Online Salon and Spa Booking System

Software Requirements Specification

Software Engineering Project

Version 1.0

Group Id: PID10

Group Members: 2018-CS-05, 2018-CS-40, 2018-CS-41

Supervisor Name : Ma'am Taliah

Department of Computer Science
University of Engineering and Technology Lahore

Table of Contents

1 Introduction

In this SRS we will discuss various topics about our project including scope,functionality, interface and technologies.

1.1 Scope

Mainly the scope of this SRS covers the domain of all the functionalities on the basis of which this system is being developed. The major aspect of developing this project is to smoothen the process of salon bookings. From a layman user to a technical user this project is going to serve the major purpose for all. In this SRS we are not only going to define our project but also the major workflow and usage of technology making it easier for everyone to understand.

1.2 Glossary

Term	Definition
Stakeholder	Any person with an interest in the project who is not a developer.
User	Reviewer or Author.
Constraints	Limitations to perform something

1.3 Technologies to be used

Technology	Description
Preceden	Preceden is a lightweight timeline maker that helps you quickly create great looking timelines and project roadmaps.
Adobe Illustrator	Illustrator is an easy to learn software that is majorly used to design illustration and a lot of other stuff. Hence we have designed wireframes for the project to drive a basic picture of how the screens are work.
VSCode	**Visual Studio Code** is a free source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.
Xampp	XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and

	interpreters for scripts written in the PHP and Perl programming languages.
CodeIgniter	CodeIgniter is an open-source software rapid development web framework, for use in building dynamic web sites with PHP.

2 General Description

2.1 User Characteristics

Basically, we have two users for this project:

- Customers
- Beauticians

These users are expected to know how to navigate across screens and look for a certain functionality. They are familiar with how the UIUX thing works throughout the system.

Customers:
We do not have an admin for the system, because it is not an LMS where we are required to maintain a lot about students and teachers. Customers' ease of usage is our first priority. Hence the customer will be able to just approach our appointment system, book their appointments and then appear on the appointed date. They can also opt for various options like priority for their appointments. Customers can also give ratings to a specific Salon and Beautician.

Beauticians:
Beauticians' major functionality is to take care of their profile for the customers, so that customers can select a specific beautician on the basis of their profile. Beauticians should also have basic knowledge of how the system works and then they are good to go.

2.2 General Constraints

Initially, the project's major focus is to go without using admin and to work for only one branch, later on it can be expanded to further branches or it might need an administrator in the future.

2.3 Assumptions & Dependencies

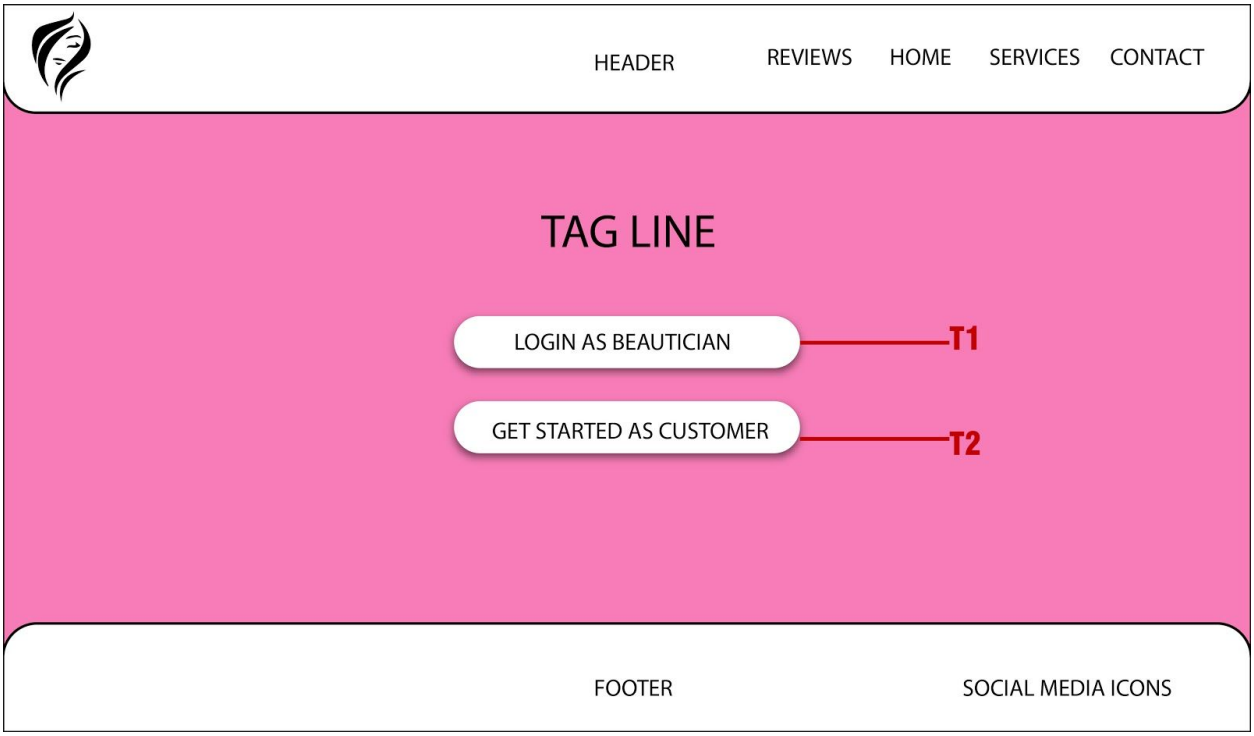
Following are some basic assumptions for this system:

- Since it is a web based project, basic internet speed of loading a web page is a critical need because all other things are dependent on it.
- Whatever data the users of the system add should be kept confidential. One can say there should be a security layer above that data so only authorized users can access it.
- As a customer there should be at least one spot available to book a beautician.

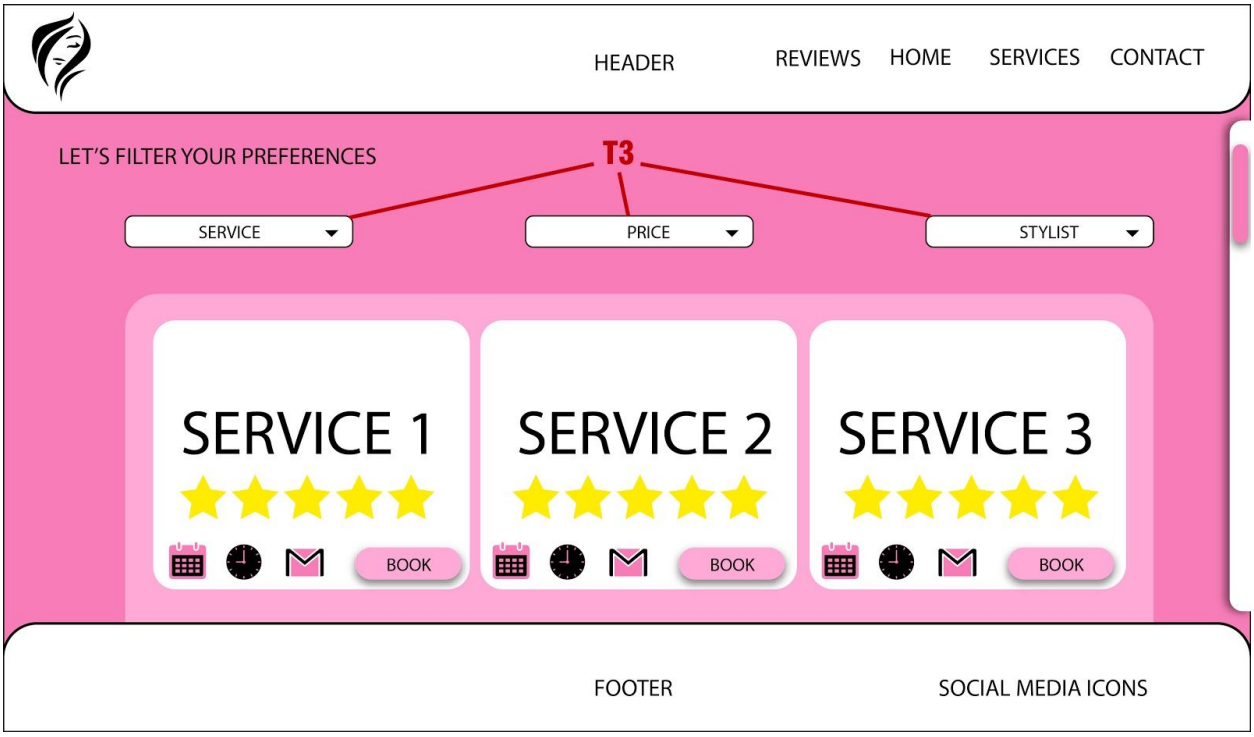
3 Specific Requirements

3.1 External Interface Requirements:

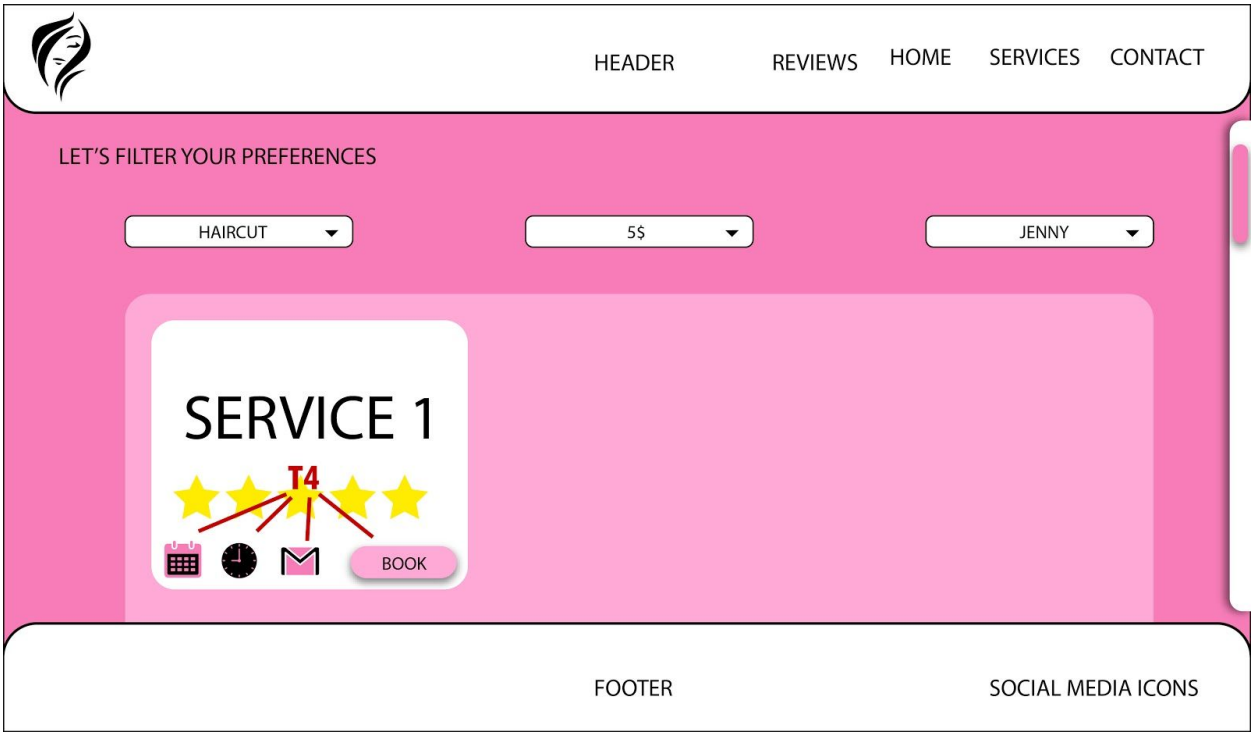
3.1.1 User Interfaces



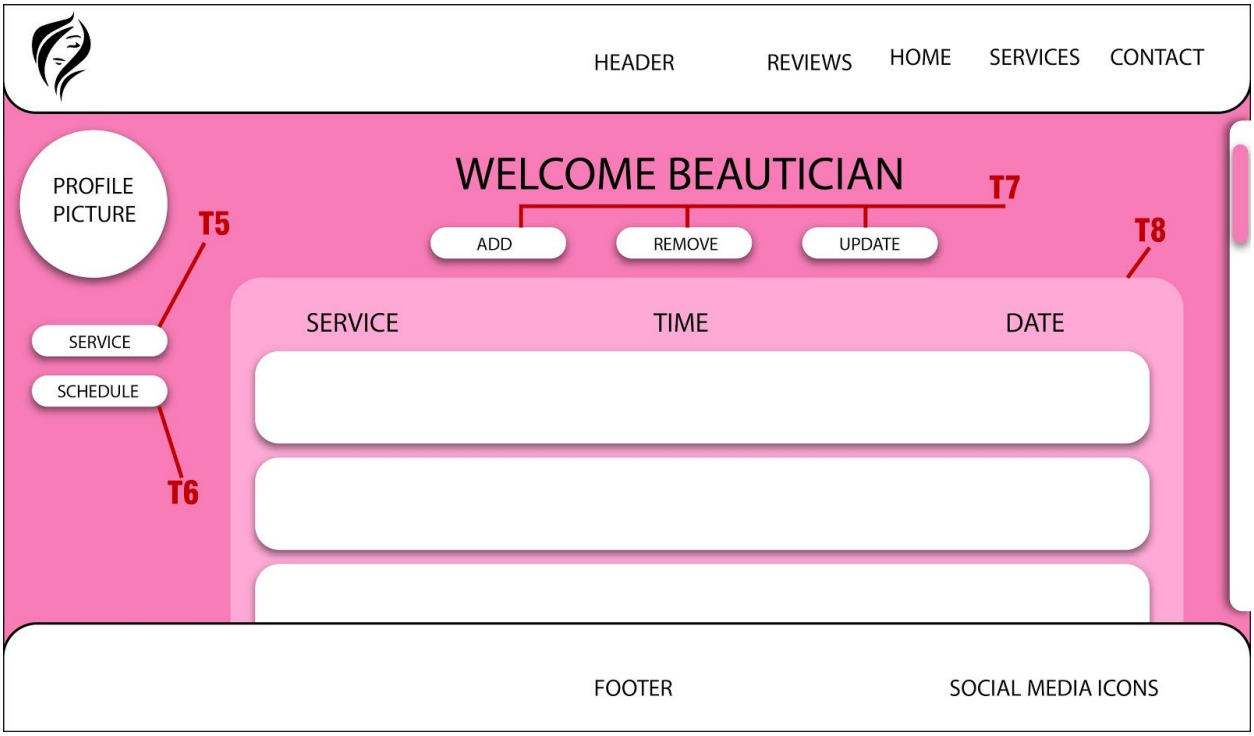
T1: As a beautician, I shall be able to login to my profile.
T2: As a customer, I shall be able to see the salon services and book appointments without any login.



