Software Requirements Specification

for

Hairdressing service reservation system

Version 1.0

Prepared by

Arman Salehi	40113421	armansalehi82@aut.ac.ir
Sadegh Khaleghi	40112014	sadegh.khaleghi@aut.ac.ir
Mehrsa Malekzadeh	40113028	mehrsamalek@aut.ac.ir
Alireza Davoudi	40113011	ali.davoudi@aut.ac.ir
Amir Mahdi Habibi	40031012	amhabibi@aut.ac.ir

Instructor:	Dr. Zahra Ghorbanali
Course:	Software Engineering
Teaching Assistant:	Majid Nami
Date:	October 25, 2024

Contents

Revisions

Version	Primary Author(s)	Description of Version	Date Completed

1 Introduction

1.1 Document Purpose

In this document, we aim to provide a comprehensive and clear overview of the purpose, functionalities, and requirements that this software addresses. This initial explanation is intended to ensure that all stakeholders gain a thorough understanding of the project’s objectives, enabling alignment toward a shared and well-defined goal.

In the subsequent sections, we will precisely and transparently examine the challenges and requirements associated with this project. This approach will ensure that the software effectively meets user expectations and supports seamless interaction, allowing users to achieve their objectives. Additionally, it will ensure the software’s responsiveness to requests, avoiding potential disruptions in operation.

1.2 Product Scope

The Salon Reservation System allows clients to reserve services online, view available slots, and select their preferred staff members. Key features include user management, appointment booking, automated reminders, and statistical reporting for managers to optimize services and improve client satisfaction. Additional features such as staff profiles, schedule management, and client feedback collection are designed to enhance the overall user experience.

This application is intended for single-location salons and requires an internet connection for use. It does not support multiple languages or currencies by default and may have limitations on the number of simultaneous users based on infrastructure constraints.

1.3 Intended Audience and Document Overview

Since this project is currently an internal university project developed for the Software Engineering course, the intended readers of this document are:

- The course instructor and teaching assistant, serving as mentors and product managers
- The Scrum Master, as the lead of the Scrum team
- Scrum team members, including project developers and designers
- Other students in the course to provide feedback and evaluation about the project

Each of these readers may review relevant sections of this document as desired. Specifically, it is recommended that the instructor, teaching assistant, and other students focus on the "Introduction" and "Overall Description" sections. The Scrum team members and developers, however, should be thoroughly familiar with all parts of the document, particularly the project requirements and software structure, to ensure an accurate and comprehensive understanding of the system.

1.4 Definitions, Acronyms and Abbreviations

Scrum: An agile framework for managing and completing complex projects, especially in software development.

Scrum Master: The person responsible for facilitating Scrum processes and guiding the team in agile principles.

Stakeholders: Individuals who are affected by or have an interest in the outcome of a project, including customers, users, and investors.

COMET: An object-oriented methodology for the analysis and design of real-time and complex systems.

UML (Unified Modeling Language): A standardized modeling language for representing and designing software systems.

MySQL: An open-source relational database management system for storing and managing data.

Python: A high-level, general-purpose programming language known for its readability and simplicity.

Django: A high-level web framework in Python for rapid and secure web application development.

MVT (Model-View-Template): An architectural pattern used in Django to separate data logic, display, and templates.

PEP 8: A set of guidelines for writing Python code in a clean and consistent manner.

HTTPS (Hypertext Transfer Protocol Secure): A secure version of HTTP that encrypts data for safe web transmission.

UI (User Interface): The interface that enables user interaction with the system.

GUI (Graphical User Interface): A visual user interface that uses graphics and visual elements for interaction.

WCAG (Web Content Accessibility Guidelines): Standards for making web content accessible to individuals with disabilities.

API (Application Programming Interface): A set of protocols and tools that allow software applications to communicate with each other.

JSON (JavaScript Object Notation): A lightweight format for exchanging data between server and client.

ZarinPal: An online payment platform in Iran that enables secure transactions.

GDPR (General Data Protection Regulation): Regulations for data protection and privacy for users in the European Union.

2FA (Two-Factor Authentication): A security method requiring two forms of identification for authentication.

RBAC (Role-Based Access Control): A model for controlling access based on users' roles within a system.

PCI-DSS (Payment Card Industry Data Security Standard): Security standards to protect credit card data.

IDS (Intrusion Detection System): A system designed to detect and alert unauthorized intrusions in networks or systems.

1.5 Document Conventions

In general, this document follows the IEEE formatting requirements. Use Arial font size 11, or 12 throughout the document for text. Use italics for comments. All the headings are according to the template suggested by IEEE and contents provided in the course.

1.6 References and Acknowledgments

- IEEE Standard 830-1998
- COMET Methodology Documentation
- UML 2.0
- ZarinPal Payment Platform Documentation
- IBM Engineering Requirements Management

2 Overall Description

2.1 Product Overview

The Salon Reservation System is a web-based application designed to streamline the process of booking appointments and managing schedules for salons. This system enables clients to easily view available services, choose preferred hairdressers, and book appointments at convenient times, all without the need for in-person visits. It also allows salon staff to effectively manage appointments, daily schedules, and client interactions.

The system is particularly beneficial for high-traffic salons, as it reduces the time and effort required for manual booking, leading to improved efficiency and customer satisfaction. Key features include automated reminders for clients, statistical reporting for managers, and the ability for clients to leave feedback on services. This application aims to simplify the reservation process, prevent schedule conflicts, and improve the overall experience for both clients and staff. Additionally, it provides salon managers with data-driven insights to enhance service quality and client satisfaction.

