**1. Introduction**:

**i. Introduction of the System:**

Expertis is an application that helps users to find the Salon and beauty parlours around by searching pin code, address, or location. The application allows users to find the best service centers in various locations based on the reviews and ratings. The primary objective of the platform includes booking a service or an appointment.

In a modern busy day-to-day life, it is always convenient to look for services that one requires through mobile phones is the best practice to save time and effort. Expertis is designed using cutting-edge technologies that help users to find solutions at their fingertips. The app is user-friendly, convenient, free of cost, and adds value to its users' needs.

1. Project Title:

Expertis

1. Category:

Cross Platform Application

1. Overview:

**ii. Objectives of the system**

* To help find the Salons and Beauty parlours nearby
* Easily choose the best center/shop based on information, reviews, and ratings
* Users can directly book appointments for their needs without hassling which helps save time

* It helps local businesses to increase more leads and sales that generate excellent opportunities as well as income

**iii. Scope of the system**

* Transforming the traditional businesses to online

* Reducing the waiting time for salon services

**iv. Software/Hardware used for the development**

* **Software Requirements:**
* Front end: Flutter
* Back end: Node.js
* Operating system: Windows 10g
* Developing tool: Visual studio code
* Database: Mongo DB
* **Hardware Requirements:**
* Processor: Intel(R) Core(TM)i3
* Ram: 4.00GB
* Hard Disk: 40GB
* Keyboard
* Mouse

**2. SRS**

**Introduction (Brief write-up about SRS)**

A software requirements specification (SRS) is a document that captures a complete description of how the system is expected to perform. It is usually signed off at the end of the requirement engineering phases. An SRS minimizes the time and effort required by the developer to achieve desired goals and reduces the development cost. A good SRS defines how an application will interact with system hardware, other programs, and human users in various real-world situations.

**ii. Overall Description**

1. **Product perspective**

This project provides an effective way to handle all the beauty & well-being needs of the customer. Helps the customer to easily find out the required services, in less time. The project is intended to reduce the time required to avail of beauty services. It helps customers to easily book the service on time as per their requirements early and helps the shop owners effectively manage their customers. Enabling holding the customer and providing ease of service to the customer. This website relay on the shop owners to get the data and there is no external data require

1. **Product Functions:**

A general abstract description of a function to be performed by a product is given in the product function. This system contains modules like User Registration & login. User verification, password recovery, and user profile for handling the User. Shop Registration, Adding services and managing the shop information in the shop modules. The appointment module handles the appointments by the users and to the shops. Users can review the shop and services.

**Users:**

**Customer**

* Registration: The user needs to register first with their basic registration first with their basic registration details and needs to create a valid login id and password.
* Login: Using valid login credentials, the user needs to log in to the website in order to log in to the website in order to enter into the website.
* Change password: This module allows users to reset their password.
* Update user details: This module Update user details.
* Search by selecting a category: The name Search by selecting the category module allows users to search services by selecting the different categories.
* View Reviews: The View Review module allows customers to view shops Comments.
* Review the store/service: This module allows the user to give feedback about the shop based on the user experience.
* Views/Make Appointments: The customer can view or make the appointment.

**Shop:**

* Registration: User needs to register first with their basic registration details and needs to create a valid login id and password.
* Login: Using valid login credentials, the user needs to log in the website in order to enter into the website.
* Change Password: This module allows users to reset their password.
* Add/Update/Delete Services: Shop can add or update/delete the services.
* View/Receive Appointments: Shop can view or make the appointments.
* Update user details: This module updates user details.

**Admin:**

* Registration: Admin needs to register first with their basic registration details and needs to create a valid login id and password.
* Login: Using valid login credentials, the admin needs to log in to the app in order to proceed further.
* Change password: This module allows the admin to reset their password.
* View users: All the registered user details will be displayed to the admin.
* View and manage Shops:
* View hop: Admin can view all the registered shops online with their details.
* Update shop: In the Update module can the admin update the shop with some shop details.
* Delete shop: The Delete module allows the admin to delete the existing shops.
* Reviews: The View Review module allows the admin to view customer reviews on shops and services.
* Manage category: Admin can manage the category of shops.
* Manage report: This module allows the admin to manage the reports generated daily/weekly.

