RETAIL MANAGEMENT SYSTEM

Gökhan Göksel Elpeze

Murat Taylan Şahin

Berhan Türkü Ay

Oktay Türkdağlı

CYG DEVELOPMENT

1. Introduction

1.1 What the Problem is

The need for a program with a colorful interface that can process data such as stock tracking, product sales, statistics and personnel information in a company with retail product sales, and which can yield meaningful output as a result.

1.2 Goals for the Project

Using RMS, you can attract more customers and attend to existing customers in real-time, manage different branches of your shop, analyze your sale data and generate meaningful reports about your company's future. In addition to this you can check your employees' performance from this program.

1.3 Motivation for the Project

Increasing productivity and speeding processes up has a direct effect on customer satisfaction in retailers. Technology plays an important role in terms of more comfortable and intuitive customer experience. Retail Management System (RMS) we develop keeps this principle in mind and lets you manage these processes in real-time. Thanks to this, it is possible to speed up processes easily ranging from receipts to products, inventory management to price management, purchases to promotions, orders to transfers. RMS is designed with needs of retailers selling a wide range of different products in mind.

1.4 Process Flow Preview

We think that it will be useful to examine the projects that have been done before and to contact the users of those projects before making this program. We also want the project to have a modular structure and to add and remove features when necessary.

For these reasons, we want to build a model based on MVP requirements. Since the project has time limitations, we plan to focus on the coding and design part of the test process by keeping it limited.

2. Analysis and Design

2.1 Plan for Requirements Engineering

Starting step for the project is identifying the needs of users. Fisrtly we want to determine what features are required for a program that will cover all the needs of our users.

Second task is deepening our understanding of the problems and some possible solutions. In this task we will draft a rough draft of the features that will be implemented. After this stage we will the project using our understanding from previous stages.

Next stage is to discuss the feature list created previously in more detail in order to prioritize them. Critical features that will be necessary in first prototypes of the project will have higher priority while less-critical features will be placed further back or given up completely.

Lastly, after getting a complete understanding of all the features, we will draw the necessary diagrams to help design the program completely.

2.2 Functional Requirements

The program will be designed to be used in a Windows desktop environment. It is recommended to use it with a keyboard, mouse and a printer for receipts. Barcode scanner support can later be implemented if there is a need.

Primary Tasks

Sell Products

- Add multiple items to cart
- Enter customer payment details
- Generate and save a digital receipt

Return products

- Search previously saved receipt
- Make the return and generate a return receipt

Order products

- Order a given quantity of product from manufacturer
- Make the transfer to the requested shop

Transfer products

- Transfer a given quantity of products between two shops
- Make required changes to the shops' inventories

Manage products

- Add new products to the system
- Edit previously added products

- Remove products from the system

Manage employees

- Add new employees to the system
- Edit previously added employees
- Remove employees from the system

Manage shops

- Add new shops to the system
- Edit previously added shops
- Remove shops from system

Manage manufacturers

- Add new manufacturers to the system
- Edit previously added manufacturers
- Remove employees from the manufacturers

Generate and view reports

- Automatically generate meaningful reports about company sales
- View reports if the user has required authority level

Save data

- Save program information to various files
- Retrieve information of previous sessions from files

Login

- Login for users with various permission levels
- Users with lower permission levels will be unable to access certain features

2.3 Non Functional Requirements

The program needs to be able to run for a long time without restarting

It should handle frequent use well.

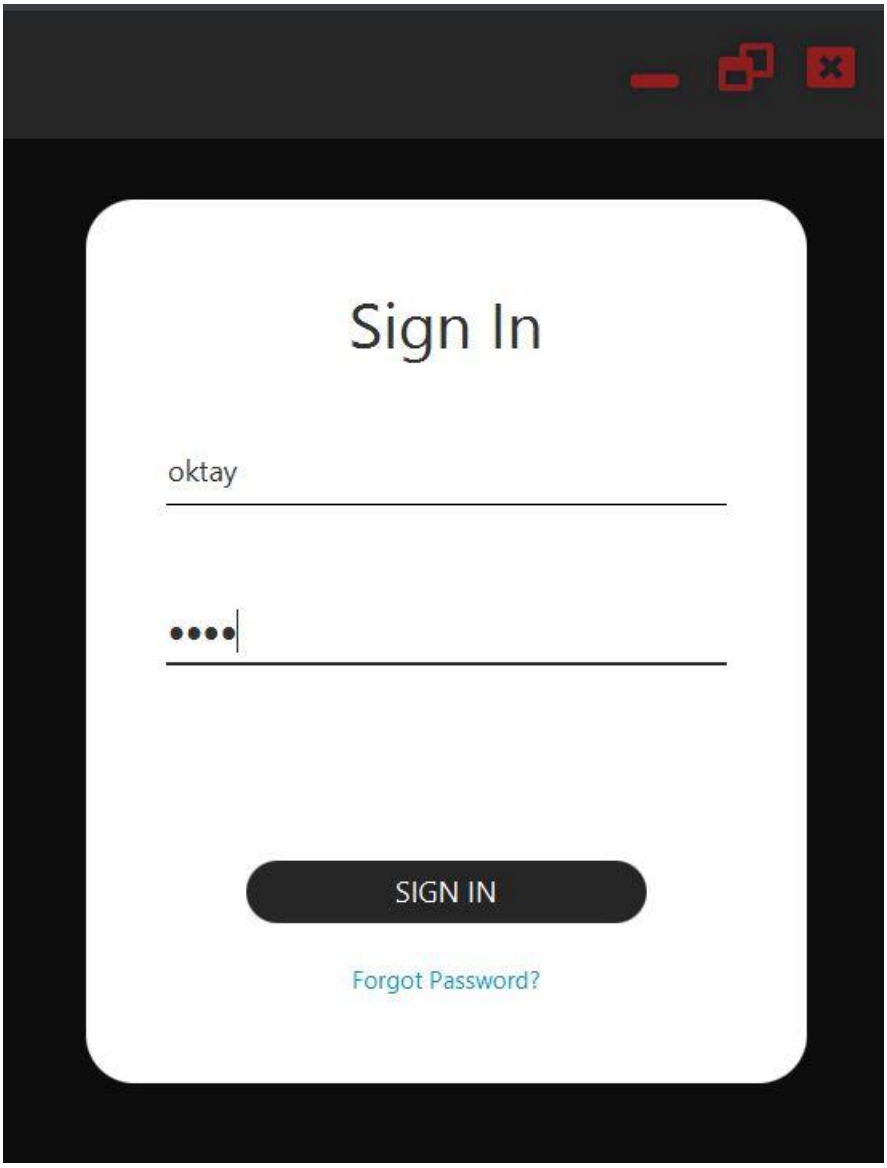
It should have an intuitive interface that will allow new users to use the program efficiently with minimal training

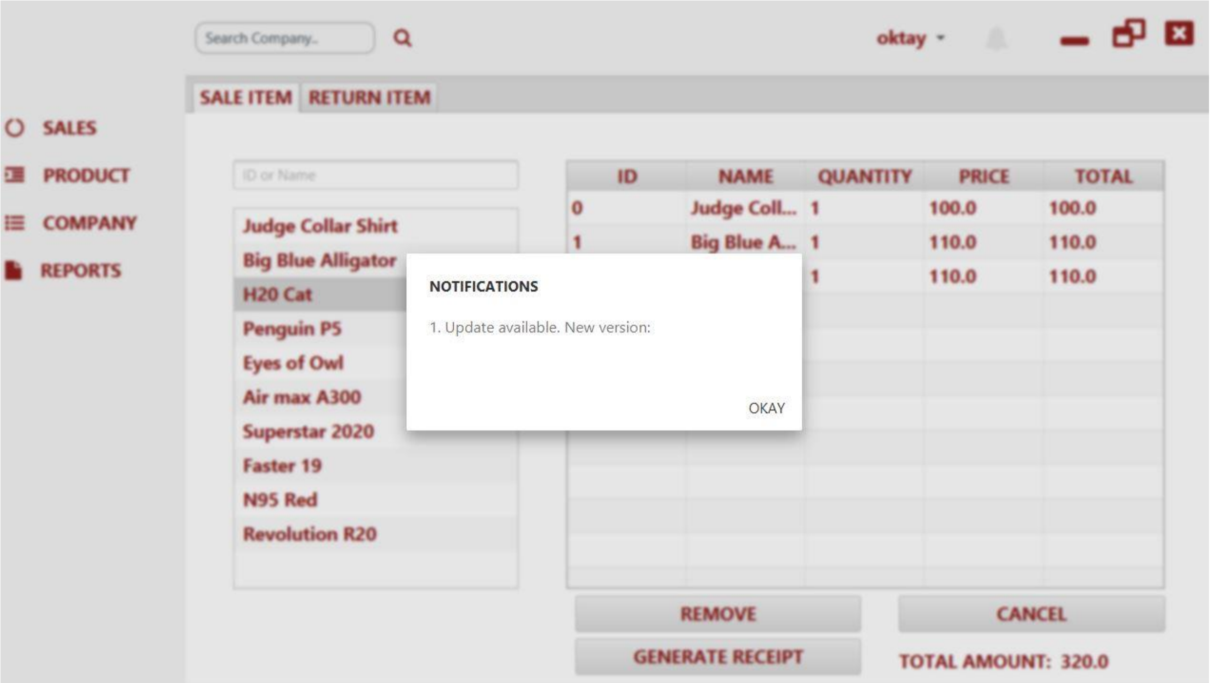
The program should handle various errors and not crash even when something goes wrong

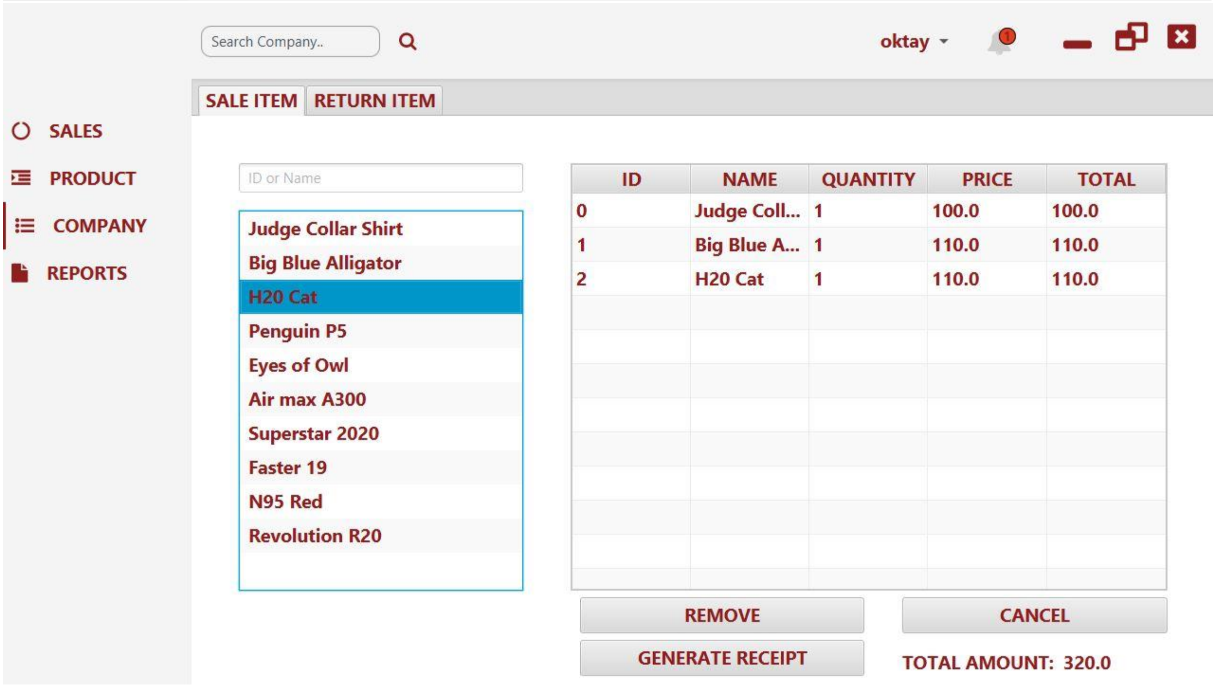
Users with different access levels should be able to access different content

