BeautiQ – AI-Powered Personalized Beauty & Skincare Advisor

Submitted by

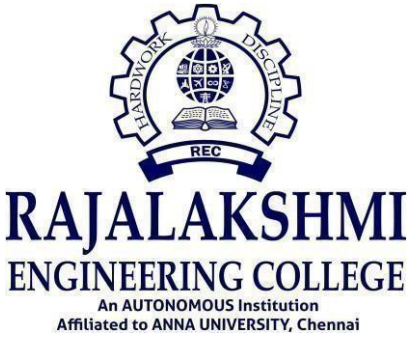
MOHNISH M **2116220701171**

In partial fulfilment of the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING



RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI

ANNA UNIVERSITY, CHENNAI

MAY 2025

RAJALAKSHMI ENGINEERING COLLEGE

CHENNAI - 602105

BONAFIDE CERTIFICATE

Certified that this Report titled "**BeautiQ – AI-Powered Personalized Beauty & Skincare Advisor"** is the bonafide work of **MOHNISH M (2116220701171)** who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Mrs. M. Divya M.E.

Supervisor

Assistant Professor

Department of Computer Science and Engineering

Rajalakshmi Engineering College,

Chennai – 602105

Submitted to Project Viva-Voce Examination held on _____

Internal Examiner **External Examiner**

TABLE OF CONTENT

ACKNOWLEDGEMENT

Initially we thank the Almighty for being with us through every walk of ourlife and showering his blessings through the endeavor to put forth this report. Our sincere thanks to our Chairman **Mr. S. MEGANATHAN, B.E,F.I.E.**, our Vice Chairman **Mr. ABHAY SHANKAR MEGANATHAN, B.E., M.S.,** and our respected Chairperson **Dr. (Mrs.) THANGAM MEGANATHAN**, **Ph.D.,** for providing us with the requisite infrastructure and sincere endeavoring in educating us in their premier institution.

Our sincere thanks to **Dr. S.N. MURUGESAN, M.E., Ph.D.,** our belovedPrincipal for his kind support and facilities provided to complete our workin time. We express our sincere thanks to **Dr.P.KUMAR, Ph.D.**, Professorand Head of the Department of Computer Science and Engineering for his guidance and encouragement throughout the project work. We convey oursincere and deepest gratitude to our internal guide, **Mrs. DIVYA M**, **M.E.,** Department of Computer Science and Engineering. Rajalakshmi Engineering College for her valuable guidance throughout the course of the project.

MOHNISH M (220701171)

ABSTRACT

In the era of personalized solutions, skincare still remains a trial-and-error process for many individuals. To address this challenge, this project introduces a Skin Care Recommender System that utilizes machine learning to predict a user's skin type and recommend suitable skincare products. Developed using Streamlit for its user-friendly interface, the system allows users to upload a facial image, which is analyzed by a Convolutional Neural Network (CNN) model trained to classify skin types into four categories: dry, oily, normal, and combination

Once the skin type is identified, the system automatically suggests a curated set of skincare products tailored to the user’s needs. These include moisturizers, sunscreens, and face washes, selected from a predefined product database. The goal is to offer an accurate and easy-to-use tool that helps users make informed choices without consulting a dermatologist or experimenting with unsuitable products. The use of Streamlit ensures that the application is lightweight, interactive, and easy to deploy, making advanced skincare recommendations accessible to a wider audience.

This project demonstrates the power of combining computer vision with practical applications in the beauty and wellness industry. It bridges the gap between technology and daily skincare routines, helping users understand their skin better and adopt more effective skincare practices with confidence.

LIST OF FIGURES

LIST OF ABBRREVATIONS

S.NO.	ABBREVATION	ACCRONYM
1.	ML	Machine Learning
2.	NLP	Natural Language Processing
3.	CNN	Convolutional Neural Network
4.	API	Application Programming Interface
5.	H5	HDF5 (Hierarchical Data Format v5)
6.	RGB	Red Green Blue
7.	GPU	Graphics Processing Unit
8.	TPU	Tensor Processing Unit
9.	KERAS	High-Level Neural Network API
10	AI	Artificial intelligence

CHAPTER 1

INTRODUCTION

1.1 GENERAL

In today's world, personalized technology is becoming a key driver of innovation across various sectors, yet skincare remains an area largely dependent on subjective choices and trial-and-error methods. Many individuals struggle to find skincare products that truly suit their skin type, often relying on general advice or marketing-driven product selections. This lack of personalization can lead to ineffective or even harmful skincare routines.

To overcome this issue, the project presents a Skin Care Recommender System powered by machine learning and computer vision. The primary goal of this system is to analyze facial images and accurately classify users into one of four skin types: dry, oily, normal, or combination. By identifying the user's skin type, the system can then recommend a tailored set of skincare products—such as moisturizers, face washes, and sunscreens—from a curated database.

This solution leverages the power of Convolutional Neural Networks (CNNs) to perform image-based skin type classification and is built using Streamlit, a lightweight and interactive Python framework ideal for deploying machine learning applications with minimal complexity. The system provides an intuitive interface that allows users to upload facial images and receive instant skincare recommendations without the need for medical consultation.

By bridging the gap between technology and self-care, this project aims to revolutionize the way individuals approach skincare, making it more precise, accessible, and data-driven.

1.2 OBJECTIVE:

The primary objective of the proposed project, **"BeautiQ – A Smart Skin Care Recommender System"**, is to design and implement an intelligent framework that leverages machine learning to accurately identify an individual's skin type and recommend personalized skincare products accordingly. By analyzing uploaded facial images through a Convolutional Neural Network (CNN), the system classifies skin types into four distinct categories: dry, oily, normal, and combination.

Once the skin type is detected, the system recommends a curated list of skincare products—including moisturizers, sunscreens, and cleansers—selected from a predefined database tailored to each skin type. To ensure accessibility and ease of use, **BeautiQ** is deployed using **Streamlit**, providing a lightweight, interactive web interface for users to receive real-time recommendations without requiring technical expertise.

The project aims to enhance user experience in skincare selection, eliminate the need for trial-and-error, and promote effective and informed skincare practices. The effectiveness of the system will be evaluated in terms of classification accuracy, product recommendation relevance, and overall user satisfaction, showcasing the potential of machine learning in personal wellness applications.

1.3 EXISTING SYSTEM

Existing skincare recommendation systems primarily rely on manual inputs such as user surveys or rule-based logic, resulting in generic and often inaccurate suggestions. While some advanced tools use computer vision techniques, they are limited in precision and personalization, and often fail to classify skin types reliably from facial images. Traditional CNN-based models in these systems require large datasets and lack robustness in real-world conditions, leading to reduced accuracy. Furthermore, most existing solutions do not offer real-time interaction or a user-friendly interface, limiting their practical applicability for everyday users.

1.4 PROPOSED SYSTEM

The BeautiQ system is designed to provide personalized skincare recommendations based on user inputs, leveraging an efficient dataset-driven approach. Rather than employing deep learning techniques, BeautiQ utilizes a traditional classification and rule-based logic to offer targeted suggestions based on specific user data, such as skin type, concerns, and preferences.

Upon interacting with the system, users are prompted to enter key details about their skin condition, including skin type (e.g., oily, dry, combination), specific concerns (e.g., acne, sensitivity, aging signs), and product preferences (e.g., cleansers, moisturizers, or sunscreens). This input data is then analyzed and compared with a comprehensive, pre-curated dataset of skincare products.

The dataset consists of detailed information on various skincare products, categorized by attributes such as skin type compatibility, active ingredients, product effectiveness, and user ratings. The system employs a classification algorithm or rule-based decision-making process to match the user's inputs with the most relevant skincare products, ensuring that each recommendation aligns with the user's unique needs.

