



Technical report

Julian David Celis Giraldo 20222020041

Catalina Ariza Ardila 20222020084

Facultad de ingeniería, Universidad Distrital Francisco José de Caldas

Group 020-82, software modeling foundations

Carlos Andrés Sierra Virguez

September 4th 2024

BOGOTÁ D.C

INDEX

1. User stories and acceptance criteria
2. Object-oriented principle analysis
3. Technical decisions
4. More diagrams

User stories and acceptance criteria

Manager (Administrator)

1. As a manager, I want to be able to register my shop in the application so that customers can learn about my business.

Acceptance Criteria:

- ❖ The administrator must be able to register their shop.
 - ❖ The user must be able to enter shop information (location, type of business, service hours).
 - ❖ Customers must be able to view this information.
 - ❖ The system must validate if the administrator already owns a shop.
-

2. As a manager, I want to have a space to list services, details, and the service hours, so that customers are well informed about these aspects.

Acceptance Criteria:

- ❖ The administrator must be able to name the services.
 - ❖ The administrator must be able to assign service details and prices.
 - ❖ The services should appear in the services list.
 - ❖ Customers must be able to select the time for their service.
-

3. As a manager, I want to have a space to showcase in detail the works or services offered in my shop, so that customers can check if it's what they are looking for.

Acceptance Criteria:

- ❖ The administrator must be able to edit, add, or delete photos of the completed works.
 - ❖ Photos must appear in an accessible and visible section for customers.
-

4. As a manager, I want to be able to show basic information about my employees on the shop's profile, so that customers can see relevant details and choose who they want to schedule with.

Acceptance Criteria:

- ❖ The administrator must be able to add employees to the shop's profile, including basic information (name, photo, specialization).
- ❖ Customers must be able to see basic employee information and select a professional according to their preference.

- ❖ The employee's personal information must be visible both on the services page and the shop's profile.
-

5. As a user, I want to register to have my profile in the app and add basic personal information.

Acceptance Criteria:

- ❖ The user must be able to fill out their data (name, birthdate, gender, phone number).
 - ❖ The basic information must appear on the user profile and be editable at any time.
 - ❖ The data must be displayed on the user's profile.
 - ❖ The user must receive a confirmation that their profile has been successfully created.
 - ❖ If any required fields are missing, the system must display a clear error message.
-

Customer

1. As a customer, I want to search for shops by name or code for convenience.

Acceptance Criteria:

- ❖ The customer must be able to easily find and view the shops the application offers in relation to what they are searching for.
 - ❖ The information should appear summarized so that the customer can quickly find what they want.
 - ❖ The customer must be able to search for a shop by name or code using a search field.
 - ❖ Search results must appear quickly and be sorted by relevance or proximity.
 - ❖ The system must show a message when no shops match the customer's search terms.
-

2. As a customer, I want to schedule one or more services to be attended.

Acceptance Criteria:

- ❖ The customer must be able to view the availability of time slots before booking a service.
- ❖ The system must validate that the selected time slot is not occupied by another customer.
- ❖ The customer must receive confirmation of their appointment once the service booking is completed.

-
3. As a customer, I want to have different payment options in the application for convenience.

Acceptance Criteria:

- ❖ The customer must be able to choose whether to pay with cash, a debit/credit card, and whether the payment will be made in person or virtually.

-
4. As a customer, I want to see a summary of my appointments, both past and future, to keep track of the services I've taken and the ones I have scheduled.

Acceptance Criteria:

- ❖ The customer must be able to see their upcoming appointments, with the date, time, location, and the booked service.
- ❖ The customer must be able to see a history of completed appointments, with information about the date and the service rendered.
- ❖ The customer must be able to access these sections from their profile or an appointments menu.

-
5. As a customer, I want to be able to rate the service I received at the shop after my appointment, to help other customers make informed decisions and provide feedback to the shop.

Acceptance Criteria:

- ❖ The customer must be able to assign a rating from 1 to 5 stars for the service received.
- ❖ The rating must be publicly visible on the shop's profile so other customers can see reviews.

Employee

1. As a shop employee, I want to receive notifications from customers who chose my service.

Acceptance Criteria:

- ❖ The employee must be able to see the main information of the customer who booked the appointment (full name, age, phone number).
 - ❖ The employee must be able to see the date and time of the appointment booked for their services.
-

2. As a shop employee, I want my professional information and the shop where I work to be visible in the application, so that customers can learn about my skills and select the services I offer.

