PROJECT PROPOSAL DOCUMENT

PROJECT TITLE

DERMACARE CONNECT – Skincare E-Commerce & Doctor Consultation System

PROBLEM STATEMENT

Many users purchase skincare products without proper knowledge of their skin type, skin tone, or existing skin conditions, which often leads to ineffective results, skin irritation, or adverse reactions. Existing e-commerce platforms mainly focus on product sales and do not provide reliable professional guidance to help users choose suitable skincare products. As a result, users are forced to depend on trial-and-error methods or external consultations. There is a need for an integrated system that combines skincare product shopping with expert medical consultation, enabling users to receive professional advice and make informed skincare decisions in a secure and user-friendly online environment.

OBJECTIVES OF THE PROJECT

- To develop an e-commerce platform focused on selling skincare products such as moisturizers, sunscreens, and serums.
- To allow users to register, log in, and search for skincare products easily.
- To provide a consultation feature where users can seek professional advice from doctors regarding skin tone and skin conditions.
- To enable doctors to view user queries and provide appropriate skincare recommendations.
- To allow the admin to manage users, doctors, and skincare products efficiently.

BRIEF DESCRIPTION OF THE PROPOSED SOLUTION

Dermacare Connect is a web-based application that combines skincare product ecommerce with a doctor consultation system. Users can browse and search for various skincare products and purchase suitable items. Since many users are unaware of their skin type and skin conditions, the system provides a consultation module where users can send skin-related queries to doctors. Doctors review these queries and respond with professional advice or product suggestions. The admin manages all users, doctors, and product listings to ensure smooth system operation. This solution helps users make informed and safe skincare decisions through expert guidance.

TECHNOLOGY STACK

- Frontend: HTML, CSS, JavaScript, Bootstrap
- Backend: Python Django Framework
- Database: SQLite / MySQL
- Development Tools: VS Code, Git
- Operating System: Windows / Linux

EXPECTED OUTCOMES

- A fully functional skincare e-commerce website with doctor consultation support.
- Improved user awareness about skin type and suitable skincare products.
- Secure and organized management of users, doctors, and products by the admin.
- A scalable and modular Django-based application suitable for real-world use and academic evaluation.

SOFTWARE REQUIREMENT SPECIFICATION

FUNCTIONAL REQUIREMENTS

1. User Registration and Authentication

- The system shall allow new users to register by providing valid details.
- The system shall allow registered users to log in using a username and password.
- The system shall allow users to log out securely.
- The system shall validate user credentials to prevent unauthorized access.

2. User Profile Management

- The system shall allow users to view their profile information.
- The system shall assign roles to users (User or Doctor).
- The system shall store user and doctor details securely.

3. Product Management

- The system shall allow the admin to add new skincare products.
- The system shall allow the admin to update existing product details.
- The system shall allow the admin to delete outdated or unavailable products.
- The system shall display product details such as name, category, price, and description.
- The system shall allow users to browse all available skincare products.

4. Product Search and Filtering

- The system shall allow users to search skincare products by name.
- The system shall allow users to search products by category or skin concern.
- The system shall display relevant search results based on user input.

5. Consultation Management

- The system shall allow users to send skin-related consultation messages to doctors.
- The system shall allow users to select a doctor while sending consultation requests.
- The system shall store consultation messages securely in the database.
- The system shall allow doctors to view consultation requests sent by users.
- The system shall allow doctors to reply to user consultation messages.
- The system shall update the consultation status as pending or replied.
- The system shall allow users to view doctor replies.
- The system shall maintain consultation history for both users and doctors.

6. Doctor Management

- The system shall allow the admin to add and manage doctor accounts.
- The system shall allow doctors to log in securely.
- The system shall restrict doctors to access only consultation-related features.

7. Admin Management

- The system shall provide a secure admin login.
- The system shall allow the admin to manage users and doctors.
- The system shall allow the admin to monitor consultations.
- The system shall allow the admin to manage product categories.

8. Role-Based Access Control

- The system shall restrict functionalities based on user roles.
- The system shall allow only admins to manage products and users.
- The system shall allow only doctors to reply to consultations.
- The system shall allow only users to request consultations.

9. Data Storage and Security

- The system shall store all user, doctor, product, and consultation data securely.
- The system shall ensure data integrity and consistency.
- The system shall prevent unauthorized data access.

10.System Usability

- The system shall provide a user-friendly interface.
- The system shall display appropriate messages for successful and failed operations.
- The system shall ensure smooth navigation across all modules.

NON FUNCTIONAL REQUIREMENTS

1. Performance Requirements

- The system shall respond to user requests within an acceptable time limit.
- Product search results shall be displayed quickly without noticeable delay.

2. Security Requirements

- The system shall provide secure authentication for users, doctors, and admin.
- User passwords shall be stored securely.
- The system shall prevent unauthorized access to sensitive data.
- Role-based access control shall be enforced to restrict features based on user roles.

3. Reliability Requirements

- The system shall function consistently without unexpected failures.
- The system shall maintain data integrity at all times.

4. Availability Requirements

- The system shall be available for use at all times except during scheduled maintenance.
- Users shall be able to access the system using a standard web browser.

4. Scalability Requirements

- The system shall support an increase in the number of users and products without performance degradation.

6. Usability Requirements

- The system shall provide a simple and intuitive user interface.
- Users shall be able to navigate the system easily without prior technical knowledge.
- The system shall display clear messages for errors and successful operations.

7. Maintainability Requirements

- The system shall be modular and easy to maintain.
- Code shall be structured and documented for future enhancements.
- Updates and bug fixes shall be easy to implement.