2.2 Product Functionality

This section provides a list of all system functionalities, which are detailed in Section 3:

- Account creation for customers and stylists
- Stylist qualification verification
- Access to salon listings
- View available services
- Appointment booking
- Electronic payment system and use of digital wallet
- Management of requested appointments
- Confirmation message delivery to users
- Communication between users and stylist
- Evaluation of the performance of stylist by users
- Cancel the reservation
- Payment distribution to stylists

2.3 Design and Implementation Constraints

The development of the Salon Reservation System is subject to several design and implementation constraints that limit available options and specify necessary requirements for the development process:

1. Design Methodology: The system will be designed using the COMET (Collaborative Object Modeling and Architectural Design Method), which is particularly well-suited for designing complex, object-oriented systems with a focus on modularity and scalability. COMET provides a structured approach for defining the system’s architecture, classes, and interactions, allowing for a cohesive and maintainable design.

2. Modeling Language: The Unified Modeling Language (UML) will be used to represent the system’s structure and behavior throughout the design phase. UML is essential for visualizing and documenting system components, interactions, and workflows in alignment with the COMET methodology, ensuring clear communication across the development team and stakeholders.

3. Hardware Limitations: The system is intended to run on standard web servers and is not designed for high-performance, resource-intensive operations. It must operate efficiently within the memory and processing capabilities typical of small-to-medium-scale servers.

4. Technology Stack: Development will use Python and Django for the backend, chosen for their flexibility and support for rapid development. The front end will be developed with HTML, CSS, and JavaScript, ensuring compatibility with modern browsers. MySQL will be the database management system, providing relational data management to meet the needs of scheduling and user data storage.

5. Parallel Operations: The system is expected to handle concurrent user sessions; however, due to resource constraints, it is not intended for very high levels of concurrent access. Concurrency control will be managed primarily at the database level.

6. Language and Localization: The system will initially support only Farsi and a single currency (Rial), which limits its use to Farsi-speaking regions or locations that operate in that currency. Localization features may be added in future versions if required.

7. Security Considerations: The system will require secure user authentication, and sensitive data (such as passwords) will be encrypted. Additionally, access to different levels of data will be controlled through role-based access to ensure that staff and clients have appropriate permissions.

8. Programming Standards: The code will follow industry-standard best practices in Python and Django, with conventions such as PEP 8 for Python and Django’s Model-View-Template (MVT) design pattern for maintainable, modular code. Adherence to these standards will facilitate code readability and simplify future maintenance.

2.4 Assumptions and Dependencies

Assumptions:

- Users of the system (including clients and salon staff) will have stable internet access, as this system is designed as a web-based application.
- The system will be accessed through modern browsers, such as Google Chrome, Firefox, or Safari, and will be compatible with the latest versions of these browsers.
- Customer and appointment data will be securely stored on a server, with access controlled and secured by system administrators.
- It is assumed that all salon staff have received adequate training to use the system effectively and are capable of managing appointments and services.

Dependencies:

- The system relies on a MySQL database server for storing user data, appointments, and system configurations.
- Since the system is developed using Python and the Django framework, up-to-date and compatible versions of these technologies must be installed and configured on the server.
- The system requires integration with external email or SMS services to send reminders and notifications to clients.
- Due to the use of the COMET method and UML for design and modeling, all members of the development team must be familiar with these methodologies and the UML modeling tools.
- Data security and privacy depend on supporting an HTTPS protocol for secure data transmission between users and the server.

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

- **Graphical Design:**

  The software will feature a user-friendly and visually appealing graphical user interface (GUI). This design will include color schemes, fonts, and UI elements such as buttons, menus, and forms.

- **User Flow:**

  Users should easily navigate from the home page to various sections (booking, user profile, and payment). This process includes the following steps:

  - User login/signup
  - Searching for stylists and selecting services
  - Booking a time and setting a date
  - Confirming the reservation and making a payment

- **Accessibility:**

  The user interface must comply with accessibility standards (such as WCAG) to ensure usability for users with special needs (such as visually impaired or disabled individuals).

3.1.2 Hardware Interfaces

The software should run on standard systems (PCs, tablets, and mobile devices). Minimum requirements include:

- CPU: 2.0 GHz or higher
- RAM: Minimum 4 GB
- Storage space: Minimum 500 MB
- Internet Connectivity

3.1.3 Software Interfaces

- **APIs and Protocols:**

  The software must utilize the Zarinpal API for processing online payments. Zarinpal is a secure and reliable payment solution in Iran that allows users to easily make payments using their bank cards.

- **Data Formats:**

  Data exchanged with Zarinpal should be in JSON format.

3.2 Functional Requirements

3.2.1 F1: Account creation for customers and stylists

The system should allow users to create accounts, log in, and manage their profiles.

- Users should be able to register using a phone number.
- Users should receive an SMS for account verification.
- Existing users should be able to log in with their credentials.

3.2.2 F2: Stylist qualification verification

F2.2: Verification

- Hairstylists register using their phone number and business details (name, address).
- A verification SMS is sent; the account is activated upon SMS confirmation.

F2.1: Login

- Hairstylists log in using their phone number.
- The system checks credentials and grants access to their dashboard if valid.

3.2.3 F3: Access to salon listings

This feature enables users to view and access all barber shops registered within the system through an integrated, interactive map. When users navigate to the map section, a digital map embedded within the application displays all available barber shops as location pins or markers. Each barber shop marker shows key details, such as the /shop name, a brief description, and service ratings when hovered over or clicked. The map is zoomable and scrollable, allowing users to explore areas by panning and adjusting the zoom level to find barber shops in specific neighborhoods or around their current location. The system uses geolocation data (if permitted by the user) to center the map on the user's current location, ensuring a personalized experience.

3.2.4 F4: View available services

The system allows clients to view available services offered by hairstylists.
Clients can browse a list of available services, including details like service name, description, duration, and price.
Clients can search for specific services or filters by category (e.g., haircuts, coloring, etc.).
Clients can select a service to view detailed information, such as duration, price, and any additional options.
The system should allow customers to see sample works of the barbers and their ratings

3.2.5 F5: Appointment booking

This feature enables users to reserve appointments at barber shops with flexibility, offering options to select specific barbers, dates, and services. Upon entering the reservation section, users first see a calendar interface that displays available dates and times. The system shows only open slots based on barber availability, working hours, and current bookings, reducing the chance of double-booking. Users can filter the list by selecting a specific barber, narrowing down to barbers they prefer or are familiar with. Alternatively, they can filter by service type (e.g., haircut, beard trim, or special grooming packages) or by date if they have a specific time in mind. Each filtered selection updates dynamically, presenting users with available options that match their preferences. Once a user selects their desired time, date, barber, and service, the reservation is locked into the system, and they receive an automatic confirmation along with an option to add it to their calendar (or a message (SMS)).