**C.** **User characteristics:**

The user needs to have basic computer knowledge to operate the system. The types of users and their characteristics are

**Admins:**

The admin has a high level of authority compared to other users. He has the authority to change the passwords and to create a new admin. Oversee all the data flow in the system. Performs necessary action to data if any kind of malfunction happens in the system.

* **Shop Owners:**

The shop owner should have basic knowledge of working with devices. Should be able to respond to clients' requests on time.

* **Users:**

Users can find a service and book an appointment and provide feedback/reviews.

**D. General Constraints:**

* GUI is only in English.
* Requires all users’ registration.
* Requires all mandatory fields to be filled with valid information.
* Minimum capacity to run the application.

**E. Assumptions:**

* Users should be familiar with using mobile applications.
* The information provided by the shop is assumed to be genuine and trustworthy.Manage report: This module allows the admin to manage the reports generated daily/weekly.
* The user has an active internet connection.
* OTP will expire after 5 minutes.

**iii. Special Requirements (Software / Hardware – if any)**

1. **External interface requirements:**

It specifies all the interfaces of the system to the people, other systems, and, hardware.

* **User interface:**
* Front End: Flutter
* Back end: node.js
* **Hardware interface:**
* Processor: Intel(R) Core(TM)i3
* RAM: 4.00GB
* HardDisk: 40GB
* **Software interface:**
* Operating System: Windows 10
* IDE’s: Visual Studio Code, Android Studio.
* Application: Chrome
* **Communication interface:**

The user must connect to the internet to access the application to allow the Email interface to send emails.

**iv. Functional requirements:**

**a. User module: 4.1**

**4.1.1 Registration**

4.1.1.1 Email String

4.1.1.2 Name String

4.1.1.3 Phone Number

4.1.1.4 Password String

**4.1.2 Login**

4.1.2.1 Email String

4.1.2.2 Password String

**4.1.3 Forgot Password**

**4.1.3.1 Send Otp**

4.1.3.1.1 Email String

4.1.3.1.2 OTP String

**4.1.3.2 Change Password**

4.1.3.2.1 New Password String

4.1.3.2. 2 Confirm Password String

**4.1.4 Reset Password**

4.1.4.1 Old Password String

4.1.4.2 New Password String

4.1.4.3 Confirm Password String

**b. Shop module: 4.2**

**4.2.1 Registration**

4.2.1.1 Owner String

4.2.1.2 Phone Number

4.2.1.3 Email String

4.2.1.4 Password String

4.2.1.5 Shop Name String

4.2.1.6 Address String

4.2.1.7 Logo String

4.2.1.8 Gallery String

4.2.1.9 Location String

**4.2.2 Login**

4.2.2.1 Email String

4.2.2.2 OTP String

**4.2.3 Forget Password**

**4.2.3.1 Send Otp**

4.2.3.1.1 Email String

4.2.3.2.2 OTP String

**4.2.3.2 Change Password**

4.2.3.2.1 New Password String

4.2.3.2.2 Confirm Password String

**4.2.4 Reset Password**

4.2.4.1 Old Password String

4.2.4.2 New Password String

4.2.4.3 Confirm Password String

**4.2.5 Services**

4.2.5.1 Service Name String

4.2.5.2 Price Number

4.2.5.3 Photo String

4.2.5.4 Time Date-time

4.2.5.5 Description String

**V. Design Constraints**

* **Hardware Constraints:**
* Processor: i3 processor
* RAM:4GB
* Hard disk: 40GB
* **Software Constraints:**
* Flutter, Nodejs, MongoDB
* Fault tolerance: Fault tolerance is achieved by every input data. If the data fails during its validation, then the input will be discarded. Only correct information will be sent.
* Standard compliance:  The software has a graphical user interface to ease users book appointments online so it requires the internet.

