

052400-1 VU Information Management and Systems Engineering (2025S)

Milestone 1

*(Please note: submission deadline: Tue 11.04.2025 13:00)*

Group 11

Student 1: Fahimy, Mustafa, matriculation number

Student 2: Nikzad, Khalifa, 12437813

*27.03.2025*, Vienna

**Contents**

[1](#_heading=h.13mf2v5qhhgj) Milestone 1 2

[1.1](#_heading=h.13mf2v5qhhgj) Team - Conceptual Modeling 2

[1.1.1](#_heading=h.s03ro0wt1fa8) Describe the Application Domain 2

[1.1.2](#_heading=h.ea5cq1u61sbz) Logical Design – ER Diagram in Chen Notation 2

[1.1.3](#_heading=h.ac7g6gje2los) Relational Modeling – SQL CREATE Statements 2

[1.2](#_heading=h.4bgm1ge0l1bs) Individual - Student 1 3

[1.2.1](#_heading=h.8yke9dc35bx5) Use Case Definition and Design 3

[1.2.2](#_heading=h.rfrga0qfa54v) Analytics Report 3

[1.2.3](#_heading=h.zfqa0owlzhe0) NoSQL Design 3

[1.3](#_heading=h.tlxymrnp9pth) Individual - Student 2 4

[1.3.1](#_heading=h.tu2pgtpyrzw6) Use Case Definition and Design 4

[1.3.2](#_heading=h.as692n11sj30) Analytics Report 4

[1.3.3](#_heading=h.4f3r7gipw15s) NoSQL Design 4

# Milestone 1

## Team - Conceptual Modeling

### Describe the Application Domain

In our system, **Users[ name,email,phone,password,...]** serves as a **supertype**, representing all individuals who interact with the platform, including both **Customers[Prefered Contact Method, Loyalty Points]** and **Employees[Job title, Salary, Hire Date, Shift]**. **Customers** are **users** who book **repair services[service\_name (e.g., Battery replacement), description,price,time\_taken]**, browse available repairs, and manage their **repair appointments[date\_time,status (Pending, Completed, Cancelled), total\_price (Final price for all selected services)]**.

On the other hand, **Employees** are **users** responsible for managing **repair appointments**, updating repair statuses, and ensuring smooth service execution. **Employees** may include **technicians**, who perform the actual repairs, **customer support staff**, who assist users with inquiries, and **admin** that manage overall the system. While both Customers and Employees share common attributes such as **name, email, phone, and password**, Employees have additional permissions, allowing them to **modify repair details, update repair appointments statuses, and manage payment[amount,status (Paid, Unpaid), payment\_method, payment\_date] processing**. This structured user hierarchy ensures clear role differentiation while maintaining an efficient repair management workflow.

The system supports different **device types[type\_name (e.g., Smartphone, Tablet…),description]**, each associated with **multiple brands[brand\_name (e.g., Samsung, Apple), country (Country of origin), founded\_year]**. Every brand, in turn, offers a variety of **device models[model\_name (e.g., Galaxy A55), release\_year (Year the model was released)]**, ensuring that **customers** can find the correct **repair service** for their specific device. Each **repair service** is tied to a particular **device model** and includes essential details such as **service name, description, price, estimated time for repair**. **Users** can select **multiple repair services** (e.g., battery replacement and screen repair) for the same device in a single **repair appointments**. Additionally, the system allows **users** to choose from different **service methods[method\_name (Visit Store, Send Device, Pickup), estimated\_time, cost],** but only **one service method** should be selected. The estimated time and cost of the service may vary depending on the selected method.