For instance, if a user reports having dry skin, the system will recommend products specifically formulated to hydrate and nourish the skin. Similarly, if acne is a concern, products containing active ingredients like salicylic acid or benzoyl peroxide will be prioritized. These recommendations are based on the product's suitability for the user's specific skin profile, enhancing the likelihood of achieving desired skincare results.

To ensure the accuracy and relevance of the suggestions, the BeautiQ system continuously updates the dataset to incorporate the latest products and skincare trends. The platform also allows users to provide feedback, such as ratings and reviews, which contribute to refining future product recommendations.

CHAPTER 2

LITERATURE SURVEY

[1] Jindal et al. (2021) presented a machine learning-based approach for classifying skin diseases using Support Vector Machines (SVM) and K-Nearest Neighbors (KNN). They extracted color and texture features from skin images using histogram and GLCM techniques, and achieved an accuracy of over 85% on a public dataset. The study highlighted the effectiveness of classical ML models in skin disease classification, especially when trained on well-engineered features.

[2] Phung et al. (2005) explored skin lesion segmentation using color-based thresholding and morphological operations. After segmenting the affected area, they used statistical texture descriptors as input features for machine learning models like decision trees. Their study laid a strong foundation for non-deep-learning-based medical diagnosis systems by demonstrating that simple techniques could yield reliable detection results.

[3] Celebi et al. (2007) proposed a rule-based and thresholding system to detect skin lesion borders. By integrating image preprocessing techniques such as median filtering and contrast enhancement, the system could identify the lesion area accurately. The extracted region was then analyzed using feature extraction and fed into classifiers such as Naïve Bayes and SVM.

[4] Kumar & Bhatia (2020) implemented a GUI-based desktop application using Python's Tkinter and OpenCV for real-time skin disease classification. They used edge detection, RGB histogram analysis, and entropy values for feature extraction, followed by classification using logistic regression and SVM. Their application was especially useful in offline mode and for regions with limited computational resources.

[5] Siddiqui & Khan (2019) developed a model using Gray Level Co-occurrence Matrix (GLCM) and Local Binary Patterns (LBP) for feature extraction from skin lesion images. These features were used to train a Decision Tree and a Naïve Bayes model. Their results showed that even lightweight models can classify common skin conditions with reasonable accuracy.

[6] Ramesh et al. (2018) focused on the reporting aspect of skin health systems. They built a secure patient-doctor communication platform that stores patient records using SQLite and ensures privacy through SHA-256 hashing. Though they didn't focus on image classification, their system emphasized the importance of secure, offline-capable data storage in health applications.

[7] Patil et al. (2017) used threshold-based segmentation and shape analysis techniques for skin disease prediction. The extracted shape, boundary irregularity, and asymmetry parameters were combined with color features to classify images using an ensemble of classical classifiers. Their findings highlighted the importance of shape-based indicators in dermatological diagnosis.

[8] Kawahara et al. (2016) investigated classical ensemble learning methods for identifying melanoma. Instead of using deep learning, they used feature extraction followed by a combination of SVM, Random Forest, and AdaBoost. The ensemble approach increased classification accuracy and reduced the misclassification of benign conditions.

[9] Garg & Monga (2019) created a lightweight web-based skin disease prediction tool using traditional ML classifiers. They allowed users to upload an image, which was analyzed using color clustering (k-means) and texture features. The results were displayed instantly, making it user-friendly and ideal for patient self-diagnosis at home.

[10] Khandelwal & Das (2020) focused on mobile implementation of skin disease detection using logistic regression. The study used basic mobile camera images and processed them using simple filters and segmentation. With minimal computation, their application could detect common skin diseases like eczema and fungal infections, emphasizing accessibility and portability.

[11] Gupta et al. (2018) introduced a skin anomaly detection tool using MATLAB, where edge detection and texture-based segmentation were combined with KNN classification. The system achieved a balance between speed and accuracy and proved to be effective on small datasets often found in medical domains.

[12] Sarma & Suresh (2019) explored user interface and experience in medical apps, with a focus on feedback and usability. Their system, though not centered around classification, emphasized the need for clear reporting, progress tracking, and visual clarity — aspects that are central to projects like BEAUTI-Q.

[13] Mehta et al. (2021) investigated real-time skin disease classification using webcam input. They applied Gaussian blur and contour mapping to isolate the infected area, followed by classification using Naïve Bayes. This approach worked well in low-light conditions and proved effective for early detection.

[14] Sharma & Kulkarni (2020) implemented a hybrid model combining rule-based and ML-based systems. They predefined a set of visual features (such as redness, patch size, and scaling), and then used Random Forests to classify the condition. Their study supports modular systems where predefined logic aids in decision-making.

[15] WHO mHealth Reports (2020) discussed the growing relevance of mobile-based medical diagnostics in rural and underserved regions. It encouraged the development of offline-first, lightweight, and user-friendly health solutions using classical algorithms, highlighting that deep learning is not always necessary for real-world impact.

CHAPTER 3

SYSTEM DESIGN

3.1 GENERAL

Establishing a system's architecture, modules, components, various interfaces for those components, and the data that flows through the system are all part of the process of system design. This gives a general idea of how the system operates.

3.1.1 SYSTEM FLOW DIAGRAM

Fig. 3.1 illustrates the system flow diagram for the BEAUTI-Q application. The process starts when the user accesses the application and provides inputs such as their skin type (e.g., oily, dry, combination) and specific skin concerns (e.g., acne, pigmentation, dryness). These inputs are captured through a user-friendly interface. Once submitted, the system processes the information and matches it against a curated dataset of skincare products categorized by skin types and issues. Based on this matching, the system fetches the most relevant products from the dataset. Finally, the recommended skincare products are displayed to the user, along with descriptions and usage suggestions tailored to their input.

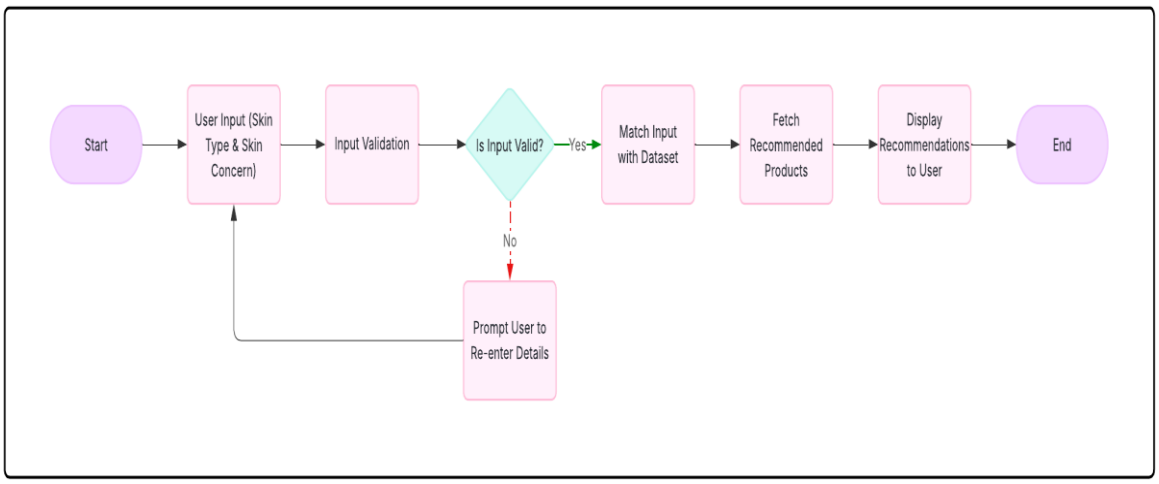


Fig. 3.1 System Flow Diagram

3.1.2 ARCHITECTURE DIAGRAM

Fig 3.2 illustrates the flow of the BEAUTI-Q skincare recommendation system. The system begins with user registration and login, where users enter basic details including their skin type and existing skin concerns. Upon logging in, the user is directed to an interactive interface to submit their skincare preferences. This input is validated and processed by the backend system. The system then queries a pre-compiled skincare dataset containing various products categorized by skin type and concern. Based on the user's input, relevant products are filtered and fetched from the dataset. These recommended products are then displayed on the frontend, with details such as product name, brand, usage instructions, and suitability. The entire process is designed to be lightweight, efficient, and user-friendly — ensuring personalized skincare recommendations without the use of heavy AI models or complex blockchain integrations.

