



Faculty of Engineering and Technology

Computer Science Department

Software Engineering – COMP433

Course's Project

Prepared by:

Student Name	Student ID
Ahmad Ghanem	1201954
Abdullah Naser	1201952
Wasim Atta	1200920
Mohammad Jarrar	1201762

Instructor: Dr. Samer Zein

Submission Date: 20/6/2024

Contents

2: Requirements Engineering	1
2.1: User Requirements [Group Task]	1
2.2: System Requirements [Group Task]	2
2.3: Effort and Cost Estimation [Group Task]	7
3: Requirements Engineering Cont.	9
3.1: Scenario Analysis [Individual Work]	9
3.1.1: Scenario of R1 [Made by Mohammad Jarrar]	9
3.1.2: Scenario of R3 [Made by Ahmad Ghanem]	11
3.1.3: Scenario of R4 [Made by Abdullah Naser]	13
3.1.4: Scenario of R6 [Made by Wasim Atta]	15
3.2: Actors and Use case (Diagram) Analysis & Modelling [Group Work]	16
3.3: Use Case Description [Individual Work]	17
3.3.1: Use Case Specification for R5 [Made by Mohammad Jarrar]	17
3.3.2: Use Case Specification for R6 [Made by Ahmad Ghanem]	19
3.3.3: Use Case Specification for R7 [Made by Abdullah Naser]	21
3.3.4: Use Case Specification for R8 [Made by Wasim Atta]	23
3.4: Activity Diagram Analysis & Modelling [Group Work]	25
3.5: Instance Activity Diagrams [Individual Work]	26
3.5.1: Instance Activity Diagram for R5 [Made by Mohammad Jarrar]	26
3.5.2: Instance Activity Diagram for R7 [Made by Abdullah Naser]	27
3.5.3: Instance Activity Diagram for R8 [Made by Wasim Atta]	28
3.5.4: Instance Activity Diagram for R6 [Made by Ahmad Ghanem]	29
4: System's Modeling and Architecture	30
4.1: Class Diagram Analysis [Group Task]	30
4.2: System Sequence Modelling [Individual Work]	31
4.2.1: Sequence Diagram of R5 [Made by Mohammad Jarrar]	31
4.2.2: Sequence Diagram of R6 [Made by Ahmad Ghanem]	32
4.2.3: Sequence Diagram of R7 [Made by Abdullah Naser]	33
4.2.4: Sequence Diagram of R8 [Made by Wasim Atta]	34
4.3: System Design Goals [Group Task]	35

4.4: System Component Diagram [Group Task]	36
4.4.1: Components Description	36
4.4.2: Component Diagram	38
4.5: System Deployment Design [Group Task]	39

2: Requirements Engineering

2.1: User Requirements [Group Task]

1. The store manager shall be able to add, edit, and remove employee profiles.
2. The Marble Shop system shall be able to generate a basic sales report showing overall sales during the month.
3. The employee(cashier) shall be able to view available raw materials and marble stone products along with their information.
4. The employee(cashier) shall be able to add a new type of raw materials add new marbles to the store or update the product availability from the store so that the system can reflect the availability of the product.
5. The employee shall be able to search for a specific marble stone product or raw material and check its status and availability.
6. The customer shall be able to add marble stone products or a quantity of raw materials to his cart and review his selections. So he can complete his purchase well.
7. The customer shall be able to ask for delivery of the purchased products to a desired location and get a refund if there is any damage.
8. The customer should have the option to pay using either cash or credit card. To provide flexibility in payment.

2.2: System Requirements [Group Task]

1- UR: The store manager shall be able to add, edit, and remove employee profiles

System Requirements:

- 1.1- The administrator should be able to add a User the attributes of the user (name, address, picture, tel-no) such as the name and the tel-no are mandatory for adding completion.
- 1.2- The name should be the first and last names, and the tel-no should provide the area code for the number.
- 1.3- The system shall enable the manager to edit the employee information (name, address, picture, tel-no) to perform edit action, by entering the change of inputs after highlighting the desired field.
- 1.4- The administrator should be able to delete an employee by highlighting the employee to delete and should pop a delete action that deletes the employee from the database using Node.js or Django, which are powerful and flexible frameworks for building server-side applications.
- 1.5- The manager portal will have features for account management, employee creation as mentioned before, and management such as: deleting an employee or editing its info. We will use RESTful APIs to interact with the backend, fetch employee data, or update its fields. These APIs will be secured using techniques like token-based authentication to ensure that only authorized users can access the data.

2. UR: The Marble Shop system shall be able to generate a basic sales report showing overall sales during the month.

System Requirements:

- 2.1- On the last working day of each month, a summary of the marbles purchased, and their cost shall be generated.
- 2.2- The system shall automatically generate the report for printing after 17:30 on the last working day of the month.
- 2.3- access to all cost reports shall be restricted to authorized users listed on a management access control list.
- 2.4- A report shall be created for the shop and shall list the marble names that were sold, the total number of each marble that was purchased, and the total cost of the marbles for each purchase.

- 2.5- We will use data visualization libraries like D3.js or Chart.js to create interactive charts and graphs for the analytics dashboard. These libraries provide a wide range of visualization options, from simple bar charts and scatter plots where each of the charts represents the relation between the days and the total profit made that day. We will also use SQL or a similar query language to fetch data from the database and calculate key metrics.

3. UR: The employee(cashier) shall be able to view available raw materials and marble stone products along with their information.

System Requirements:

- 3.1- The cashier should have an inventory management system that has the data, such that the inventory is the accounting of things, marbles, and raw materials that the shop uses or sells in its manufacturing process. The system should display all the available materials.
- 3.2- The system should display the available material sorted alphabetically.
- 3.3- the cashier should be able to view the marbles or materials that are available along with a column that contains a description of the material (name of the material, cost by kg, supplier, picture of the material, origin of the material, weight in (kg), color), and status (available, out of stock, on hold). The on-hold term means that the product is sold but still in the warehouse. And the marble description info (name of the marble, cost, supplier, picture of the marble, origin of the marble, weight in (kg), width, length, height)

4. UR: The employee(cashier) shall be able to add a new type of raw materials and marbles to the store or update the product availability from the store so that the system can reflect the product's availability.

System Requirements:

- 4.1- the employee should be able to add a new type to raw material the attributes of raw materials (name of the material, cost by kg, supplier, picture of the material, origin of the material, weight in (kg), color) such as the name and the cost and the supplier are mandatory for adding completion.
- 4.2- the employee should be able to add new marbles the attributes of marbles are (name of the marble, cost, supplier, picture of the marble, origin of the marble, weight in (kg), width, length, height) such as the name and the cost and the supplier are mandatory for adding completion.

- 4.3- The employee(cashier) shall be able to reflect the number of materials available in stock by decreasing the purchased items from the inventory system by selecting sold and how many pieces were deducted from the existing number.
- 4.4- If a material is out of stock the employee should highlight the material as out of stock by deducting the available pieces number to zero.

5. UR: The employee shall be able to search for a specific marble stone product or raw material and check its status and availability.

System Requirements:

- 5.1- The system must show an option scheme that lets the employee choose if he wants to search for a marble product or a raw material.
- 5.2- In search of a marble product, the employee enters the name of the marble product, if there are many marble products with the same product name then show results from different suppliers.
- 5.3- If the marble product was found. The system shall generate a report with its price (New Israeli Shekel) and the materials made from sorted alphabetically and its dimensions (width, length, height) and the supplier and its status (available, out of stock, on hold), and its picture.
- 5.4- If an employee wants to search for raw material the system shall show the raw material with the available quantity of that material and price per (kg) by (New Israeli Shekel) and the color and text description of the material with a picture to it. If no marble product with the name was searched by the employee, then it will show a page with “No marble product was found”. Also, if the raw material doesn’t exist will show the message “material is not available in the shop”.
- 5.5- If no marble product with the name was searched by the employee, then it will show a page with “No marble product was found”. Also, if the raw material doesn’t exist will show the message “material is not available in the shop”.
- 5.6- There will be an option to put this product in the cart of the customer if he wants to buy it.
- 5.7- The search query shall not exceed three seconds to show the page of the marble product.

- 6. UR: The customer shall be able to add marble stone products or a number of raw materials to his cart and review his selections. So he can complete his purchase well.**

System Requirements:

- 6.1- If the quantity of the raw material exists, then the employee shall be able to add quantity as a number in (kg) unit. If the employee enters an invalid quantity, the system shall display a message that shows the quantity in stock. However, if the transaction is finished the product's availability will change to on-hold and the quantity of the material will decrease by the amount purchased.
- 6.2- when the customer ends shopping, the system shall show a review page containing the customer's name and his ID and each product or raw material with their name and types sorted in decreasing price order, and showing the total price of all the purchase and the current date of the transaction. Also, the system should show an option to go back and continue shopping, and another option to confirm the purchase and finally, there should be a free text if he has any notes on the service.
- 6.3- There must be an option for the customer to choose whether he wants to get his items by delivery then the status of the items will be changed to on-hold and an extra fee of 2% will be added to the total price, however, if the customer wants to get his items directly then the quantity purchased will be only reflected.