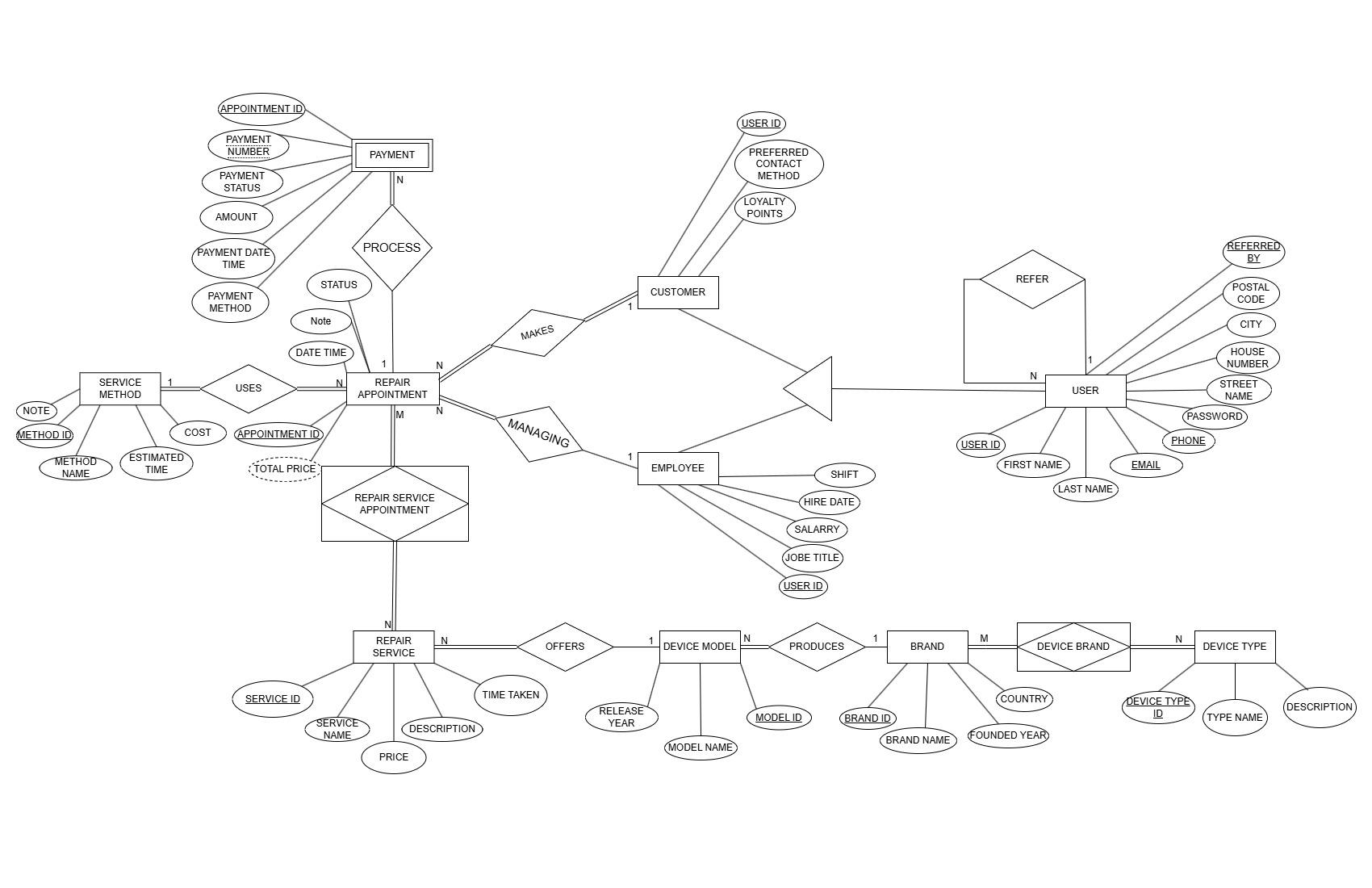
The **repair appointments** process includes selecting a **date and time** for the repair, ensuring convenience for the **users**. Once a **repair appointments** is confirmed, it is assigned a **status** (e.g., Pending, Completed, Canceled) that updates throughout the service process. **Users** also receive information about the **total cost** of the selected services and can proceed with **payment** using various methods such as **credit card, PayPal, or cash on delivery**.

