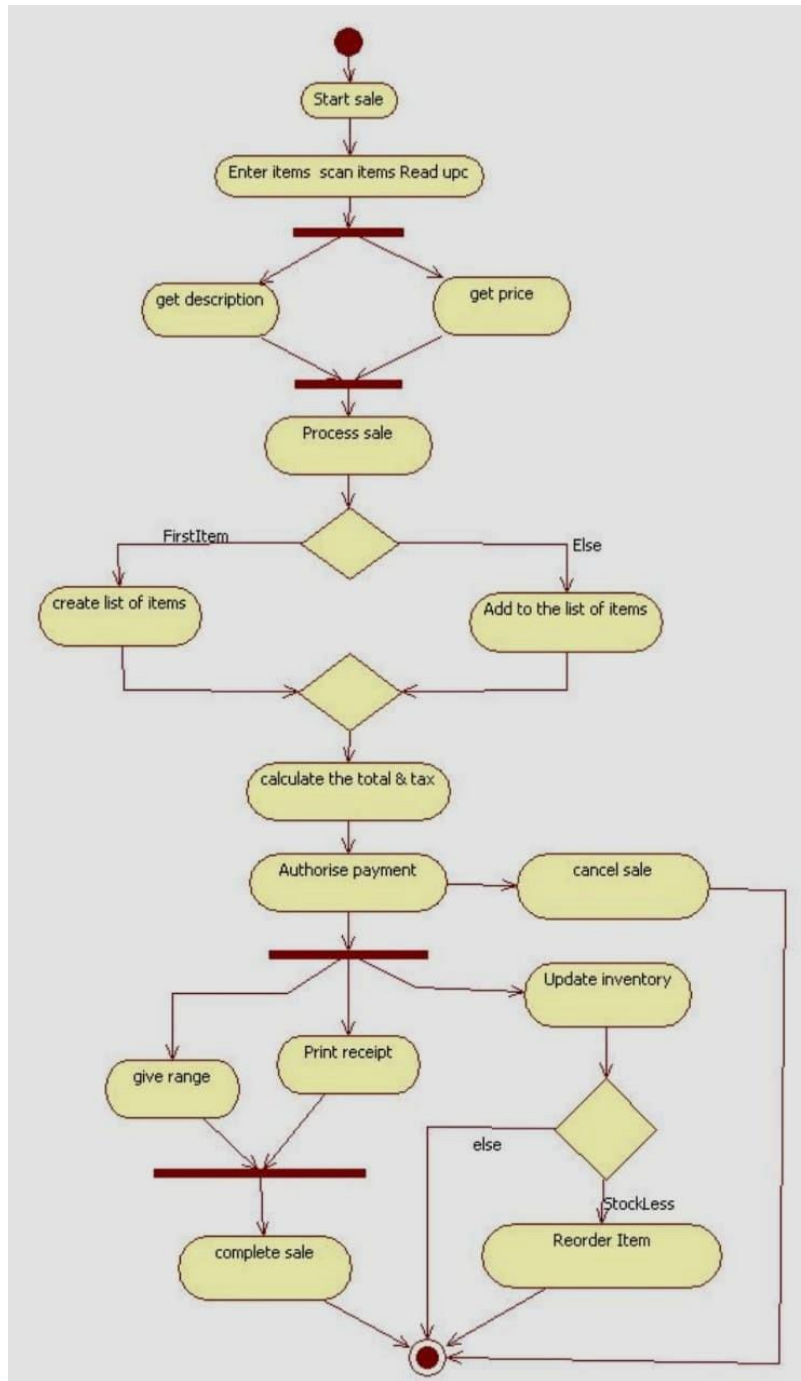
User Interfaces

The user-interface of the system shall be designed as shown in the user-interface prototypes.