- 7. UR: The customer shall be able to ask for delivery of the purchased products to a desired location and get a refund if there is any damage.**

System Requirements:

- 7.1- If the customer chooses the delivery option the system shall generate a new page to ask for specifications as the desired location by the city name street name street number house number customer phone and date delivery was booked.
- 7.2- every delivery shall have a shipment's status attribute indicating whether the package was delivered, in progress or marked as lost.
- 7.3- If there are any damages reported by the customer. For each product that was damaged, the system shall be able to access the financial status deduct the price of all products damaged from grand total of the monthly report, and return the deducted price to the customer as a refund for the damaged items.
- 7.4- If there is a delivery that is overdue by 3 days without reaching the customer change its status to delivered. The system shall send a notification message to the delivery employee that there is a delivery that hasn't yet been delivered. If the delivery wasn't delivered for more than a week the delivery is marked as lost and a refund is sent to the customer.

- 8. UR: The customer should have the option to pay using either cash or credit card. To provide flexibility in payment.**

System Requirements:

- 8.1- The customer should have the option to pay using either cash, checks, or credit card. To provide flexibility in payment.
- 8.2- After the customer confirms the purchase. The system shall generate one of the two options either cash or credit card.
- 8.3- If the customer chooses to pay in cash, the transaction will be confirmed and will end the purchase.
- 8.4- If the customer chooses to pay with a credit card, instead of having to enter his card number and passcode along with other specifications (expiration date, cardholder name, and CVC code) he swipes his credit card through a reader to complete the payment. The external requirement is derived from the need for the system to conform to privacy law.

2.3: Effort and Cost Estimation [Group Task]

Assign function points to each function based on its complexity and size, the range is from 1-5, 1 being with little complexity and 5 being most complex.

Identify functionality:

- 1- Employee Registration
- 2- Generate sales
- 3- Marble or raw material catalog browsing
- 4- add to stock
- 5- searching for a marble or a row of material
- 6- Add to cart
- 7- Delivery process
- 8- payment method

Points:

- 1- Employee Registration:2
- 2- Generate sales :1
- 3- Marble or raw material catalog browsing:3
- 4- add to stock :3
- 5- searching for a marble or a row of materials: 5
- 6- Add to cart:4
- 7- Delivery process:4
- 8- payment method:3

Total Points: 25 Points

effort:

we will assume that the conversion factor is 6 Hours per function point

$$\text{Total effort} = \text{Total function points} * \text{Conversion factor}$$

$$= 25 * 6$$

$$= 150 \text{ working hours}$$

Hence, we estimate that our shop will approximately take 150 working hours.

The following table shows the distribution of the functionality points and the number of days.

Table 1: Functions distribution

Feature/User Req.	Function point	Total Function Points (*6)	# of Days
UR1	2	12	2
UR2	1	6	1
UR3	3	18	3
UR4	3	18	3
UR5	5	30	5
UR6	4	24	4
UR7	4	24	4
UR8	3	18	3
Total	25	150	25

3: Requirements Engineering Cont.

3.1: Scenario Analysis [Individual Work]

3.1.1: Scenario of R1 [Made by Mohammad Jarrar]

UR1. The store manager shall be able to add, edit, and remove employee profiles.

Actors:

Assume Participating actors are:

- 1- Store Manager: Adel
- 2- Employee: Majed

A) Add Employee Profile:

Initial assumption:

Adel (store Manager) select “Add new Employee” to add new employee (Majed) to the system. Then the system will show attributes which he can add (name: Majed Kiswani, address: Ramallah: grove street , telephone:05968912 , picture).

Normal Flow:

When Adel selects “Add new Employee” he will add the mandatory attributes which are (name: Majed Kiswani and telephone number: 05968912) of the Majed. The store manager will enter the employee’s first name (Majed) and last name (Kiswani) and telephone number with area code. Adel can optionally add additional information (address: Ramallah: grove street, picture) of Majed. After Adel confirms the entry if the system validates the entered data, then the Majed’s profile will be added to the database, and the Adel will be notified by message said, “Employee added successfully” and Majed will be assigned to a unique id by system (for ex: 5).

What can go wrong:

If Adel tries to end the adding process without entering one of mandatory attributes, the system will print error message ask him to enter data to the field.

If there is validation error such as invalid telephone number format (if number less than 10 digits and doesn’t start with 059) the system will display message.

System state on completion:

Majed’s profile stored successfully in database.
The system will return to previous state, then Adel will be able to add new Employee if he want.

B) Edit Employee Profile:

Initial assumption:

If Adel wants to edit Majed's profile he will select option "Edit Employee Profile" option.

Normal Flow:

Adel typed the unique id (5) of Majed so the system will display his profile on the screen. Adel will be able to modify one or more of Majed's fields (name: Majed Al-Kiswani, address: Ramallah: Salah Deen Street, picture, telephone number: 059824221). Then after selecting save option if the system validates data fields, then system display "Employee information updated successfully".

What can go wrong:

If Adel tries to save options without making any changing the system will display message "no modification was made".

If Adel tries to add non-valid input data field, the system will notify him by message said, "input not valid".

System state on competition:

1-Adel is informed about successful update.

2-The system returns to previous state ready to receive other actions from Adel.

C) Deleting Employee Profile:

Initial assumption:

If Adel want to delete Majed' profile from the system. He will select "Delete Employee" option.

Normal Flow:

After Adel selects "Delete Employee" option, the system will display blank of Employee form. Then to find Majed's profile, Ahed will type Majed's id (5). Then the system will display Adel's information (name, address, telephone number, photo) in the form. If Ahed selects "delete" option, then a delete verification dialog will be displayed on the screen to confirm the deletion. If Ahed select yes Majed's profile will be deleted from the system, and Ahed will get message "Employee profile is successfully deleted".

What can go wrong:

If Ahed didn't confirm the deletion process by enter "no" option, he will go the previous state before the dialog.

If there are any issues with deleting Majed's profile (such as database error). Ahd will get message "Error in deleting the Employee".

3.1.2: Scenario of R3 [Made by Ahmad Ghanem]

UR3. The employee(cashier) shall be able to view available raw materials and marble stone products along with their information.

Actors:

Say that the employee is Ahmad.

Initial Assumption:

Ahmad views the raw materials available at the shop and the system shall display the information of the material when clicked on it (name of the material: sand, cost by kg: \$100, supplier: sandes industry, picture of the material: sand picture, the origin of the material: Jordan, weight in (kg): 100kg, color: brown) as well as its status and status (available, out of stock, on hold) available for example and Ahmad shall choose to view the page of marble stone products of the shop (name of the marble: Quartz, cost: \$20, supplier: Anova Quartz Stone, picture of the marble: quartz picture, the origin of the marble: Palestine, weight in (kg): 10 kg, width:2m, length:1m, height:1m).

Normal Flow:

The employee Ahmad after logging into the system with his credentials, then clicks on the available raw material or the marbles stones button, a loading indicator in a popup appears to the employee while fetching the data stored at a local server at the shop a table shall then be displayed with each raw material or marble stones available with their description next to them as mentioned previously.

Alternative Flow:

The employee Ahmad after logging into the system with his credentials, then clicks on the available raw material or the marbles stones button, a loading indicator in a popup appears to the representative while fetching the data stored at a local server at the shop a table shall then be displayed with each raw material or marble stones then the employee clicks on the "Status" filter and choose the out-of-stock items, then the system adapts to the changing filter by showing a progress bar for couple seconds and then reflect the changes to the table, Ahmad then reviews the out-of-stock items only along with their information.

Error Flow:

After Ahmad logs into the system and tries to click on available marble stones or raw materials The system displays a loading indicator while attempting to fetch data, the system faces an issue while retrieving the data from the server due to a server error, then the system

displays a pop-up message “ERROR RETRIEVING THE DATA, Please try again later” then a Retry button and repeats the process after logging into the system if the issue persists Ahmad contacts the technical support for the issue.

Error Flow:

The employee Ahmad after logging into the system with his credentials, then clicks on the available raw material or the marbles stones button, a loading indicator in a popup appears to the employee while fetching the data stored at a local server at the shop, the network connection is lost and the system is unable to retrieve the data due to network issue, the system shall pop an error window "Network Error: Unable to fetch data. Please check your connection and try again.", Ahmad checks the connection and clicks on try the button “retry again” and repeats the process after logging into the system if the issue persists Ahmad contacts the technical support for the issue.

System State on Completion:

Ahmad is still logged in to the system, the system showed the item with its details with filters and without and handled the flows and errors and the system is ready for another action to be performed.

3.1.3: Scenario of R4 [Made by Abdullah Naser]

UR4. The employee(cashier) shall be able to add a new type of raw materials, add new marbles to the store or update the product availability from the store so that the system can reflect the availability of the product.

Actors:

Assume that the actor is the employee Abdullah.

Initial assumption:

Abdullah (the employee) wants to add a new inventory item such as raw materials or marble stones to the system. He will enter relevant details including (item name, cost, supplier, image of the material, origin of the material, weight, and dimensions) ensuring accurate inventory management within the system.

Normal flow:

The employee Abdullah, after logging into the system with his credentials, navigates to the inventory section and then clicks on add new raw material or clicks on add new marble stone to the shop. A new window is opened to add new raw material or new marble stone. He enters the new data to the shop as follows for example for raw material he should add (name of the material: sand, cost by kg: \$100, supplier: sandes industry, picture of the material: sand picture, the origin of the material: Jordan, weight in (kg): 100kg, color: brown) and could change the status or if he wants to add a marble stone shop (name of the marble: Quartz, cost: \$20, supplier: Anova Quartz Stone, picture of the marble: quartz picture, the origin of the marble: Palestine, weight in (kg): 10 kg, width:2m, length:1m, height:1m). and the entered data was correct and had no flows then the system database reflected the material added along with its availability status.