The system maintains an organized structure of its relationships: **Users** can **place multiple repair appointments,** and each repair appointments can include **multiple repair services**. Additionally, the system supports an **administration role** where **admins** can **manage repair services,** update pricing, and oversee **repair appointments**. This structure ensures an efficient, user-friendly experience while allowing service providers to maintain a well-organized repair management system

After the description of the application domain using color-coding (see guidelines):

* List all entities and all relationships (fill in the table on the previous page)
* For the weak entity: List the weak entity, the strong entity, and the identifying relationship between weak and strong entity

### Logical Design – ER Diagram in Chen Notation



### Relational Modeling – SQL CREATE Statements

1. **User**: user\_id, first\_name, last\_name, email, phone, password, street\_name, house\_number, city, postal\_code, referred\_by
2. **Customer**: user\_id, preferred\_contact\_method, loyalty\_points
3. **Employee**: user\_id, job\_title, salary, hire\_date, shift
4. **Repair Appointment**: appointment\_id, customer\_id, employee\_id, method\_id, date\_time, status, total\_price
5. **Repair Service**: service\_id, service\_name, description, price, time\_taken
6. **Repair Service Appointment**: service\_id, appointment\_id
7. **Service Method**: method\_id, method\_name, estimated\_time, cost, note
8. **Payment**: appointment\_id, payment\_number, amount, payment\_status, payment\_method, payment\_date\_time
9. **Device Type**: device\_type\_id, type\_name, description
10. **Brand**: brand\_id, brand\_name, country, founded\_year
11. **Device Type Brand**: device\_type\_id, brand\_id

## Individual - Student 1

Student 1: last name, first name, matriculation number

### Use Case Definition and Design

State if you are using version 1 (weak entity) or version 2 (IS-A)

#### Textual Description

##### *Title (weak entity/IS-A relationship)*

**Trigger**:

**Preconditions**:

**Main Flow**:

**Postconditions**:

**Entities**:

#### Graphical Representation

<Insert your diagram here>

### Analytics Report

#### Concept

After describing your analytics report, list the three entities involved.

#### Proof of Concept

Add screenshots and descriptions of executing your analytics report.

### NoSQL Design

List entities involved in the use case (use case, not report[!]) in order to improve readability.

#### Design Overview

#### Expected Execution and Possible Changes

#### Five Rules of Thumb

## Individual - Student 2

Student 1: Nikzad, Khalifa, 12437813

### Use Case Definition and Design

I am using version 2 (IS-A).