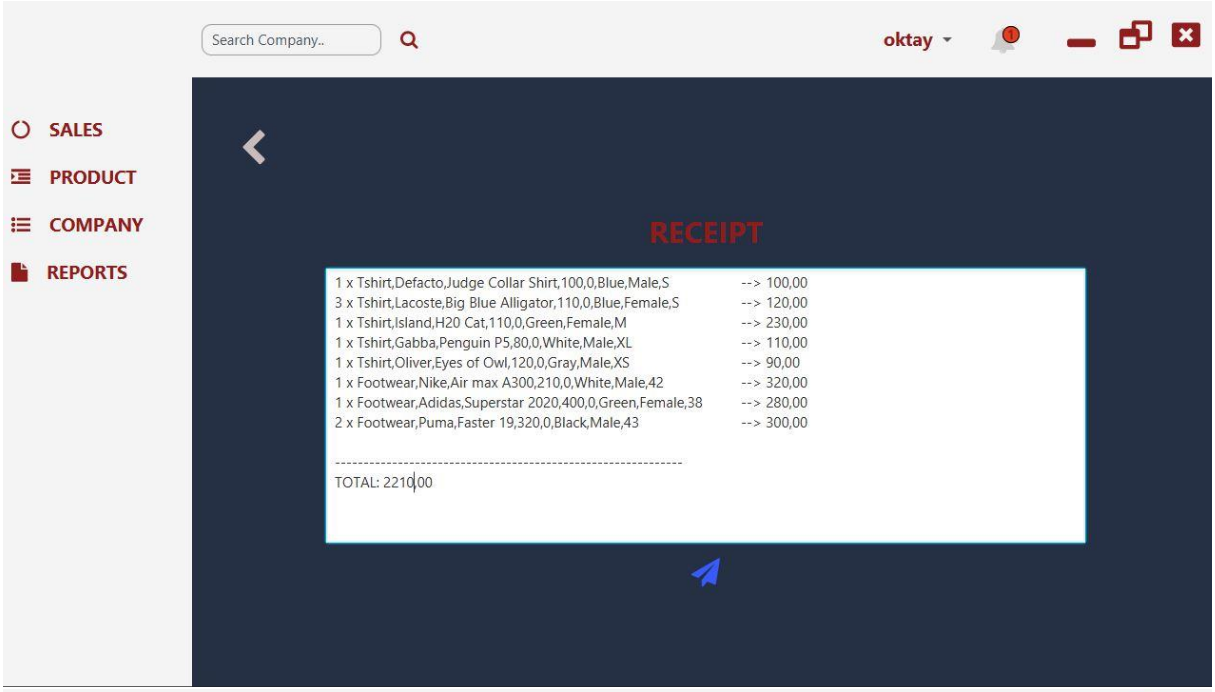
It should be modular and easy to improve in the future in case additional functions are requested

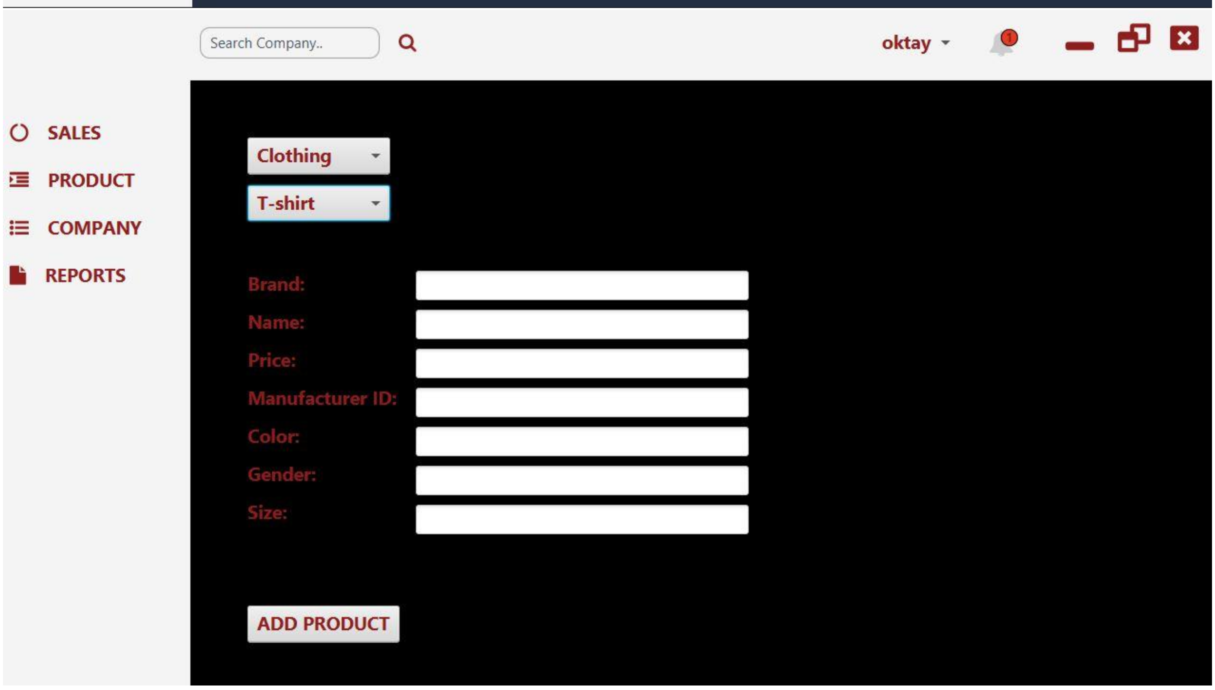
Program Screenshots:

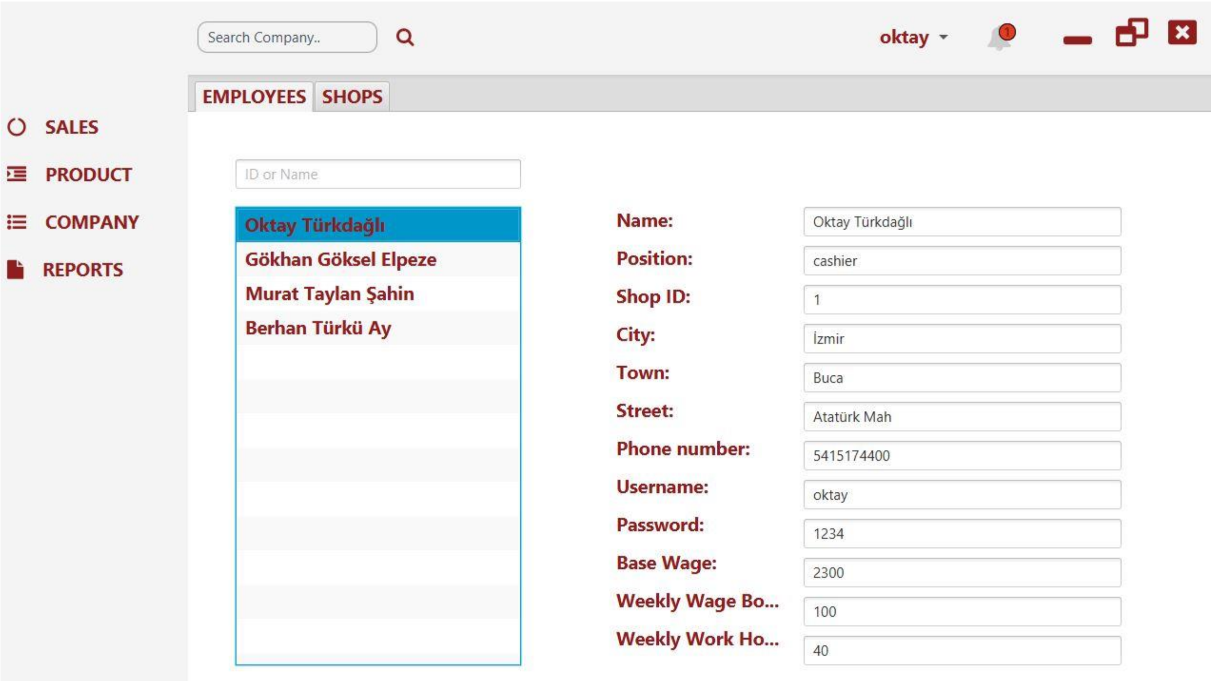


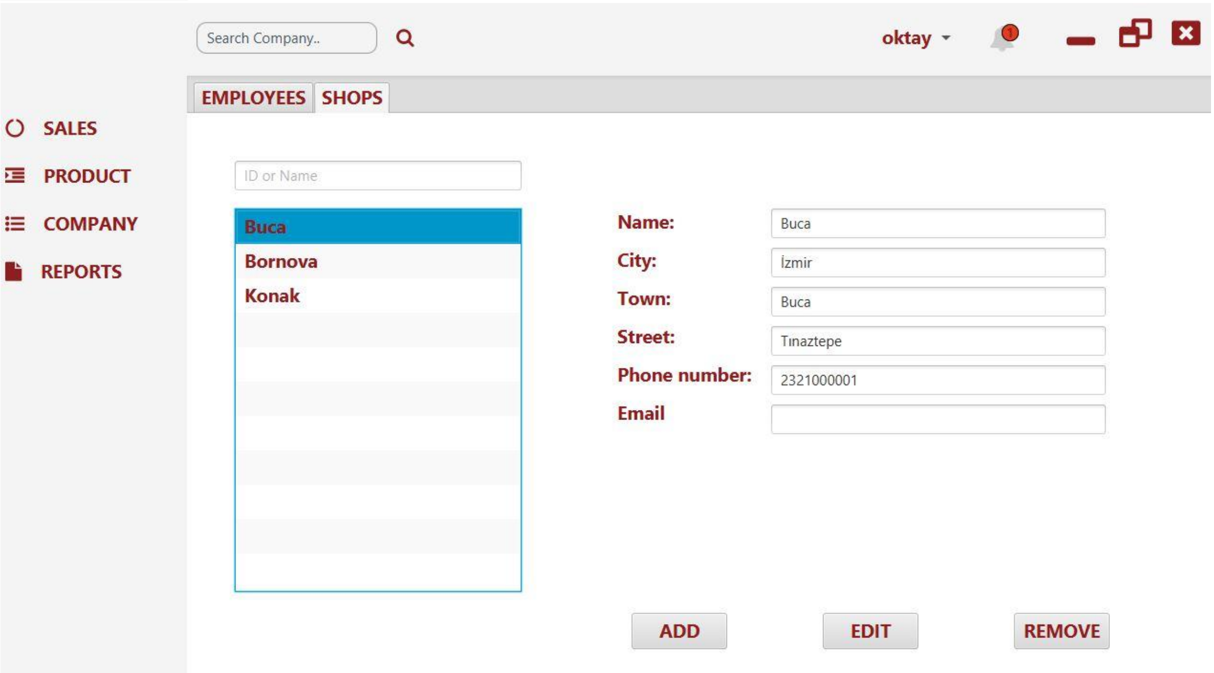

Search Company..
oktay
SALE ITEM
RETURN ITEM
SALES
PRODUCT
COMPANY
REPORTS
ID or Name
Judge Collar Shirt
Big Blue Alligator
H20 Cat
Penguin P5
Eyes of Owl
Air max A300
Superstar 2020
Faster 19
N95 Red
Revolution R20
ID
NAME
QUANTITY
PRICE
TOTAL
0 Judge Coll... 1 100.0 100.0
1 Big Blue A... 1 110.0 110.0
1 110.0 110.0
NOTIFICATIONS
1. Update available. New version:
OKAY
REMOVE
CANCEL
GENERATE RECEIPT
TOTAL AMOUNT: 320.0

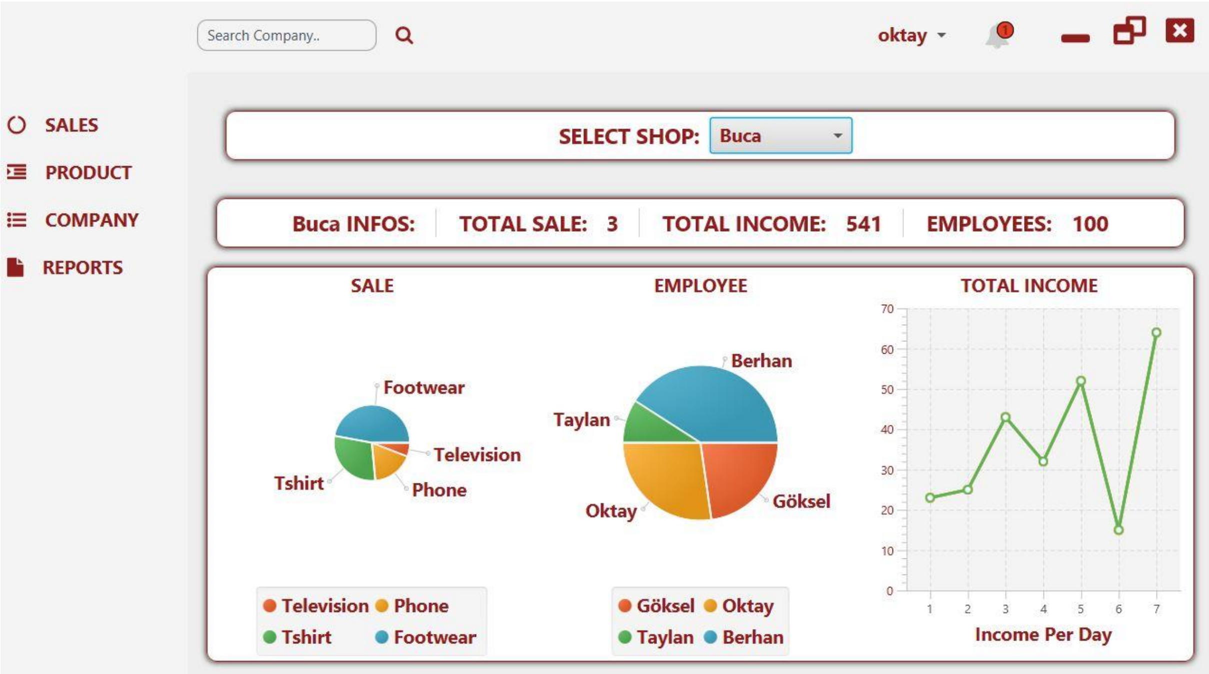

Search Company..
oktay
SALE ITEM
RETURN ITEM
SALES
PRODUCT
COMPANY
REPORTS
ID or Name
Judge Collar Shirt
Big Blue Alligator
H20 Cat
Penguin P5
Eyes of Owl
Air max A300
Superstar 2020
Faster 19
N95 Red
Revolution R20
ID
NAME
QUANTITY
PRICE
TOTAL
0 Judge Coll... 1 100.0 100.0
1 Big Blue A... 1 110.0 110.0
2 H20 Cat 1 110.0 110.0
REMOVE
CANCEL
GENERATE RECEIPT
TOTAL AMOUNT: 320.0


Search Company..
oktay
SALES
PRODUCT
COMPANY
REPORTS
RECEIPT
1 x Tshirt,Defacto,Judge Collar Shirt,100,0,Blue,Male,S --> 100,00
3 x Tshirt,Lacoste,Big Blue Alligator,110,0,Blue,Female,S --> 120,00
1 x Tshirt,Island,H20 Cat,110,0,Green,Female,M --> 230,00
1 x Tshirt,Gabba,Penguin P5,80,0,White,Male,XL --> 110,00
1 x Tshirt,Oliver,Eyes of Owl,120,0,Gray,Male,XS --> 90,00
1 x Footwear,Nike,Air max A300,210,0,White,Male,42 --> 320,00
1 x Footwear,Adidas,Superstar 2020,400,0,Green,Female,38 --> 280,00
2 x Footwear,Puma,Faster 19,320,0,Black,Male,43 --> 300,00
TOTAL: 2210,00


Search Company..
oktay
SALES
PRODUCT
COMPANY
REPORTS
Clothing
T-shirt
Brand:
Name:
Price:
Manufacturer ID:
Color:
Gender:
Size:
ADD PRODUCT


Search Company..
oktay
EMPLOYEES
SHOPS
SALES
PRODUCT
COMPANY
REPORTS
ID or Name
Oktay Türkdağlı
Gökhan Göksel Elpeze
Murat Taylan Şahin
Berhan Türkü Ay
Name:
Oktay Türkdağlı
Position:
cashier
Shop ID:
1
City:
İzmir
Town:
Buca
Street:
Atatürk Mah
Phone number:
5415174400
Username:
oktay
Password:
1234
Base Wage:
2300
Weekly Wage Bo...
100
Weekly Work Ho...
40


Search Company..
oktay
EMPLOYEES
SHOPS
SALES
PRODUCT
COMPANY
REPORTS
ID or Name
Buca
Bornova
Konak
Name:
Buca
City:
İzmir
Town:
Buca
Street:
Tınaztepe
Phone number:
2321000001
Email
ADD
EDIT
REMOVE


Search Company..
oktay
SALES
PRODUCT
COMPANY
REPORTS
SELECT SHOP:
Buca
Buca INFOS:
TOTAL SALE: 3
TOTAL INCOME: 541
EMPLOYEES: 100
SALE
Footwear
Television
Tshirt
Phone
Television
Phone
Tshirt
Footwear
EMPLOYEE
Berhan
Taylan
Oktay
Göksel
Göksel
Oktay
Taylan
Berhan
TOTAL INCOME
Income Per Day

2.4 Use Cases

A use case diagram is used to represent use cases and actors of the program in detail (figure 11). There are 4 main actors on the system. Administrator for managing the overall system and company, Branch manager for overseeing a branch and managing sales, sales associate for completing sales and a customer who makes purchases through the sales associate. Management Screen, Company screen, product screen, sale screen and menu represent 5 main screens of the system and other use cases under them shows their main functionalities. Blue dotted line is used to imply <<Extends>>, black dotted line is for <<Include>> and straight black line shows association.

