A

MAJOR PROJECT REPORT ON

FACIAL BEAUTY ATTRACTIVENESS WITH PREDICTABLE ANALYSIS USING DEEP LEARNING NETWORKS

Submitted in partial fulfilment of the requirement for the award of degree of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

SUBMITTED BY

T. NEHA	**218R1A04P2**
U. SUHITHA	**218R1A04P3**
V. JEEVAN	**218R1A04P4**
Y. ANIL KUMAR	**218R1A04P5**

Under the Esteemed Guidance of

Mr.A. GOPI
Associate professor

CMR ENGINEERING COLLEGE
EXPLORE TO INVENT

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

CMR ENGINEERING COLLEGE

UGC AUTONOMOUS

(Approved by AICTE, Affiliated to JNTU Hyderabad, Accredited by NBA)

Kandlakoya(V), Medchal(M), Telangana – 501401

(2024-2025)

CMR ENGINEERING COLLEGE

UGC AUTONOMOUS

(Approved by AICTE, Affiliated to JNTU Hyderabad, Accredited by NBA)

Kandlakoya(V), Medchal Road, Hyderabad - 501 401

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CERTIFICATE

This is to certify that the major-project work entitled **"FACIAL BEAUTY ATTRACTIVENESS WITH PREDICTABLE ANALYSIS USING DEEP LEARNING NETWORKS"** is being submitted by **T. NEHA** bearing Roll No **218R1A04P2, U. SUHITHA** bearing Roll No **218R1A04P3, V. JEEVAN** bearing Roll No **218R1A04P4, Y. ANIL KUMAR** bearing Roll No **218R1A04P5** in B.Tech IV-II semester, Electronics and Communication Engineering is a record Bonafide work carried out by then during the academic year 2024-25.The results embodied in this report have not been submitted to any other University for the award of any degree.

INTERNAL GUIDE
Mr.A.GOPI

HEAD OF THE DEPARTMENT
Dr. SUMAN MISHRA

EXTERNAL EXAMINER

ACKNOWLEDGEMENTS

We sincerely thank the management of our college **CMR Engineering College** for providing required facilities during our project work. We derive great pleasure in expressing our sincere gratitude to our Principal **Dr. A. S. Reddy** for his timely suggestions, which helped us to complete the project work successfully. It is the very auspicious moment we would like to express our gratitude to **Dr. SUMAN MISHRA,** Head of the Department, ECE for his consistent encouragement during the progress of this project.

We take it as a privilege to thank our project coordinator **Dr. T. SATYANARAYANA**, Associate Professor, Department of ECE for the ideas that led to complete the project work and we also thank him for his continuous guidance, support and unfailing patience, throughout the course of this work. We sincerely thank our project internal guide **MR.A. GOPI,** Associate Professor of ECE for guidance and encouragement in carrying out this project work.

DECLARATION

We hereby declare that the major project entitled **"FACIAL BEAUTY ATTRACTIVENESS WITH PREDICTABLE ANALYSIS USING DEEP LEARNING NETWORKS"** is the work done by us in campus at **CMR ENGINEERING COLLEGE,** Kandlakoya during the academic year 2024-2025 and is submitted as major project in partial fulfilment of the requirements for the award of degree of **BACHELOR OF TECHNOLOGY** in **ELECTRONICS AND COMMUNICATION ENGINEERING FROM JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY,HYDERABAD .**