#### Textual Description

##### **Use Case Name**: Book Repair Appointment (I*S-A relationship)*

**Trigger**: The customer selects a repair service and chooses a service method and time slot.

**Preconditions**: The customer has selected the device type, brand, and model.

**Main Flow**:

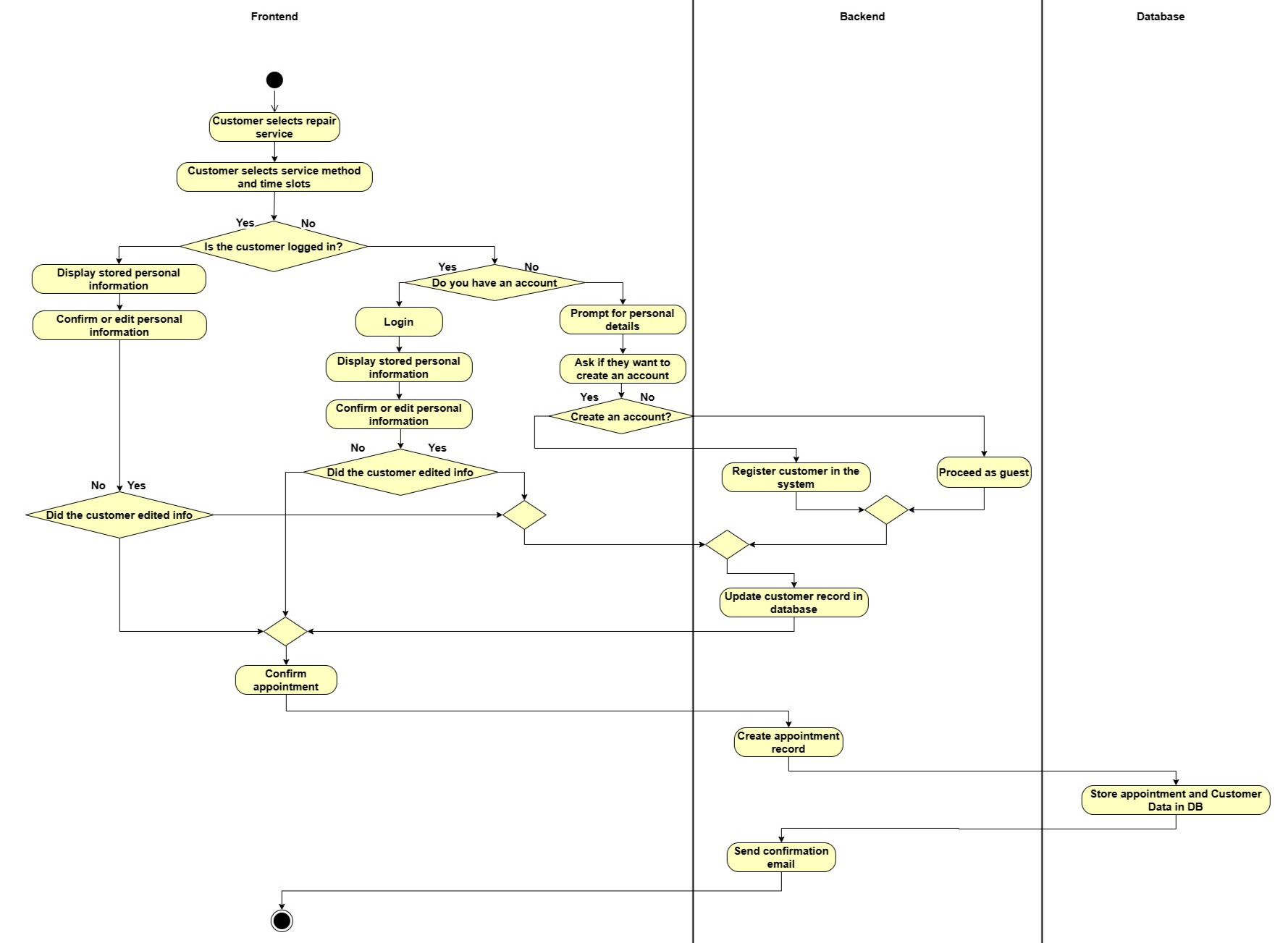
1. Customer selects the repair service and chooses the service method and time slot.
2. The system checks if the customer is logged in:
   * If logged in, their stored personal details are displayed for review or editing.
   * If not logged in, the system asks whether the customer already has an account:
     + If yes, they are prompted to log in.
     + If not, they must provide personal details and decide whether to create an account or proceed as a guest.
3. After personal details are provided, the system asks if the customer has edited their information:
   * If yes, the system updates the customer record in the database.
   * If not, the process continues.
4. The customer confirms the appointment.
5. The backend creates an appointment record and stores customer and appointment data in the database.
6. A confirmation email is sent to the customer.

**Postconditions**:

* The customer’s appointment is successfully recorded in the database.
* A confirmation email is sent to the customer.
* If the customer opted to create an account, they are now registered in the system.

**Entities**: User (Supertype), Customer (Subtype of User), Repair\_Appointment (Outside the IS-A Relationship)

#### Graphical Representation



### Analytics Report

#### Concept

After describing your analytics report, list the three entities involved.

#### Proof of Concept

Add screenshots and descriptions of executing your analytics report.

### NoSQL Design

List entities involved in the use case (use case, not report[!]) in order to improve readability.

#### Design Overview

#### Expected Execution and Possible Changes

#### Five Rules of Thumb