2.5 Models

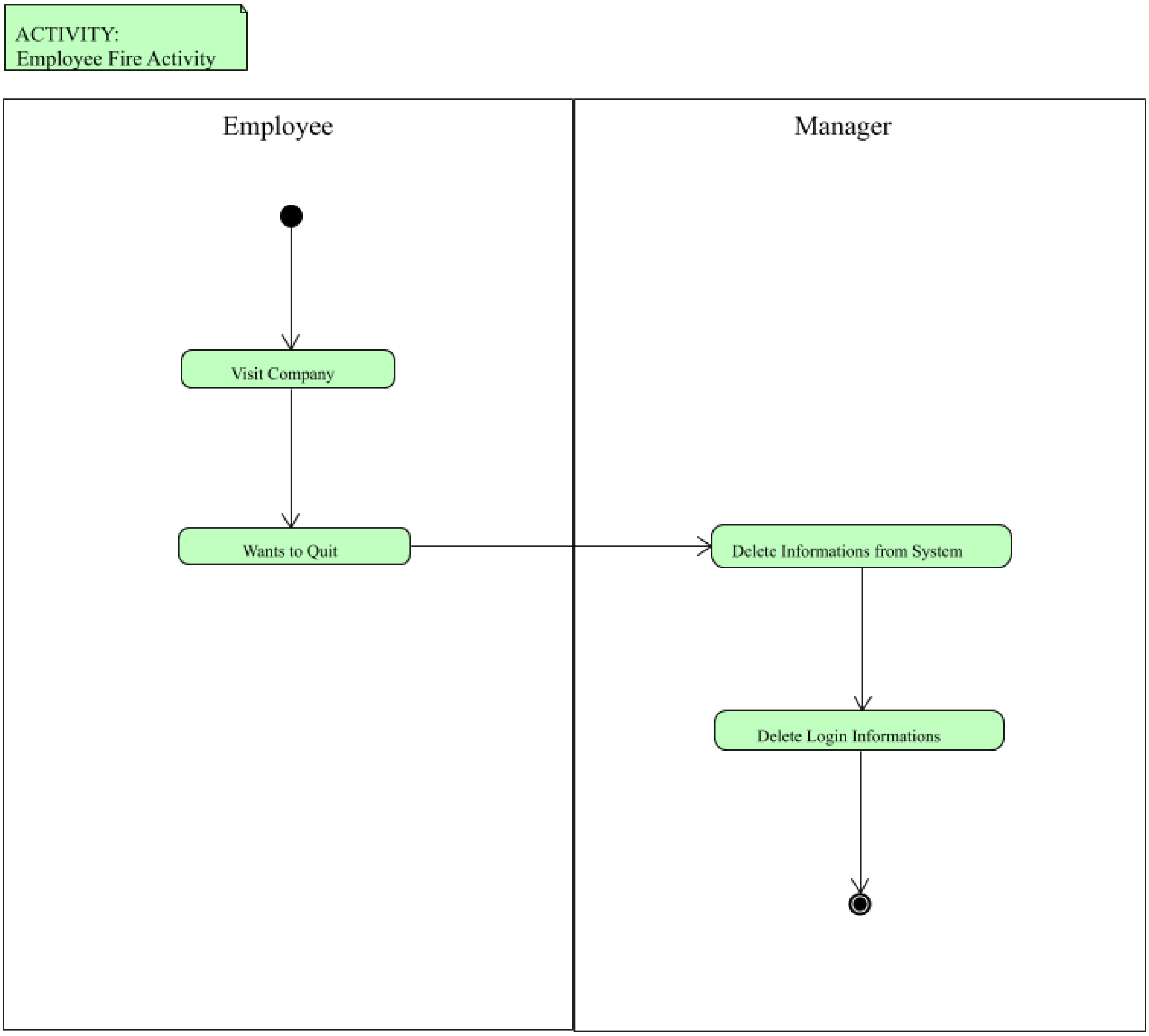


Figure 1

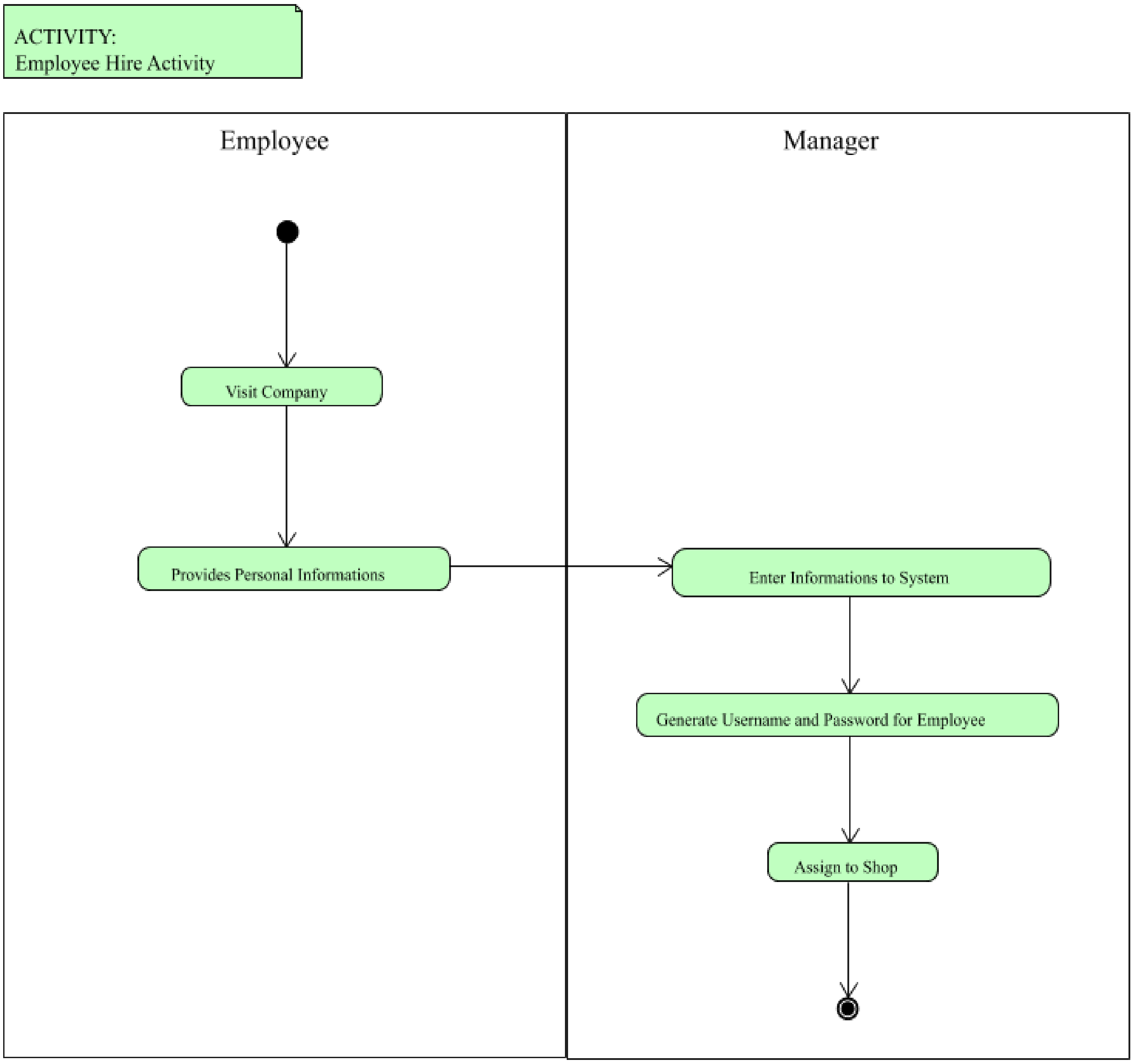

ACTIVITY:
Employee Hire Activity
Employee
Manager
Visit Company
Provides Personal Informations
Enter Informations to System
Generate Username and Password for Employee
Assign to Shop

Figure 2

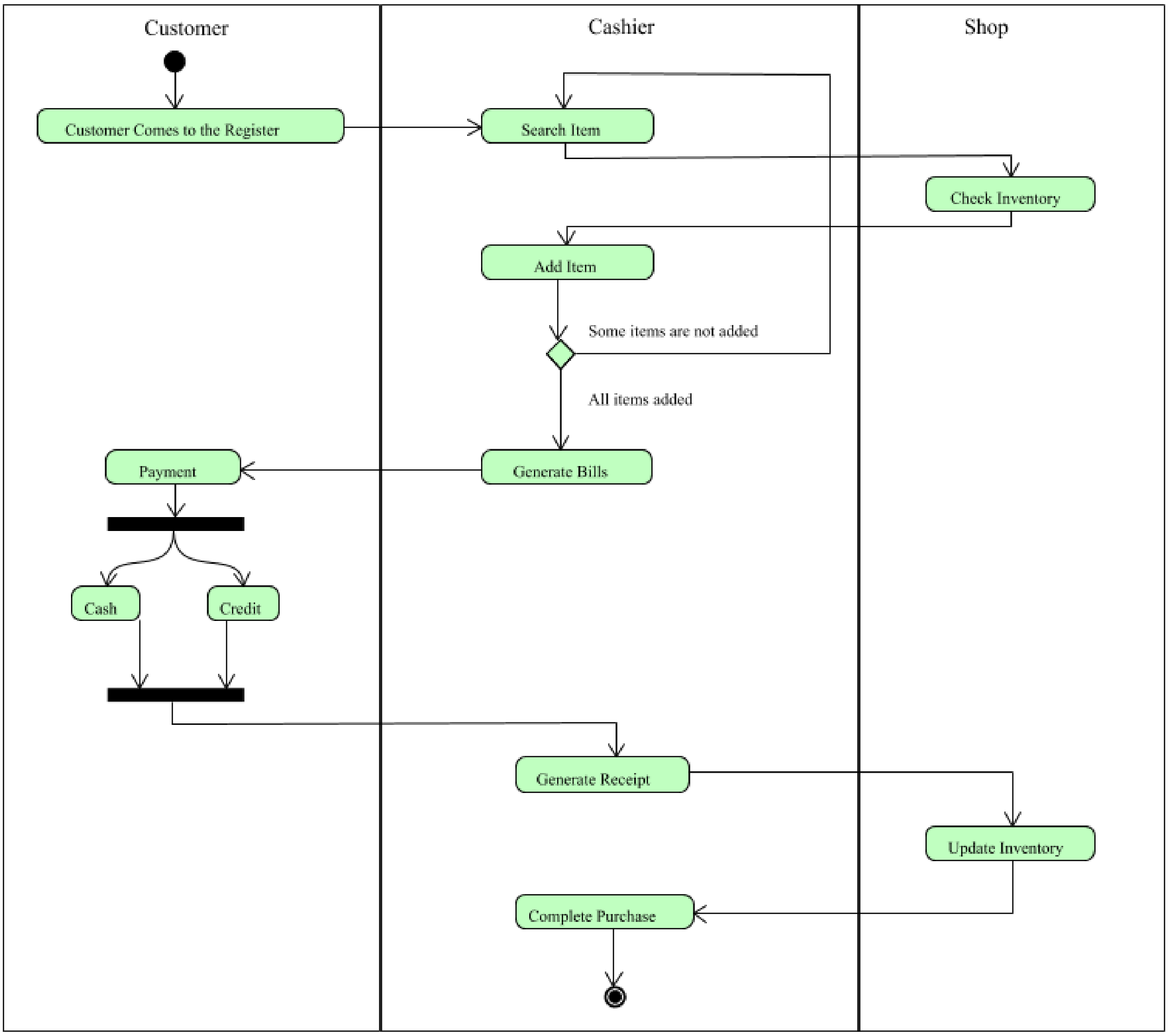


Figure 3

ACTIVITY

FIRE
Employee resignes, program user removes their personal information from system, then removes his username and password.

HIRE
Employee comes to the company and gives his personal information for the hiring process. Program user enters this information to the system and generates unique username and password. Assigns him to a shop.

SALE
Customer comes to register cashier searchs item from the system, shop checks inventory, then cashier adds item and repeats this until all customers items has been scanned, customer chooses payment option (cash or credit card), cashier generates receipt, shop updates inventory, cashier completes purchase.