**Vi. System Attributes**

**Reliability:**

* The main pillar of the reliability of the system is the backup of the database continuously maintained and updated to reflect the most recent changes.
* The system provides all databases on redundant applications with automatic switchover.
* The reliability of the overall program depends on the reliability of the separate components

**Availability:**

* The system should be available at all times, meaning the user can access it using the application, only restricted by the downtime of the server on which the system runs.
* In case of a hardware failure or database corruption, backups of the database should be retrieved from the server and saved by the administrator.
* Then the service will be restarted. It means 24/7 availability.

**Security:**

* Security of the app managed with privacy as the main goal. Security of the system is achieved by user name and password. The only authorized user can access the website.
* The system backend server shall only be accessible to authenticated administrations

**Maintainability:**

The admin will maintain the whole application. In case of failure, re-initialization of the program will be done. Software design is done with modularity in mind so that maintainability can be done easily.

**Portability:**

* The application is portable to any system having 4Gb RAM for better performance
* The application is a flutter-based cross-platform app so the end-user part is fully portable.
* Any web browser should be able to run the features of the system. The system can be used on any platform

**Vii. Other Requirements (if any)**

* Databases should use sharing to be redundant to prevent loss of data.
* Backups of the databases should be done hourly and be kept for one week.

**3. System Design (Functional Design)**

**i. Introduction**

System design is a solution to creation of a new system. This important phase is compares to

Several steps. It provides the understanding and procedural details necessary for implementing the system recommended in the feasibility study. Emphasis is on translating the performance requirements into design specification. Design goes through logical and physical development. Logical design previews the present physical systems, prepare input and output specification and detail the implementation plan and internal logic of the modules. The physical design maps out the details of the physical system, plans the system specifies any new hardware and software.

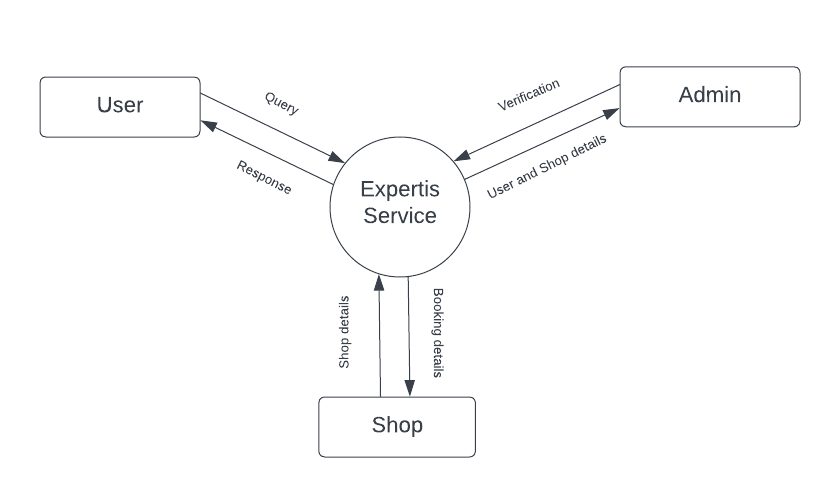
**ii. Assumption and Constraints**

* GUI is only in English.
* Requires all users’ registration.
* Requires all mandatory fields to be filled with valid information.
* Minimum capacity to run the application.
* Users should be familiar with using mobile applications.
* The information provided by the shop is assumed to be genuine and trustworthy Manage report: This module allows the admin to manage the reports generated daily/weekly.
* The user has an active internet connection.
* OTP will expire after 5 minutes.