3.2.6 F6: Electronic payment system

This feature facilitates a seamless payment experience by integrating a digital wallet system for both users and barbers within the platform. Users can load funds into their digital wallet through various payment methods, such as credit/debit cards, bank transfers, or third-party services. The wallet balance can then be used to pay for reservations directly, reducing dependency on cash payments or external apps. Additionally, each barber has their own digital wallet within the system, where they receive payments for completed services. When a user confirms a reservation, they have the option to pay from their digital wallet, ensuring instant and secure transactions. Upon payment, the amount is instantly credited to the barber’s wallet (minus any service fees), and both the user and barber receive an automatic confirmation of payment. Users can also set up auto-reload options, ensuring they have sufficient funds for future appointments.

3.2.7 F7: Management of requested appointments

F7.1: Pending Appointments

Newly booked appointments are set to "Pending" status and require confirmation by the salon manager or hairstylist.

F7.2: Confirm Appointments

Allows the manager or hairstylist to confirm "Pending" appointments, changing their status to "Confirmed" and notifying the client.

- The manager views and selects a "Pending" appointment.
- The system updates the status to "Confirmed" and sends a notification to the client.

3.2.8 F8: Confirmation message delivery to users

The system shall send an SMS notification to the client based on the appointment status:

- If the appointment is **confirmed**, an SMS will be sent notifying the client of the confirmation.
- If the appointment is **canceled**, an SMS will be sent notifying the client of the cancellation.

3.2.9 F9: Communication between users and stylists

Once an appointment request is confirmed by the stylist, both the customer and the stylist should be able to access each other's contact information. In the user's panel, the name and contact number of their stylist should be visible for each scheduled appointment. Similarly, in the stylist's panel, they should see the name and contact number of each client for the appointments booked, facilitating direct communication if needed.

3.2.10 F10: Evaluation of the performance of stylists by users

After completing their appointment and receiving the services, users can provide feedback on the quality and other attributes of the stylist and salon. Either the user or the stylist will press the "Complete Service" button on the appointment page, triggering a feedback form for the user. Users can leave comments, upload photos of the services or salon, and provide an overall quality rating. Ratings can include options such as "Excellent," "Good," "Average," "Poor," or "Very Dissatisfied." All reviews are displayed in the salon’s profile, indicating the specific stylist and service for each review. An aggregate score from all reviews is also displayed as the stylist's rating.

3.2.11 F11: Appointment Cancellation Policies

- **F11.1: Cancellation by Users**
  As per the payment requirements, users pay the full appointment fee upfront. They can cancel their reservation up to one day (24 hours) before the appointment for a full refund to their digital wallet. If the cancellation occurs within 24 to 12 hours of the appointment, 70% of the fee is refunded. For cancellations less than 12 hours before the appointment, only 50% of the fee is refunded. Additionally, if a user cancels three appointments within one month, their account will be suspended for one week.

- **F11.2: Cancellation by Stylists**
  Stylists can cancel an appointment up to two days (48 hours) before the scheduled time. If a stylist cancels within 48 hours, a message of apology will be sent to the user, and alternative stylists with availability at the same time will be suggested. If the user does not accept an alternative, they will receive a 20% discount code for their next appointment with the same stylist. If stylists cancel appointments twice within a one-month period, their account will also be suspended for one week.

3.2.12 F12: Payment distribution to stylists

To encourage stylists to increase their customer base and enhance service quality, as well as to achieve project profitability, payments will be based on a percentage of each booking. For each appointment, 70% of the fee will go to the stylist, with the remaining 30% allocated to the management and development team.

- **F12.1: Scheduled Daily Payouts**
  At the end of each day, the system will automatically transfer the accumulated earnings to each stylist’s account. Currently, instant payouts are not supported due to limited resources and banking integration constraints. However, real-time payment transfers may be introduced in the future.

- **F12.2: Potential Performance-Based Bonuses**
  As the project’s profitability increases, a performance-based bonus system may be established to further reward stylists who maintain high customer satisfaction and service quality.