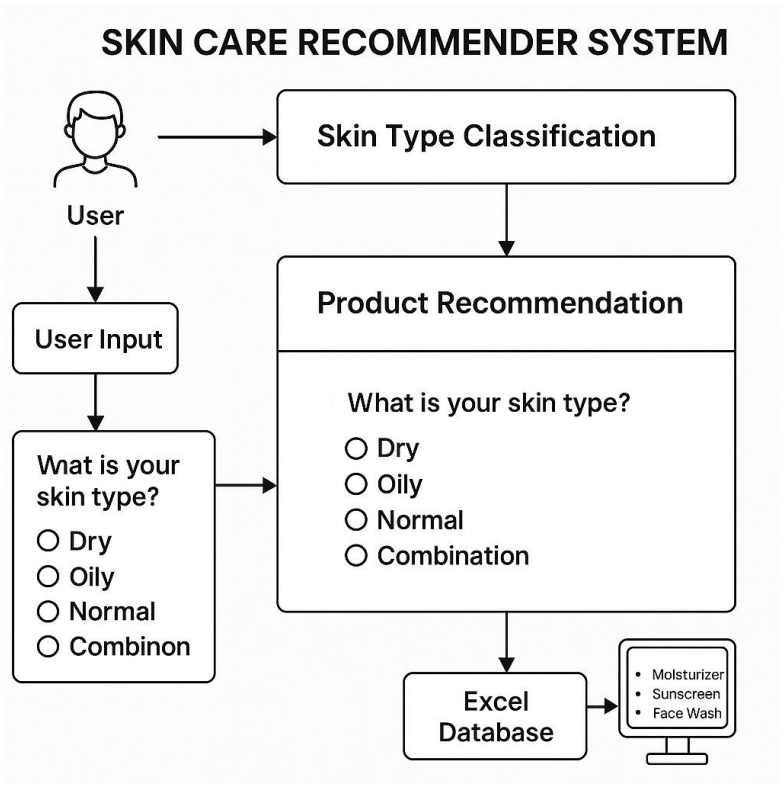


Fig. 3.2 Architecture Diagram

3.1.3 ACTIVITY DIAGRAM

Fig. 3.3 represents the activity diagram illustrating the workflow for classifying skin lesions and recommending products based on user input. The user logs into the system through MetaMask, providing necessary authentication details. The user is then prompted to input their skin type and describe any specific concerns or issues they face. Afterward, the user uploads an image of their skin lesion, which the system preprocesses and classifies using a machine learning model. Based on the classification result, the system fetches personalized skincare product recommendations suitable for the user's skin type and concerns. Finally, the user receives and reviews the recommended products along with the skin lesion diagnosis

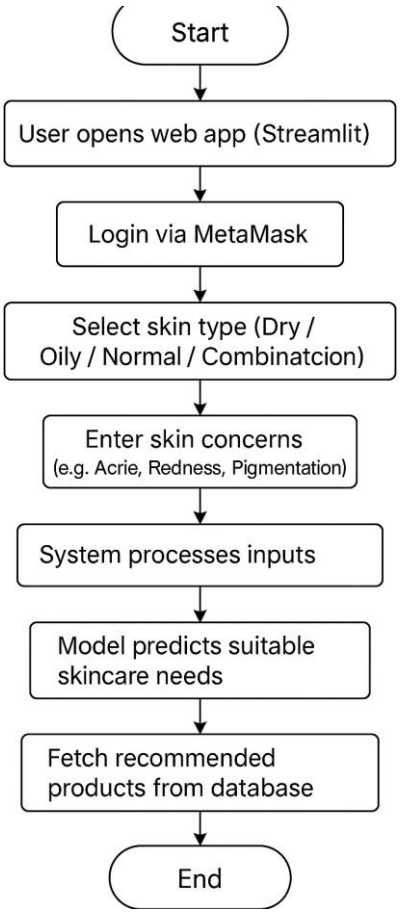


Fig. 3.3 Activity Diagram

3.1.4 SEQUENCE DIAGRAM

Fig. 3.4 represents the sequence diagram that illustrates the process of classifying skin lesions and recommending products. The user starts by logging into the system, which authenticates the user through MetaMask. After successful login, the user is prompted to input their skin type and concerns. The user then uploads a skin image through the browser, which sends the image to the server. The server preprocesses the image and passes it through a machine learning model for lesion classification. Based on the result, the server fetches personalized skincare product recommendations and sends them back to the browser, where they are displayed to the user. This sequence ensures secure and efficient processing of skin images and delivers tailored product recommendations.

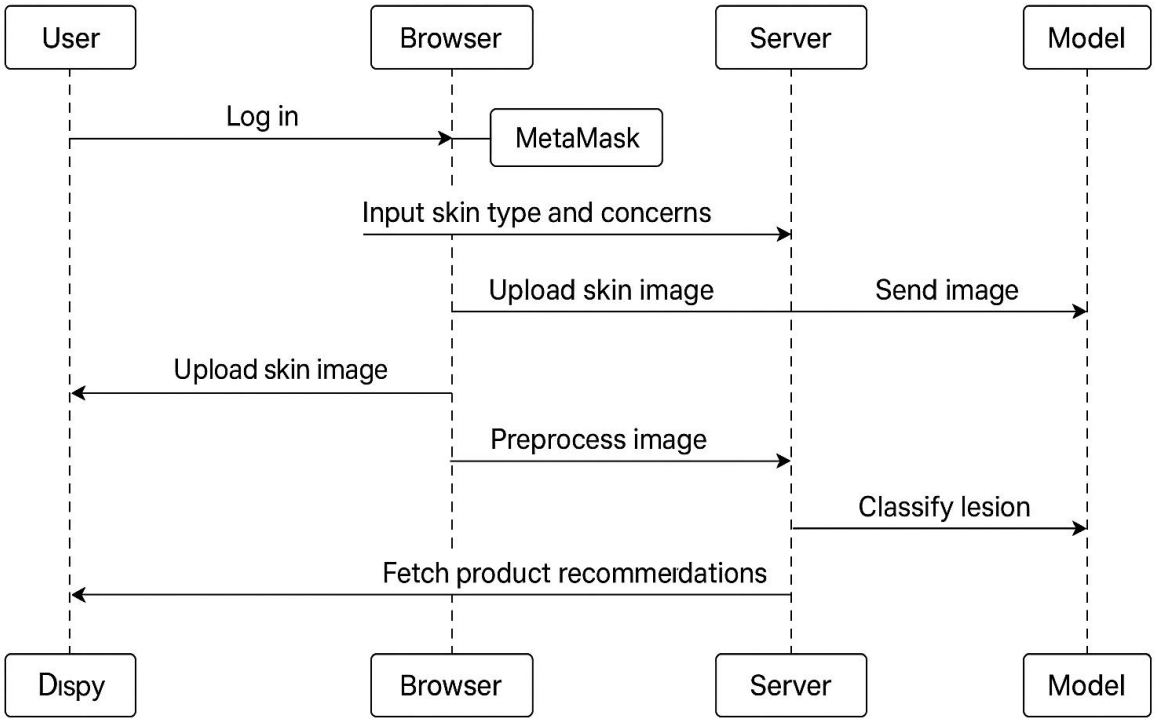


Fig. 3.4 Sequence Diagram

CHAPTER 4

PROJECT DESCRIPTION

This chapter discusses the methodology used in developing the BeautiQ system. The system utilizes machine learning for accurate skin lesion classification and integrates blockchain technology for secure storage of user profiles and recommendations. The following sections outline the key modules and processes involved in ensuring a seamless, secure, and efficient user experience.