Acceptance Criteria:

- ❖ The employee must be able to add their professional experience and be linked to the shop where they work (name, role, specialization, experience, shop location).
 - ❖ Customers can only view the employee's professional information (no access to personal details).
 - ❖ The employee's information will be accessible from the shop's profile.
-

3. As a shop employee, I want to be able to rate customers after completing a service, to help maintain a history of customer behavior and improve service quality in the future.

Acceptance Criteria:

- ❖ The employee must be able to rate the customer after the service has been completed.
 - ❖ The rating must be recorded in the customer's history and visible only to the administrator and employees.
-

Object-oriented principle analysis

The object-oriented paradigm allows us to represent real-world ideas in software as realistically as possible, using abstraction and concepts of class-object.

In this project, this paradigm is very useful because the application aims to solve a real-world problem (scheduling appointments in different shops in a fast and organized way). Additionally, since this is a complex and scalable system, OOP's modular approach to organizing and structuring the code makes problem development easier.

In our project, several OOP concepts are already implemented, such as polymorphism when scheduling a service, as the process varies depending on the shop.

The SOLID principles are also followed, which can be seen in the application's class diagram.

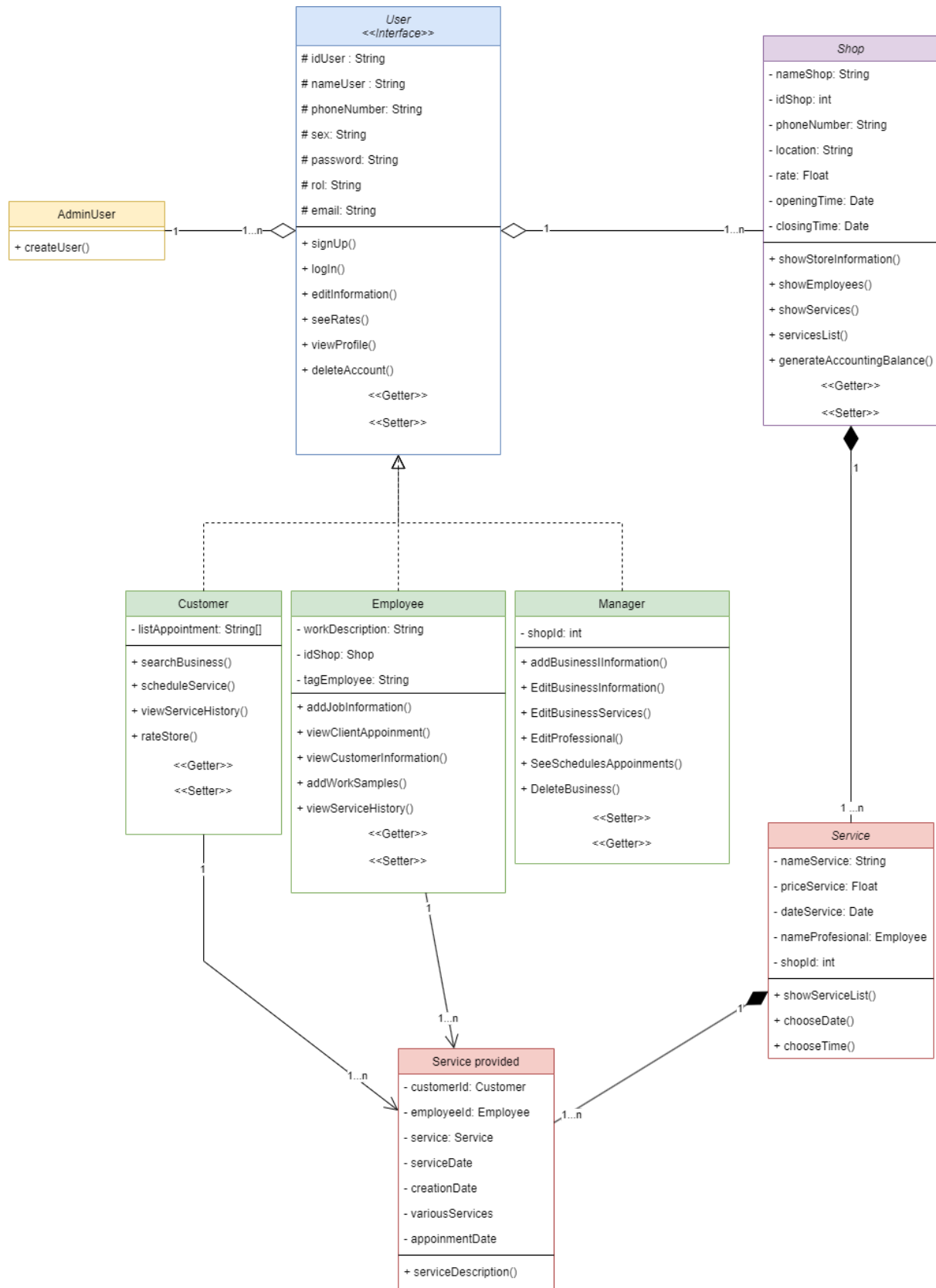


Figure 1: Class Diagram.