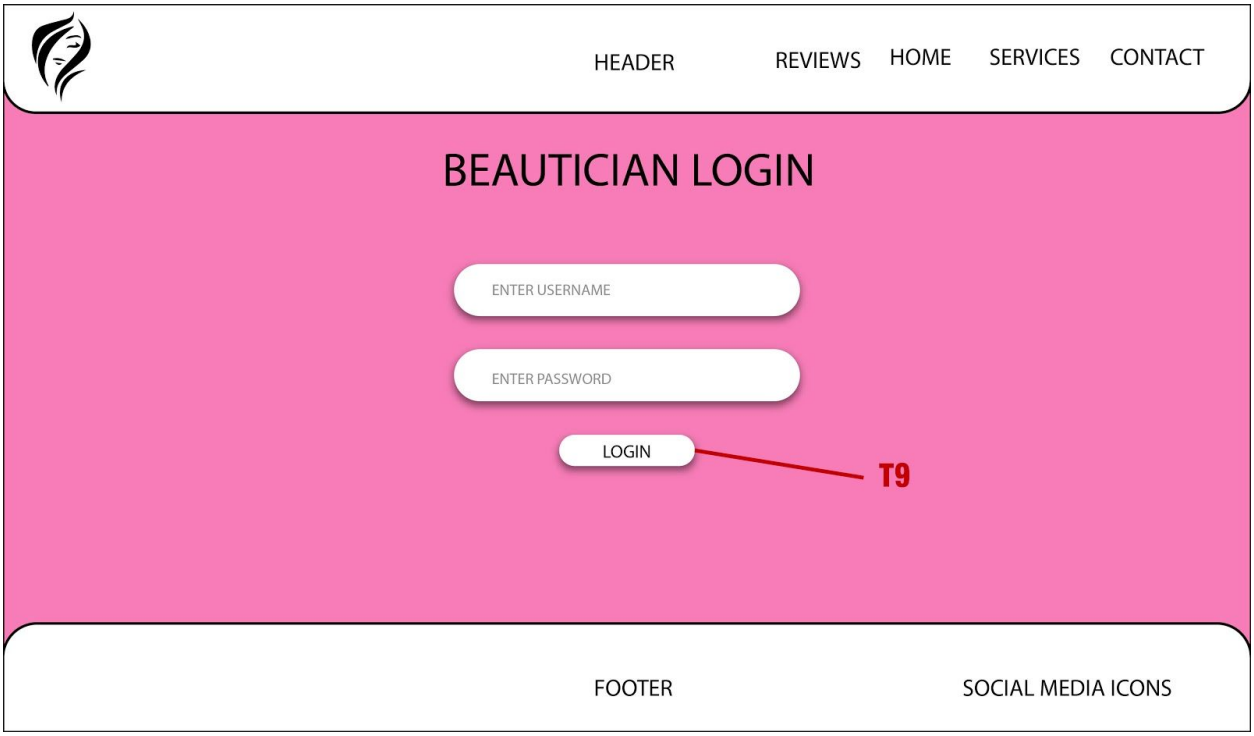
T3: As a customer, I shall be able to filter the services on the basis of service name, price and stylist.



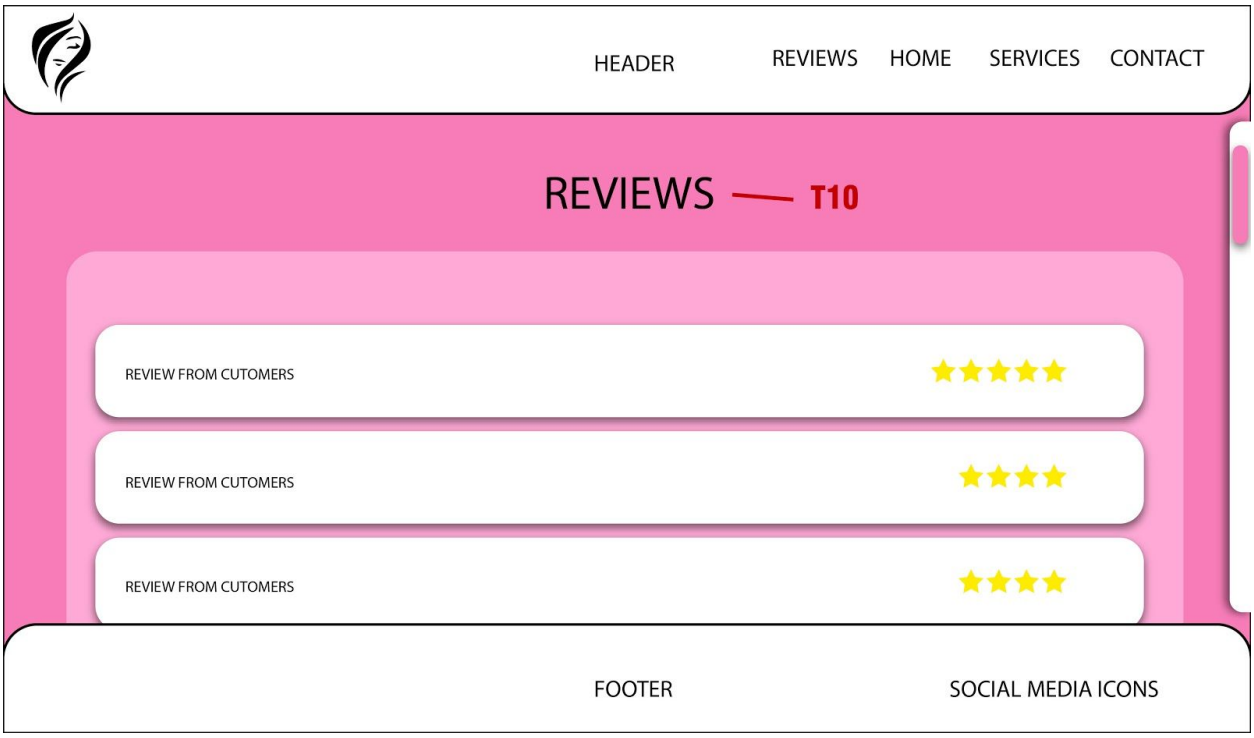
T4: As a customer, I shall be able to book an appointment according to the desired date and time and by giving my email.