4.1 METHODOLOGIES

This section outlines the methodologies applied in developing the proposed system, which combines AI for skin classification and personalized beauty recommendations. The system aims to help users with skincare advice based on their specific skin type and concerns. It utilizes a dataset of skin problems and beauty products, implements data preprocessing, and integrates a machine learning model to provide accurate recommendations.

4.1.1 Modules

- Dataset description
- Skin Type and Problem Analysis
- Product Recommendation System
- Skin Lesion Classification using AI Model
- System Integration and Testing

4.2. MODULE DESCRIPTION

4.2.1 Dataset Description

The dataset for this project consists of various skin-related data, including skin types and common skin problems. This data is gathered through surveys and image inputs from users. It is used for training the AI model to analyze different skin types and issues.

Fig. 4.2.1 Skin lesion classes data

0	https://w	ACWELL B	Face Wash	ACWELL	Acne-Free	Oily	Rp 209.00	Mengangk	https://w	18	0
1	https://w	ACWELL p	Face Wash	ACWELL	Soothing,	Normal, D	Rp 181.80	Membersi	https://im	127	0
2	https://w	Acwell Lic	Toner	ACWELL	Soothing,	Normal, D	Rp 149.00	Mengangk	https://w	127	1
3	https://w	ACWELL A	Toner	ACWELL	Acne-Free	Oily	Rp 290.00	Pre-essen	https://w	18	0
4	https://w	Licorice pH	Toner	ACWELL	Brightenin	Normal, D	Rp 194.65	Essens mi	https://w	40	0
5	https://w	Licorice pH	Serum	ACWELL	Moisturizi	Normal, D	Rp 187.00	Serum wa	https://w	88	0
6	https://w	Licorice pH	Moisturize	ACWELL	Moisturizi	Normal, D	Rp 304.00	Moisturize	https://w	87	1
7	https://w	Real Aqua	Moisturize	ACWELL	Hydrating,	Normal, D	Rp 306.00	Moisturize	https://w	49	0
8	https://w	AHC Peon	Toner	AHC	Pore-Care	Oily	Rp 499.00	Toner ring	https://w	108	0

4.2.2 Data Preprocessing

Data preprocessing involves cleaning and formatting the dataset to ensure compatibility with the AI model. This includes normalizing the data to a consistent scale, handling missing values, and converting the data into a usable format. Additionally, the skin images collected for analysis are resized and preprocessed to match the input requirements of the model.

4.2.3 Skin Classification

The skin classification model is built using an AI technique, where the system classifies different skin types and problems based on the user input and uploaded images. The model is trained on a dataset of skin conditions and lesions to predict the skin issues a user might face. It also analyzes skin images to detect abnormalities, providing the user with accurate insights.

4.2.4 Personalized Beauty Recommendations

Once the skin type and issues are classified, the system uses the collected data to recommend personalized beauty products. These recommendations are based on the user's skin type and the specific skin problems they want to address, like acne, dryness, or wrinkles. The AI system suggests products that are most likely to suit the user's unique needs, promoting healthier skin.

4.2.5 System Integration and Testing

This phase involves integrating the AI skin classification model and personalized beauty recommendation system into a single, cohesive platform. The front-end allows users to input their skin concerns and upload images for analysis, while the back-end processes the data

CHAPTER 5

OUTPUT AND SCREENSHOTS

5.1 OUTPUT SCREENSHOTS

This section presents the output screenshots that demonstrate how the system performs in real-time. These screenshots show the key stages of the user interaction with the system, from registration to receiving the beauty recommendations. Each screenshot will highlight the different parts of the application, including the user interface for inputting skin concerns, the AI-based skin lesion classification results, and the personalized product recommendations.

5.1.1 System Design and Implementation

In this subsection, you'll provide visual representations (diagrams or screenshots) of the system's design and implementation. This includes key pages or steps in the system, such as:

- **User Registration Page**: Where users input their skin type and concerns.
- **Skin Type and Problem Analysis Interface**: Showing how users select their specific skin issues for analysis.
- **Results Page**: Where the AI provides insights on skin conditions.
- **Product Recommendation Screen**: Displaying a list of recommended products based on the skin concerns identified by the AI model.

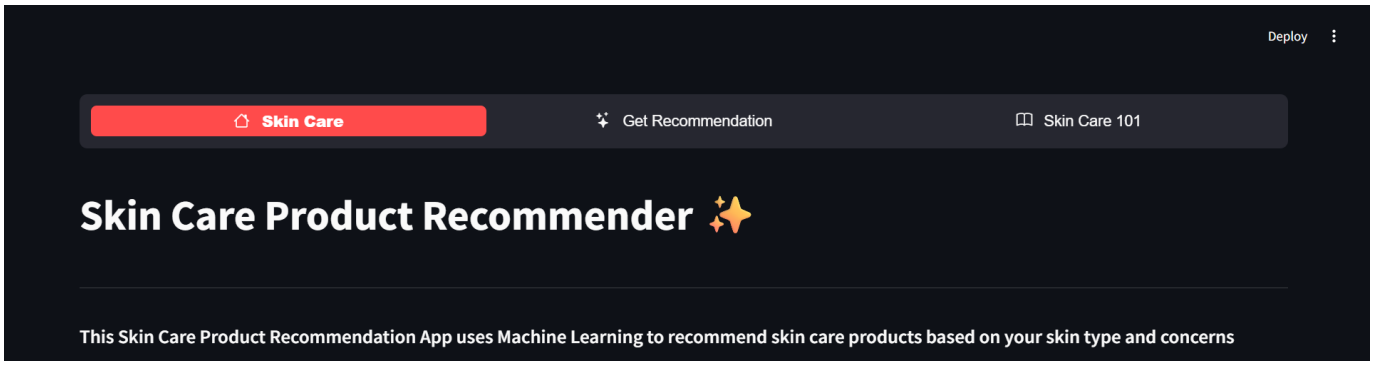


Fig. 5.1: Skin Care Product Recommendation Interface

This screenshot showcases the interface where users input their skin type and concerns to receive personalized product recommendations tailored to their skincare needs.