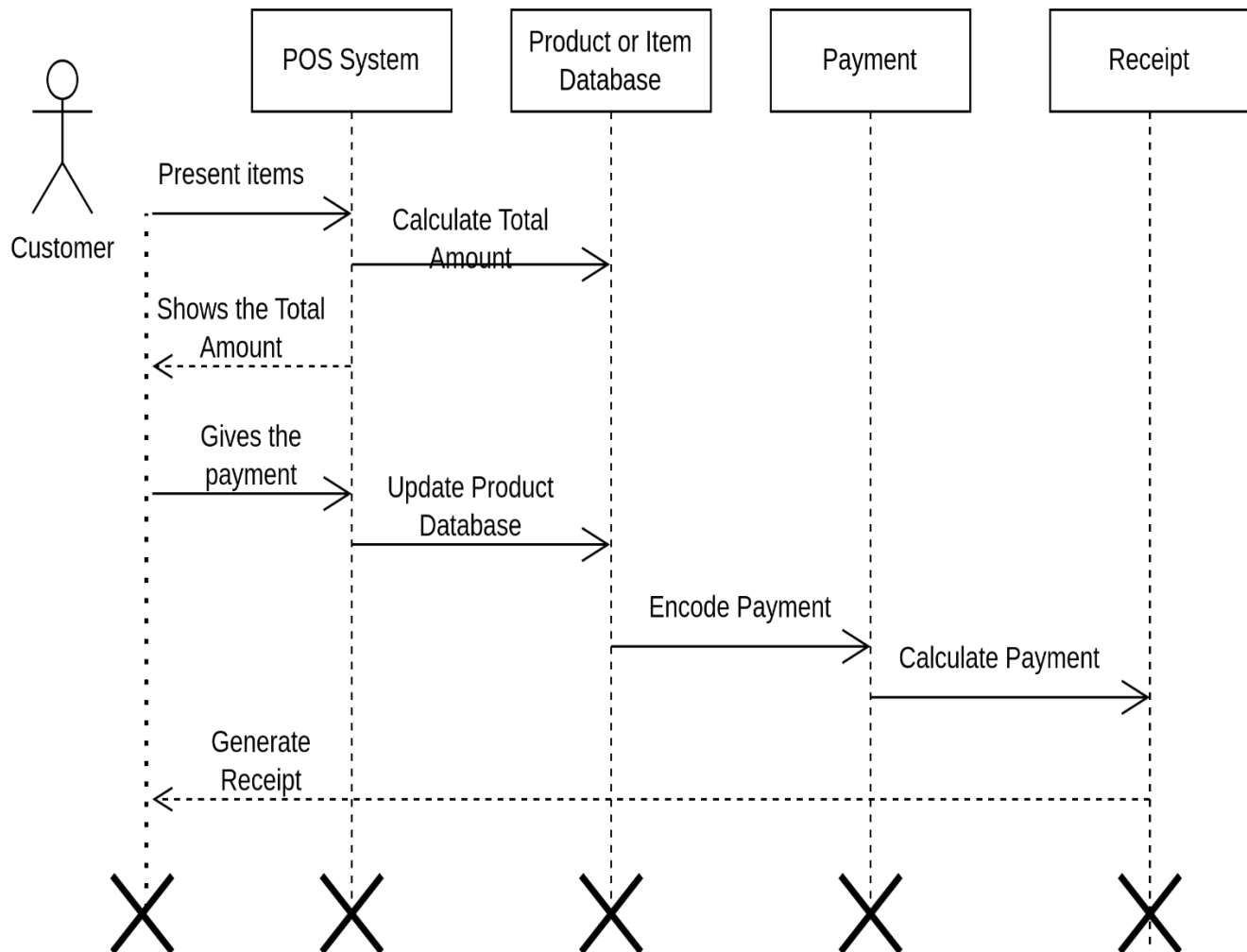
Result: Thus the Software Requirement Specification Document for Point Of Sale System has been completed

Ex.No. 3**MODELING****(i) Design model –UML diagrams****Use case diagram****Step 1:start draw.io->Create-> Use Case Diagram**