**iii. Description of Programs**

1. **Context Flow Diagram (CFD)**

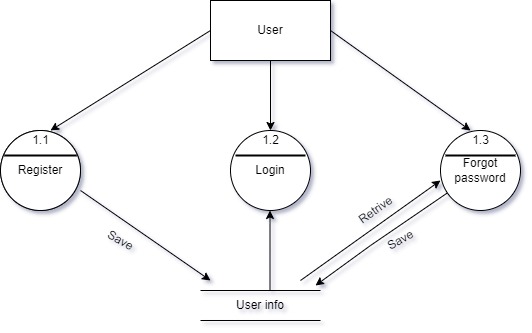
**CFD (Level 0):**



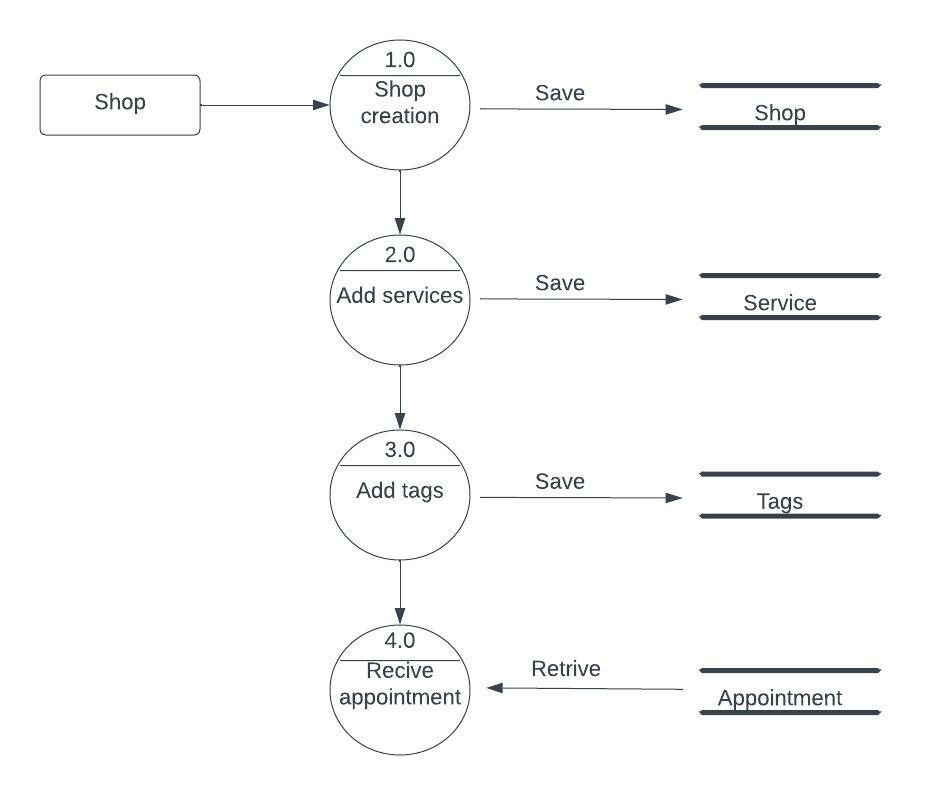
1. **Data Flow Diagrams**

**Level 1 (User)**

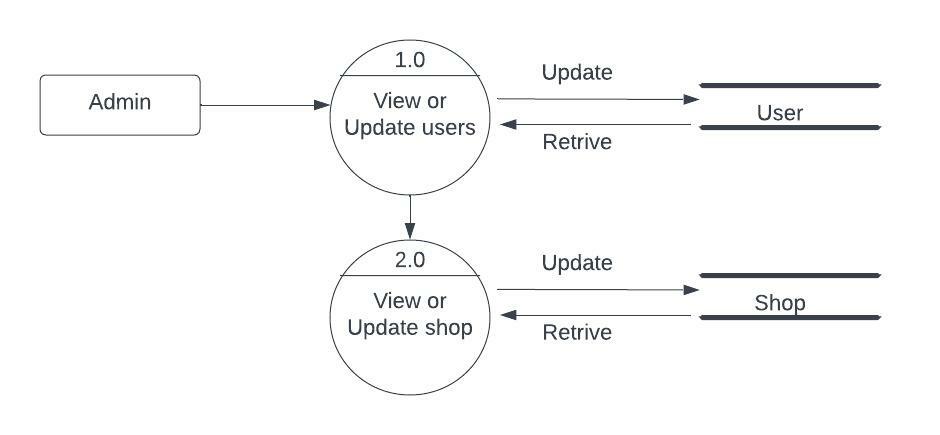
**Level 2 (User)**



**Level 1 (Shop)**



**Level 1 (Admin)**



**iv. Description of components**

* **Registration/login:**
  1. **Registration**

Allows new users to register on the application

**b. Login**

                       Helps existing users to login into the application

* **Shop Module:**

1. **create shop**

This allows shop owners to create their shop on Expertis and add services they provide