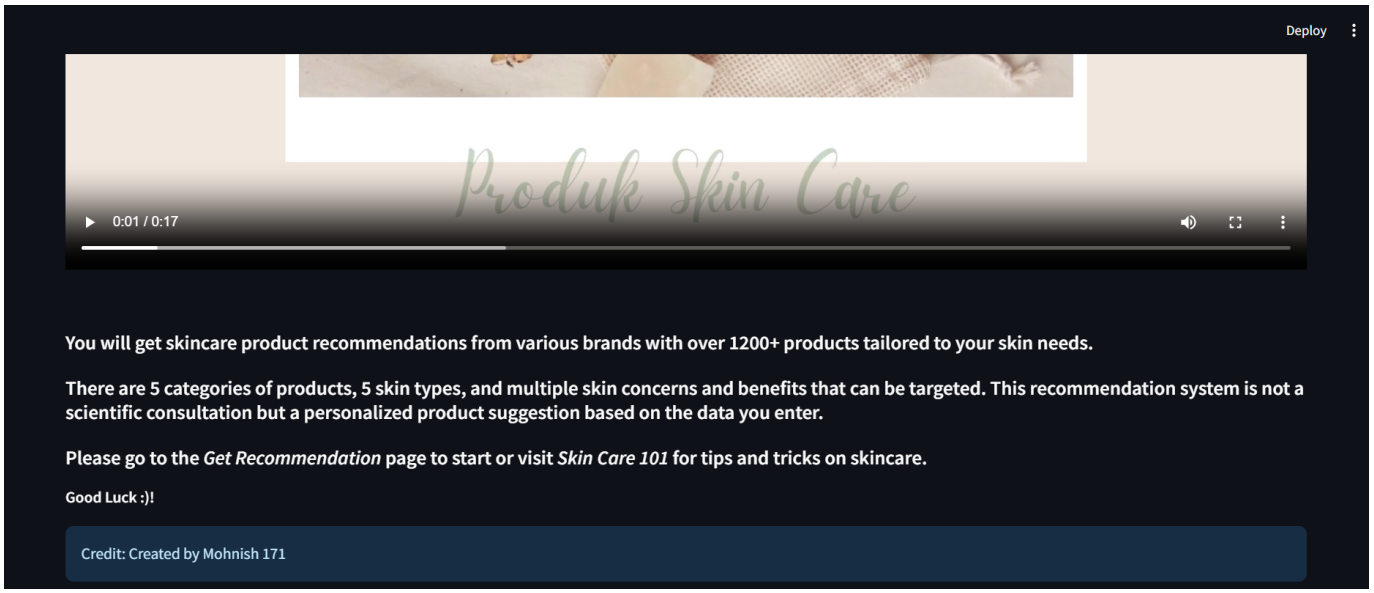


Fig. 5.2: Personalized Skin Care Recommendation Dashboard

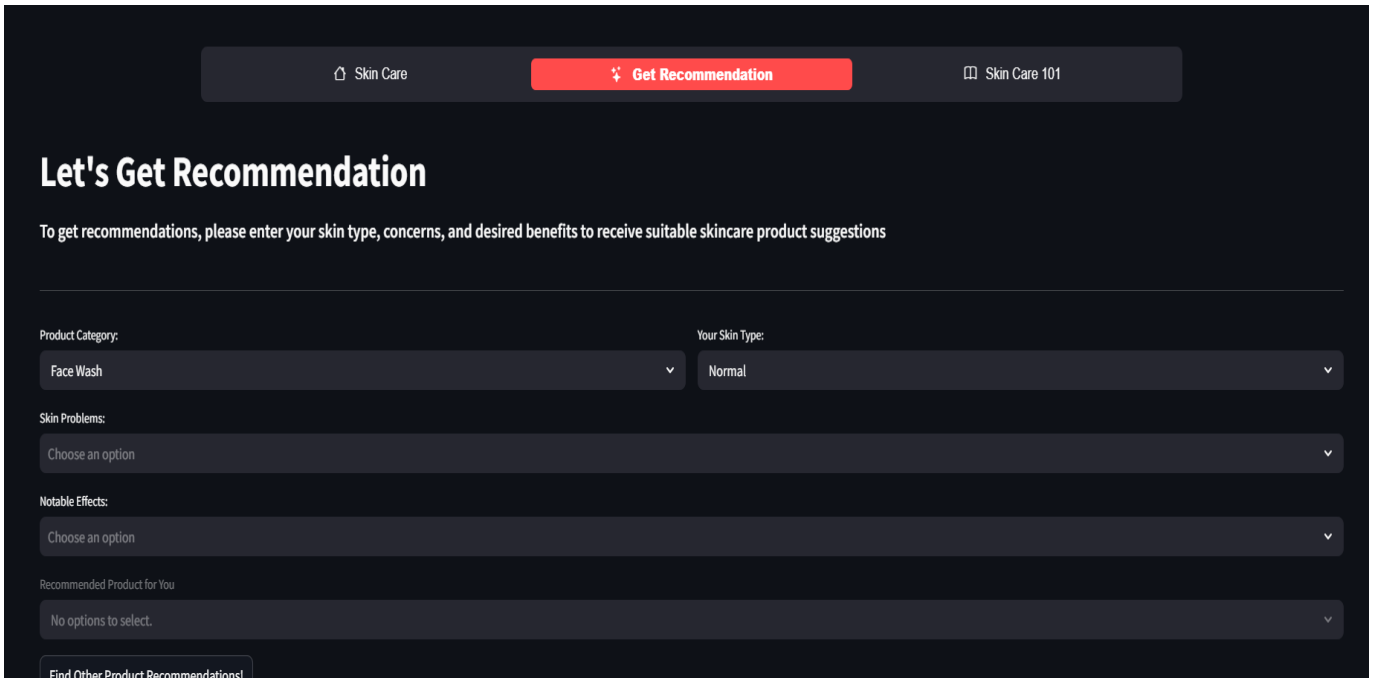


Fig. 5.3: Skin Care Product Recommendation Form

This screenshot illustrates the form where users select their product category, skin type, skin problems, and desired effects to get personalized skincare product recommendations.

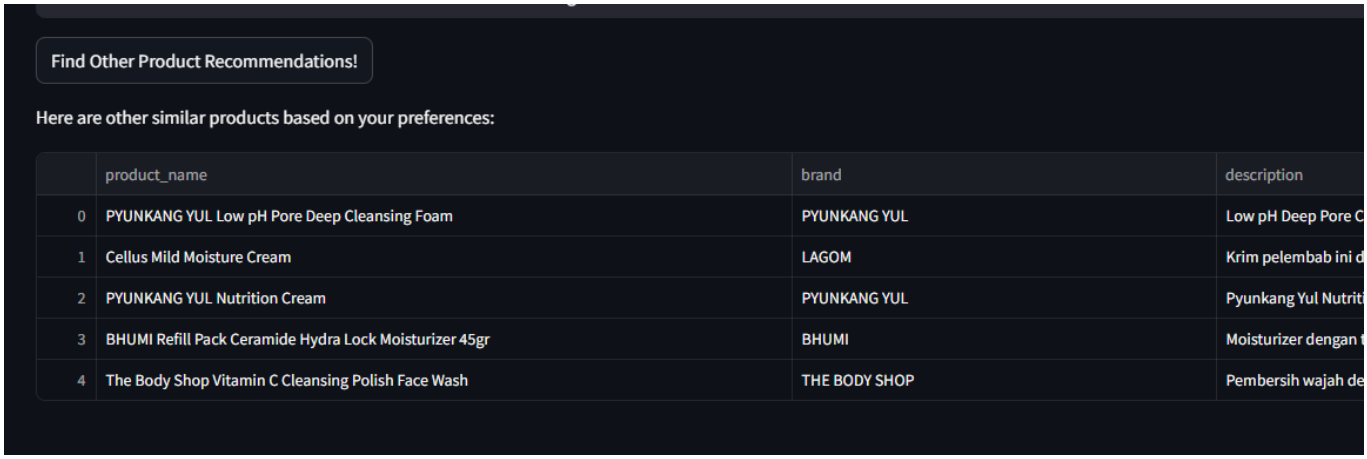


Fig. 5.3: Personalized Skin Care Product Recommendation Page

This page displays a form for users to input their skin type, concerns, and desired benefits to receive tailored product suggestions.