3.3 Use Case Model

Use case shape key

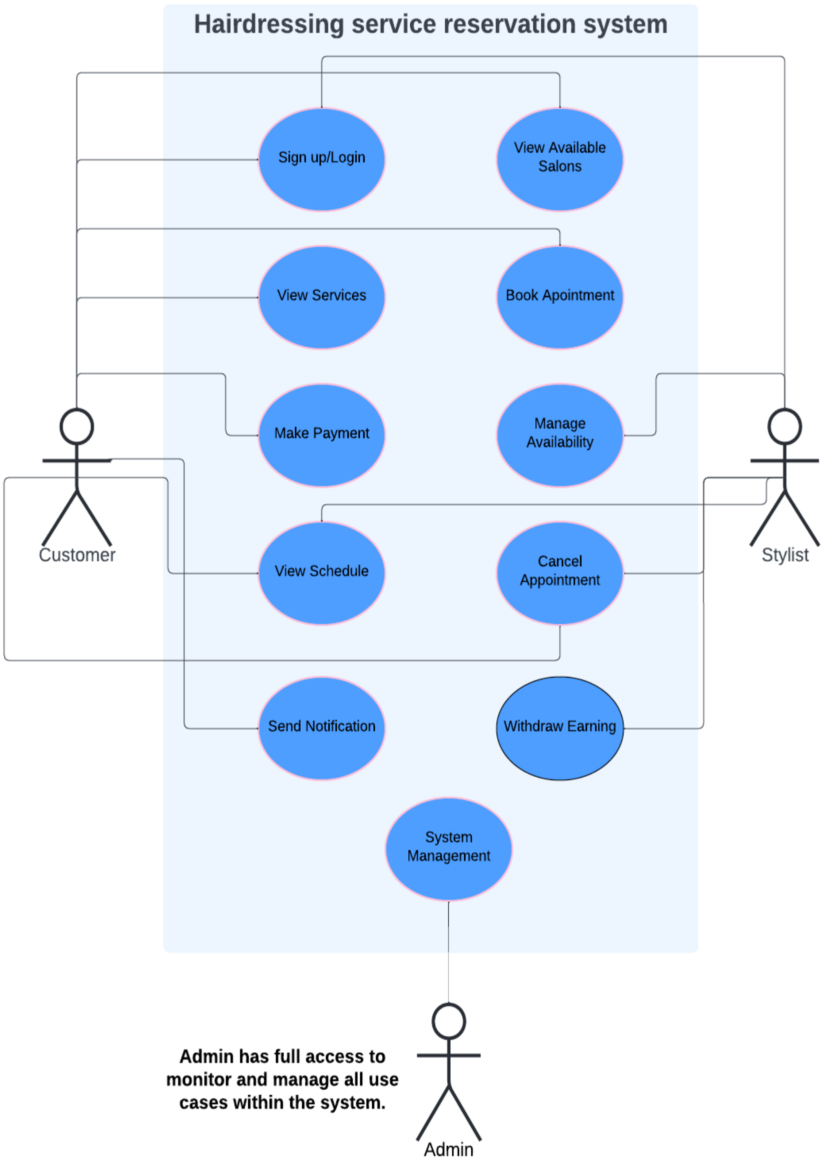
웃 Actor

● Use case

□ System

Hairdressing Service Reservation System

This flowchart depicts the use cases for a hairdressing service reservation system, including customer actions such as booking appointments, viewing available salons and services, making payments, and cancelling appointments. Stylists can manage their availability and view their schedules, while the admin has full access to monitor and manage all use cases within the system.



Admin has full access to monitor and manage all use cases within the system.

3.3.1 Sign up / Login – U1

Author: Sadegh Khaleghi

Purpose:
To allow clients to log in or sign up using their phone number, receiving a verification code via SMS for authentication. This simplified process ensures security and ease of access for clients.

Requirements Traceability:

- R01: Clients must be able to register or log in using only their phone number.
- R02: The system should send an SMS verification code to the client's phone number.
- R03: Clients must enter the received verification code to complete the login or registration process.

Priority: High

Preconditions:

- The client has a valid phone number and can receive SMS messages.

Postconditions:

- The client is authenticated and logged in to the system, with access to their profile and booking functionalities.

Actors:

- Client (user logging in or signing up)
- System (handles phone number verification, code generation, and SMS sending)

Extends:

- None

Flow of Events:

1. **Basic Flow**

- Step 1: The client opens the app or website and selects the "Login / Sign Up" option.
- Step 2: The client enters their phone number in the provided field.
- Step 3: The system generates a unique verification code and sends it to the client's phone number via SMS.
- Step 4: The client receives the SMS and enters the verification code in the app or website.
- Step 5: The system verifies the code:
    - ▪ If correct, the system logs the client in and displays the client's dashboard.
    - ▪ If incorrect, the system prompts the client to re-enter the code or request a new code.

2. **Alternative Flow**

- A1: If the client doesn't receive the code, they can request a resend after a short delay.
- A2: If the client enters an incorrect code multiple times, the system temporarily locks the account and notifies the client to try again later for security purposes.

3. **Exceptions**

- E1: If the phone number is invalid, the system notifies the client and prompts them to enter a valid phone number.
- E2: If there is an issue with the SMS service, the system notifies the client and suggests trying again later.

Includes (other use case IDs):

- None

Notes/Issues:

- Ensure compliance with privacy and security standards for storing and managing phone numbers.
- Consider implementing a CAPTCHA for added security against automated login attempts.

3.3.2 View Available Salons – U2

Author: Mehrsa Malekzadeh

Purpose:
To simplify the user experience by allowing users to locate nearby barber shops using an interactive map.

Requirements Traceability: Linked to FR-1 (Map-Based Access to Barber Shops)

Priority: High

Preconditions:

- User must have an active internet connection.
- Users should enable location services if they want to view nearby barber shops.
- Barber shops must be registered within the system database.

Post conditions:

- User can view the locations of all nearby barber shops.
- Map displays accurate information, such as the name and distance of each barber shop.
- User can navigate to additional details or proceed to booking from this view.

Actors: User, Admin

Extends: None

Flow of Events:

1. **Basic Flow**

- The user accesses the “Find Barber Shops” section.
- The system request’s location permissions and centers the map based on the user’s location.
- The map displays nearby barber shops as pins or markers.
- The user clicks on a pin to view details and access additional options, such as reserving an appointment.

2. **Alternative Flow**

- If location permissions are denied, the map defaults to a general area view, and users can manually pan or zoom to find barber shops.

3. **Exceptions**

- If the map service is unavailable, the system displays an error message, and users are directed to a list view instead.

- If no barber shops are nearby, a “no results found” message is displayed with options to expand the search area.

Includes: FR-1, FR-2 (if it integrates with the reservation system)

Notes/Issues:
- Ensure user data privacy when accessing location.
- Map load times should be optimized for smooth user experience.

3.3.3 View Services – U3

Author: Alireza Davoudi

Purpose:
To allow users (clients) to browse through the list of available services offered by barbers, including details such as service descriptions, prices, duration, and ratings, to facilitate booking decisions.

Requirements Traceability:
- R7: Users must be able to view a list of all available hairdressing services.
- R8: Users must be able to see details of each service, including price, duration, and barber profile information.
- R9: Users should be able to filter and sort services based on criteria like price, duration, rating, and availability.