What can go wrong:

Abdullah's entered item is a duplicate and already exists in the system, the system shall display an error message "Error: item already exists" and Abdullah should recheck his entered item and if necessary, update an existing item rather than adding a new one.

What can go wrong:

Abdullah entered the data in an incorrect format (e.g., text in a numerical field, invalid picture file type), and the system should display an error message. "Error: data field– is an incorrect format" and Abdullah should recheck his entered Field values and try again.

System state on completion:

The system added new raw material or new marble stone to the database and after clicking on add pop up message shall pop up “item is added successfully” and Abdullah can see his changes reflected in the database.

3.1.4: Scenario of R6 [Made by Wasim Atta]

UR6. The customer shall be able to add marble stone products or a number of raw materials to his cart and review his selections. So, he can complete his purchase well.

Actors:

Assume that the actor is the Employee (Wasim)

Initial assumption:

The employee can review all the selected items to be purchased before he confirms the purchase and proceeds with the payment.

Normal flow:

Wasim logs into the system and navigates through the available marble stones or raw materials and if a customer(rami) wants to purchase an item he adds the item to the cart along with the desired quantity and the system shall display a pop-up message “added to the cart with the details of the product(name, cost, supplier, origin, weight, dimensions)” when the rami finish shopping Wasim clicks view cart button to finalize the purchase and inspect the added items to the cart so he can confirm that all items are added successfully.

What can go wrong:

Wasim logs into the system and navigates through the available marble stones or raw materials and if a customer(rami) wants to purchase an item he adds the item to the cart along with the desired quantity the system displays a pop-up message indicating that “product out of stock and cannot be added to cart” and return to the normal flow.

What can go wrong:

Wasim logs into the system and navigates through the available marble stones or raw materials and if a customer(rami) wants to purchase an item he adds the item to the cart along with the desired quantity then Wasim reviews the cart contents and then the rami want to make a change either to the quantity or the items in the cart if he wants to remove an item for example then he clicks on the delete button next to the item and the system shall reflect the changes and the new total cost.

System state on completion:

After successfully reviewing the cart and modifying if needed the system shall have the updated cart with all the desired items and the total cost of them.

3.2: Actors and Use case (Diagram) Analysis & Modelling [Group Work]

Actors:

1. Store Manager
2. Employee
3. GPS System

Use Case Diagram:



Figure 1: Use-Case diagram of the system

3.3: Use Case Description [Individual Work]

3.3.1: Use Case Specification for R5 [Made by Mohammad Jarrar]

Brief Description:

This use case makes the employee has the ability to search for a specific marble stone product or raw material in the system and check its status and availability.

Flow of Events:

2.1: Basic Flow- Search Marble Stone product or raw material

- 1- Employee chooses between searching for Marble product or Raw material by either selecting option “Search for product” to search marble stone product or “Search Raw Material”.
- 2- If employee choose “Search product”, the system will show search page and allowing employee to input a field to search by (name of product , supplier (if name not found or multiple similar product names)).
- 3- If employee choose “Search Raw Material”, the system will show search page for raw materials allowing employee to input a field to search by, which is the type of raw material.
- 4- For both cases, the employee select “Search” option after specify the search fields.
- 5- The system will find matching product or raw materials based on specified fields. Then display there information . For marble product the system will display (name, cost, supplier , picture, origin, weight in (kg), width , length , height) of the marble product. However, for if raw material was found after searching the system will display (type (name) , cost by kg , supplier , picture, origin, weight in kg, color) for raw material. Finally , for both the system will display their status and availability information.
- 6- The employee reviews the search results to find the desired product or raw material.

2.2: Alternative Flows:

- 1-If the system doesn't find any marble product or raw material match the search fields, the system will display message “no result found”.
- 2- The employee can modify search fields, or cancel the search operation.

Special Requirements:

The system should retrieve search results within a maximum of 2 seconds to ensure efficiency performance.

Entry Conditions:

Before accessing this use case, the employee must login to the store system.

Exit Conditions:

There are no specific exit conditions with this use case, since there is no modifying data, only searching information of raw material or marble product.

3.3.2: Use Case Specification for R6 [Made by Ahmad Ghanem]

UR6. The customer shall be able to add marble stone products or a number of raw materials to his cart and review his selections. So, he can complete his purchase well.

Use case specification:

Brief Description:

The employee reviews the added items to the cart before proceeding to the purchase phase.

Flow of Events:

2.1: Basic Flow- Add Marable Stone Products or Raw Materials

- 1- If the added quantity as a number in (kg) unit to the cart exists
- 2- The system shall add the items to the cart and after the purchase is complete it shall deduct the amount of purchased items.
- 3- If the customer ends his shopping and the employee clicks on the go-to-cart page
- 4- The system shall display a page containing the ID and each stone or raw material with their name and types sorted in decreasing price order and showing the total price of all the purchases and the current date of the transaction. Also, the system shall show an option to go back and continue shopping, and another option to confirm the purchase and finally, there should be a free text if he has any notes on the service.
- 5- At the cart page there should be a check box specifying whether he wants to have the items directly or on delivery.
- 6- Then the system shall change the status of the items to on-hold and an extra fee of 2% will be added to the total price, however, if the customer wants to get his items directly then the quantity purchased will be only reflected.

2.2: Alternative Flows:

- 1- if the quantity that the customer wanted didn't exist
- 2- the system shall display a popup message "The quantity is insufficient for this order"

Special Requirements:

- The font of the system must be very clear to the employee from a distance of 1,5 meters
- The add to cart page should be clear of all the items in it and all the quantities and the corresponding prices

Entry Conditions:

There is no entry condition the employee shall be able to view the cart on any page of the system.

Exit Conditions:

The exit conditions of this use case are that the purchased items are listed for the employee to see and after submitting the purchase we should go to the payment page.

3.3.3: Use Case Specification for R7 [Made by Abdullah Naser]

UR7. The customer shall be able to ask for delivery of the purchased products to a desired location and get a refund if there is any damage.

Use Case Specification:

Brief Description:

This use case enables the customer to ask for delivery after purchasing the item from the shop, and he can file a claim that the delivered item is damaged.

Flow of Events:

2.1: Basic Flow – Customer is Asking for Delivery

- 1- After finishing the purchase process.
- 2- The system shall display an option on whether the customer wants to have the products on-site or deliver them.
- 3- If the employee chooses the option to deliver
- 4- The system shall open a new page that the employee fills in specifying the details of the delivery as the desired location by the (city name, street name, street number, house number), customer phone, and the date on which the delivery was booked.
- 5- If the employee selects the delivery option
- 6- The system shall attach the ID of the items to be delivered to items to be delivered to the delivery company and send it automatically.
- 7- After the delivery is complete if the customer reaches out to the employee to report any issues or damages that happened
- 8- The employee should notify the system that there is damage to the item and the system shall open a new window to fill out the details of the damage.

2.2: Alternative Flows:

- 1- If the delivery is overdue the expected delivery date
- 2- The system shall automatically send a notification to the employee that there is a package that has an overdue date of 3 days, and the employee should reach out to the delivery company.
- 3- If the delivery is overdue by an entire week after the expected delivery date
- 4- The system shall automatically send a notification to the employee that the package is marked as lost and send the package ID to the lost packages category and the employee should file a refund form if the customer requested a refund.
- 5- If the customer wants to change the delivery location, he/she calls the employee to change the delivery location.
- 6- The employee navigates to the order settings and the system prompts the employee to change the delivery location.

- 7- Then he enters the new location and clicks update, then the system shall update the delivery location, and confirm the change to the delivery company.

Special Requirements:

- 1- The font of the system must be very clear to the employee from a distance of 1.5 meters.
- 2- The system should be connected to the delivery system to be able to track the items in real time as they are delivered.

Entry Conditions:

Before accessing this use case, the employee must complete the purchase process.

Exit Conditions:

The exit conditions of this use case are that the purchased items are listed to the delivery company.

3.3.4: Use Case Specification for R8 [Made by Wasim Atta]

UR8. The customer should have the option to pay using either cash or credit card. To provide flexibility in payment.

Use case specification:

Brief Description:

The customer should have the option to pay using either cash, checks, or credit card. To provide flexibility in payment.

Flow of Events:

2.1: Basic Flow

- 1- After finishing the purchase at the cashier the customer
- 2- The system shall generate one of the two options either cash or credit card.
- 3- If the customer chooses to pay in cash.
- 4- The employee should choose cash and the transaction will be confirmed with the amount and will end the purchase.
- 5- If the customer chooses to pay with a credit card
- 6- The employee should choose the credit card and the transaction instead of having to enter his card number and passcode along with other specifications (expiration date, cardholder name, and CVC code) he swipes his credit card through a reader to complete the payment. The external requirement is derived from the need for the system to conform to privacy law

2.2: Alternative Flows:

- 1- If the delivery is overdue the expected delivery date
 - a. The system shall automatically send a notification to the employee that there is a package that has an overdue date of 3 days and the employee should reach out to the delivery company.
 - b. If the delivery is overdue by an entire week after the expected delivery date the system shall automatically send a notification to the employee that the package is marked as lost and send the package ID to the lost packages category and the employee should file a refund form if the customer requested a refund.
- 2- If the customer wants to change the delivery location
 - a. The customer calls the employee to change the delivery location the employee navigates to the order settings and the system prompts the employee to change the delivery location then he enters the new location and clicks update, then the system shall update the delivery location, and confirm the change to the delivery company

Special Requirements:

- 1- The font of the system must be very clear to the employee from a distance of 1,5 meters.
- 2- The system should be connected to the delivery system to be able to track the items in real time as they are delivered.