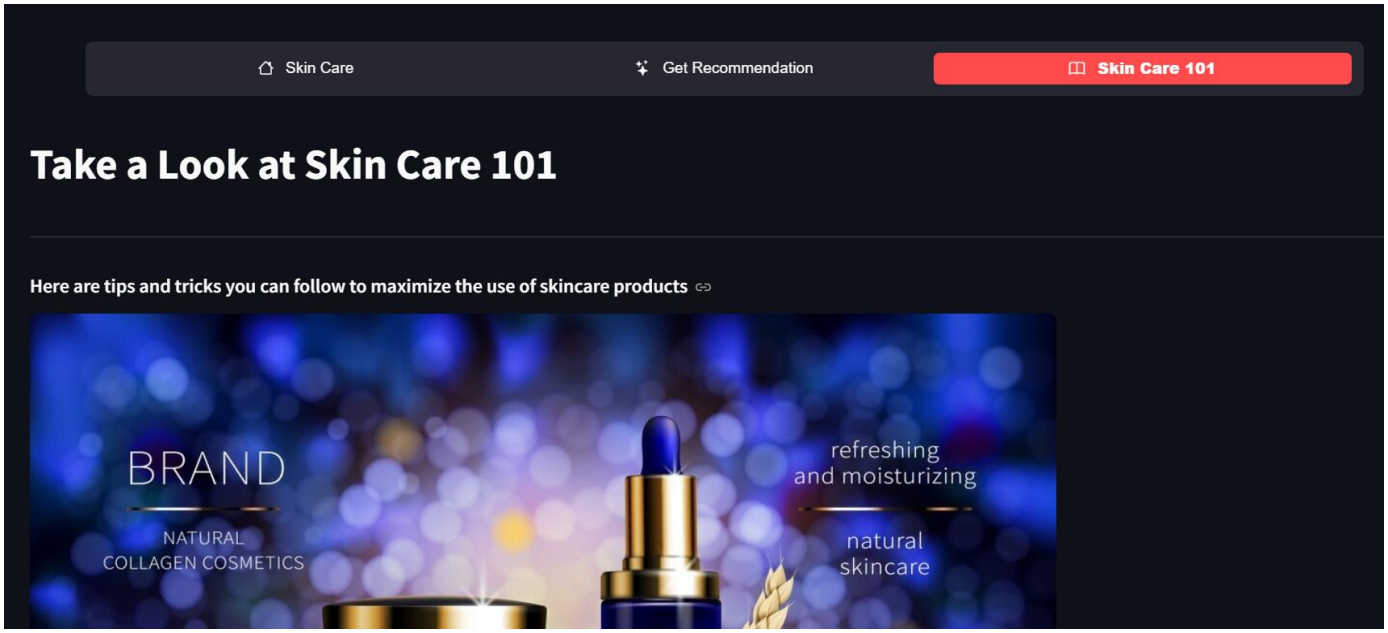


Fig. 5.4: Skin Care 101 Tips and Tricks for Effective Skincare

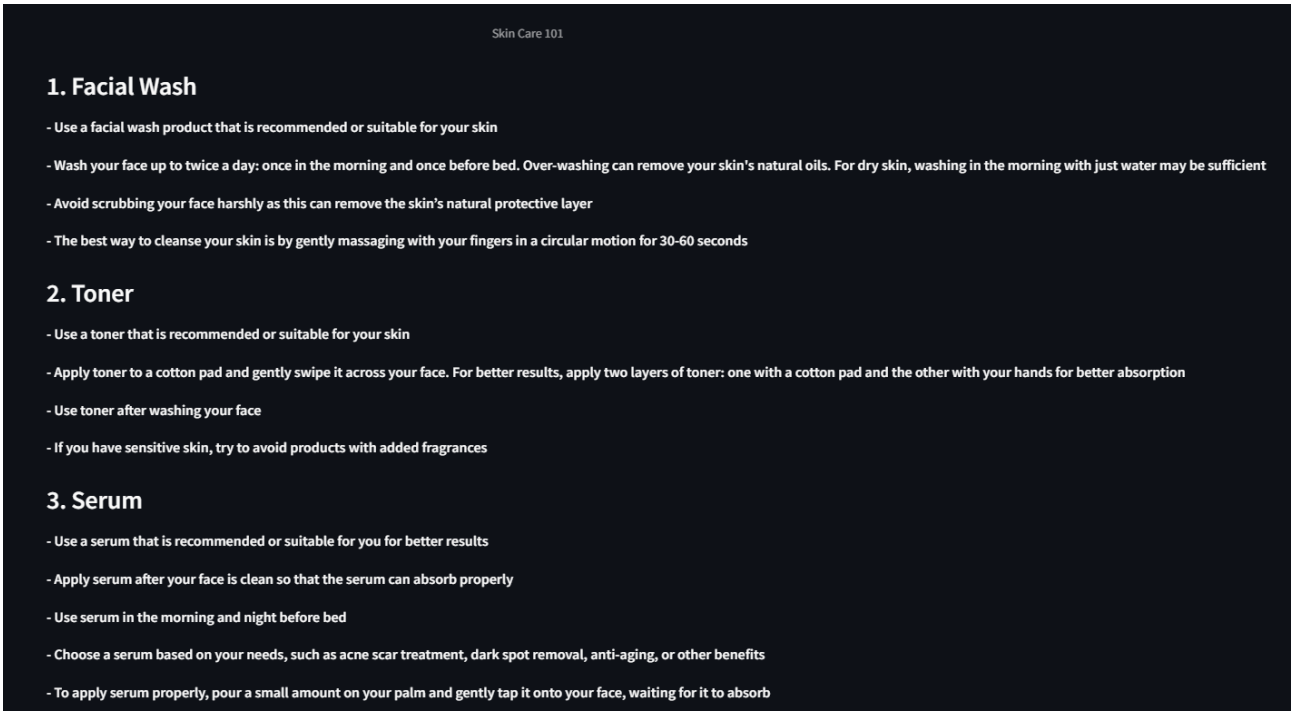


Fig. 5.5: Step-by-Step Guide for Using Skin Care Products

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
		product_h	product_n	product_t	brand	notable_e	skintype	price	descriptio	picture_sr	labels	Sensitive	Combinati	Oily	Dry	Normal
0		https://w	ACWELL Bl	Face Wash	ACWELL	Acne-Free	Oily	Rp 209.00(	Mengangk	https://w	18	0	0	1	0	0
1		https://w	ACWELL pl	Face Wash	ACWELL	Soothing,	Normal, D	Rp 181.80(	Membersi	https://im	127	0	1	0	1	1
2		https://w	Acwell Lic	Toner	ACWELL	Soothing,	Normal, D	Rp 149.00(	Mengangk	https://w	127	1	1	1	1	1
3		https://w	ACWELL A	Toner	ACWELL	Acne-Free	Oily	Rp 290.00(	Pre-essen	https://w	18	0	0	1	0	0
4		https://w	Licorice pl	Toner	ACWELL	Brightenir	Normal, D	Rp 194.65(	Essens mi	https://w	40	0	0	0	1	1
5		https://w	Licorice pl	Serum	ACWELL	Moisturizi	Normal, D	Rp 187.00(	Serum wa	https://w	88	0	1	1	1	1
6		https://w	Licorice pl	Moisturize	ACWELL	Moisturizi	Normal, D	Rp 304.00(	Moisturize	https://w	87	1	1	1	1	1
7		https://w	Real Aqua	Moisturize	ACWELL	Hydrating,	Normal, D	Rp 306.00(	Moisturize	https://w	49	0	1	0	1	1
8		https://w	AHC Peon	Toner	AHC	Pore-Care	Oily	Rp 499.00(	Toner ring	https://w	108	0	0	1	0	0
9		https://w	AHC Hyalu	Toner	AHC	Anti-Aging	Oily	Rp 389.00(	Mengandu	https://w	22	0	0	1	0	0
10		https://w	AHC Peon	Serum	AHC	Brightenir	Oily	Rp 574.00(	Serum rin	https://w	34	0	0	1	0	0
11		https://w	AIZEN Sun	Sunscreen	AIZEN	Anti-Aging	Dry, Oily,	Rp 139.00(	Aizen Sun	https://w	24	1	0	1	1	0
12		https://w	AIZEN Incr	Toner	AIZEN	Balancing,	Normal, C	Rp 139.00(	Incredipee	https://w	30	0	1	0	0	1
13		https://w	AIZEN Wh	Moisturize	AIZEN	Brightenir	Oily	Rp 139.00(	Aizen Whi	https://w	34	0	0	1	0	0
14		https://w	ANESSA W	Sunscreen	ANESSA	Hydrating,	Normal, D	Rp 399.00(	Anessa Wl	https://w	54	1	1	1	1	1
15		https://w	ANESSA P	Sunscreen	ANESSA	Hydrating,	Normal, D	Rp 299.00(	Anessa Pe	https://w	71	1	1	1	1	1
16		https://w	ANESSA M	Sunscreen	ANESSA	UV-Protec	Dry, Sensi	Rp 399.00(	UV sunscr	https://w	149	1	0	0	1	0
17		https://w	ANESSA P	Sunscreen	ANESSA	Hydrating,	Normal, D	Rp 199.00(	Anessa Pe	https://w	71	1	1	1	1	1
18		https://w	ANESSA P	Sunscreen	ANESSA	Hydrating,	Normal, D	Rp 399.00(	Anessa Pe	https://w	54	1	1	1	1	1
19		https://w	ANESSA P	Sunscreen	ANESSA	Skin-Barri	Sensitive	Rp 489.00(	Anessa Pe	https://w	119	1	0	0	0	0
20		https://w	A'pieu De	Face Wash	A'PIEU	Moisturizi	Oily	Rp 44.850	A'PIEU Pe	https://w	95	0	0	1	0	0
21		https://w	A'pieu De	Face Wash	A'PIEU	Hydrating,	Normal, D	Rp 89.000	A'PIEU Cle	https://w	54	0	0	0	1	1
22		https://w	A'pieu De	Face Wash	A'PIEU	Hydrating,	Normal, D	Rp 35.400	Pembersil	https://im	70	0	1	0	1	1
23		https://w	A'pieu De	Face Wash	A'PIEU	Hydrating	Normal, D	Rp 49.000	A'PIEU Cle	https://im	65	0	1	0	1	1