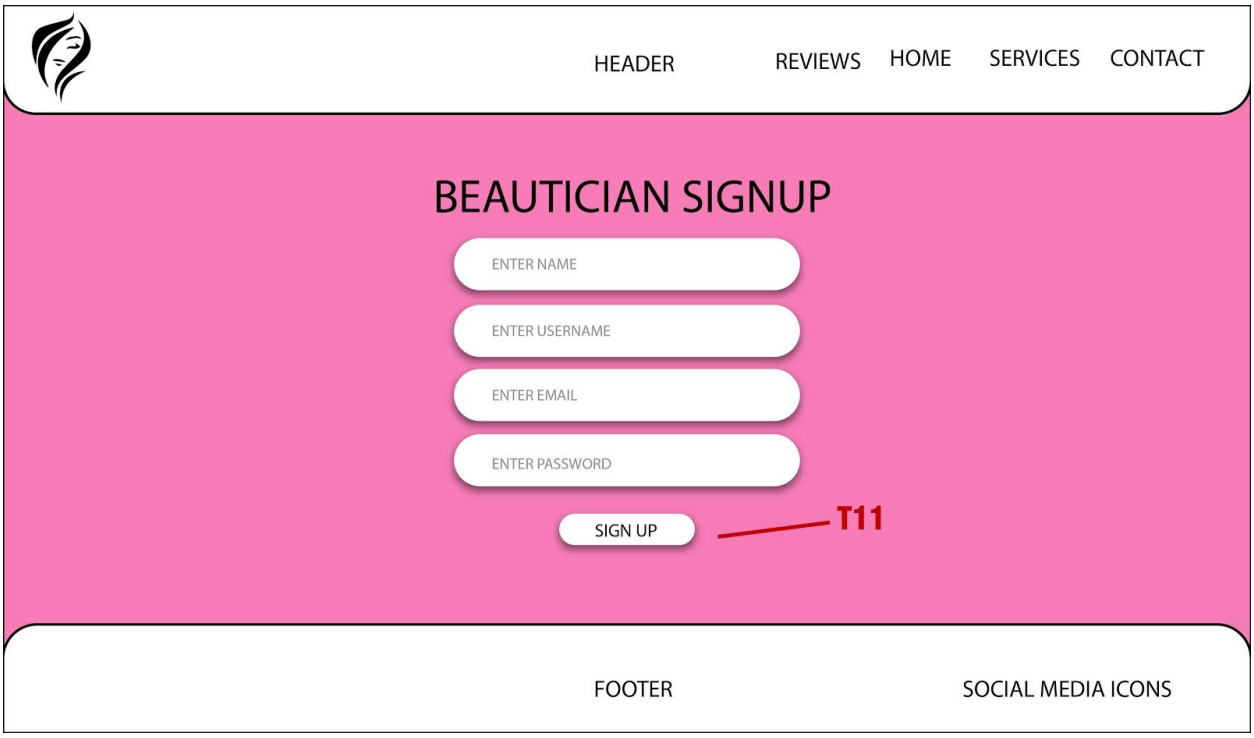
T5: As a beautician, I shall be able to view my services panel.
T6: As a beautician, I shall be able to view my schedules with various customers which can be further edited like reschedule. This rescheduling will automatically inform the customer that the appointment has been rescheduled.
T7: As a beautician, I shall be able to add, remove and update the services that I provide..
T8: As a beautician, I shall be able to view and scroll down the list of all of my schedules.



T9: As a beautician, I shall be able to login to my account by entering my username and password.



T10: As a customer, I shall be able to see the reviews.



T11: As a beautician, I shall be able to create my account and maintain my profile.