Entry Conditions:

Before accessing this use case, the employee must complete the purchase process.

Exit Conditions:

The exit conditions of this use case are that the purchased items are listed to the delivery company.

3.4: Activity Diagram Analysis & Modelling [Group Work]

The following is the main activity diagram that shows the overall system use-cases and how the interaction occurs between them in order to achieve main business service.

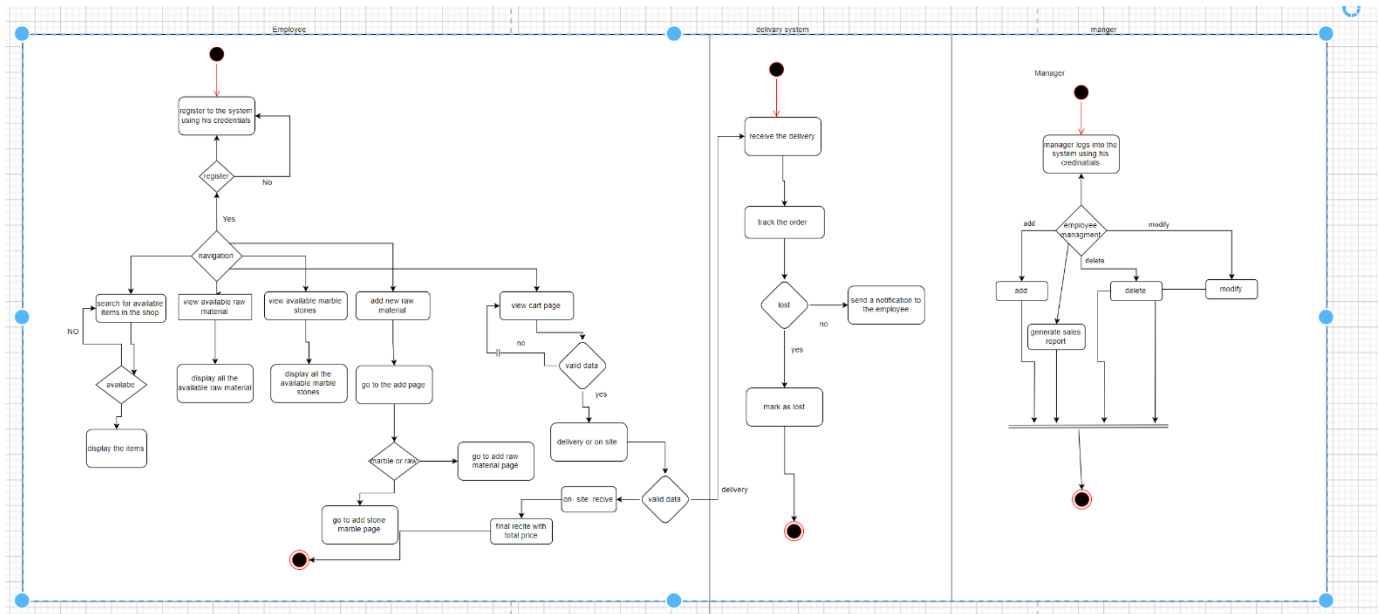


Figure 2: Activity Diagram of the system.

3.5: Instance Activity Diagrams [Individual Work]

3.5.1: Instance Activity Diagram for R5 [Made by Mohammad Jarrar]

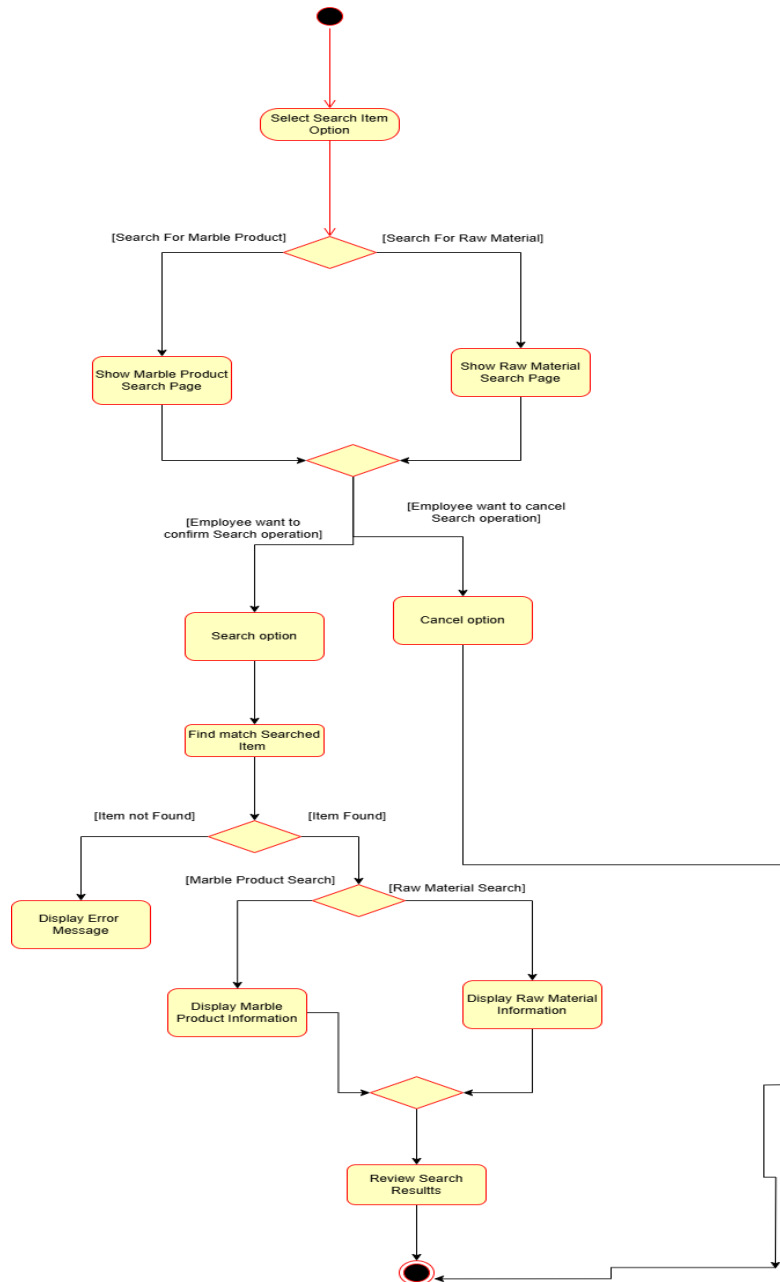


Figure 3: Instance Activity Diagram of R5 [by Mohammad Jarrar]

3.5.2: Instance Activity Diagram for R7 [Made by Abdullah Naser]