S: The classes comply with the single responsibility principle. This is best observed in the CRC cards that were created.

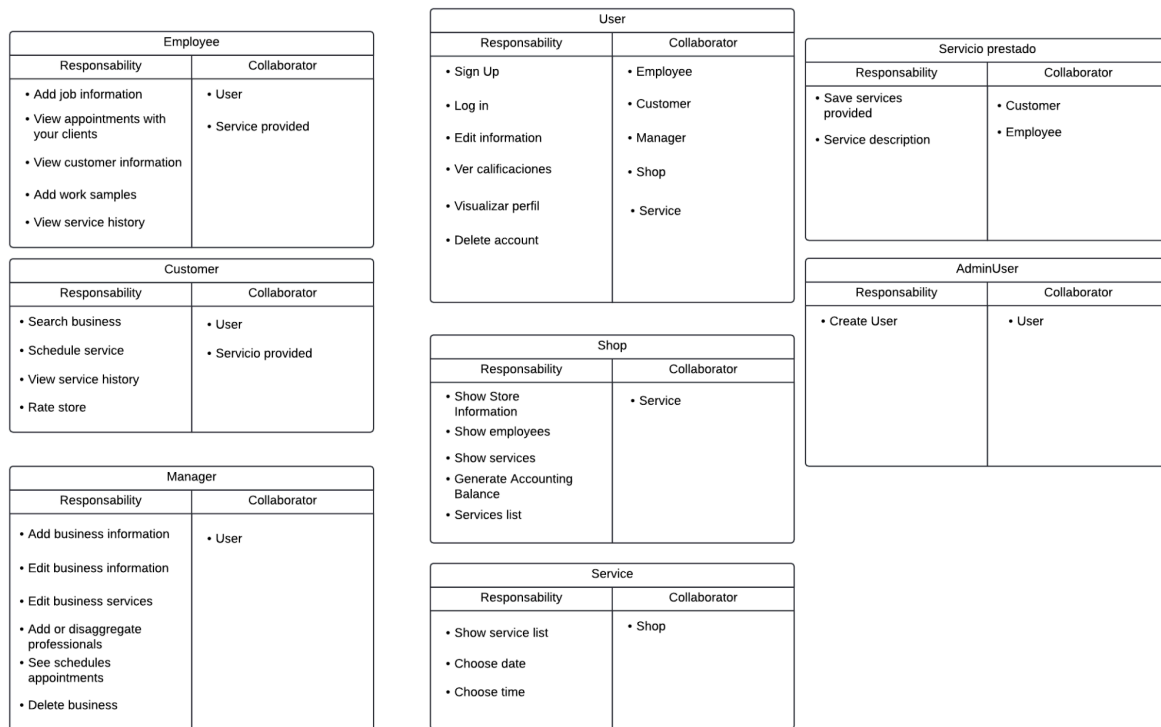


Figure 2: CRC Cards.

O: By implementing inheritance, the abstract class meets the criterion of being closed to modification, and the concrete classes comply with the criterion of being open to extension.

L: By implementing inheritance, the subclasses can be used as if they were the superclass, thus fulfilling Liskov's substitution principle.

I: The interface segregation principle, although not perfectly implemented, is partially fulfilled with the "User" interface, which forces the child classes to use its methods.

D: The user only connects to high-level classes, thus ensuring the dependency inversion principle.

As the project progresses, we aim to implement design patterns, which are an important part of programming with OOP.

Technical decisions

- ❖ As a team, we decided to limit the aggregation of multiple shops to a single administrator, as this would add a level of complexity to the application.
- ❖ We also decided that the "Manager" class should be independent of the "Employee" class due to the amount of responsibilities the manager has.
- ❖ Since we don't have clients yet, we tried to create the user stories based on our own experiences.

More diagrams

Activity diagram:

- [Click here to see the diagram](#)
- [Click here to see the diagram](#)
- [Click here to see the diagram](#)

Sequence diagram:

- [Click here to see the diagram](#)

Deployment diagram:

- [Click here to see the diagram](#)