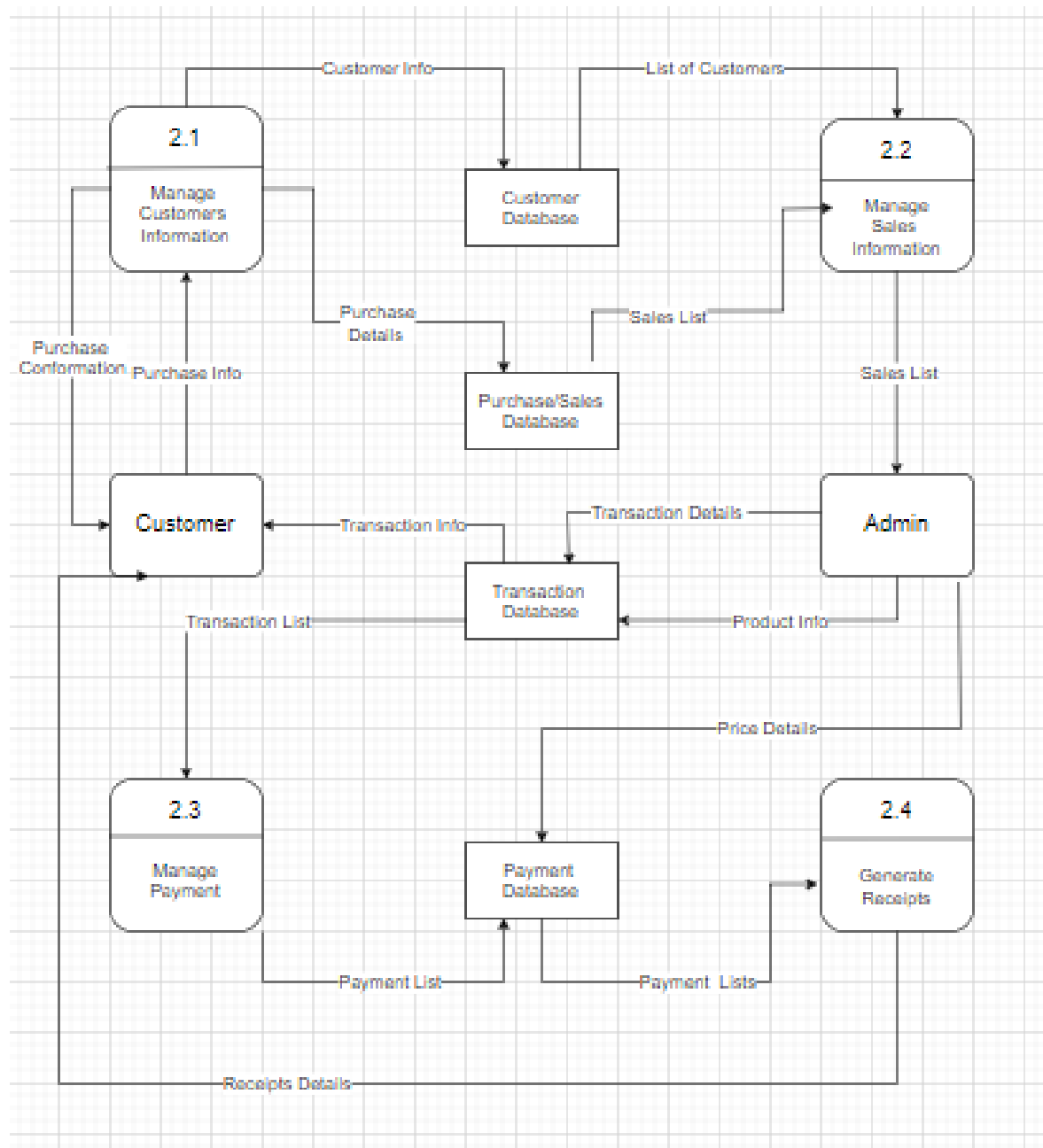
Activity Diagram for POS



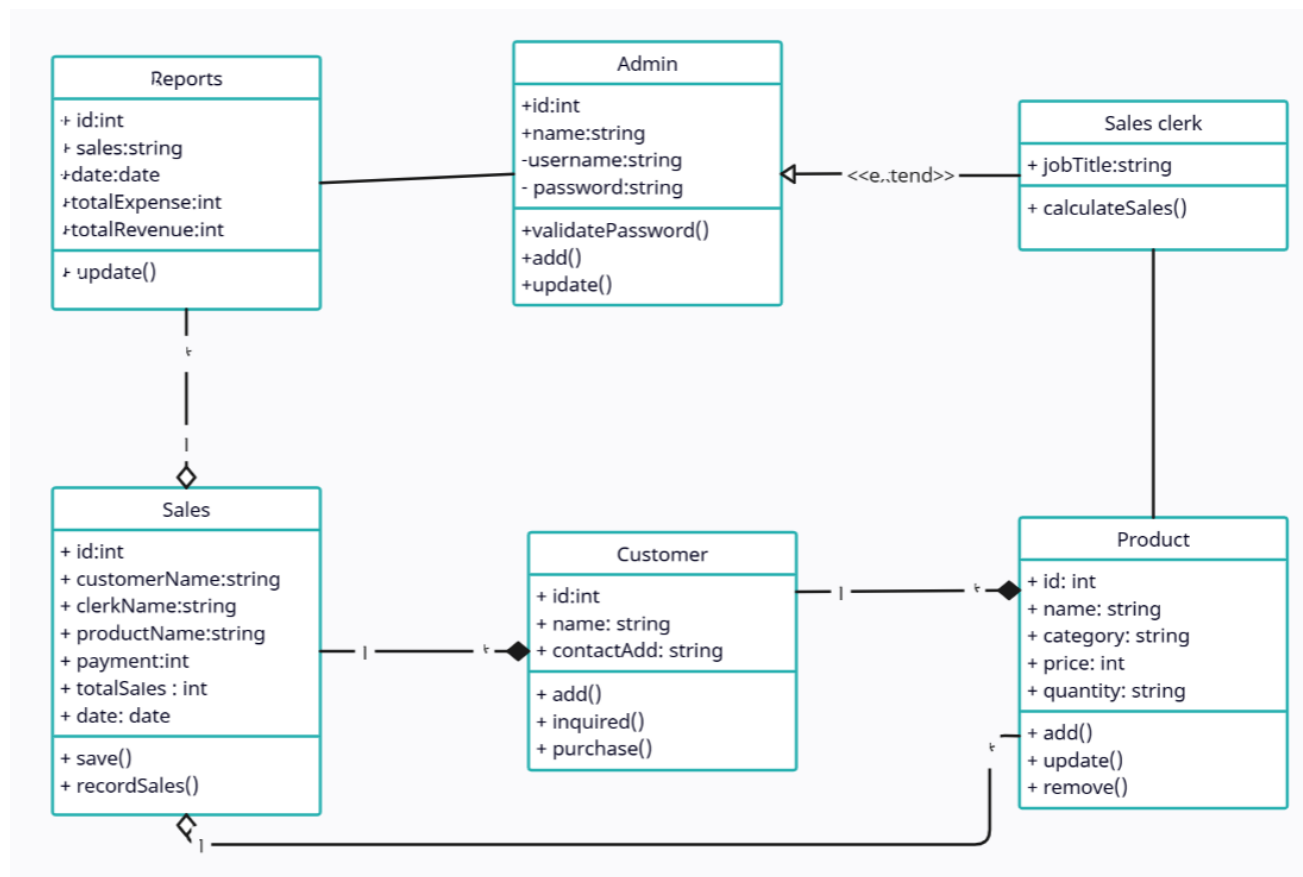
Sequence Diagram for POS



Data Flow Diagram for Point Of Sale Terminal



Class Diagram for POS



Ex.No.3 (b) DATA DICTIONARY**Product Details**

S. No	Name	Alias Name	Where Used/How Used	Supplementary Data	
				Data Type	Limitations
1	Product name	-	Purchase, Generate display	string	Up to 20 char
2	Product ID	-	Purchase, Generate display.	integer	Up to 2 digits
3	Customer Name	-	Purchase, Display	string	Up to 20 char
4	Phone Number	-	Purchase, Display	integer	Up to 10 digits
5	Price	-	Purchase, Display	integer	Up to 16 digit