T. NEHA	**(218R1A04P2)**
U. SUHITHA	**(218R1A04P3)**
V. JEEVAN	**(218R1A04P4)**
Y. ANIL KUMAR	**(218R1A04P5)**

ABSTRACT

The Facial Beauty Attractiveness Prediction and Cosmetics Recommendation App is a web-based application that utilizes machine learning algorithms to predict a user's Facial Beauty Attractiveness and provide personalized cosmetics recommendations. The app aims to address the problem of identifying the right Facial care products for individuals with different Facial Beauty Attractivenesss, which can be a challenging and time-consuming task. The app uses a deep learning model to analyze user-uploaded images and predict their Facial Beauty Attractiveness based on various features such as texture, color, and tone. The model is trained on a large dataset of Facial images and their corresponding Facial Beauty Attractivenesss, allowing it to accurately classify Facial Beauty Attractivenesss with high precision. Once the user's Facial Beauty Attractiveness is determined, the app provides personalized cosmetics recommendations tailored to their specific Facial needs. The recommendations are based on a comprehensive Facial care routine that includes cleansing, moisturizing, sun protection, and optional exfoliation steps. The app also provides recommendations for specific products that are suitable for the user's Facial Beauty Attractiveness, taking into account factors such as ingredients, texture, and price. The app is designed to be user-friendly and accessible, with a simple and intuitive interface that allows users to easily upload their images and view their Facial care recommendations. The app also includes a market page where users can browse and purchase recommended products from various brands.

CONTENTS

CHAPTER-4

CHAPTER-5

LIST OF FIGURES

Page no

CHAPTER-1
INTRODUCTION

This project, "Facial Beauty Attractiveness Prediction and Cosmetics Recommendation App," is a web application that uses deep learning to predict facial beauty attractiveness and offer personalized skincare recommendations. By analyzing user-uploaded facial images, the app identifies individual skin types and conditions such as oily, dry, combination, or normal—and recommends products accordingly. Utilizing advanced Convolutional Neural Networks (CNNs) and the Xception model, the system achieves high accuracy in facial feature analysis, which enhances the reliability of its predictions. The app's backend is built with Python, Flask, and Django, while the frontend uses HTML, CSS, and JavaScript, supported by a MySQL database for user data and recommendations.

1.1 OVERVIEW OF THE PROJECT

In addition to personalized insights, the app integrates an e-commerce marketplace, allowing users to browse and purchase recommended products. This project not only simplifies skincare choices but also showcases how deep learning can elevate user experiences by delivering tailored, data-driven beauty solutions. Future improvements may include expanding the product range and refining attractiveness scoring, making this app a valuable tool for personalized skincare.

1.2 OBJECTIVE OF THE PROJECT

The objective of the "Facial Beauty Attractiveness Prediction and Cosmetics Recommendation App" is to create an AI-driven system that analyzes user-uploaded facial images to predict beauty attractiveness scores accurately while classifying skin types such as oily, dry, combination, and normal. By leveraging advanced deep learning models like CNNs and the Xception model, the app provides personalized skincare and cosmetic recommendations tailored to individual skin characteristics.

Additionally, it offers a user- friendly interface that allows users to upload images, receive customized skincare advice, and access a marketplace for browsing and purchasing recommended products. This project aims to improve consumer decision-making by offering personalized, data-driven skincare solutions for a more effective beauty routine**.**

1.3 ORGANIZATION OF THE PROJECT

1. Objective of the Project

The primary objective is to develop an AI-powered system that accurately predicts facial attractiveness based on user-uploaded images and offers personalized skincare and cosmetic recommendations. By analyzing facial features, the app aims to help users make informed decisions about their skincare routines and product choices.

2. Technology Stack

The system leverages a robust technology stack including Python for the backend with frameworks like Flask and Django, a MySQL database for data storage, and HTML/CSS and JavaScript for the frontend interface. The deep learning component is powered by Convolutional Neural Networks (CNNs) and the Xception model to ensure high accuracy in facial image analysis.

3. Image Preprocessing and Feature Extraction

Image preprocessing is a critical step that includes grayscale conversion, median blur application, image resizing, and thresholding.

These methods enhance data clarity and reduce noise, enabling the CNN and Xception models to extract meaningful features for accurate classification of skin types and conditions.

4. Personalized Recommendations

Based on the attractiveness prediction and skin type classification, the system provides tailored product recommendations.

These recommendations cover various skincare categories, allowing users to address specific needs like moisturizing for dry skin or oil control for oily skin.