Priority: Medium

Preconditions:
- The user is logged in to access personalized service recommendations (if applicable).

Postconditions:
- The user has viewed detailed information about the services and can proceed to book a service if desired.

Actors:
- User (client browsing available services)
- System (fetches and displays services and related details)

Extends:
- None

Flow of Events:

1. **Basic Flow**

- Step 1: The user selects the "View Services" option from the main interface.
- Step 2: The system displays a list of available services with brief details (e.g., service name, price, duration, and barber rating).
- Step 3: The user clicks on a specific service to view detailed information, which includes service description, full price, estimated duration, barber profile, and user ratings.
- Step 4: The user may filter or sort the service list based on their preferences (e.g., lowest price, highest rating, shortest duration).
- Step 5: The user reviews the service details and may proceed to book a service or return to the main service list.

2. **Alternative Flow**

- A1: If the user applies specific filters, the system updates the displayed services accordingly to reflect the criteria (e.g., displaying only services within a certain price range).

3. **Exceptions**

- E1: If there are no available services that match the user’s filter criteria, the system displays a "No matching services" message.
- E2: If a service's details cannot be loaded (e.g., network error), the system displays an error message and suggests the user try again.

Includes (other use case IDs):

- None

Notes/Issues:

- Consider optimizing the service list for faster loading times, especially if there is many services.

Ensure secure access to barber profiles to protect sensitive information while allowing users to make informed decisions based on ratings and reviews.

3.3.4 Book Appointment - U4

Author: Mehrsa Malekzadeh

Purpose: To allow users to schedule appointments by selecting preferred dates, times, barbers, and services.

Requirements Traceability: Linked to FR-2 (Flexible Appointment Scheduling and Filtering Options)

Priority: High

Preconditions: User must have an active account.

- Barber shop schedules and availability must be up to date in the system.

Post conditions:

- User successfully schedules a reservation.
- Reservation details are saved, and a confirmation is sent to both user and barber.

Actors: User, Admin, Barber

Extends: None

Flow of Events

1. **Basic Flow**

- The user navigates to the reservation section.
- They select a barber, date, time, and desired service.
- The system verifies availability and presents a “Confirm Reservation” button.
- The user confirms, and a reservation is created and saved in the system.

2. **Alternative Flow**

- If a preferred barber is unavailable, the system suggests other available barbers with matching services.

3. **Exceptions**

- If a time slot is booked before confirmation, the system notifies the user, prompting them to select a different time.
- If the system fails to connect to the scheduling database, an error message is displayed, and the user cannot proceed.

Includes: FR-2, FR-3 (if integrating with payment system)

Notes/Issues:

- Provide clear error messages to enhance user experience.
- Ensure reservation accuracy by preventing double-booking.

3.3.5 Make Payment – U5

Author: Mehrsa Malekzadeh

Purpose: To enable secure and convenient payments via a digital wallet for both users and barbers.

Requirements Traceability: Linked to FR-3 (Digital Wallet for Users and Barbers with Electronic Payment Integration)

Priority: High

Preconditions:

- User must have a verified payment method to fund their digital wallet.
- Barber must have a registered account in the system.

Post conditions:

- Payment is successfully completed, and funds are transferred to the barber’s digital wallet.
- User and barber receive confirmation.

Actors: User, Admin

Extends: None

Flow of Events

1. **Basic Flow**

- User selects the option to pay via the digital wallet during reservation confirmation.
- System verifies if the user's wallet has sufficient balance.
- If funds are sufficient, payment is processed, and the barber's wallet is credited.
- Both user and barber receive a payment confirmation.

2. **Alternative Flow**

- If the user's balance is insufficient, they are prompted to add funds through their chosen payment method.
- After adding funds, they complete the reservation payment.

3. **Exceptions**

- If the payment gateway fails, the system alerts the user and provides alternative payment options.
- If the barber's account cannot receive payments due to any system error, the payment is temporarily held until resolved.

Includes: FR-3 (Wallet funding, Reservation payment)

Notes/Issues:

- Ensure transaction security and comply with financial regulations.
- Implement refund mechanisms in case of payment issues or reservation cancellations.

3.3.6 Manage Availability – U6

Author: Sadegh Khaleghi

Purpose:
To allow barbers to manage their availability for appointments by setting available times and dates, as well as to review and approve or decline appointment requests based on their availability and personal preferences.

Requirements Traceability:

- R21: Barbers must be able to set their available dates and time slots.
- R22: Barbers must be able to mark specific dates and times as unavailable.
- R23: Barbers must receive notifications of new appointment requests and be able to approve or decline them.
- R24: Clients should only be able to see time slots marked as available by the barber.
- R25: Barbers and clients should be notified of approved or declined appointments.

Priority: High

Preconditions:

- The barber is logged in and verified.

Postconditions:

- The barber’s availability is updated, and any appointments the barber has approved or declined are recorded in the system, with notifications sent to clients as needed.

Actors:

- Barber (service provider managing their availability and approving/declining appointments)

- System (facilitates availability management, appointment requests, and notifications)

Extends:

- None

Flow of Events:

1. Basic Flow

- Step 1: The barber selects the "Manage Availability" option from their dashboard.

- Step 2: The system displays a calendar view or list where the barber can set their available dates and time slots.

- Step 3: The barber selects specific dates and times to mark as available for client bookings.

- Step 4: The barber also marks any dates and times as unavailable when they do not wish to take appointments.

- Step 5: When a new appointment request is made, the system notifies the barber.

- Step 6: The barber reviews the appointment details (client name, service type, date, time) and decides to approve or decline the request.

- Step 7: If approved, the system confirms the booking, updates the barber’s schedule, and sends a notification to the client.

- Step 8: If declined, the system marks the slot as unavailable for that client, and a notification is sent to inform the client.

2. Alternative Flow

- A1: If the barber has recurring availability (e.g., every Monday from 9 AM to 5 PM), they can apply a repeating schedule for availability.