User Details

S. No	Name	Alias Name	Where Used/How Used	Supplementary Data	
				Data Type	Limitations
1	Customer name	-	Membership, Purchase	string	Up to 20 char
2	Customer id	-	Membership	string	Up to 2 digits
3	Address	-	Membership	string	Up to 20 char
4	Phone Number	-	Membership	string	Up to 10 digits
5	Credit card no	-	Membership	integer	Up to 16 digit

Ex.No.4 **IMPLEMENTATION****Main.py**

```
_author_ = "macaw"  
import os  
from tkinter import *  
from tkinter import message box  
  
main = Tk()  
main.geometry("1366x768")  
main.title("Big Bazaar")  
main.resizable(0, 0)  
def Exit():  
    sure = messagebox.askyesno("Exit","Are you sure you want to exit?", parent=main)  
    if sure == True:  
        main.destroy()  
  
main.protocol("WM_DELETE_WINDOW", Exit)  
  
def emp():  
    main.withdraw()  
    os.system("python employee.py")  
    main.deiconify()
```

```
def adm():  
    main.withdraw()  
    os.system('python admin.py')  
    main.deiconify()  
  
label1 = Label(main)  
label1.place(relx=0, rely=0, width=1366, height=768)  
  
img = PhotoImage(file='./images/main.png')  
label1.configure(image=img)  
  
button1 = Button(main)  
button1.place(relx=0.316, rely=0.446, width=146, height=90)  
button1.configure(relief='flat')  
button1.configure(overrelief='flat')  
button1.configure(activebackground='#ffffff')  
button1.configure(cursor='hand2')  
button1.configure(foreground='#ffffff')  
button1.configure(background='#ffffff')  
button1.configure(borderwidth='0')  
img2 = PhotoImage(file='./images/1.png')  
button1.configure(image=img2)  
button1.configure(command=emp)  
  
button2 = Button(main)  
button2.place(relx=0.566, rely=0.448, width=146, height=90)  
button2.configure(relief='flat')
```

21IT401 SOFTWARE ENGINEERING LABORATORY

```
button2.configure(overrelief="flat")
button2.configure(activebackground="#ffffff")
button2.configure(cursor="hand2")
button2.configure(foreground="#ffffff")
button2.configure(background="#ffffff")
button2.configure(borderwidth="0")
img3 = PhotoImage(file="./images/2.png")
button2.configure(image=img3)
button2.configure(command=adm)

main.mainloop()
```