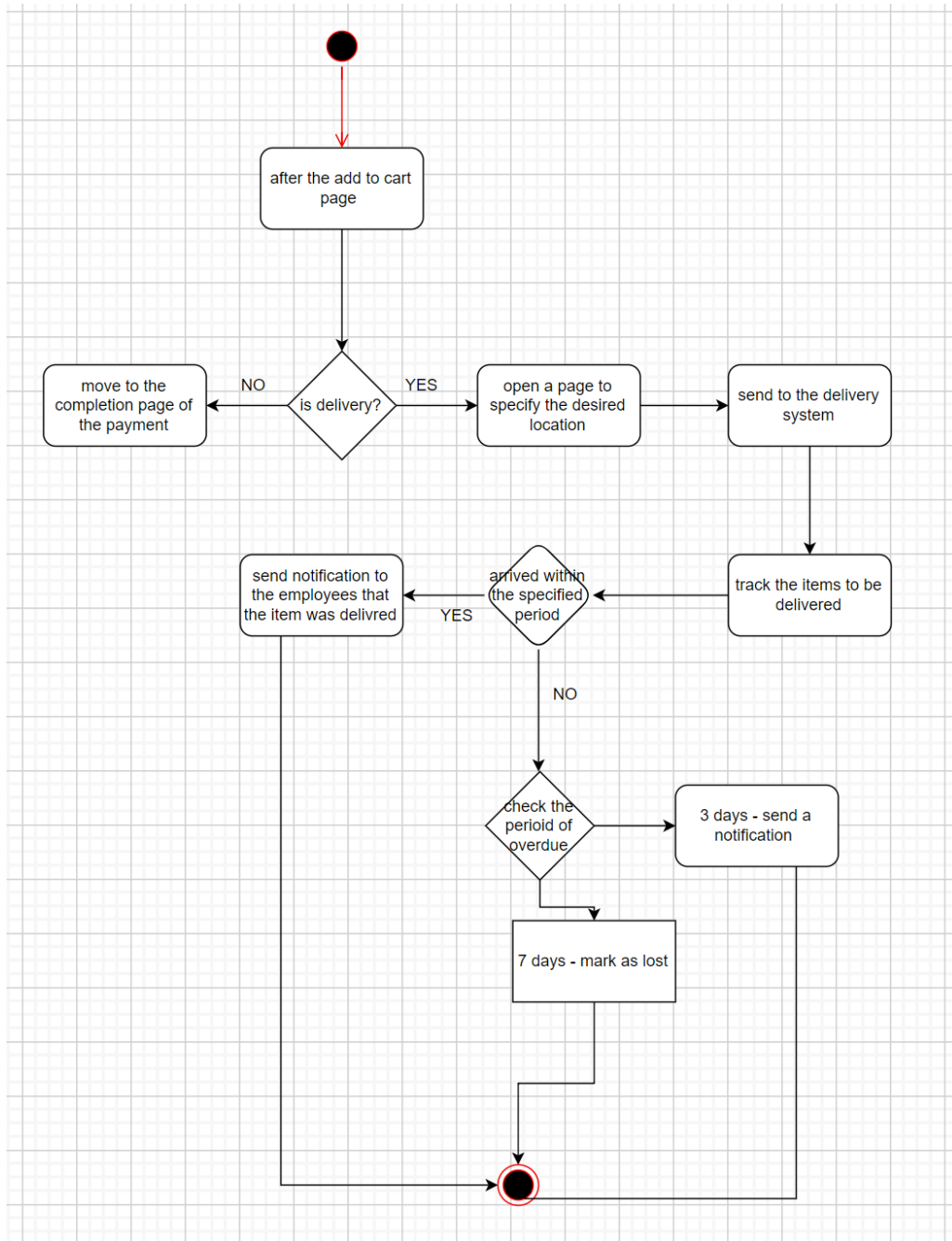


Figure 4: Instance Activity Diagram of R7 [by Abdullah Naser]

3.5.3: Instance Activity Diagram for R8 [Made by Wasim Atta]

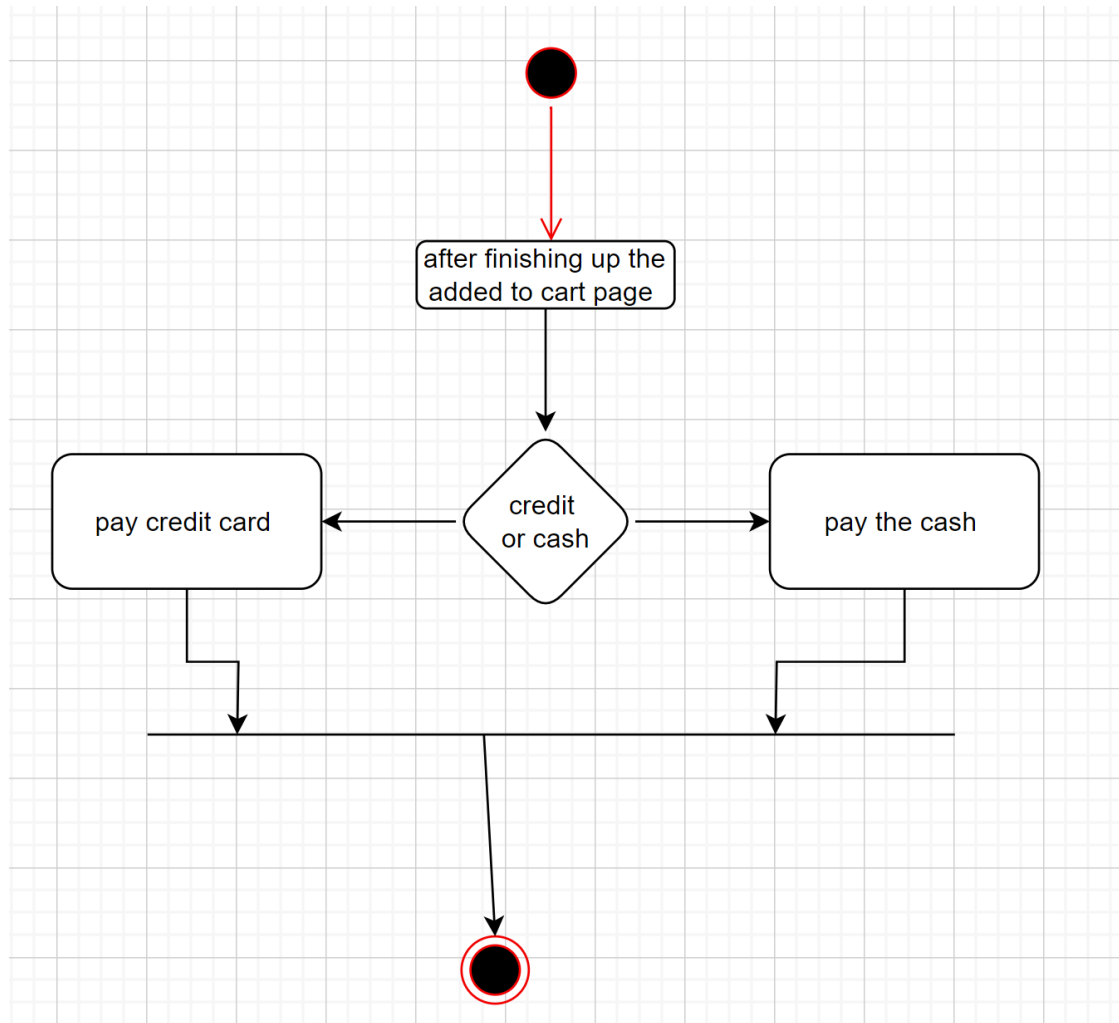
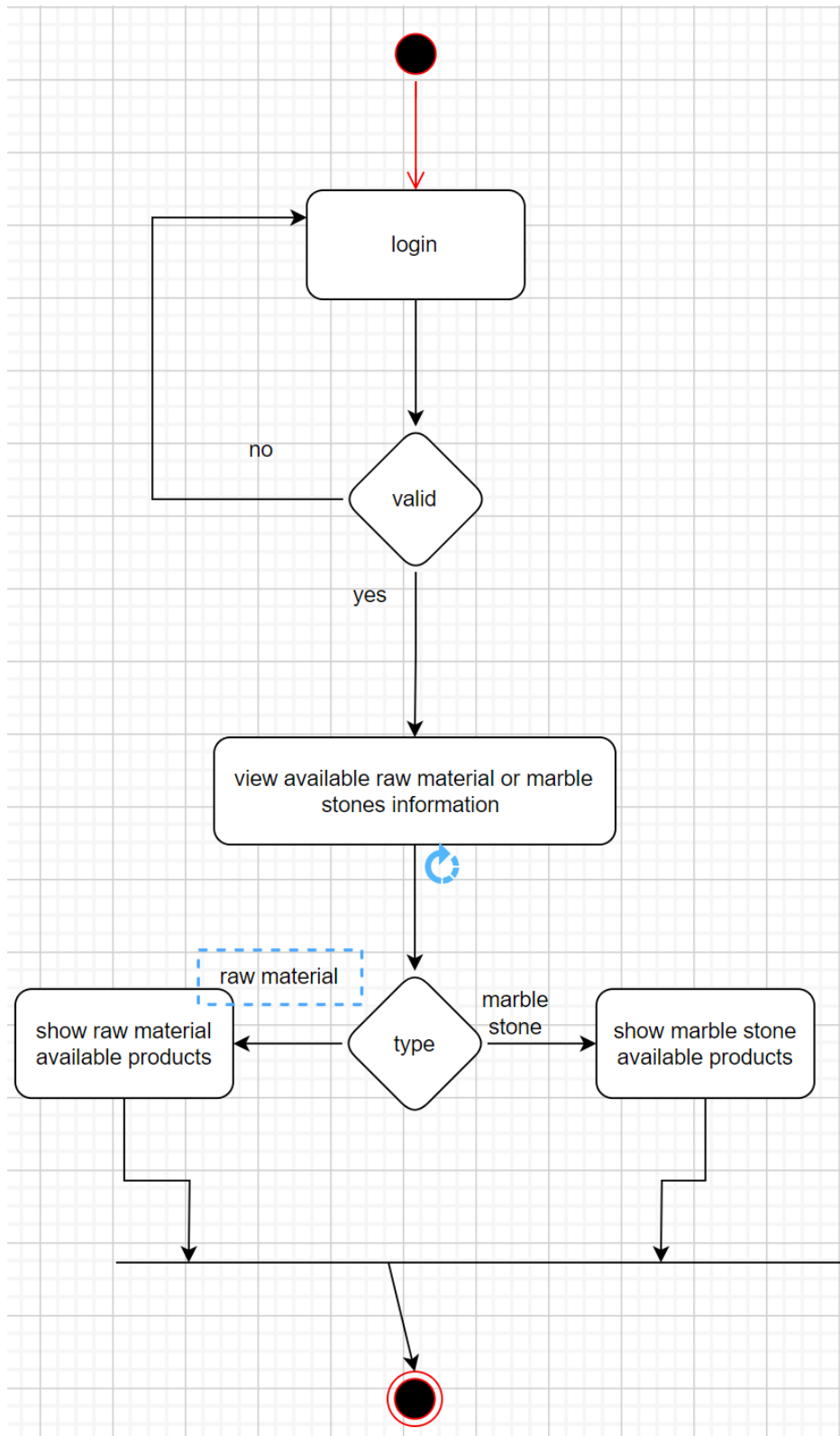


Figure 5: Instance Activity Diagram of R7 [by Wasim Atta]

3.5.4: Instance Activity Diagram for R6 [Made by Ahmad Ghanem]



4: System's Modeling and Architecture

4.1: Class Diagram Analysis [Group Task]