5. Future Enhancements

The project envisions future improvements, such as expanding the product database, refining attractiveness scoring algorithms, and enhancing model accuracy. Additionally, integrating more comprehensive skincare features and expanding applicability for both men and women could further improve the user experience.

The "Facial Beauty Attractiveness Prediction and Cosmetics Recommendation App" project is organized into comprehensive chapters that guide the reader through its development process and functionality. Starting with an Introduction that outlines the project's purpose, objectives, and the significance of AI in personalized skincare, the document proceeds to a Literature Survey that reviews relevant research and current approaches in facial beauty analysis and recommendation systems. The Software and System Architecture chapter details the technical framework, including front-end and back- end components, and explains the integration of deep learning models for accurate facial feature extraction.

The Hardware Requirements and Preprocessing sections follow, covering the necessary hardware and image preprocessing steps for optimal performance. The project then delves into Model Training and Evaluation, explaining the training process, dataset management, and evaluation metrics. System Implementation describes the functional flow from user registration to recommendations, while Testing and Results document the system's performance and accuracy. The project concludes with a Summary and Future Work section, discussing achieved goals and potential improvements, followed by References that provide sources supporting the project's development. This organized structure ensures clarity and depth in each stage of the project, from planning to execution and evaluation.

CHAPTER-2

LITERATURE SURVEY

2.1 AcneNet- A Deep CNN Based Classification Approach for Acne Classes and Facial Beauty Attractiveness

Facial diseases are very common and nowadays easy to get remedy from. But, sometimes properly diagnosing these diseases can be quite troublesome due to the stiff hard-todiscriminate nature of the symptoms they exhibit. Deep Neural Networks, since its recent advent, has started outperforming different algorithms in almost every sectors. One of the problem domains, where Deep Neural Networks are really thriving today, is Image Classification and Object and Pattern Discovery from images. A special type of Deep Neural Network is Convolutional Neural Networks (CNN), which are being extensively used for different sorts of computer vision and image classification related problems. Hence, we have proposed a novel approach, where we have developed and used a Deep Residual Neural Network model for classifying five classes of Acnes from images. Our model has achieved an approximate accuracy as much as 99.44% for one class, and the rest were also above 94% with fairly high precision and recall score.

2.2 Facial Net:A Deep Learning Framework for Facial Lesion Segmentation

There has been a steady increase in the incidence of Facial cancer worldwide, with a high rate of mortality. Early detection and segmentation of Facial lesions is crucial for timely diagnosis and treatment, necessary to improve the survival rate of patients. However, Facial lesion segmentation is a challenging task due to the low contrast of lesions and their high similarity in terms of appearance, to healthy tissue. This underlines the need for an accurate and automatic approach for Facial lesion segmentation. To tackle this issue, we propose a convolutional neural network (CNN) called Facial Net. The proposed CNN is a modified version of U-Net. The network employs dilated and densely block convolutions to incorporate multi-scale and global context information during training. We compared the performance of our approach with other state-of-the-art techniques, using the ISBI 2017 challenge dataset. Our approach outperformed the others in terms of the Dice coefficient, Jaccard index and sensitivity, evaluated on the held- out challenge test data set, across 5-fold cross validation experiments. Facial Net achieved an average value of 85.10, 76.67 and 93%, for the DC, JI and SE, respectively.

2.3 U-Net: Convolutional Networks for Biomedical Image Segmentation

There is large consent that successful training of deep networks requires many thousand annotated training samples. In this paper, we present a network and training strategy that relies on the strong use of data augmentation to use the available annotated samples more efficiently. The architecture consists of a contracting path to capture context and a symmetric expanding path that enables precise localization. We show that such a network can be trained end-to-end from very few images and outperforms the prior best method (a sliding-window convolutional network) on the ISBI challenge for segmentation of neuronal structures in electron microscopic stacks. Using the same network trained on transmitted light microscopy images (phase contrast and DIC) we won the ISBI cell tracking challenge 2015 in these categories by a large margin. Moreover, the network is fast. Segmentation of a 512x512 image takes less than a second on a recent GPU. The full implementation (based on Caffe) and the trained networks are available at http://lmb.informatik.uni-freiburg.de/people/ronneber/u-net .