- A2: If an appointment conflicts with personal plans or unforeseen events, the barber can mark that slot as unavailable or manually decline the appointment request.

3. Exceptions

- E1: If the barber tries to update availability for a time slot that already has an approved appointment, the system prevents changes to avoid conflicts.

- E2: If there is an issue with network connectivity while updating availability or approving/declining an appointment, the system notifies the barber and prompts them to try again later.

Includes (other use case IDs):

- None

Notes/Issues:

- Consider adding an option for barbers to add notes when declining appointments, which could be sent to clients for clarity.

- Ensure real-time updates to availability to prevent overbooking or double-booking in case of high demand.

3.3.7 View Schedule – U7

Author: Alireza Davoudi

Purpose:
To allow users to view their scheduled appointments and details, including the date, time, barber, service type, and status, facilitating easy tracking and management of upcoming and past bookings.

Requirements Traceability:
- R15: Users must be able to view a list of their upcoming and past appointments.
- R16: Users must be able to view appointment details, including service type, date, time, barber name, and location.
- R17: Users should be able to see the status of each appointment (e.g., confirmed, pending, or completed).

Priority: High
Preconditions:
- The user is logged in to view their personalized appointment history.

Postconditions:
- The user has successfully viewed their booked appointments and relevant details, enabling them to keep track of their scheduling with barbers.

Actors:
- User (client and hairstylists viewing their booked appointments)
- System (retrieves and displays appointment information)

Extends:

- None

Flow of Events

1. **Basic Flow**

- Step 1: The user selects the "View Booked Appointments" option from their profile or dashboard.
- Step 2: The system displays a list of all upcoming and past appointments for the user.
- Step 3: The user clicks on a specific appointment to view more detailed information, which includes service type, barber name, date, time, and location.
- Step 4: The system shows additional details, such as the appointment status (e.g., confirmed, pending, or completed) and any relevant notes or instructions from the barber.
- Step 5: The user can navigate back to the main list or view other appointments as needed.

2. **Alternative Flow**

- A1: If the user has no booked appointments, the system displays a "No appointments booked" message and may suggest booking a new appointment.

3. **Exceptions**

- E1: If there is an error retrieving appointment data (e.g., network issues), the system notifies the user and suggests they try again later.
- E2: If an appointment was recently canceled or modified, the system refreshes the list to display the updated appointment status.

Includes (other use case IDs):

- None

Notes/Issues:

- Consider a calendar view option to enhance usability for users with multiple appointments.
- Ensure data privacy so users only see their own appointments and not those of others.

3.3.8 Cancel Appointment – U8

Author: Arman Salehi

Purpose:
Allow users (customers and stylists) to cancel an existing salon appointment according to the defined cancellation policy.

Requirements Traceability:
Traceable to the requirement that users need the flexibility to manage and cancel their appointments when necessary, ensuring the system reflects real-time availability.

Priority:
High - This functionality is essential for user experience and resource management within the system.

Preconditions:
- The user must be logged in.
- The user must have an existing appointment scheduled in the system.

Postconditions:
- The appointment status is updated to "Canceled" in the system.
- Notifications are sent to all relevant parties (the user and the stylist).
- The slot previously occupied by the canceled appointment is marked as available for other users.

Actors:
- Customer
- Stylist
- System (for automatic notifications and availability updates)

Flow of Events:
Basic Flow:
1. The user navigates to the “My Appointments” section.
2. The user selects the appointment they wish to cancel.
3. The system displays the details of the selected appointment and prompts the user with cancellation options (including any relevant policies, fees, or penalties).
4. The user confirms the cancellation.
5. The system updates the appointment status to "Canceled."
6. Notifications are sent to both the user and the stylist about the cancellation.
7. The system marks the time slot as available for other users.

Alternative Flow:
- **Early Cancellation:**
- If the user cancels the appointment within a predefined timeframe (e.g., 24 hours in advance), the system may waive any cancellation fees.
- **Last-Minute Cancellation:**
- If the user cancels at the last minute, the system notifies the user of any applicable cancellation fee and confirms their intent to proceed with the cancellation.

Exceptions:
- **No Active Appointment Found:**
If the user selects an appointment that doesn’t exist or has already been canceled, the system displays an error message and returns the user to the "My Appointments" section.
-Cancellation Restrictions:
If the user has reached a maximum limit of cancellations within a set period, the system blocks the cancellation and notifies the user of the restriction.

Includes Other Use Cases:
- **Notification:** Sending notification of cancellation to involved parties.
- **Update Availability:** Marking the canceled appointment slot as open for other bookings.

Notes:
- Cancellation policies (e.g., timeframe, fees) should be clearly defined to avoid confusion.
- Consider implementing cancellation limits to prevent abuse of the feature.
- Refund or charge adjustments may be triggered as per the cancellation policy.

3.3.9 Send Notification – U9

Author: Alireza Davoudi

Purpose:
To notify users (clients and barbers) of important updates such as appointment confirmations, cancellations, reminders, and changes to appointment details, ensuring effective communication and reducing missed appointments.

Requirements Traceability:

- R24: The system must send notifications to clients and barbers upon key events (e.g., booking confirmation, cancellation).
- R25: Notifications must include relevant details such as date, time, service type, and any changes or special instructions.
- R26: Notifications should be customizable, allowing users to choose preferred channels (e.g., email, SMS, in-app).

Priority: High

Preconditions:

- A notification-worthy event occurs, such as a new booking, modification, or cancellation of an appointment.
- User contact information is verified, and preferences are set.

Postconditions:

- The user (client or barber) has been notified of the event and can take action if needed (e.g., confirming attendance or modifying plans).