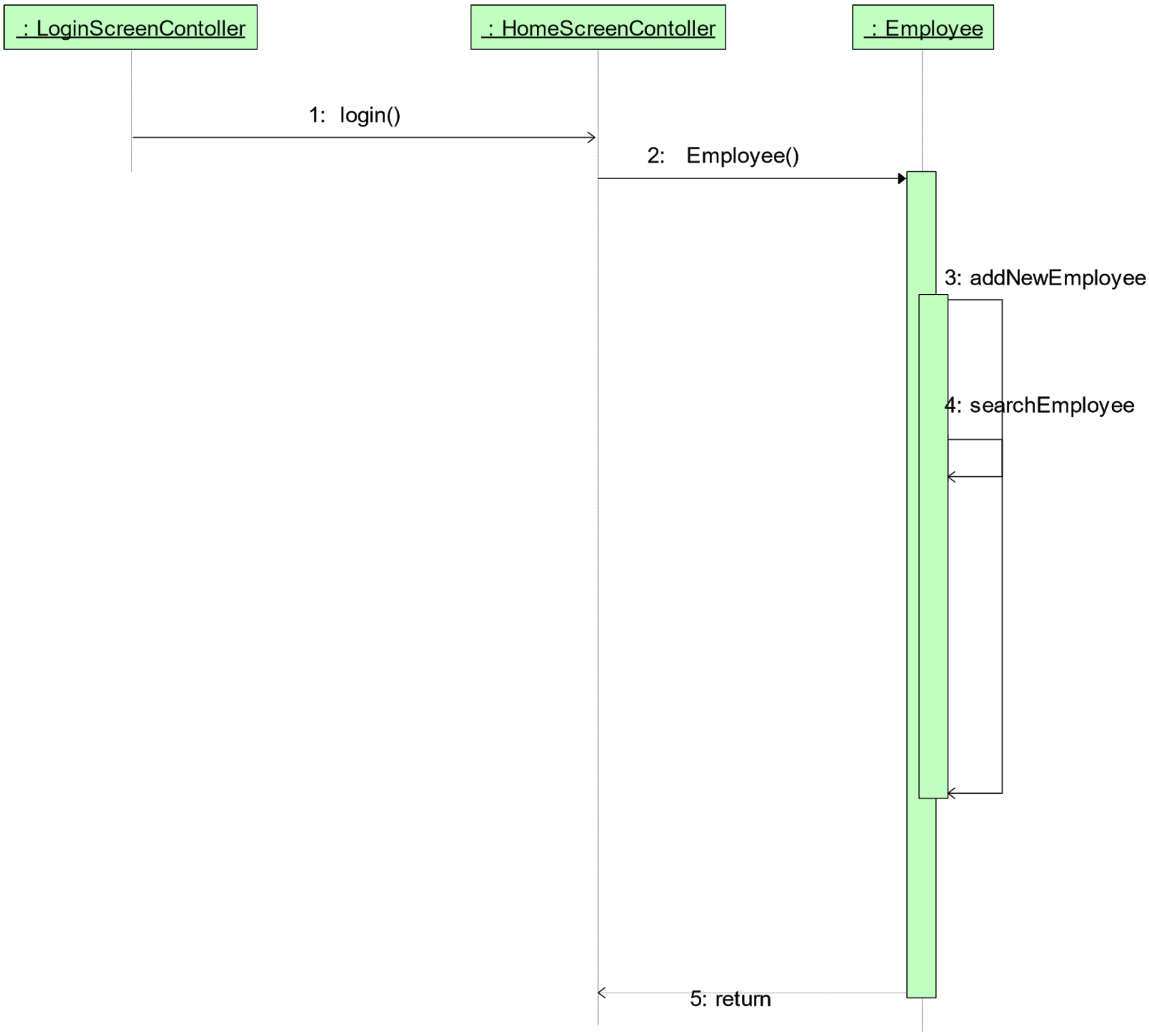

: LoginScreenContoller
: HomeScreenContoller
: Employee
1: login()
2: Employee()
3: addNewEmployee
4: searchEmployee
5: return

Figure 4

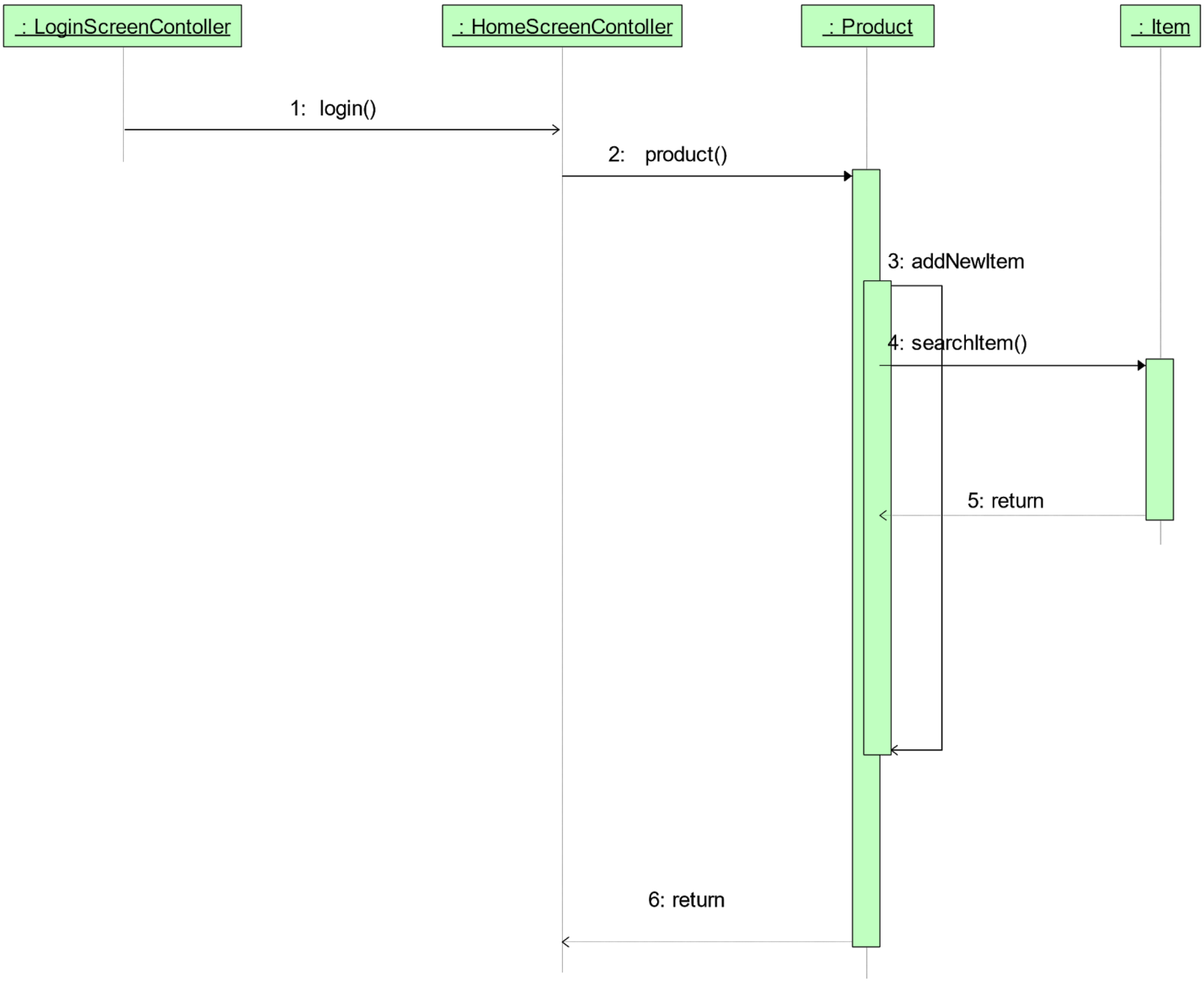

: LoginScreenContoller
: HomeScreenContoller
: Product
: Item
1: login()
2: product()
3: addNewItem
4: searchItem()
5: return
6: return

Figure 5

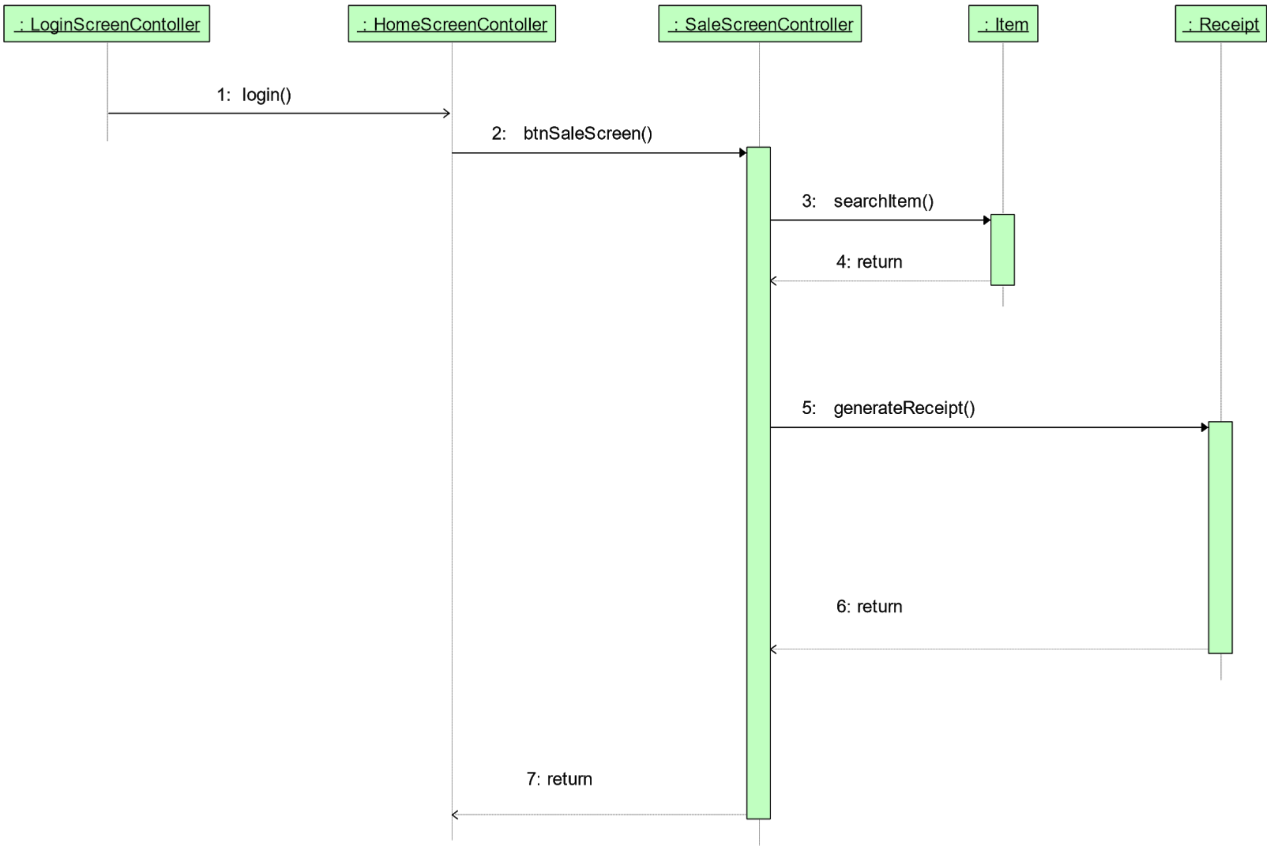

: LoginScreenContoller
: HomeScreenContoller
: SaleScreenController
: Item
: Receipt
1: login()
2: btnSaleScreen()
3: searchItem()
4: return
5: generateReceipt()
6: return
7: return