3.2 Functional Requirements

A Functional requirement document defines the functionality of a system .It depends upon the type of software, expected user and the system where the software is used.

3.2.1 Beautician

Beauticians will be able to login.
Beauticians will be able to create services.
Beauticians will be able to maintain (Update/Delete) their details such as price,hours etc.
Beauticians will be able manage their appointments.

3.2.2 Customers

Customers will be able to make appointments
Customers will provide their information (name,e-mail address)
Customers can give feedback by giving star to the service after availing it.

3.3 Use Cases

Use Case Title	**Schedule Appointment**
Use Case Id	1
Requirement Id	1
Description: This use case is about scheduling appointments by customers.	
Pre Conditions: 1. Databases should load all the services that the salon is offering to the user. 2. The user should be able to view this page containing services. 3. All offered services should be functional.	
Normal Flow	
1. Customer navigates to the service window successfully.	
2. Customers can schedule the service.	
Alternative Flow	
1a. There is a problem in the loading of services ○ Customer must reload the page. Customer continues from step 2.	
Post Conditions: – Customer successfully reaches the services page for scheduling.	
Open issues: What if the internet connection is lost. What if the web page gives error of 404	
Authority: Customer	

<Provide here the usage scenarios of all use cases in table format explaining Use Case title, Use Case Id, Actions, Description, Alternative Paths, Pre and Post Conditions, Author, Exceptions. You are supposed to provide a usage scenario for each use case shown in the use case diagram. Following is an example for a usage scenario>

Use Case Title	**Add Service**
Use Case Id	2
Requirement Id	2
Description: This use case is about adding a new service to the system .	
Pre Conditions: Beautician must have a profile Databases should be available in online mode.	
Normal Flow	
1. Beautician log in to the account (if already exists).	
2. If the beautician is new he/she must sign up.	
3. Beautician clicks the button to add a new service.	
4. System asks for the required information.	
5. System after confirmation adds the service to the system.	
Alternative Flow	
1a. There is a problem in the credentials; some credentials need to be corrected. ● Forgot password functionality or reset password should be given. Beautician continues from step 3. 4a. There is a problem in the data provided; some data is incorrect ● Beautician checks the entered data and corrects the error Beautician continues from step 5	
Post Conditions: - A new user service is successfully added.	
Open issues:	
Authority: Beautician	

Use Case Title	**Add Beautician**
Use Case Id	3
Requirement Id	3
Description: This use case is about adding a new beautician to the salon.	
Pre Conditions: Databases should be available in online mode.	
Normal Flow	
1. Beautician opens the sign up window.	
2. System asks for the required information.	
3. System after confirmation adds the service to the system.	
Alternative Flow	
1a. There is a problem in the data provided; some data needs to be corrected. ● Beautician checks the entered data and corrects the error. Beautician continues from step 3.	
Post Conditions:	

– A new user beautician is successfully added.
Open issues:
Authority: Beautician

Use Case Title	**Update Service**
Use Case Id	4
Requirement Id	4
Description: This use case is about updating an existing service in the system .	
Pre Conditions: 1. Beautician must have a profile. 2. The service to be updated must exist. 3. Databases should be available in online mode.	
Normal Flow	
1. Beautician log in to the account.	
2. He/She chooses the service.	
3. Beautician clicks the button to update the service.	
4. System asks for the required information.	
5. System after confirmation updates the service to the system.	
Alternative Flow	
4a. There is a problem in the data provided; some data needs to be corrected. ● Beautician checks the entered data and corrects the error. Beautician continues from step 5.	
Post Conditions: ● The service is successfully updated.	
Open issues:	
Authority: Beautician	

Use Case Title	**Delete Service**
Use Case Id	5
Requirement Id	5
Description: This use case is about deleting a service from the system .	

Pre Conditions: 1. Beautician must have a profile. 2. The service to be deleted must exist. 3. Databases should be available in online mode.
Normal Flow
1. Beautician log in to the account.
2. He/She selects the service.
3. Beautician clicks the button to delete the service.
4. System asks for the confirmation.
5. System after confirmation deletes the service from the system.
Alternative Flow
Post Conditions: ● The service is successfully deleted.
Open issues:
Authority: Beautician

Use Case Title	**Giving review**
Use Case Id	6
Requirement Id	6
Description: This use case is about giving the review to the service .	
Pre Conditions: Customer must have availed that service. Databases should be available in online mode.	
Normal Flow	
1. Customer schedule a service.	
2. Once the time of service has passed, the review button becomes active .	
3. Customer clicks the button to add review to the service.	
4. System asks for the required information.	
5. System after confirmation adds the review to the service..	
Alternative Flow	

Post Conditions: ● A review to the service is successfully added.
Open issues:
Authority: Customer

3.4 Non Functional Requirements

Basically, non-functional requirements relate to qualities of the system that cut across user facing features, such as security, reliability, and performance. They are mostly considered optional.