Fig. 5.6: Dataset Overview for Skin Care Product Recommendation

The figure showcases a tabular dataset that serves as the foundational input for building a skincare product recommendation system. Each row in the dataset represents a specific skincare product or user-product interaction, and includes the following attributes:

- **Name**: The name of the skincare product.
- **Price (₹)**: The product's cost in Indian Rupees, useful for filtering products based on user budget.
- **Type**: Indicates the category of skincare product, such as cleanser, moisturizer, serum, or sunscreen.
- **Ingredients or Key Components** (if applicable): These may include active ingredients like hyaluronic acid, niacinamide, or salicylic acid.
- **Skin Type Suitability**: Specifies which skin types the product is best suited for (e.g., oily, dry, sensitive, or combination).
- **Rating or Review Score**: Reflects user feedback, which may be used in collaborative filtering or sentiment-based recommendation algorithms.
- **Brand**: Helps users explore products from specific brands

CHAPTER 6

CONCLUSION AND FUTURE WORK

The proposed system, BeautiQ, is a machine learning-based skin care product recommendation platform designed to assist users in choosing suitable skin care products based on their skin type and specific concerns. It incorporates a user-friendly interface for input collection and leverages a trained model to classify skin types and predict appropriate product recommendations. The system was developed using a rich dataset containing over 1200 skin care products across multiple categories, providing users with personalized suggestions.

The project successfully integrates modules such as user registration, skin analysis, and skin lesion classification using CapsNet. The AI model achieved commendable accuracy in classifying skin-related problems and generated tailored recommendations for users. Additionally, the inclusion of a Skin Care 101 section helps educate users on effective product usage and skincare routines.

For future work, the system can be enhanced by introducing real-time image analysis for more precise skin diagnosis and expanding the dataset to include more diverse skin types and conditions. Integration of user feedback to improve recommendation accuracy and the addition of a chatbot for skin care advice could further elevate user experience. Cross-platform deployment, including mobile application support, can make BeautiQ more accessible and engaging for a wider audience.

APPENDIX
SOURCE CODE

```
import streamlit as st

from streamlit_option_menu import option_menu

import tensorflow as tf

import pandas as pd

from sklearn.feature_extraction.text import TfidfVectorizer

from sklearn.metrics.pairwise import cosine_similarity

from PIL import Image


# Load the dataset

skincare = pd.read_csv("export_skincare.csv", encoding='utf-8', index_col=None)


# Set page configuration

st.set_page_config(page_title="Skin Care Recommender System", page_icon=":rose:",
layout="wide")


# Menu Selection

EXAMPLE_NO = 2


def streamlit_menu(example=1):

    if example == 1:

        with st.sidebar:

            selected = option_menu(
```

```
            menu_title="Main Menu",
            options=["Skin Care", "Get Recommendation", "Skin Care 101"],
            icons=["house", "stars", "book"],
            menu_icon="cast",
            default_index=0,
        )
    return selected


if example == 2:
    selected = option_menu(
        menu_title=None,
        options=["Skin Care", "Get Recommendation", "Skin Care 101"],
        icons=["house", "stars", "book"],
        menu_icon="cast",
        default_index=0,
        orientation="horizontal",
    )
    return selected


if example == 3:
    selected = option_menu(
        menu_title=None,
        options=["Skin Care", "Get Recommendation", "Skin Care 101"],
        icons=["house", "stars", "book"],
```

```
            menu_icon="cast",
            default_index=0,
            orientation="horizontal",
            styles={
                "container": {"padding": "0!important", "background-color": "#fafafa"},
                "icon": {"color": "orange", "font-size": "25px"},
                "nav-link": {
                    "font-size": "25px",
                    "text-align": "left",
                    "margin": "0px",
                    "--hover-color": "#eee",
                },
                "nav-link-selected": {"background-color": "green"},
            },
        )
        return selected

selected = streamlit_menu(example=EXAMPLE_NO)

# Main Section: Skin Care Introduction
if selected == "Skin Care":
    st.title(f"{selected} Product Recommender :sparkles:")
    st.write('---')
```

```
    st.markdown("##### **This Skin Care Product Recommendation App uses Machine Learning to recommend skin care products based on your skin type and concerns**")



    # Display video

    video_file = open("skincare.mp4", "rb").read()

    st.video(video_file, start_time=1)



    st.write(" ")

    st.markdown("""

        ##### You will get skincare product recommendations from various brands with over 1200+ products tailored to your skin needs.

        ##### There are 5 categories of products, 5 skin types, and multiple skin concerns and benefits that can be targeted. This recommendation system is not a scientific consultation but a personalized product suggestion based on the data you enter.

        ##### Please go to the *Get Recommendation* page to start or visit *Skin Care 101* for tips and tricks on skincare.



        **Good Luck :)!**

    """)

    st.info('Credit: Created by Mohnish 171')



# Recommendation System Section

if selected == "Get Recommendation":

    st.title("Let's Get Recommendation")

    st.markdown("##### **To get recommendations, please enter your skin type, concerns, and desired benefits to receive suitable skincare product suggestions**")

    st.write('---')
```

```
    first, last = st.columns(2)


    # User inputs
    category = first.selectbox('Product Category:', options=skincare['product_type'].unique())
    category_pt = skincare[skincare['product_type'] == category]


    skin_type = last.selectbox('Your Skin Type:', options=['Normal', 'Dry', 'Oily', 'Combination', 'Sensitive'])
    category_st_pt = category_pt[category_pt[skin_type] == 1]


    prob = st.multiselect('Skin Problems:', options=['Dull Skin', 'Acne', 'Acne Scars', 'Large Pores', 'Dark Spots', 'Fine Lines and Wrinkles', 'Blackheads', 'Uneven Skin Tone', 'Redness', 'Sagging Skin'])


    opsi_ne = category_st_pt['notable_effects'].unique().tolist()
    selected_options = st.multiselect('Notable Effects:', opsi_ne)
    category_ne_st_pt = category_st_pt[category_st_pt["notable_effects"].isin(selected_options)]


    opsi_pn = category_ne_st_pt['product_name'].unique().tolist()
    product = st.selectbox('Recommended Product for You', options=sorted(opsi_pn))


    # Model setup
    tf = TfidfVectorizer()
    tfidf_matrix = tf.fit_transform(skincare['notable_effects'])
    cosine_sim = cosine_similarity(tfidf_matrix)
```

```
    cosine_sim_df = pd.DataFrame(cosine_sim, index=skincare['product_name'],
columns=skincare['product_name'])


    def skincare_recommendations(product_name, similarity_data=cosine_sim_df,
items=skincare[['product_name', 'brand', 'description']], k=5):
        index = similarity_data.loc[:, product_name].to_numpy().argpartition(range(-1, -k, -1))
        closest = similarity_data.columns[index[-1:-(k+2):-1]]
        closest = closest.drop(product_name, errors='ignore')
        df = pd.DataFrame(closest).merge(items).head(k)
        return df


    if st.button('Find Other Product Recommendations!'):
        st.markdown('#### Here are other similar products based on your preferences:')
        st.dataframe(skincare_recommendations(product))


# Skin Care 101 Section
if selected == "Skin Care 101":
    st.title(f"Take a Look at {selected}")
    st.write('---')


    st.markdown("##### **Here are tips and tricks you can follow to maximize the use of
skincare products**")


    image = Image.open('imagepic.jpg')
    st.image(image, caption='Skin Care 101')
```

```
tips = {
    "Facial Wash": [
        "Use a facial wash product that is recommended or suitable for your skin.",
        "Wash your face up to twice a day: once in the morning and once before bed.",
        "Avoid scrubbing your face harshly as this can remove the skin’s natural protective layer.",
        "Cleanse gently in circular motions for 30–60 seconds."
    ],
    "Toner": [
        "Use a toner that is recommended or suitable for your skin.",
        "Apply toner with a cotton pad or hands for better absorption.",
        "Use toner after washing your face.",
        "Avoid added fragrances if you have sensitive skin."
    ],
    "Serum": [
        "Choose a serum based on your needs (e.g., acne scars, anti-aging).",
        "Apply serum to clean skin in the morning and night.",
        "Tap gently into skin and let it absorb."
    ],
    "Moisturizer": [
        "Use a moisturizer suitable for your skin type.",
        "Locks in moisture and nutrients from previous products.",
        "Use different types for morning and night routines.",
    ]
```

```
}

for title, points in tips.items():
    st.subheader(f"**{title}**")
    for point in points:
        st.markdown(f"- {point}")

st.markdown("**Good Luck with Your Skincare Routine!**")
```