Figure 6

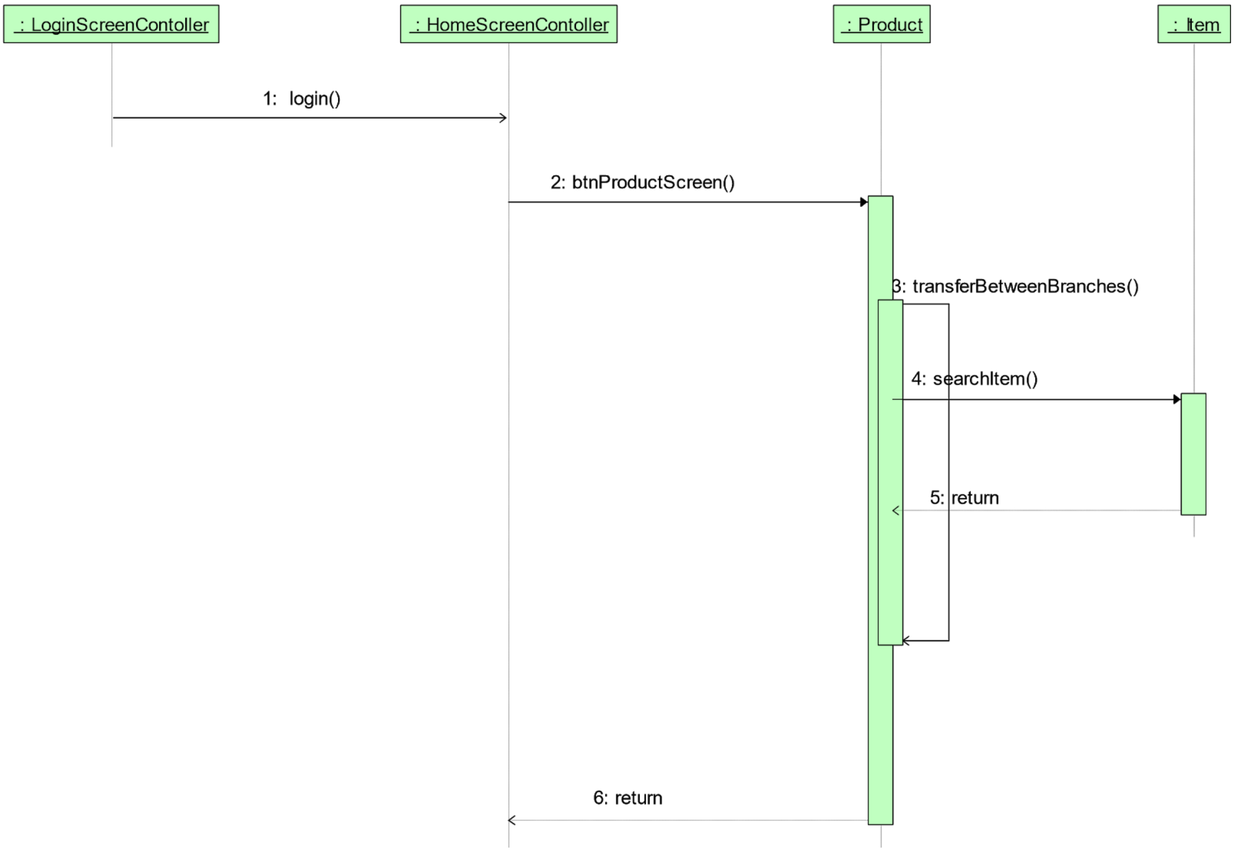


Figure 7

SEQUENCE DIAGRAM:

ADD NEW EMPLOYEE
User logs in, enters the home screen. Activates employee function and contacts employee class a new user is added with addNewEmployee function, before this function finishes, checks if this employee has been added before.

ADD NEW ITEM
User logs in, enters the home screen. Activates product function and connects to product class. A new product is added with addNewItem, before this function ends, checks if this item has been added before.

SALE SCREEN
User logs in, enters the home screen. Starts saleScreen function and connects to SaleScreenController class. Adds items to sale screen with searchItem function, after all items are added generates receipt with generateReceipt function

TRANSFER SCREEN
User logs in, enters the home screen. Starts productScreen function and connects to product class. Adds a new transfer with transferBetweenBranches funtion, before this function ends coonects to Item class with searchItem function and adds necessary items.

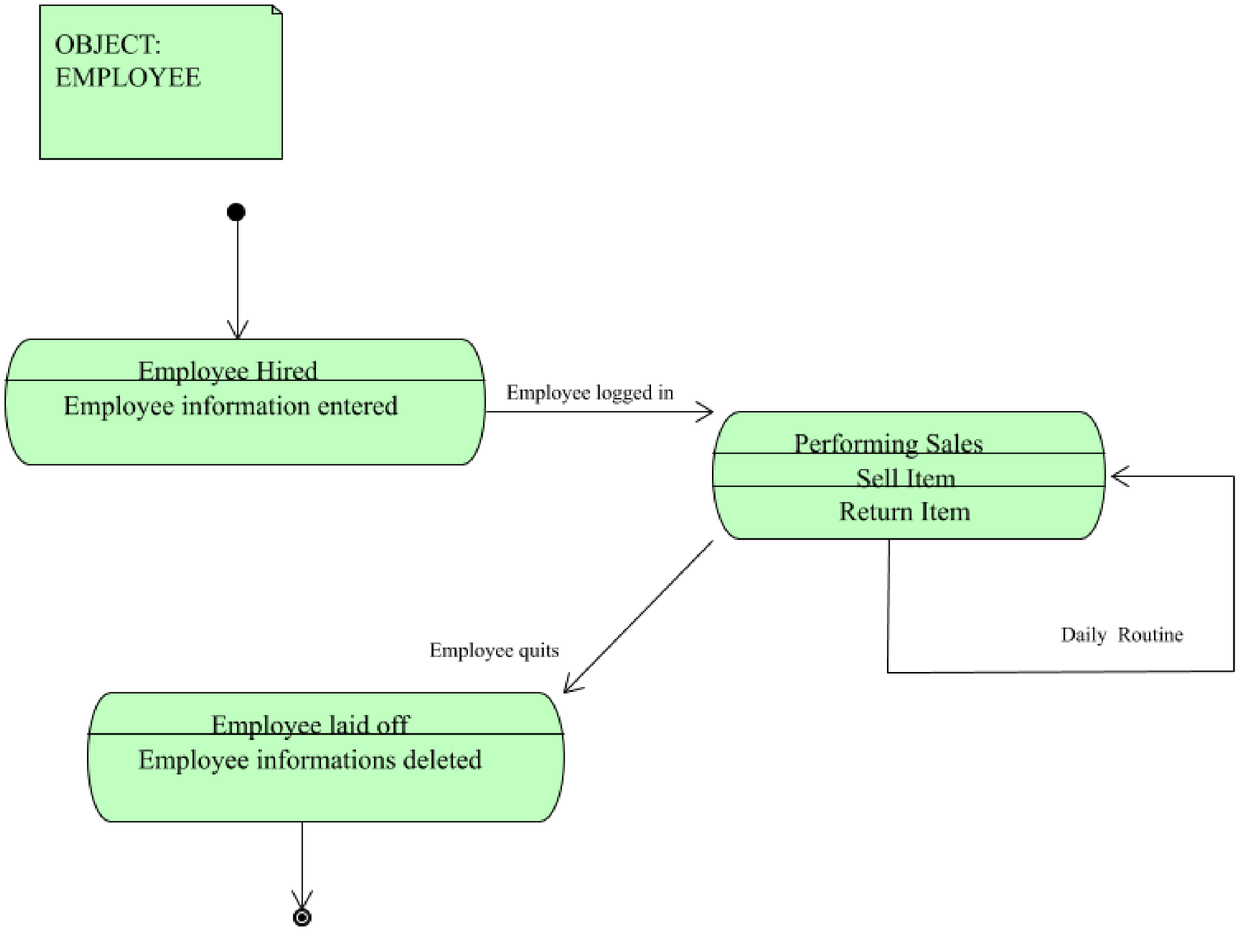

OBJECT:
EMPLOYEE
Employee Hired
Employee information entered
Employee logged in
Performing Sales
Sell Item
Return Item
Daily Routine
Employee quits
Employee laid off
Employee informations deleted

Figure 8

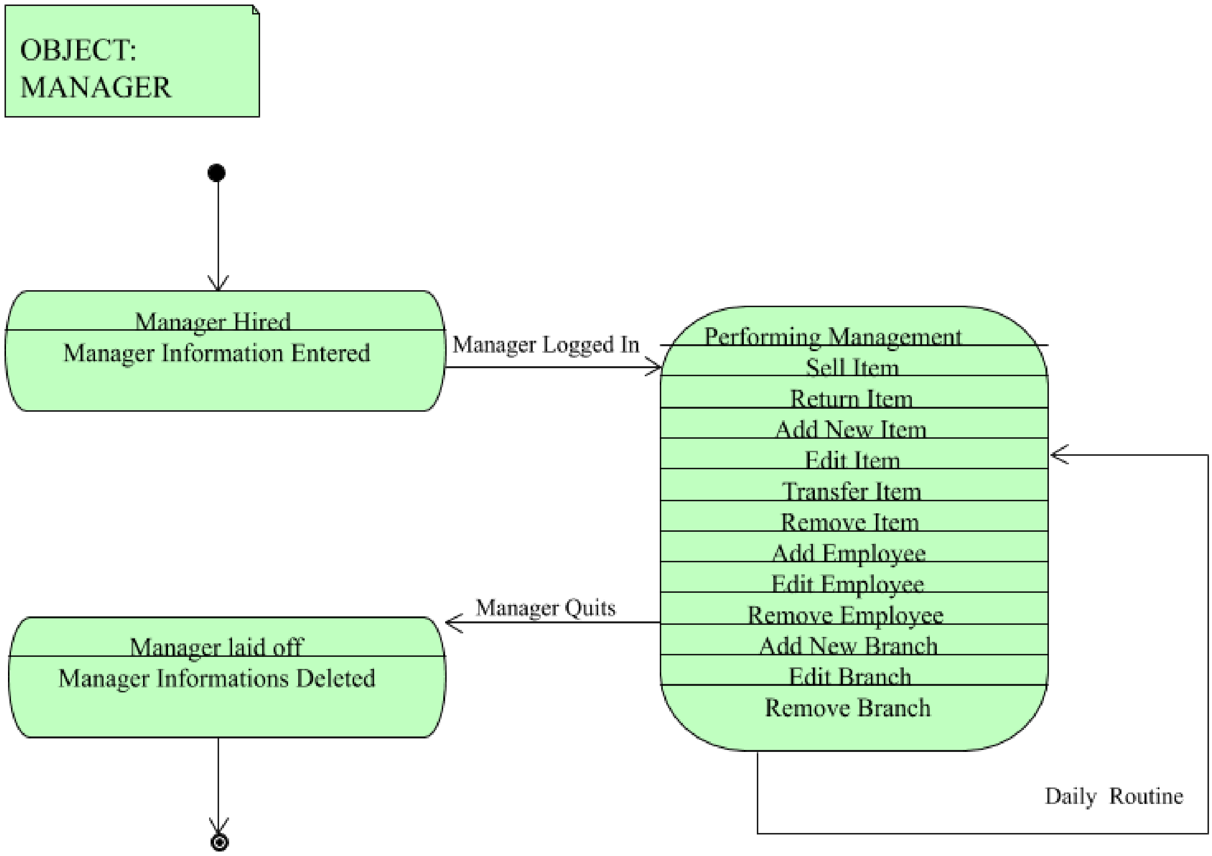


Figure 9

STATE

EMPLOYEE
Employee is hired, his daily routine is selling items and returning items, then when employee quits his information is deleted.