Admin.py

```
class login_page:
    def __init__(self, top=None):
        top.geometry("1366x768")
        top.resizable(0, 0)
        top.title("Retail Manager(ADMIN)")

        self.label1 = Label(root)
        self.label1.place(relx=0, rely=0, width=1366, height=768)
        self.img = PhotoImage(file="./images/admin_login.png")
        self.label1.configure(image=self.img)
```

```
self.entry1 = Entry(root)
self.entry1.place(relx=0.373, rely=0.273, width=374, height=24)
self.entry1.configure(font="-family {Poppins} -size 10")
self.entry1.configure(relief="flat")
self.entry1.configure(textvariable=user)

self.entry2 = Entry(root)
self.entry2.place(relx=0.373, rely=0.384, width=374, height=24)
self.entry2.configure(font="-family {Poppins} -size 10")
self.entry2.configure(relief="flat")
self.entry2.configure(show="*")
self.entry2.configure(textvariable=passwd)

self.button1 = Button(root)
self.button1.place(relx=0.366, rely=0.685, width=356, height=43)
self.button1.configure(relief="flat")
self.button1.configure(overrelief="flat")
self.button1.configure(activebackground="#D2463E")
self.button1.configure(cursor="hand2")
self.button1.configure(foreground="#ffffff")
self.button1.configure(background="#D2463E")
self.button1.configure(font="-family {Poppins SemiBold} -size 20")
self.button1.configure(borderwidth="0")
self.button1.configure(text="LOGIN")
self.button1.configure(command=self.login)
```

```
def login(self, Event=None):  
    username = user.get()  
    password = passwd.get()  
  
    with sqlite3.connect("./Database/store.db") as db:  
        cur = db.cursor()  
    find_user = "SELECT * FROM employee WHERE emp_id = ? and password = ?"  
    cur.execute(find_user, [username, password])  
    results = cur.fetchall()  
    if results:  
        if results[0][6]=="Admin":  
            messagebox.showinfo("Login Page", "The login is successful.")  
            page1.entry1.delete(0, END)  
            page1.entry2.delete(0, END)  
  
            root.withdraw()  
            global adm  
            global page2  
            adm = Toplevel()  
            page2 = Admin_Page(adm)  
            #page2.time()  
            adm.protocol("WM_DELETE_WINDOW", exitt)  
            adm.mainloop()  
    else:  
        messagebox.showerror("Oops!!!", "You are not an admin.")
```

else:

messagebox.showerror("Error", "Incorrect username or password.")

page1.entry2.delete(0, END)

def exitt():

sure = messagebox.askyesno("Exit", "Are you sure you want to exit?", parent=root)

if sure == True:

adm.destroy()

root.destroy()

def inventory():

adm.withdraw()

global inv

global page3

inv = Toplevel()

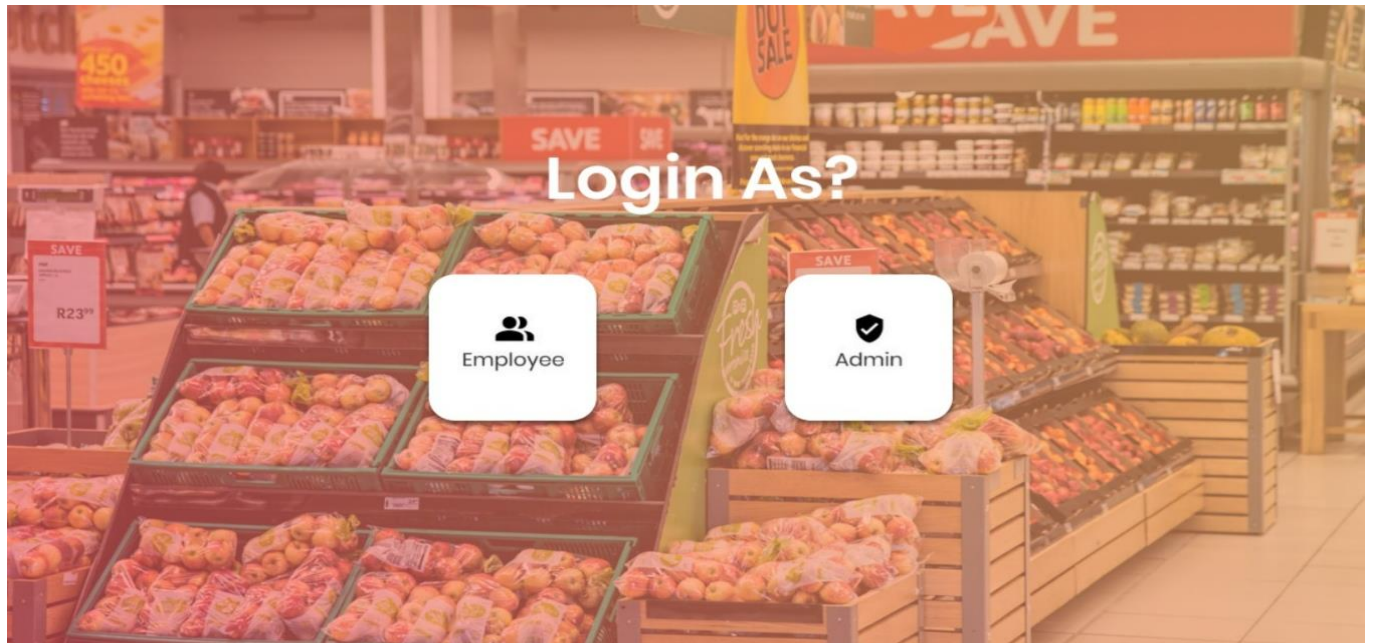
page3 = Inventory(inv)