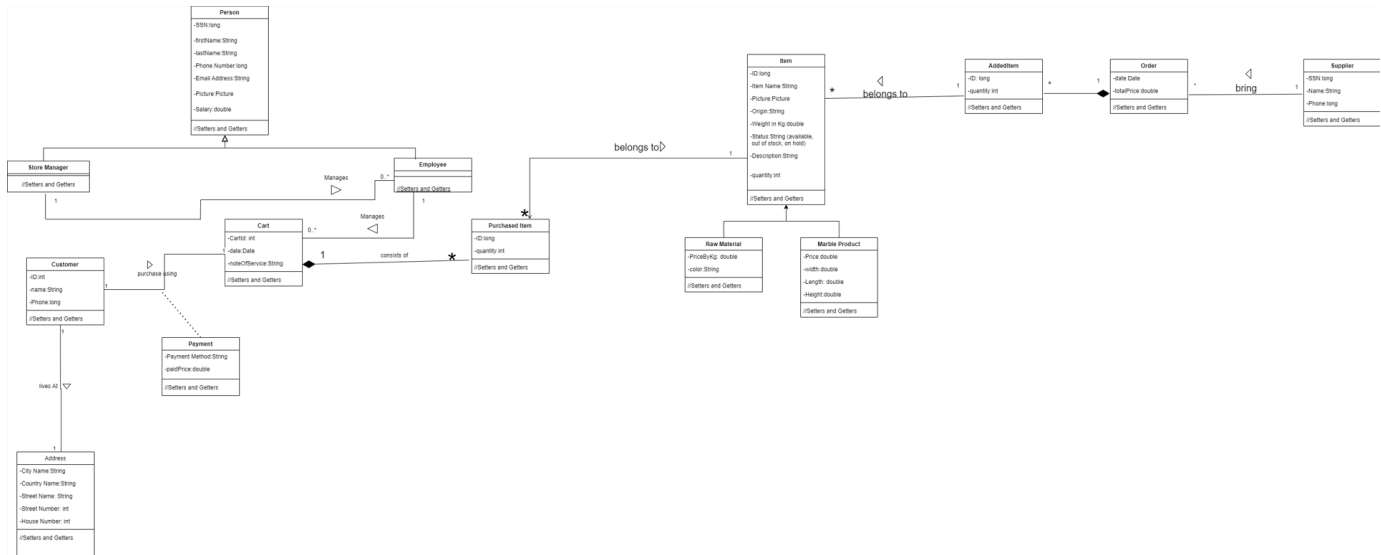
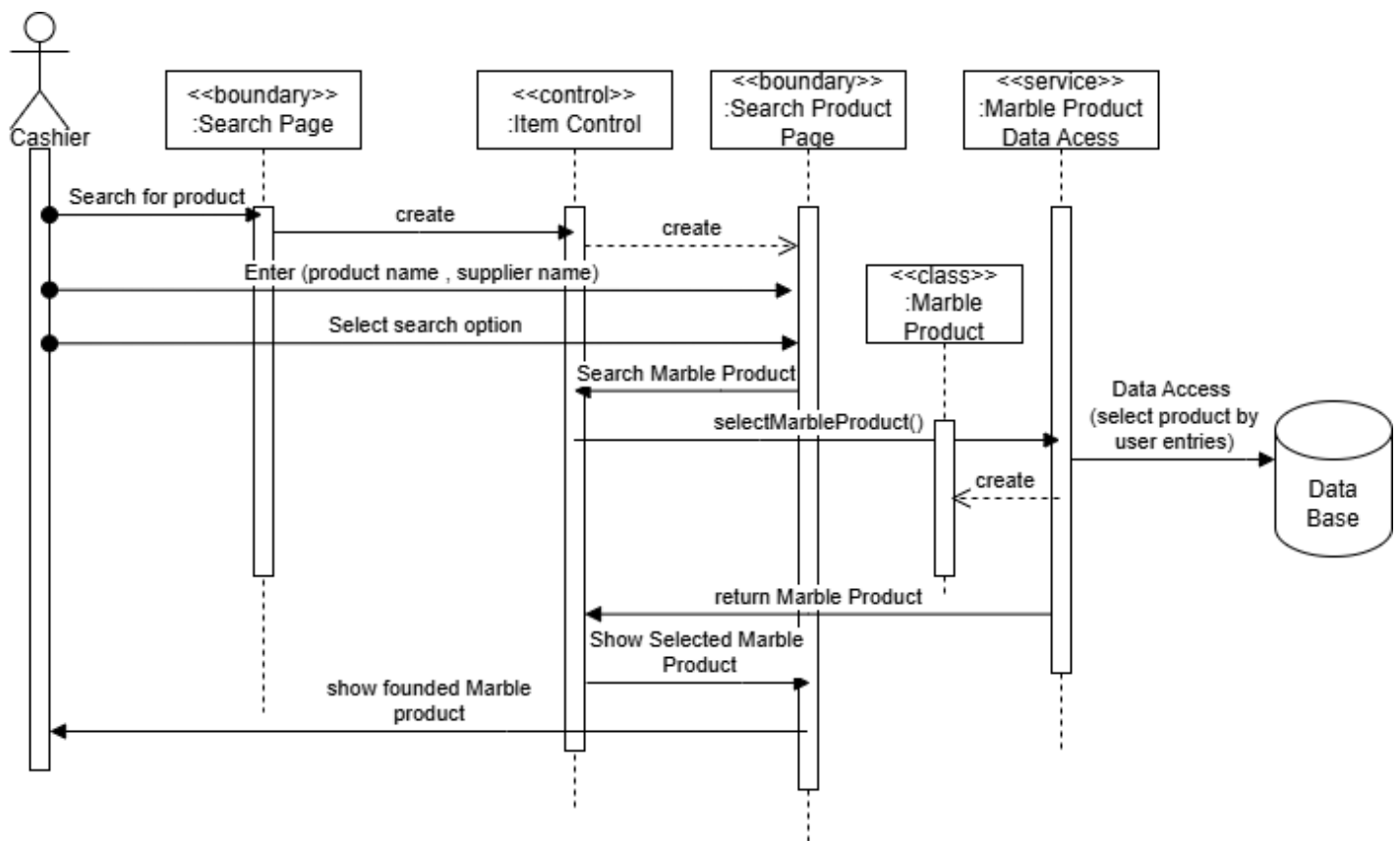


Figure 6: System Class Diagram

4.2: System Sequence Modelling [Individual Work]

4.2.1: Sequence Diagram of R5 [Made by Mohammad Jarrar]



Actors: Employee

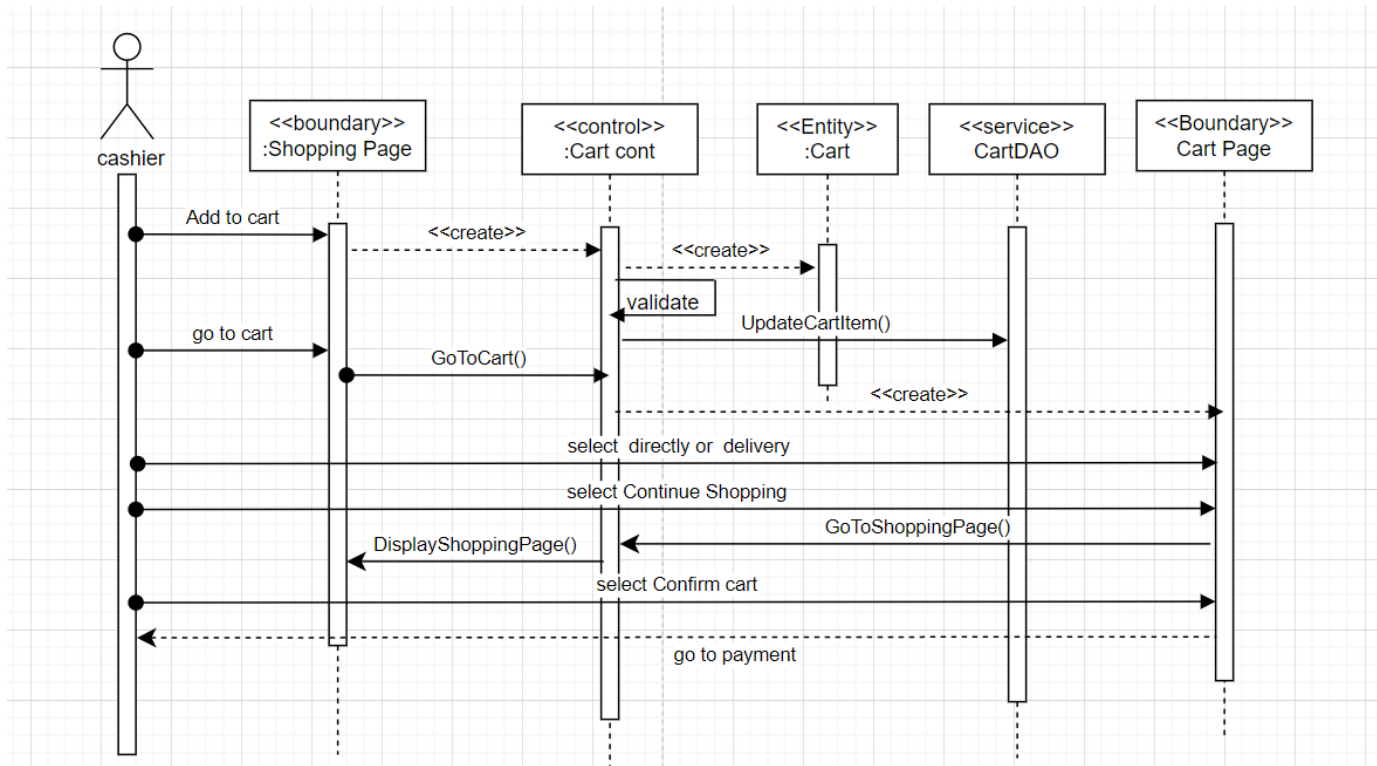
Views: Search Page, Search Marble Product Page