2.4 Fully Convolutional Networks for Semantic Segmentation

Convolutional networks are powerful visual models that yield hierarchies of features. We show that convolutional networks by themselves, trained end-to-end, pixelsto-pixels, exceed the state- of-the-art in semantic segmentation.

Our key insight is to build “fully convolutional” networks that take input of arbitrary size and produce correspondingly-sized output with efficient inference and learning. We define and detail the space of fully convolutional networks, explain their application to spatially dense prediction tasks, and draw connections to prior models.

We adapt contemporary classification networks (AlexNet , the VGG net , and GoogLeNet) into fully convolutional networks and transfer their learned representations by fine- tuning to the segmentation task.

We then define a skip architecture that combines semantic information from a deep, coarse layer with appearance information from a shallow, fine layer to produce accurate and detailed segmentations.

Our fully convolutional network achieves stateo f-the-art segmentation of PASCAL VOC (20% relative improvement to 62.2% mean IU on 2012), NYUDv2, and SIFT Flow, while inference takes less than one fifth of a second for a typical image

2.5 Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation

Spatial pyramid pooling module or encode-decoder structure are used in deep neural networks for semantic segmentation task. The former networks are able to encode multi-scale contextual information by probing the incoming features with filters or pooling operations at multiple rates and multiple effective fields-of-view, while the latter networks can capture sharper object boundaries by gradually recovering the spatial information. In this work, we propose to combine the advantages from both methods. Specifically, our proposed model, DeepLabv3+, extends DeepLabv3 by adding a simple yet effective decoder module to refine the segmentation results especially along object boundaries.

We further explore the Xception model and apply the depthwise separable convolution to both Atrous Spatial Pyramid Pooling and decoder modules, resulting in a faster and stronger encoder-decoder network. We demonstrate the effectiveness of the proposed model on PASCAL VOC 2012 and Cityscapes datasets, achieving the test set performance of 89.0\% and 82.1\% without any post-processing.

CHAPTER-3

SYSTEM REQUIREMENTS AND DESIGN

The Unified Modeling Language (UML) is a powerful tool for visualizing, specifying, constructing, and documenting the artifacts of a system. It uses diagrams to represent various aspects of a system.

3.1 SOFTWARE REQUIREMENTS:

- **Operating system** : Windows 7 Ultimate.
- **Coding Language** : Python.
- **Front-End** : Python.
- **Back-End** : Django-ORM
- **Designing** : Html, css, javascript.
- **Data Base** : MySQL (WAMP Server).

3.2 UNIFIED MODELLING LANGUAGE DIAGRAMS

UML is a method for describing the system architecture in detail using the blue print. UML represents a collection of best engineering practice that has proven successful in the modeling of large and complex systems. The UML is very important parts of developing object-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. Using the helps UML helps project teams communicate explore potential designs and validate the architectural design of the software.

The Unified Modeling Language (UML) is a powerful tool for visualizing, specifying, constructing, and documenting the artifacts of a system. It uses diagrams to represent various aspects of a system. There are two main categories:

1. Structural Diagrams

Class Diagram: Represents the structure of a system by showing classes, their attributes, methods, and the relationships among objects.

Component Diagram: Illustrates how components are wired together to form larger systems.

Deployment Diagram: Shows the physical deployment of artifacts on nodes.

Object Diagram: Similar to a class diagram but represents specific instances.

2. Behavioral Diagrams

Use Case Diagram: Captures the functionality of a system from a user's perspective. Sequence Diagram: Depicts object interactions arranged in a time sequence.

Activity Diagram: Represents workflows of stepwise activities and actions. State Diagram: Shows the states and transitions of a particular object.