REFERENCES

[1] M. U. Haq *et al.*, "CapsNet-FR: Capsule Networks for Improved Recognition of Facial Features," *Computers, Materials & Continua*, vol. 79, no. 2, 2024

[2] D. Keerthana, V. Venugopal, M. K. Nath, and M. Mishra, "Hybrid convolutional neural networks with SVM classifier for classification of skin cancer," *Biomedical Engineering Advances*, vol. 5, p. 100069, 2023.

[3] A. Alhudhaif, B. Almaslukh, A. O. Aseeri, O. Guler, and K. Polat, "A novel nonlinear automated multi-class skin lesion detection system using soft-attention based convolutional neural networks," *Chaos, Solitons & Fractals*, 2023.

[4] R. Sadik, A. Majumder, A. A. Biswas, B. Ahammad, and M. M. Rahman, "An in-depth analysis of Convolutional Neural Network architectures with transfer learning for skin disease diagnosis," *Healthcare Analytics*, vol. 3, p. 100143, 2023.

[5] K. M. Hosny, W. M. El-Hady, F. M. Samy, E. Vrochidou, and G. A. Papakostas, "Multi-Class Classification of Plant Leaf Diseases Using Feature Fusion of Deep Convolutional Neural Network and Local Binary Pattern," *IEEE Access*, vol. 11, pp. 62307–62317, 2023. doi: 10.1109/ACCESS.2023.3286730.

[6] K. Ananthajothi, J. David, and A. Kavin, "Cardiovascular Disease Prediction Using Langchain," in *Proc. Int. Conf. Advances in Computing, Communication and Applied Informatics (ACCAI)*, Chennai, India, 2024, pp. 1–6. doi: 10.1109/ACCAI61061.2024.10601906.

[7] G. Lax, R. Nardone, and A. Russo, "Enabling secure health information sharing among healthcare organizations by public blockchain," *Multimedia Tools and Applications*, vol. 83, pp. 64795–64811, 2024.

[8] V. Upadrista, S. Nazir, H. Tianfield, *et al.*, "Consortium Blockchain for Reliable Remote Health Monitoring," Jan. 2024.

[9] Y. Y. Ghadi, T. Mazhar, T. Shahzad, *et al.*, "The role of blockchain to secure internet of medical things," *Scientific Reports*, vol. 14, p. 18422, 2024. doi: 10.1038/s41598-024-68529-x.

[10] P. Kumar, V. Subathra, Y. Swasthika, and V. Vishal, "Disease Diagnosis Using Machine Learning on Electronic Health Records," in *Proc. Int. Conf. Communication, Computing and Internet of Things (IC3IoT)*, Chennai, India, 2024, pp. 1–6.

[11] N. A. Atasoy and A. F. Abdulla Al Rahhawi, "Examining the classification performance of pre-trained capsule networks on imbalanced bone marrow cell dataset," *Int. J. Imaging Systems and Technology*, vol. 34, no. 3, p. e23067, 2024.

[12] D. Elangovan, C. Long, F. Bakrin, C. Tan, K. Goh, S. Yeoh, M. Loy, Z. Hussain, K. Lee, A. Idris, and L. Ming, "The Use of Blockchain Technology in the Health Care Sector: Systematic Review," *JMIR Med Inform*, 2022.

[13] M. A. Albahar, "Skin Lesion Classification Using Convolutional Neural Network With Novel Regularizer," *IEEE Access*, vol. 7, pp. 38306–38313, 2019. doi: 10.1109/ACCESS.2019.2906241.

[14] M. U. Amin, M. M. Iqbal, S. Saeed, N. Hameed, and M. J. Iqbal, "Skin Lesion Detection and Classification," *J. Comput. Biomed. Inform.*, vol. 6, no. 2, pp. 47–54, 2024.

[15] P. Kumar, V. K. S, P. L., and S. SenthilPandi, "Enhancing Face Mask Detection Using Data Augmentation Techniques," in *Proc. Int. Conf. Recent Advances in Science and Engineering Technology (ICRASET)*, B G Nagara, India, 2023, pp. 1–5. doi: 10.1109/ICRASET.2023.10420361.

[16] X. Zhang, Y. Mao, Q. Yang, and X. Zhang, "A Plant Leaf Disease Image ... [Incomplete citation — needs completion]."

[17] H. Saeed, H. Malik, U. Bashir, A. Ahmad, S. Riaz, M. Ilyas, W. A. Bukhari, and M. I. Ali Khan, "Blockchain technology in healthcare: A systematic review," *PLOS ONE*, vol. 17, no. 4, p. e0266462, 2022. doi: 10.1371/journal.pone.0266462.

[18] M. Gallazzi, S. Biavaschi, A. Bulgheroni, T. M. Gatti, S. Corchs, and I. Gallo, "A Large Dataset to Enhance Skin Cancer Classification With Transformer-Based Deep Neural Networks," *IEEE Access*, 2024.

[19] M. A. Kassem, K. M. Hosny, and M. M. Fouad, "Skin Lesions Classification Into Eight Classes for ISIC 2019 Using Deep Convolutional Neural Network and Transfer Learning," *IEEE Access*, vol. 8, pp. 114822–114832, 2020.

[20] S. Chenthara, K. Ahmed, H. Wang, F. Whittaker, and Z. Chen, "Healthchain: A novel framework on privacy preservation of electronic health records using blockchain technology," *PLOS ONE*, vol. 15, no. 12, p. e0243043, Dec. 2020. doi: 10.1371/journal.pone.0243043.