Entities: Marble Product

Services: Marble Product Data Access

Controller: Item Controller

4.2.2: Sequence Diagram of R6 [Made by Ahmad Ghanem]



Actors: Cashier(Employee)

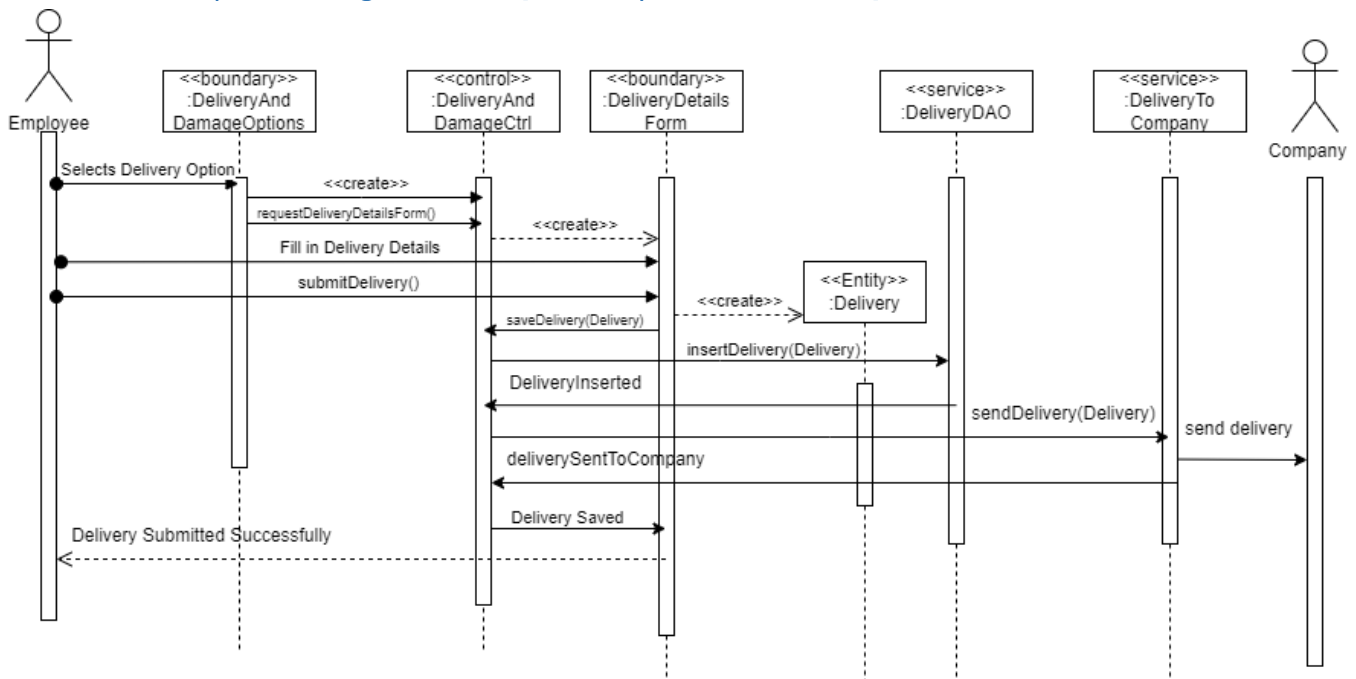
Views: ShoppingPage, CartPage

Entities: Cart

Services: CartDAO

Controller: CartController

4.2.3: Sequence Diagram of R7 [Made by Abdullah Naser]



Actors: Employee, Company

Views: DeliveryAndDamageOptionsView, DeliveryDetailsForm

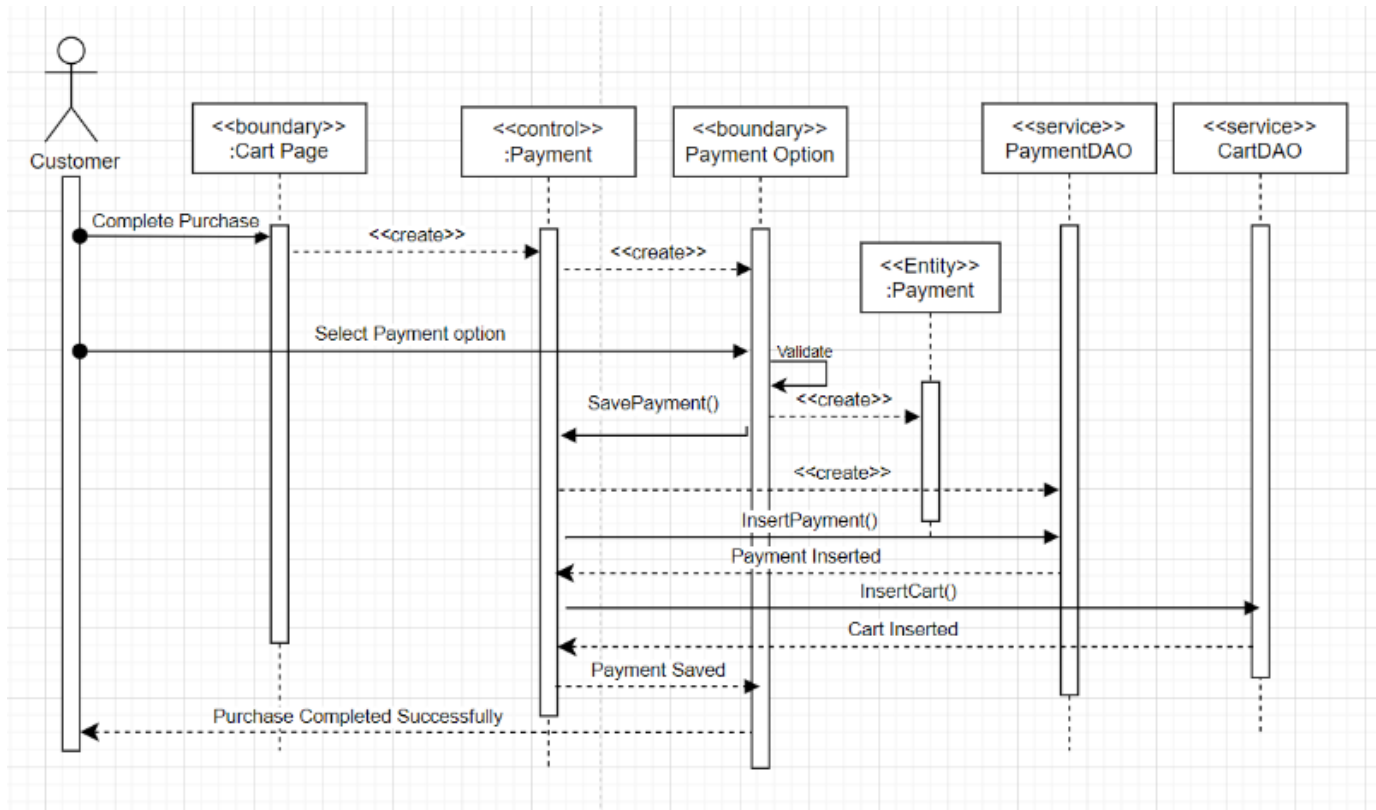
Entities: Delivery

Services: DeliveryDAO, DeliveryToCompanyService

Controller: DeliveryAndDamageController

Note: The basic flow is asking for delivery to a desired location and refund if there is damage. But the sequence diagram captures only the delivery's flow since if I added the damage, it would make the sequence diagram redundant without any additional information to the developers. The goal of this sequence is to make the developers follow the SOLID and MVC design principles and know how the dynamic behavior of the code will be.

4.2.4: Sequence Diagram of R8 [Made by Wasim Atta]



Actors: Customer

Views: PaymentOption, CartPage

Entities: Payment

Services: PaymentDAO

Controller: PaymentController

4.3: System Design Goals [Group Task]

- **Low coupling:** The components should have the least dependency on each other, so that the system will be more maintainable.
- **High cohesion:** Each component should have functions with the highest dependencies to each other, and all should have a single purpose.
- **High performance:** The cashier should be able to retrieve every marble stone or material information with a time not exceeding three seconds.
- **Reliability:** the cashier should be able to continue updating the system even the network/back-end server go down.
- **Security:** The system's inner components must be protected, so we want to hide inner data from the end-user.

4.4: System Component Diagram [Group Task]

4.4.1: Components Description

A- User Interface Layer:

1- Cashier UI:

This component give service to the end-user(cashier) by manage the user-interface for the cashier. Also, this component request services from Sales Manager and order Management components in order to handle cashier interactions (add, update, delete or sell any marble stone product or raw materials) and get functions of business logic for the items in the shop.

2- Manager UI:

This component gives service for the manager of the store by manage the description and showing the data to the manager. This component request services from Manager Logic component that handle manager interactions (add, update, delete on any employee, and see reports) and access services that do the business logic in Manager Logic component.