Actors:

- System (triggers and sends notifications)
- Client (receives notifications about their appointments)
- Barber (receives notifications about client appointments)

Extends:

- None

Flow of Events

1. **Basic Flow**

- Step 1: A notification-triggering event occurs (e.g., an appointment is booked, modified, or canceled).
- Step 2: The system identifies the user(s) (client and/or barber) to notify based on the event.
- Step 3: The system generates a notification message containing relevant details (e.g., appointment time, date, barber name, service type).
- Step 4: The system sends the notification through the user's preferred channel (e.g., email, SMS, in-app notification).
- Step 5: The system confirms the notification has been sent and logs the notification details.

2. **Alternative Flow**

- A1: If the user has multiple notification preferences set, the system sends notifications via all selected channels (e.g., both SMS and email).

- A2: For reminders, the system sends notifications a set time before the appointment (e.g., 24 hours prior), based on user preference.

3. **Exceptions**

- E1: If a notification fails to send (e.g., network error or invalid contact details), the system retries and, if unsuccessful, logs the failure and notifies the system admin or sends an alert to the user within the app.

- E2: If the user has no contact preference set, the system defaults to an in-app notification and suggests the user update their contact preferences.

Includes (other use case IDs):

- None

Notes/Issues:

- Consider allowing users to customize notification timing (e.g., reminders 1 hour, 6 hours, or 24 hours before appointments).

- Ensure notifications comply with data protection and privacy laws, especially for SMS and email notifications.

3.3.10 Withdraw Earning – U10

Author: Arman Salehi

Purpose:
Allow stylists to withdraw their earnings from completed appointments in the salon booking system.

Requirements Traceability:
Linked to the requirement for stylists to access and manage their earnings, ensuring transparency and ease of use in financial transactions.

Priority:
High - This functionality is essential for stylist satisfaction and trust in the platform.

Preconditions:
- The stylist must be logged in.
- The stylist must have a balance greater than or equal to the minimum withdrawal amount.

Postconditions:
- The withdrawal request is processed, and the balance is updated in the system.
- A confirmation notification is sent to the stylist.
- The earnings are transferred to the stylist’s registered bank account.

Actors:
- Stylist
- System (for balance updates and transaction processing)

Flow of Events

Basic Flow:

1. The stylist navigates to the “Earnings” section.
2. The stylist views their available balance and selects the option to withdraw earnings.
3. The system prompts the stylist to confirm the withdrawal amount.
4. The stylist confirms the withdrawal request.
5. The system processes the withdrawal, deducts the amount from the stylist’s balance, and transfers it to the stylist's bank account.
6. A confirmation notification is sent to the stylist.

Alternative Flow:

- **Partial Withdrawal:**
- If the stylist chooses to withdraw only part of their earnings, the system allows them to input the specific amount (within minimum and maximum limits).
- The system processes the partial withdrawal and updates the balance accordingly.

Exceptions:

- **Insufficient Balance:**
- If the stylist’s balance is below the minimum withdrawal amount, the system displays an error message and prevents the withdrawal.

Bank Account Not Set Up:

- If the stylist hasn’t registered a bank account, the system prompts them to enter their account details before proceeding with the withdrawal.
- **System Error:**
- If there is an issue processing the withdrawal or updating the balance, the system displays an error message and advises the stylist to try again later.

Includes Other Use Cases:

- **Notification:** Sending a confirmation notification to the stylist upon successful withdrawal.
- **Transaction Processing:** Handling the financial transaction and updating the stylist's balance.

Notes:

- Withdrawal limits and processing times should be communicated to the stylist to manage expectations.
- Security measures, such as two-factor authentication, may be required to complete the withdrawal process for added safety.

4 Other Non-functional Requirements

4.1 Performance Requirements

1. Response Time:

- The system must process user requests with a maximum delay of **two seconds.** This applies to actions like viewing available appointments, booking appointments, and processing online payments.
- For payment transactions, the system should confirm and respond to requests within **five seconds.**

2. Scalability:

- The system must support **at least 1,000 concurrent users** without performance degradation, allowing for increased users over time.
- Infrastructure should accommodate future scaling to maintain performance even with higher traffic.

3. Throughput:

- The system should process **at least 200 booking requests per minute** to handle high-demand periods efficiently.
- Each booking and payment transaction should be managed with minimal delay and interference, ensuring smooth handling of concurrent requests.

4. Reliability and Availability:

- The system must be available **99.9% of the time,** especially during peak usage hours, with measures in place to prevent unexpected downtime.
- In the event of a system failure, it should recover to normal operations within **ten minutes.**

5. Data Storage and Retrieval:

- Data access times for user information, booking history, and payment statuses should be less than **one second,** enabling users to quickly retrieve essential information.
- The system should be designed for continuous data storage and retrieval, with fast access to historical data when needed.

6. Efficiency in Resource Usage:

- The system must efficiently use hardware resources such as memory and CPU to maintain stability without unnecessary consumption.
- Resources should be managed to ensure consistent performance even during maximum user activity.

4.2 Safety and Security Requirements

1. Data Privacy and Protection:

- All user data, including personal information, booking details, and payment information, must be stored securely and encrypted to prevent unauthorized access.
- Compliance with relevant data protection laws, such as GDPR (General Data Protection Regulation), must be ensured for any data storage, transmission, or processing involving user information.

2. User Authentication and Authorization:

- A secure authentication process, including password protection and optional two-factor authentication (2FA), should be implemented to verify user identities.
- Role-based access control (RBAC) will ensure that only authorized personnel (e.g., admins and stylists) can access sensitive areas and perform specific actions within the system.

3. Payment Security:

- Secure payment protocols, such as SSL/TLS encryption, must be used to protect data during online transactions.
- Payment processing must adhere to PCI-DSS (Payment Card Industry Data Security Standard) guidelines to ensure the safe handling of financial information.

4. Data Backup and Recovery:

- Regular backups must be performed to protect against data loss in case of system failures or other disruptions. Backups should occur daily, with storage offsite if feasible.
- A recovery plan should be established, allowing the system to restore data and resume normal operations within 10 minutes after any major failure or breach.