**b. Appointment**

Allows shop owners to book appointments with users

* **User Module:**

1. **Profile**

It allows users to create and update their profile

1. **Book appointment**

This module allows users to make an appointment

1. **Review**

This module allows users to add reviews to a service or shop.

* **Admin Module:**

1. **Verification**

This module allows the admin to verify registered shops.

**4. Database Design (or Data structure)**

1. **Introduction**

Database design is first step of design activities. That is conducted during software engineering.

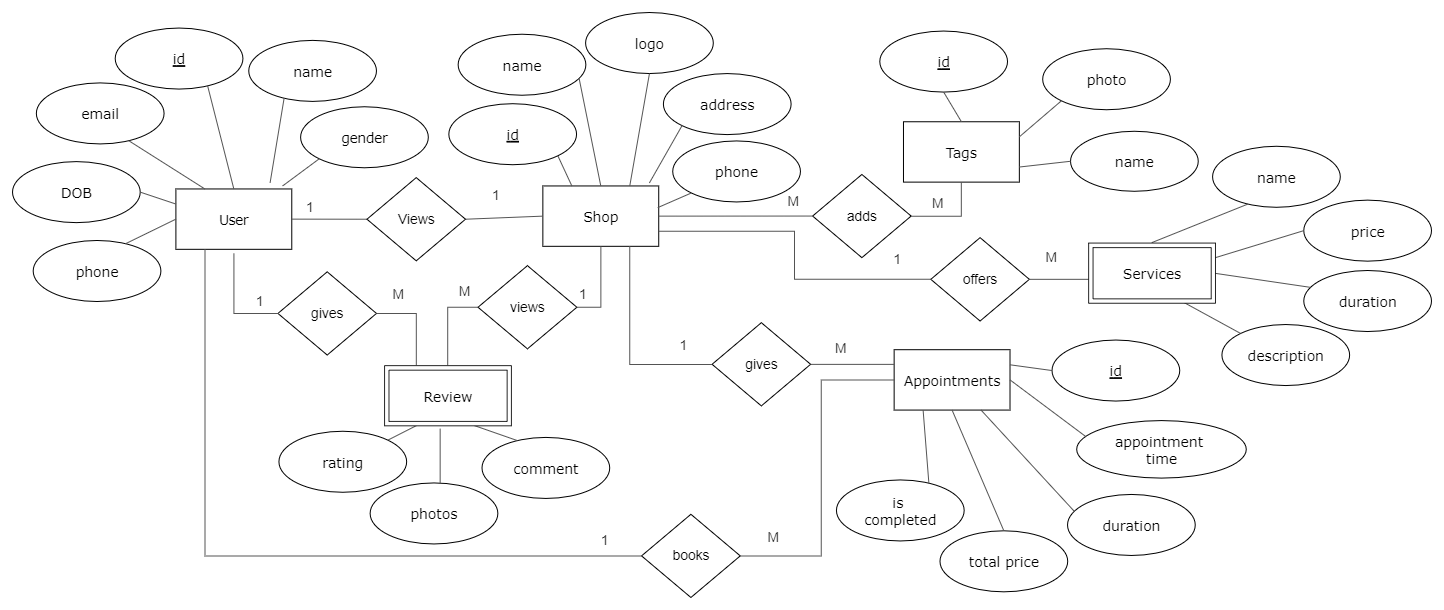
Regardless of the design techniques to be used, well designed data can lead to better structure, effective modularity, and reduced procedural complexity. The requirement analysis will reveal the actual data to be stored and the direction of flow of the data.