B- User Interface Management Layer:

1- Sales Manager:

This component gives services to Cashier UI by handling some of cashier interactions with UI, and have functions that will update on the products or raw materials in sale operations of the customers by requesting services from Data Access on Data Base component. Also, will request Local Data Access component if there is any drop in the back-end of the main server. Finally, at the begging will access Cashing Items components to get all important information for higher speed access to the server.

2- Order Manager:

This component also gives services to the Cashier UI by handling some of cashier interactions with UI, and have functions that will update on the marble stones and raw materials in adding any order to the store by requesting services from Data Access on Data Base component that will access data base server.

3- Manager Logic:

This component gives services for Manager UI by handling manager interactions with UI, by (add, delete, update) an employee or showing reports of the shop. This component can access the database using the Data Access on Data Base component.

C- Data Access Management Layer:

1- Data Access on Data Base:

This component gives services for all the component in User interface management layer. Since this component is the main component that is responsible to access the back-end data base component. This component request services from back-end of database component to access database logic. Also, this component provides services for Data Synchronization component to transfer data to the Local Data Access component and also transfer data to the Cashing Items component.

2- Local Data Access:

This component is made in order to achieve Reliability design goal by making Local Data base that will be accessed if the main Back-end data base got dropped. So that this component will give service to Sales Manager component that just responsible to access marble item or raw materials in customer sales. This component will synchronize and get the data from Data Access on Data Base component using the component Data Synchronization.

3- Data Synchronization:

This component was made to synchronize data between the local data base and main data base by requesting services from both components in order to synchronize data between the two components.

4- Cashing Items:

This component was made in order to achieve Performance design goal by cashing the items quickly when system is open. So that, this component gives services to Sales Manager component at the begging to cashed items by requesting services from Data Access on Data Base component that will access the main data base component.

C- Back End Data Base Management Layer:

1- Data Access on Data Base:

This component will request services directly from the database server, and give services to Data Access on Data Base component that will request services as operations on the database.

2- Local Data Base:

This component will request services directly from the local server, and give services to Local Data Access component that will request services as operations on the local data base.

Therefore:

Using components with each single responsibility achieve the design goals low coupling and high cohesion. Also using local data access and local data base to achieve high speed response, these components are made to achieve performance design goal. Reliability in our project will be achieved using Cashing items component, that will cash items when pc open. Finally, the security design goals is achieved using layered structure that hide inner components.

4.4.2: Component Diagram

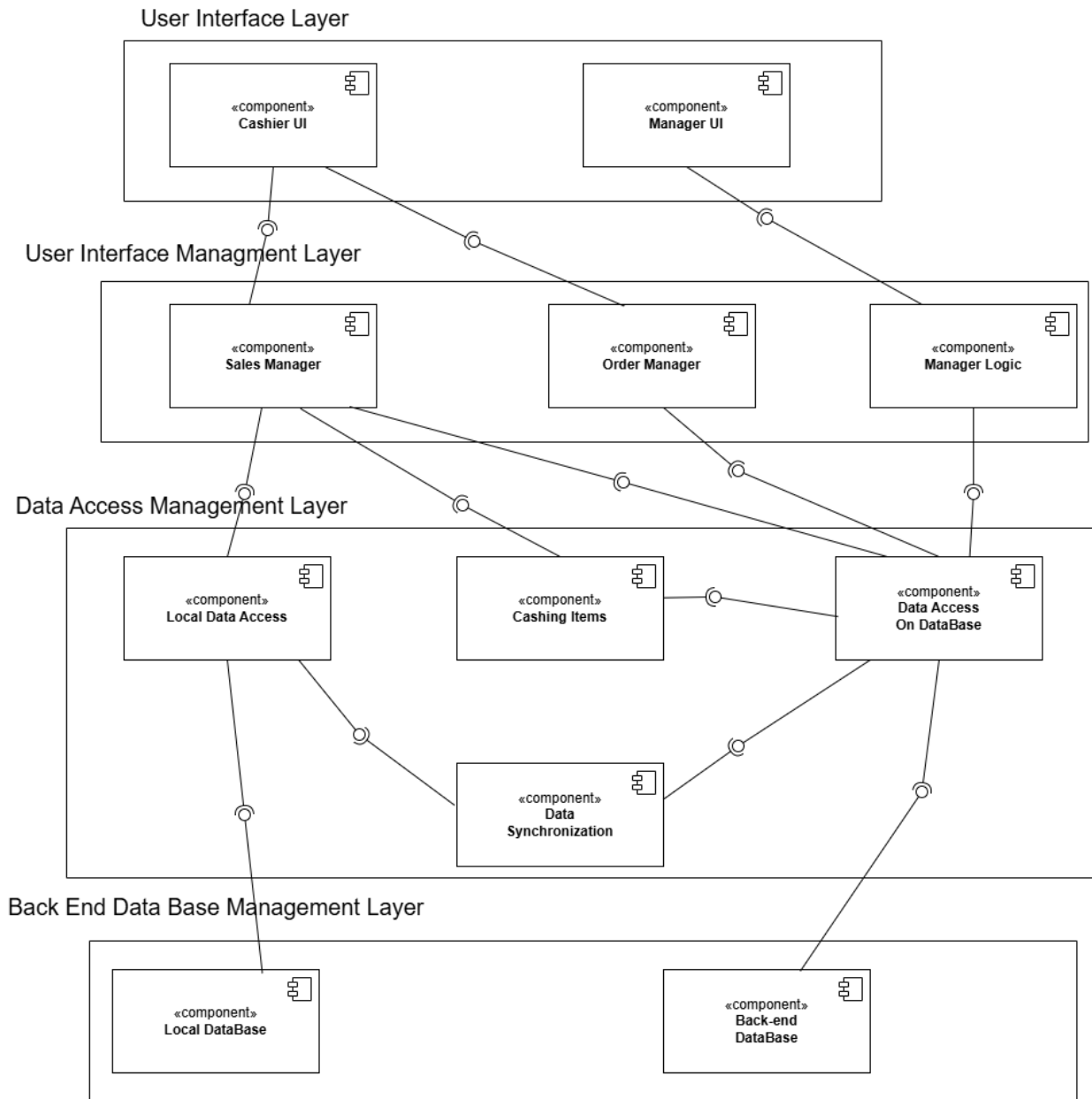


Figure 7: System Component Diagram

4.5: System Deployment Design [Group Task]

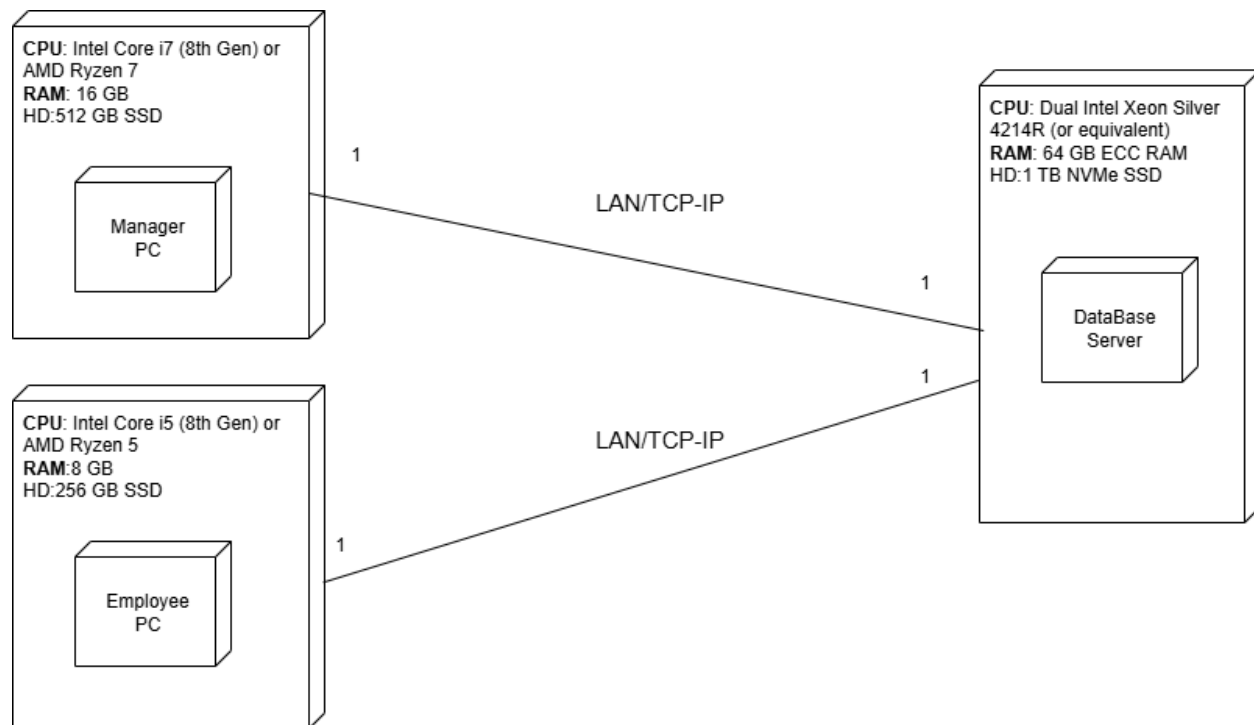


Figure 8: Deployment Diagram