5. Network Security:

- Firewalls and intrusion detection systems (IDS) must be set up to monitor and protect the system from malicious attacks or unauthorized access.
- Security protocols, including IP whitelisting and secure VPN connections for remote admin access, should be applied to minimize network-based risks.

6. Logging and Monitoring:

- The system must implement detailed logging of access, transactions, and administrative activities, enabling the detection of unusual behavior or potential breaches.
- Security alerts should be configured to notify the admin team of any suspicious activity, allowing for quick investigation and response.

7. Session Management:

- User sessions must automatically expire after a set period of inactivity (e.g., 15 minutes) to reduce the risk of unauthorized access.
- Security tokens should be utilized to protect session data, and invalid tokens should automatically terminate sessions upon user logout.

4.3 Software Quality Attributes

1. Reliability:

- The system must perform consistently under specified conditions, handling bookings, cancellations, and payments accurately without errors.
- The system should maintain an uptime of **99.9%,** ensuring that service interruptions are minimal, especially during peak hours.

2. Usability:

- The interface must be user-friendly and intuitive, enabling users to book appointments, view available stylists, and make payments with minimal training or guidance.
- The design should accommodate both desktop and mobile devices, offering a responsive experience for various screen sizes.

3. Maintainability:

- The codebase should be modular and follow established coding standards to facilitate easy maintenance and updates.
- Documentation must be provided for both the code and system design to support future developers and ensure that any changes or enhancements can be implemented smoothly.

4. Scalability:

- The system architecture should support future expansion, allowing for an increased number of users, stylists, and services without a decline in performance.
- Additional servers or resources should be easy to add to handle increased traffic if needed.

5. Security:

- The system must adhere to strict security measures, such as data encryption and secure authentication, to protect user data and prevent unauthorized access.
- Regular security audits should be conducted to identify and address potential vulnerabilities, keeping the system safe from cyber threats.

6. Performance:

- The system should provide a response time of **two seconds or less** for user actions such as booking or viewing schedules, and **five seconds or less** for payment processing.
- Even with peak user activity, the system should maintain efficient performance and avoid lag or slow loading times.

7. Flexibility:

- The system should be designed with flexibility in mind, allowing customization of features (such as services offered or pricing) as the salon's needs evolve.
- It should support integration with additional third-party services or plugins if new functionalities are required in the future.

8. Testability:

- The software design should support easy testing of modules and overall functionality, with test cases covering all major features and use cases.
- Automated testing should be enabled where possible to quickly identify issues during development and after updates.

Appendix A – Case Study

1. Introduction

The salon industry is highly competitive, and customers often experience challenges in booking appointments, finding the right stylist, and managing schedules conveniently. This project aims to create an online salon reservation system that streamlines the booking process, ensures availability management for stylists, and offers a seamless payment solution for clients. By addressing these needs, the system enhances the overall customer experience and supports stylists in managing their appointments and earnings efficiently.

2. Background

In recent years, the demand for on-demand and flexible booking services has increased, especially in the beauty and wellness sector. Traditional methods of booking appointments via phone calls or walk-ins can be inconvenient for both customers and salon professionals. This case study is inspired by the need for an accessible, easy-to-use platform for reserving salon services, inspired by an increasing trend in digital transformation within the service industry.

3. Objective of the System

The salon reservation system has three core objectives:

- Simplify the Booking Process: To allow customers to browse available stylists, view available slots, and book appointments online without hassle.
- Support Stylist Availability Management: To give stylists control over their schedules, allowing them to update their availability easily.
- Streamline Payment and Earnings Distribution: To enable secure payment processing for customers and facilitate earnings withdrawal for stylists.

4. Target Users

The system has three main user groups:

- **Customers:** Individuals who wish to book salon services based on the stylist's availability and location.
- **Stylists:** Professionals who work at the salon and need an organized way to manage their schedules and track their earnings.
- **Admin:** The system administrator who oversees and manages system functionality, ensuring smooth operations.

5. Challenges

The salon industry faces several challenges that this system aims to solve:

- Inconsistent Booking Methods: Traditional booking methods are time-consuming and can lead to miscommunications or overbooking.
- Manual Schedule Management for Stylists: Many stylists struggle to balance their schedules due to a lack of structured systems, leading to missed appointments or double bookings.
- Delayed Payments: In some cases, stylists face delays in receiving payments, which can affect their financial planning and satisfaction with their workplace.

6. Proposed Solution

The salon reservation system addresses the above challenges with a comprehensive digital platform:

- User-Friendly Booking Interface: Customers can easily search for stylists, select available time slots, and confirm bookings with just a few clicks.
- Availability and Schedule Management for Stylists: Stylists can update their schedules, set breaks, and view upcoming appointments, ensuring better time management.
- Automated Payment System: The system integrates with a local payment platform (ZarinPal) to handle payments securely, allowing customers to pay online and stylists to withdraw their earnings directly.

7. Implementation Strategy

The project will be developed in three phases:

- Phase 1: Designing the system architecture and database, focusing on the reservation and payment modules.
- Phase 2: Developing a user interface that is intuitive for all user groups, with particular emphasis on ease of use for both customers and stylists.
- Phase 3: Implementing testing, feedback, and security measures to ensure the system is secure and efficient.

8. Expected Outcomes

The salon reservation system is expected to achieve the following:

- Increase in Customer Convenience and Satisfaction: By reducing booking times and improving appointment reliability, customer satisfaction is expected to rise.
- Enhanced Schedule Management for Stylists: Stylists will be able to organize their appointments with minimal errors or conflicts, reducing missed bookings.
- Improved Financial Management: Automated payment processing ensures that stylists receive their earnings promptly and securely, improving financial planning.

9. Conclusion

By digitalizing the booking and payment process, the salon reservation system creates a seamless experience for both customers and stylists, streamlining operations and reducing administrative overhead. This system is a practical example of how technology can transform traditional services, offering enhanced convenience, security, and efficiency in the beauty and wellness industry.