**ii. Purpose and Scope**

A good database design is, therefore, one that: Divides your information into subject-based tables to reduce redundant data. Provides Access with the information it requires to join the information in the tables together needed. Helps support and ensure the accuracy and integrity of your information.

**iii. Schema information**

**iv. ER diagram:**



**5. Detailed Design**

1. **Introduction**

Detailed design starts after the system design phase is completed. This basic goal in the detailed design is to specify the logic for the different modules that have been specified during the system design. The first step before the detailed design or code for module can developed is that the specification of the module be given precisely. Once the module is precisely specified the internal logic for the module that will implement the given specification can decided.

Detailed Design Description of the module adds much value and the programmer as step in coding will use experienced programmers feel that they can go directly to coding, so detailed design. A few techniques exist for detailed design. The most common are design walkthroughs and critical design reviews.

**ii. Structure of the software package**

**iii. Modular decomposition of the System**

1. **Registration/login:**

* **Input:** Registration / Login Details
* **Procedural details:** When admin open this website it will check if user is already login or if not then will directed to this web application, here he has to enter his Email id and password which will be matched with stored Id and password. If he is a valid user he will be redirected to home page otherwise it will show Error Message.
* **Output:** Registration/Login successfully

1. **Shop Module**

* **Create shop:**
* **Input:** Create shop details
* **Procedural details:** This module allows shop owners to create their shop on Expertis and add services they provide
* **Output:** Shop created successfully
* **Appointment:**
* **Input:** Appointment details
* **Procedural details:** This module allows shop owner to book appointments with users
* **Output:** Appointment booked successfully

1. **User Module**

* **Profile:**
* **Input:** Profile details
* **Procedural details:** This module allows users to create and update their profile
* **Output:** Profile created successfully
* **Book appointment**:
* **Input**: Appointment details
* **Procedural details:** This module allows users to make an appointment
* **Output:** Appointment booked successfully
* **Review:**
* **Input:** Review details
* **Procedural details:** This module allows users to add reviews to a service or shop
* **Output:** Review added successfully

1. **Admin Module:**

* **Verification:**
* **Input:** Verification details
* **Procedural details:** This module allows the admin to verify registered shops
* **Output:** Verification successfully

**6. Program code listing**

1. **Database connection**

**7. User Interface**

**i. Login**

**8. Testing**

1. **Introduction**

Testing is the phase where errors from the earlier phase also must be tested. If the proce of examine something with intension of finding errors. Testing program consists of providing the program with the set of test inputs and observing if the program behaves has expected. If the program files then the condition under debugging which failure of occurs in noted from debugging and correction.

**ii. Test Reports**

1. **Unit Testing:**

Unittesting is the smallest unit of testing that focuses on verification. Using the concept of detailed design, the process of specification testing is carried out to uncover the errors within the boundary of the modules. All modules must be successful in the testing before the integration test begins.

1. **Integrate Testing :**

In this application developer test the programs as system. Software unit in a system are the modules and routines that are assembled and integrated to form a specific union.

1. **System Testing:**

Software testing is a critical element of software quality assurance and represent the ultimate review of specification, design and coding. The testing phase involves the testing of system using various test data; preparation of test data places a vital role in the system testing. After preparing the test data, the system is tested.

Those test data, errors were found and corrected by following testing steps and corrections are recorded for future reference. Thus a series testing is performed on the system before it is ready for implementation.

* **Conclusion**
* **Limitations**
* Requires internet connectivity to use the application
* Shop owner should be active in the application.

* **Scope for enhancement**
* Google map integration for calculating distance and direction
* Payment Gateway Integration using Razor Pay
* Quick chat option between the user shop
* **Abbreviations and Acronyms**
* SRS: Software Requirement Analysis and Specifications
* GUI: Graphical user interface
* RAM: Random Access Memory
* Bibliography/ References
* **Book and Author**:
* An integrated approach to software engineering Pankaj Jalote, 2nd  Narosa Publishing House.

* **Websites:**
* [WWW.google.com](http://WWW.google.com)
* [WWW.w3school.com](http://WWW.w3school.com)

## 

## 