MANAGER
Manager is hired, his daily routine is selling, adding, returning, editing, transfering and removing items; adding, editing, removing employees. Adding, editing, removing branches. when manager quits his information is deleted.

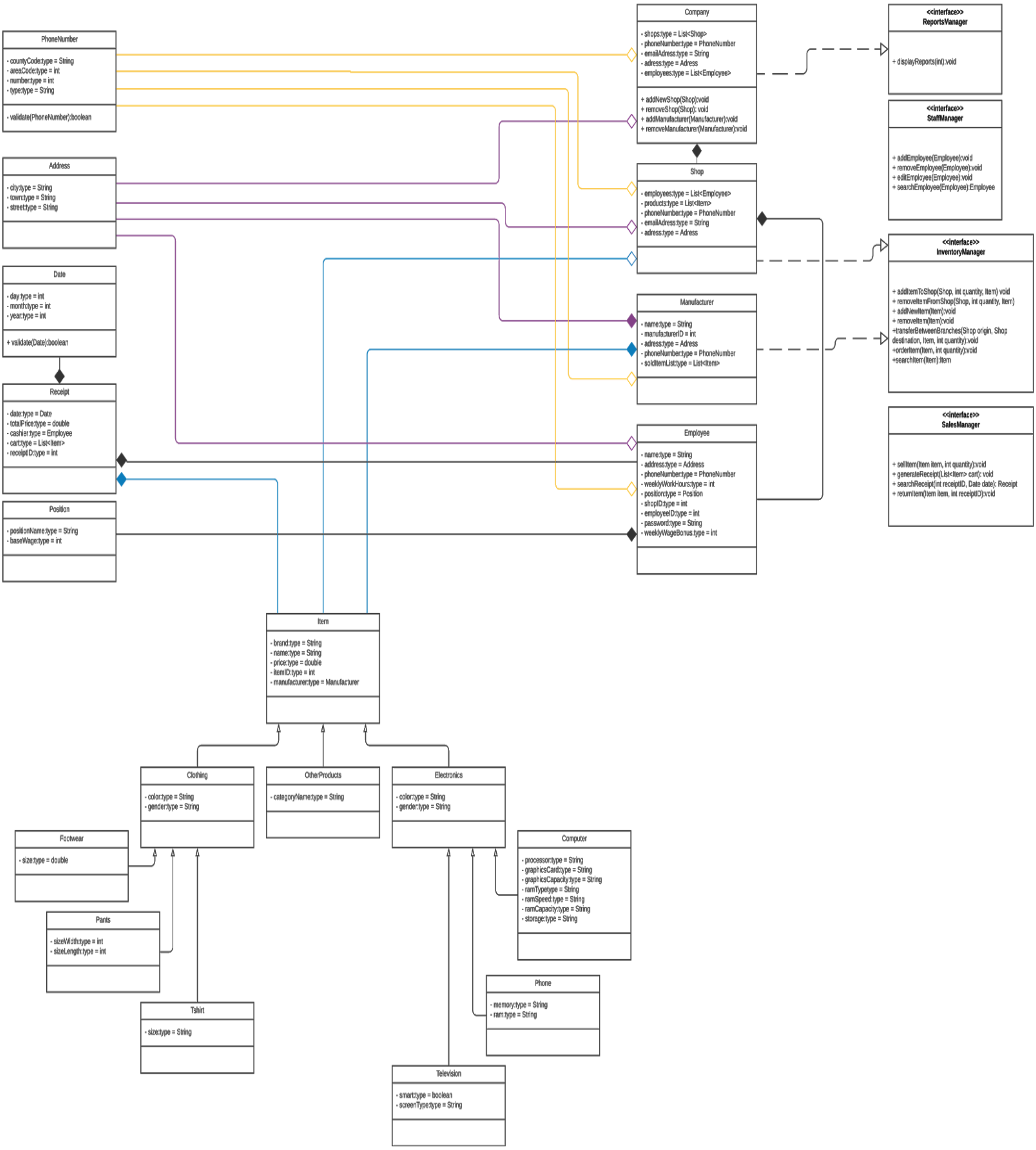


Figure 10

CLASS DIAGRAM:

Diagram shows all the classes inside the system and their attributes and functions. Item class and all its children represent product classes. Phone number, address, date, receipt and position represents utility classes. Company, shop, employee and manufacturer represents main classes. 4 interfaces correspond to management classes.

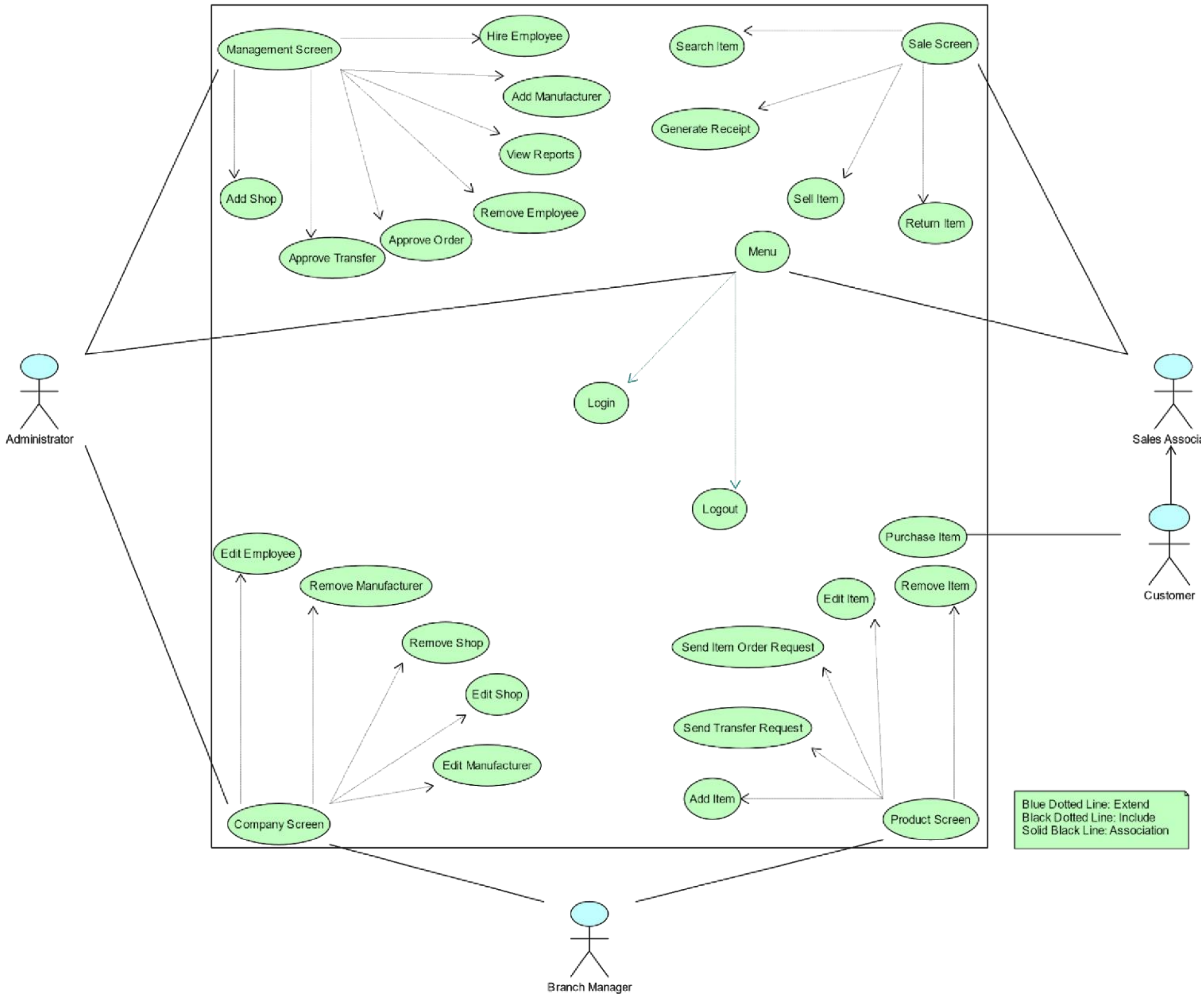


Figure 11

USE CASE DIAGRAM:

There are 4 main actors on the system. Administrator for managing the overall system and company, Branch manager for overseeing a branch and managing sales, sales associate for completing sales and a customer who makes purchases through the sales associate. Management Screen, Company screen, product screen, sale screen and menu represent 5 main screens of the system and other use cases under them shows their main functionalities.

Blue dotted line is used to imply <<Extends>>, black dotted line is for <<Include>> and straight black line shows association.

3. Project Plan

3.1 Task Descriptions

Determining Requirements

Designing GUI

Designing Project Structure

Implementing Functions
- Inventory Manger
- Sales Manager
- Report Manager
- Staff Manager
- Other Functions

Implementing Classes
- Management Classes
- Product Classes
- Company Classes
- Utility Classes
- GUI Classes

Drawing Diagrams

Database Designing and Implementing

3.2 Task Assignment

Everyone:
- Determining Requirements
- Designing GUI
- Other Functions
- Drawing Diagrams

M.Taylan Şahin:
- Designing Project Structure
- Staff Manager
- Management Classes
- Utility Classes

G.Göksel Elpeze:
- Designing Project Structure
- Report Manager
- GUI Classes
- Utility Classes
- Database Designing and Implementing

Berhan T. Ay:

- Inventory Manger
- Staff Manager
- Product Classes
- Company Classes

Oktay Türkdağlı:

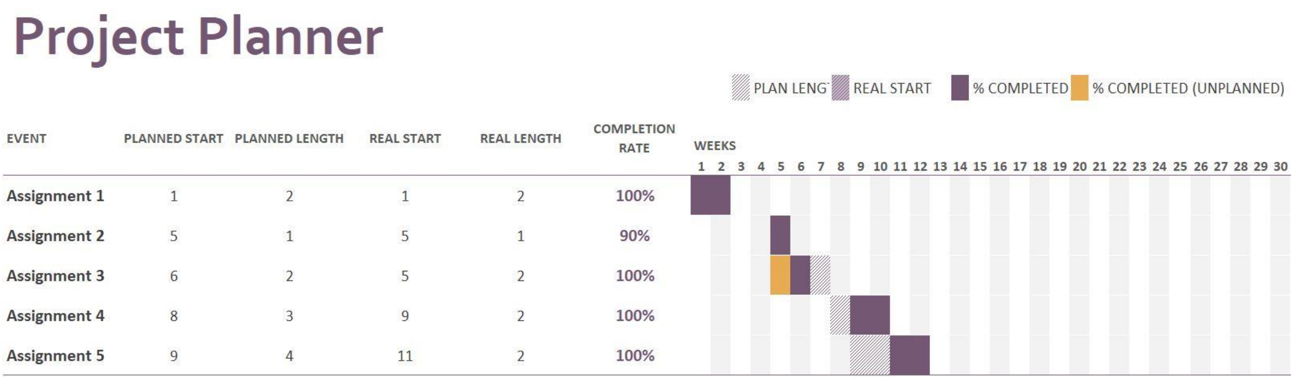
- Sales Manager
- Report Manager
- Product Classes
- Company Classes

3.3 Deliverables and Milestones

We had 5 milestones in this project:

1. Discussing and deciding on the topic, talking about the details of the project and updating the relevant parts of the project.
2. Designing the system and drawing activity, sequence, class, state and use case diagrams. Updating the relevant parts of the report.
3. Creating required classes and deciding class variables. Updating the relevant parts of the report.
4. Filling in the functions of the classes created previously. Updating the relevant parts of the report.
5. Completing GUI and project report. Fully finishing the project.