8. Portability Requirements

- The system shall run on different operating systems such as Windows and Linux.
- The system shall be accessible through major web browsers.

9. Compatibility Requirements

- The system shall be compatible with modern web browsers.
- The system shall work correctly with Django-supported databases.

ASSUMPTIONS

- Users have basic knowledge of using web applications and internet browsers.
- Users and doctors have access to a stable internet connection.
- Doctors registered in the system are assumed to be certified and authorized by the admin.
- Users provide accurate information while registering and submitting consultation requests.
- The system is intended to provide general skincare guidance and not a replacement for physical medical diagnosis.
- Admin actively manages users, doctors, and product data to ensure system reliability.

CONSTRAINTS

- The system is developed using the Django framework, which defines the architectural structure of the application.
- The project is limited to skincare products and does not include other cosmetic or medical categories.
- The consultation feature is limited to message-based interaction and does not support real-time chat or video calls.
- The application scope is restricted to academic and demonstration purposes.

ER DIAGRAM

Entites and Attributes

1. User

Primary Key: User_ID

Attributes

- User_ID (PK)
- Name
- Email
- Password
- Phone
- Address

2. Doctor

Primary Key: Doctor_ID

Attributes

- Doctor_ID (PK)
- Name
- Email
- Password
- Specialization
- Experience

3. Admin

Primary Key: Admin_ID

Attributes

- Admin_ID (PK)
- Name
- Email
- Password

4. Product

Primary Key: Product_ID

Attributes

- Product_ID (PK)
- Product_Name
- Category
- Price
- Description
- Stock

5. Consultation

Primary Key: Consultation_ID

Attributes

- Consultation_ID (PK)
- Message
- Reply
- Status (Pending / Replied)
- Date
- User_ID (FK)
- Doctor_ID (FK)

RELATIONSHIP BETWEEN ENTITIES AND CARDINALITY

User - Consultation

- One user can send many consultations
- Each consultation belongs to one user

Cardinality:
1 ——— N

Doctor - Consultation

- One doctor can handle many consultations
- Each consultation is answered by one doctor

Cardinality:
1 ——— N

Admin — Product

- One admin can manage many products
- Each product is added/updated by one admin

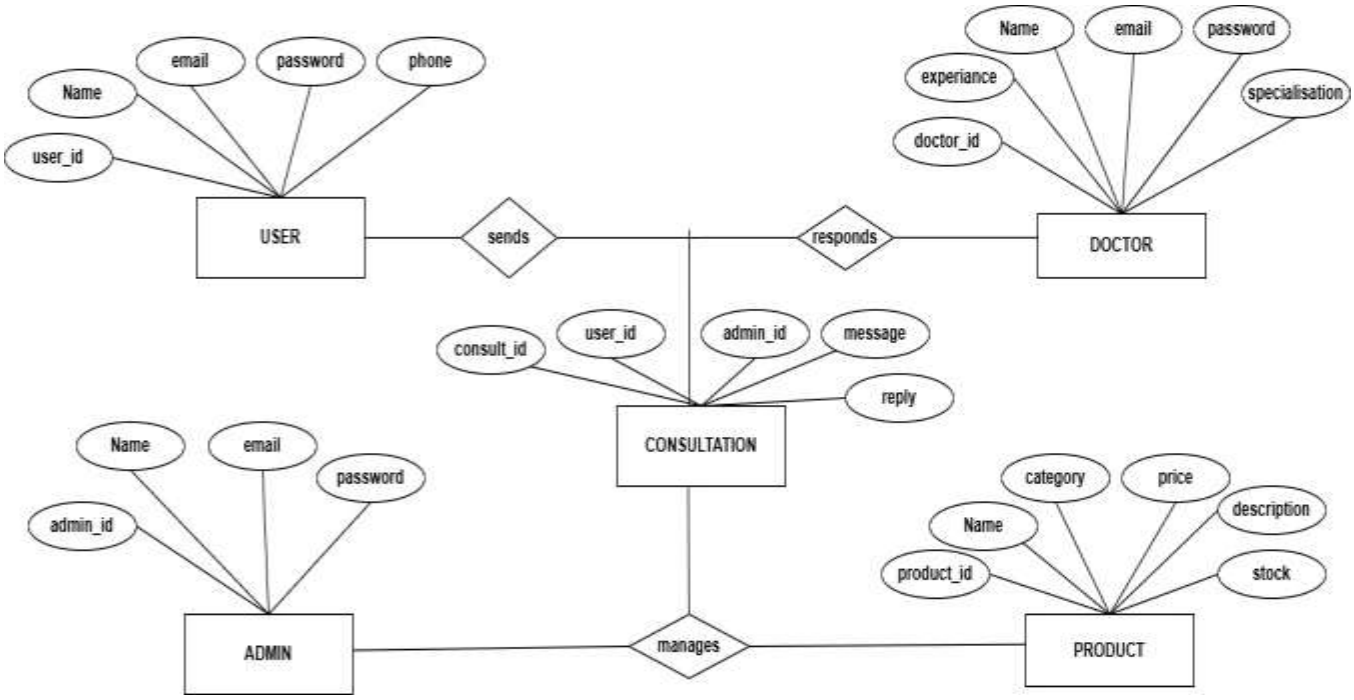
Cardinality:
1 ——— N

User -Product

- Users can view/search many products
- A product can be viewed by many users

Cardinality:
M ——— N

ER DIAGRAM


Name
email
password
phone
user_id
USER
sends
responds
Name
email
password
experiance
specialisation
doctor_id
DOCTOR
consult_id
user_id
admin_id
message
reply
CONSULTATION
Name
email
password
admin_id
ADMIN
manages
category
price
description
Name
product_id
stock
PRODUCT