3.2.1 Use Case Diagram

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

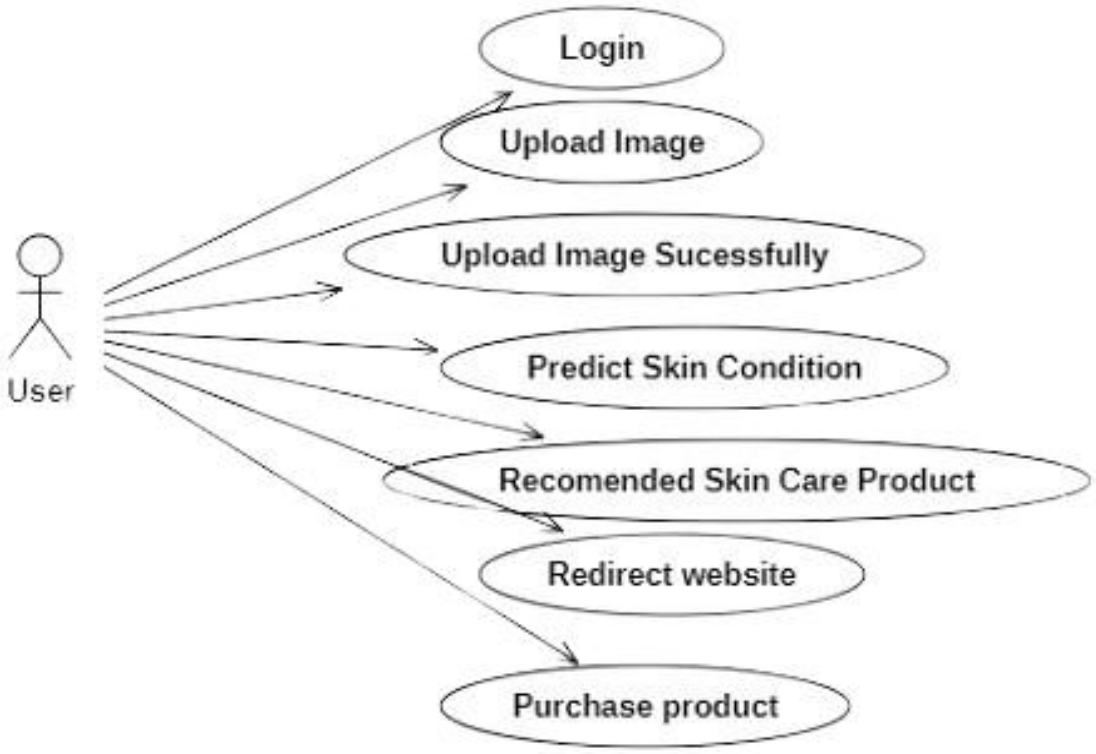


Fig 3.1.1 use case diagram

3.2.2 Activity Diagram

Activity diagrams are graphical representations of work flows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step work flows of components in a system. An activity diagram shows the overall flow of control.