3.4 Project Schedule



EVENT	PLANNED START	PLANNED LENGTH	REAL START	REAL LENGTH	COMPLETION RATE
Assignment 1	1	2	1	2	100%
Assignment 2	5	1	5	1	90%
Assignment 3	6	2	5	2	100%
Assignment 4	8	3	9	2	100%
Assignment 5	9	4	11	2	100%

4. Testing

4.1 Features to be tested

Every new feature added will pass some test before the changes are made permanent. Some of these test are as follows:

- Program code will be checked for any syntax or logic errors.
- Program will be run if possible to see if it works
- Newly implemented functions will be used repeatedly to see if it works as intended

4.2 Test Cases

Some example test cases are:

- Do fields display correct information
- Is GUI responsive
- Is the speed up to requirements
- Are there proper checks for wrong types of input inplace
- Does the program consistently crash while using any feature
- Is interface usable in different displays
- Are there any data loss between sessions
- Does the program give appropriate feedback in case of errors

.

.

.

4.3 Testing Schedule

Testing will be done side-by-side with implementation of new features. Every feature added will be tested to ensure the quality. After the whole project is completed these test will be repeated to check if there were any errors in integration.

5. Conclusion

5.1 The Problem and Solution

The problem that The need for a program with a colorful interface that can process data such as stock tracking, product sales, statistics and personnel information in a company with retail product sales, and which can yield meaningful output as a result.

The solution was to provide Using RMS, attract more customers and attend to existing customers in real-time, manage different branches of shop, analyze sale data and generate meaningful reports about your company's future. In addition to this you can check your employees' performance from this program

5.2 The Team and the SE Process

Bu projede Waterfall yazılım geliştireme süreci kullanıldı. Böylece Şelalenin her basamağında yer alan aktiviteler eksiksiz olarak yerine getirildi ve bir sonraki basamağa geçildi. Burada yapı oluşturulurken MVP (Minimum viable product) baz alındı. Waterfall modeli ile kullanıcı katılımı başlangıç safhasında mümkündür. Kullanıcı gereksinimleri bu safhada tespit edildi ve detaylandırıldı.

5.3 Engagement of Umbrella Activities

Two of the main umbrella activities we used were as follows:

Software Project Management: Which was used to lead the project and ensure that the project was controlled, monitored, and on schedule. “Trello” was used for these processes.

Version control System: A component of software configuration management, version control, also known as revision control or source control, management of changes to documents, computer programs, large web sites, and other collections of information. Each revision is associated with a timestamp and the person making the change. Revisions can be compared, restored, and with some types of files, merged. “gitlab” was used for these processes.

5.4 The Organization's Benefits

Our organization has gained many experiences with this project. He first designed many components such as interface design, creating a user-based management system and statistical reporting. In addition, we have mastered many issues such as working remotely, using the version control system and creating workflows.

6. User Manual

6.1 Software Description

Increasing productivity and speeding processes up has a direct effect on customer satisfaction in retailers. Technology plays an important role in terms of more comfortable and intuitive customer experience. Retail Management System (RMS) we develop keeps this principle in mind and lets you manage these processes in real-time. Thanks to this, it is possible to speed up processes easily ranging from receipts to products, inventory management to price management, purchases to promotions, orders to transfers.

RMS is designed with needs of retailers selling a wide range of different products in mind.

Using RMS, you can attract more customers and attend to existing customers in real-time, manage different branches of your shop, analyze your sale data and generate meaningful reports about your company's future. In addition to this you can offer promotions for customers and check your employees’ performance from this program.

6.2 How to Use the Software

Selling a Product:

- After Logging in with your username and password, click on SALES menu.
- In the screen that opens up, enter product id to the SEARCH bar. Click on the correct product or press ENTER.
- Chosen product will appear on the Cart Preview Window.
- Repeat these steps for all other products in your customer’s cart.
- Click on GENERATE RECEIPT button.
- Pick the correct payment option (Cash or Credit Card) and press COMPLETE TRANSACTION button.
- You can press CANCEL on any of these steps in case you want to cancel the transaction.
- Sale successful notification appears upon successful sale.

Adding a New Product:

- After login, if you have admin privileges PRODUCT tab will activate, click on PRODUCT menu.
- If you want to add a new item, leave SEARCH bar on the newly opened screen blank.

- Product’s attributes are filled (Example TV attributes):
    - o Category
    - o Name
    - o Size
    - o Build Date
    - o Name of Manufacturer
    - o Name of Distributor
    - o Price
    - o Quantity
    - o Date Product enters company (Passive)
    - o Person adding the item (Passive)
- Click on SAVE.
- Item added successfully notification appears upon successful entry.

Removing a Product:

- After login, if you have admin privileges PRODUCT tab will activate, click on PRODUCT menu.
- Type id of the product that needs to be removed on the newly opened screen.

  Click on the correct item or press ENTER.
- Attributes of the product should fill automatically. Click on REMOVE button
- Click YES on the “Are you sure?” popup.
- Item successfully removed notification appears upon successful removal.

Editing an Existing Product:

- After login, if you have admin privileges PRODUCT tab will activate, click on PRODUCT menu.
- Type id of the product that needs to be edited on the newly opened screen. Click on the correct item or press ENTER
- Attributes of the product should fill automatically. Modify the attributes you need to change.
- Click on EDIT button.

- If the edited product is the same as an already existing product, “Would you like to merge these items?” notification appears. If you do not want to merge the items, click no and edit items attributes again.
- Item successfully edited notification appears upon a successful edit..

Transfering Products:

- After login, if you have admin privileges PRODUCT tab will activate, click on PRODUCT menu.
- Pick TRANSFERS tab on the newly opened screen, Type item name or id to the SEARCH bar and pick the correct item or press ENTER
- Pick the origin and destination branches.
- Enter the quantity (Quantity may not exceed origin branch’s maximum stock).
- “Transfer request is sent.” notification appears upon successful entry.

- An admin should approve of transfer request.
    - If user has logged in with an admin account, MANAGEMENT tab will activate, click on MANAGEMENT tab.
    - Pick TRANSFER REQUESTS tab from newly opened window.
    - Transfer requests should be listed here. Pick the request you want to approve and click APPROVE.
    - Transfer approved notification appears upon successful approval.

Adding a New Employee:

- After login, if you have admin privileges MANAGEMENT tab will activate, click on MANAGEMENT menu.
- Navigate to EMPLOYEES tab on the newly opened window and pick ADD option. Do not fill in the SEARCH bar.
- Following attributes of the Employee is filled
    - Employee ID (Passive)
    - National Identification Number

- o Taxpayer ID
- o Name
- o Surname
- o Branch id where employee works in
- o Position
- o Phone Number
- o Address
- o E-mail
- o Salary (If this is left empty base salary of the position will be applied)
- o Birth Date
- o Birthplace
- o Start date(Passive)
- o Current user (Passive)

- Click on SAVE button.
- Employee added notification appears upon successful entry.

Editing Employees:

- After login, if you have admin privileges MANAGEMENT tab will activate, click on MANAGEMENT menu.
- Navigate to EMPLOYEES tab on the newly opened window and pick ADD option. Fill in the SEARCH bar with employee id or name and pick the employee or press ENTER.
- Employee attributes should automatically fill up, edit the attribute you want to change.
- Click EDIT button.
- Edit successful notification appears upon successful edit.

Removing Employees:

- After login, if you have admin privileges MANAGEMENT tab will activate, click on MANAGEMENT menu.

- Navigate to EMPLOYEES tab on the newly opened window and pick ADD option. Fill in the SEARCH bar with employee id or name and pick the employee or press ENTER.
- Employee attributes should automatically fill up, check if employee is the one you want to remove, then Click EDIT button.
- Answer yes to “Are you sure?” notification.
- Employee successfully removed notification appears upon successful removal.

Adding a new branch:

- After login, if you have admin privileges MANAGEMENT tab will activate, click on MANAGEMENT menu.
- Navigate to BRANCHES tab on the newly opened window and pick ADD option. Do not fill in the search bar.
- Fill in the following attributes of the new branch:
  - Name
  - Phone number
  - E-mail
  - Address
  - Opening Date (Passive)
- Click on SAVE button
- Successfully saved notification appear upon successful entry.

Removing a Branch:

- After login, if you have admin privileges MANAGEMENT tab will activate, click on MANAGEMENT menu.
- Navigate to BRANCHES tab on the newly opened window and pick REMOVE option. Fill the search bar with branch id.
- Click on REMOVE button
- Answer yes to “Are you sure?” notification.
- Branch successfully removed notification appears upon successful removal.

Editing a Branch:

- After login, if you have admin privileges MANAGEMENT tab will activate, click on MANAGEMENT menu.
- Navigate to BRANCHES tab on the newly opened window. Fill the search bar with branch id.
- Attributes should fill automatically. Edit the attributes you want to change.
- Answer yes to “Are you sure?” notification.
- Branch successfully edited notification appears upon successful edit.

6.3 Troubleshooting Common Problems

Problem: Loading incorrectly / Session has Timed Out

Clear your system’s cache and re-launch the program.

Problem: Invalid Login

The credentials used to log in was not found in the database, input the credentials again in case of mistype.

Problem: No Product Found

Check the product name and id parameter, you may have typed an incorrect product name or the product may not be in the system. You can ask your administrator to add the product.

Problem: No Employees Found

Check the employee name and id parameter, you may have typed an incorrect product name or it may not be present in the running system. You can ask your manager for help to add the employee.

Problem: such a product already exists

You may get this error when trying to add new products, if there is another product with the same name and same features as the product you added, you will not be allowed to add new products. Try changing the name and properties of the product.

Problem: such an employee already exists

You may get this error while trying to add a new employee, you will not be allowed to add another employee with the same name and same features as the employee you added. Try changing the name and characteristics of the employe