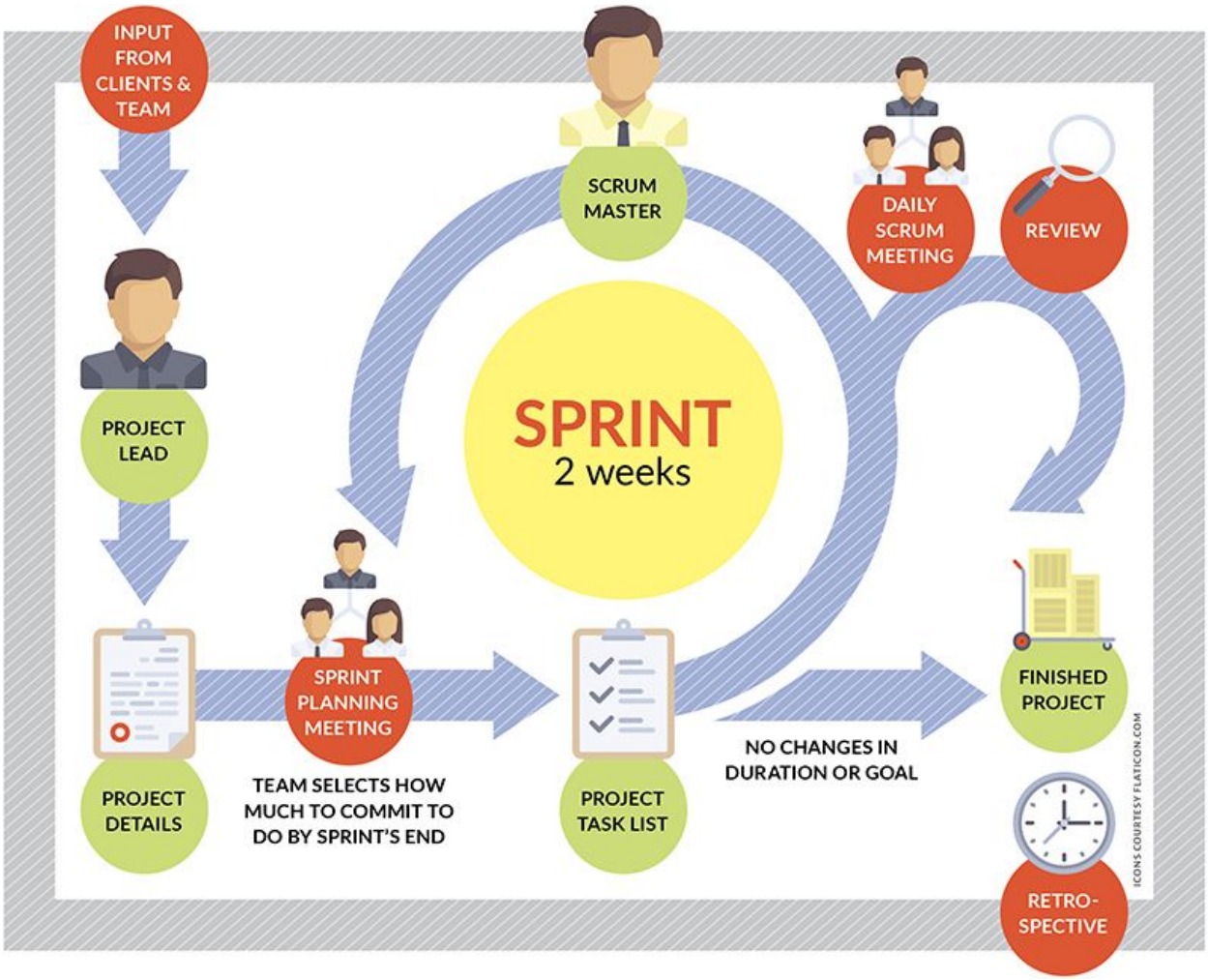
1. Accessibility – The system is accessible anywhere at any time by the authorized users.
2. Accuracy – The correctness of data inputs to the system was ensured.
3. Availability – System is available within working hours.
4. Efficiency – Users were given the facility to perform the salon management processes correctly.
5. Effectiveness – Users were given the facility to perform correct salon management processes via the suggesting system.
6. Maintainability – This is a considerable factor especially for a non-technical user. Maintainability of the system is not more complex.
7. Privacy – The confidentiality of the data inputs to the system has been assured.
8. Reliability - Ability of the suggested system to function under stated conditions for a specified period of time has been assured.
9. Security – The data feeds to the system has been protected by controlling the user access privileges.

4 Methodology

4.1 Adopted Methodology

After a lot of study we have finally decided to adopt SCRUM as a methodology for our project that will allow us to cope with our changing requirements with the passage of time. Scrum is an agile methodology or framework used primarily for software development projects with the goal of delivering new software capability every 2-4 weeks. It is one of the approaches that influenced the Agile Manifesto, which articulates a set of values and principles to guide decisions on how to develop higher-quality software faster. As compared to waterfall, agile methodology works in a different way. Here, a version of the project is released after every iteration. This is called SPRINTS. It offers flexibility of covering up missing requirements in the next phase (whereas possible). Agile methods break tasks into smaller iterations, or parts. It does not directly involve long term planning. The project scope and requirements are laid down at the beginning of the development process. Plans regarding the number of iterations, the duration and the scope of Each iteration is clearly defined in advance.