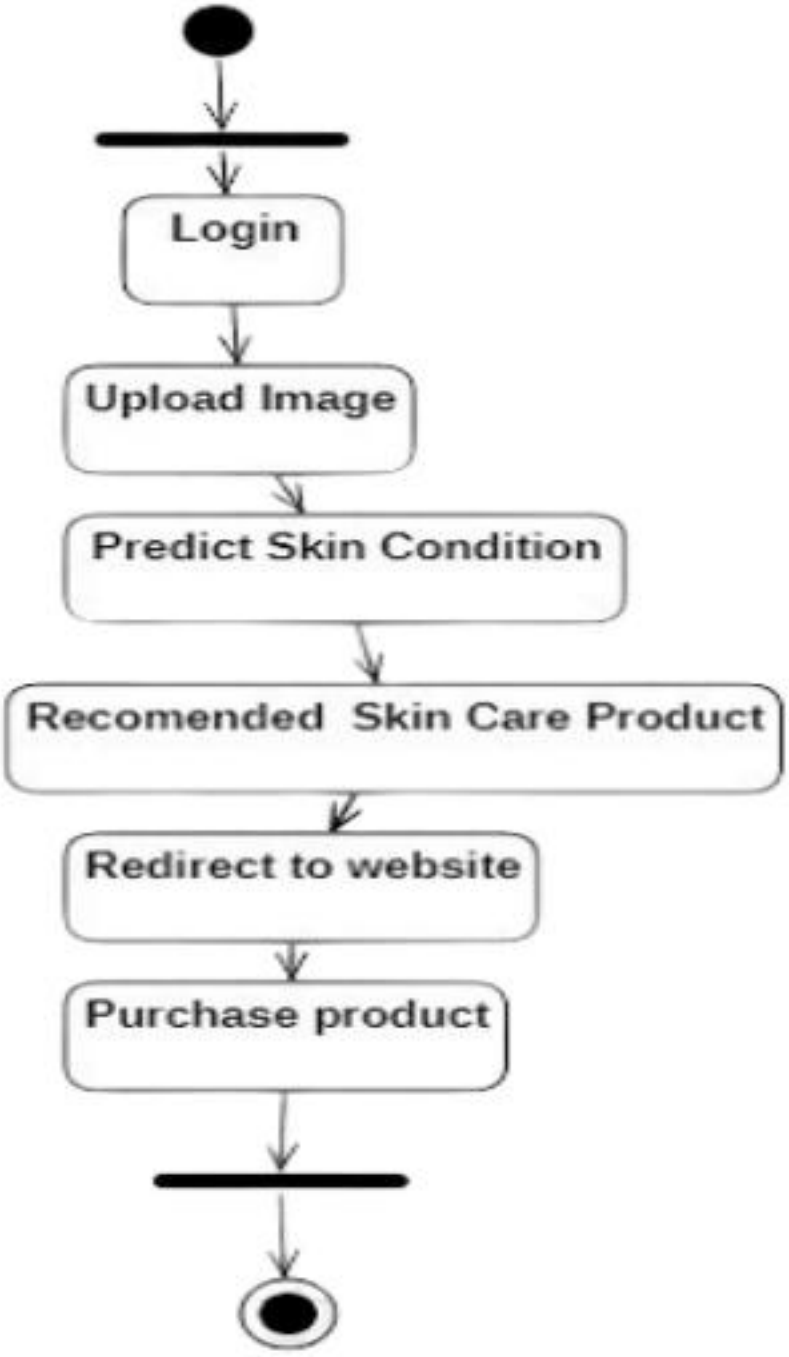


Fig: 3.2.2 activity diagram

3.2.3 Sequence Diagram

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagram.