page3.time()

inv.protocol("WM_DELETE_WINDOW", exitt)

inv.mainloop()

OUTPUT

A login form overlay on a gradient background transitioning from red on the left to orange on the right. The form is a white rounded rectangle with the title "Login" in bold black text. Below the title are two input fields: "Username" with a user icon and "Password" with a lock icon. At the bottom of the form is a red rounded rectangular button with the text "LOGIN" in white.

EMP0000

Logout

Billing System

23:20:48 PM

Customer Details

Bill Number

Search

Customer Name

Contact Number

Products

Select Category

Sub Category

Product

Quantity

Add To Cart

Remove

Clear

Bill Window

BIG BAZAAR (FUTURE RETAIL LTD.)

BB-THANE-DAHISAR THAKUR MALL

THANE - 401104

TEL NO: 022-28284707

HELPLINE: 1800-200-255

GSTIN: 35AABCS1429B1ZX we.f 01.07.2017

Customer Name:

Bill Number:

Phone Number:

Date:

Product Name	Quantity(Qty)	Price
--------------	---------------	-------

Bill Options

Total

Generate

Clear

Exit

EMP0000

Logout

Billing System

23:21:49 PM

Customer Details

Bill Number

Search

Customer Name swetha

Contact Number 9342653440

Products

Select Category

Pantry

Sub Category

Grain

Product

Kashi Company GoLean Crunch! Honey Almond Flax Cereal

Quantity

23

In Stock: 241

Add To Cart

Remove

Clear

Bill Window

BIG BAZAAR (FUTURE RETAIL LTD.)

BB-THANE-DAHISAR THAKUR MALL

THANE - 401104

TEL NO: 022-28284707

HELPLINE: 1800-200-255

GSTIN: 35AABCS1429B1ZX we.f 01.07.2017

Customer Name: swetha

Bill Number: BBU4U4DA

Phone Number: 9342653440

Date: 2023-04-30

Product Name	Quantity(Qty)	Price
Kashi Company GoLean Crunch! Honey Almond Flax Cereal	23	1495.0
Total		Rs. 1495.0

Bill Options

Total

Generate

Clear

Exit

Ex. No 5**TESTING****Test cases:**

Name	Requirement	Description	Input	Expected o/p	Actual o/p
Login	Username & password should be entered	Used to verify if actor is the authorized person or not	User name, password	Valid msg	Valid msg
Login	Username & password should be entered	Used to verify if actor is the authorized person or not	User name, password	Invalid msg	Invalid msg
Modify product	Product details	Store the product details	Product id, Product _name, price, quantity	Valid msg	Valid msg
Customer details	Customer details should be entered	Store the customer details	Customer name ,customer id	Valid msg	Valid msg
Availability	Check if the required no of product are available	Used to sell product	Product id, Product name	Valid msg	Valid msg
Availability	Check if the required no of product are available	Used to sell product	Product id, Product name	Invalid msg	Invalid msg
Generate bill	Check for product cost and tax.	Calculate the total cost of the purchase	Customer name, customer phone number, product id, product name	Valid msg	Valid msg
Search	All information stored in database	View product details	Product name ,product id,price, Quantity available.	Valid msg	Valid msg