4.2 Diagram of Methodology



4.3 Reason of Choosing Methodology

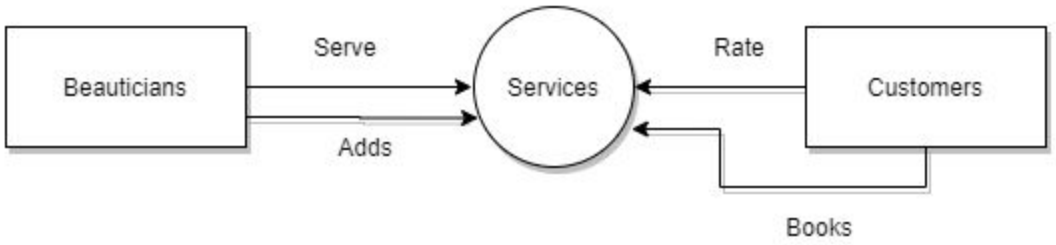
The basic reason for choosing this method is to deal with uncertain changes in the requirements. The division of the entire project into smaller parts helps to minimize the project risk and to reduce the overall project delivery time requirements.

5 Project Work Plan

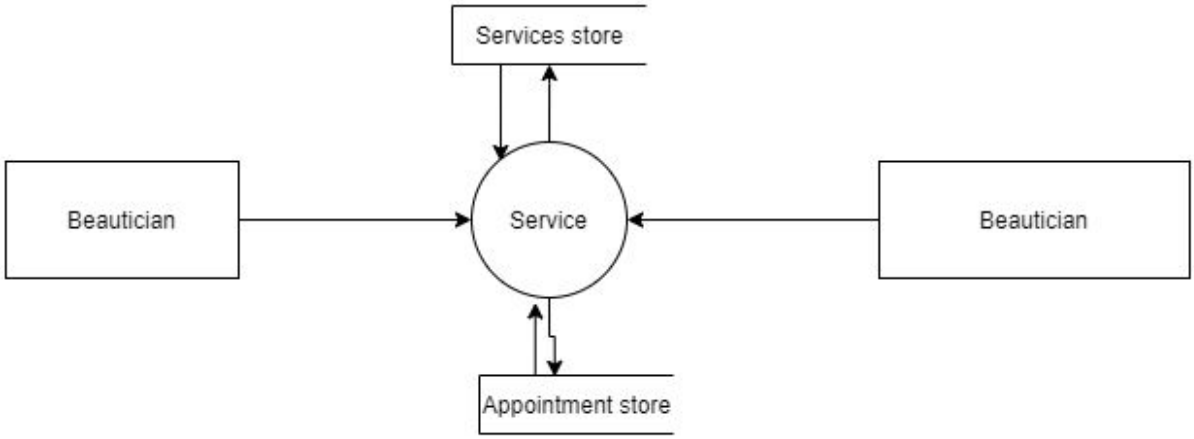
6 Design Phase

Data Flow Diagram

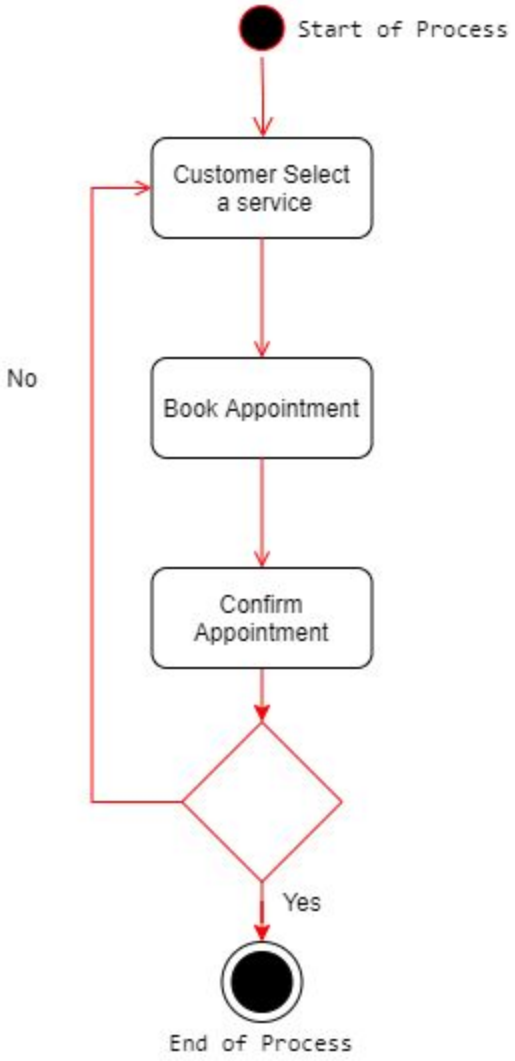
Level 0



Level 1


Services store
Beautician
Service
Beautician
Appointment store

Activity Diagram



Sequential Diagram