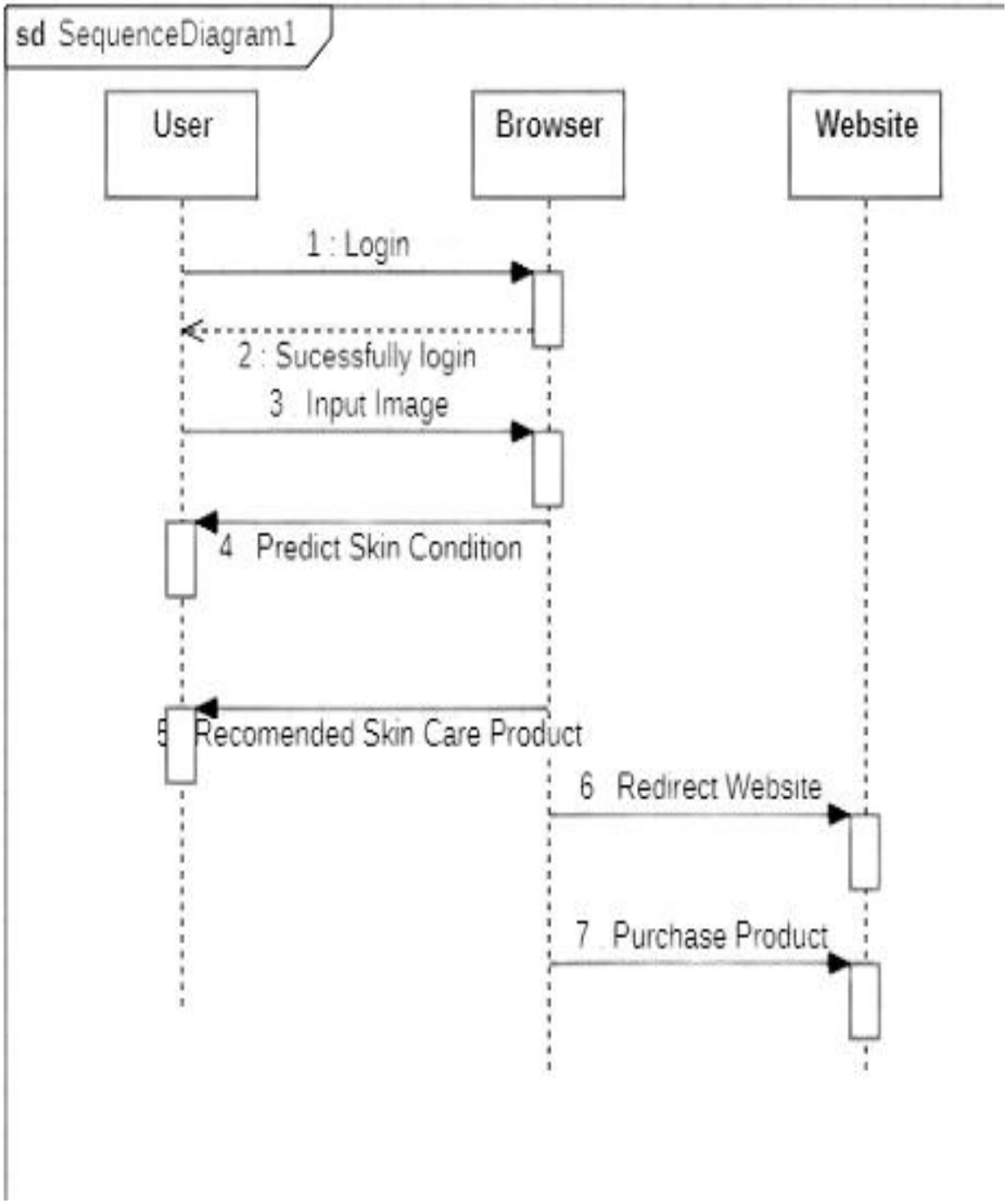


Fig 3.2.3 sequence diagram

3.2.4 Class Diagram

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

CHAPTER-4

SYSTEM ANALYSIS

4.1 EXISTING SYSTEM

ML Facial Beauty Attractiveness Prediction Relies on predefined rules and heuristics for Facial Beauty Attractiveness prediction. Limited adaptability to diverse Facial characteristics. May not accurately account for subtle nuances in Facial conditions. Requires manual updates to accommodate new data and trends. Utilises a machine learning model for Facial Beauty Attractiveness prediction. Learns from data, allowing adaptability to diverse Facial characteristics. Can capture subtle nuances in Facial conditions for more accurate predictions. Automatic learning reduces the need for frequent manual updates.

Rule-based approach uses predefined rules for Facial Beauty Attractiveness prediction.
Deep learning approach uses a model that learns from data for accurate predictions.
Deep learning adapts to diverse Facial characteristics and subtle nuances.
Deep learning requires less manual updates compared to rule-based approach.

Rule-Based Approach:

- Uses predefined rules for predictions based on fixed criteria.
- Requires manual updates when new conditions or variations arise.
- Limited in adapting to subtle nuances and diverse facial characteristics.
- More suitable for simpler systems with specific parameters.

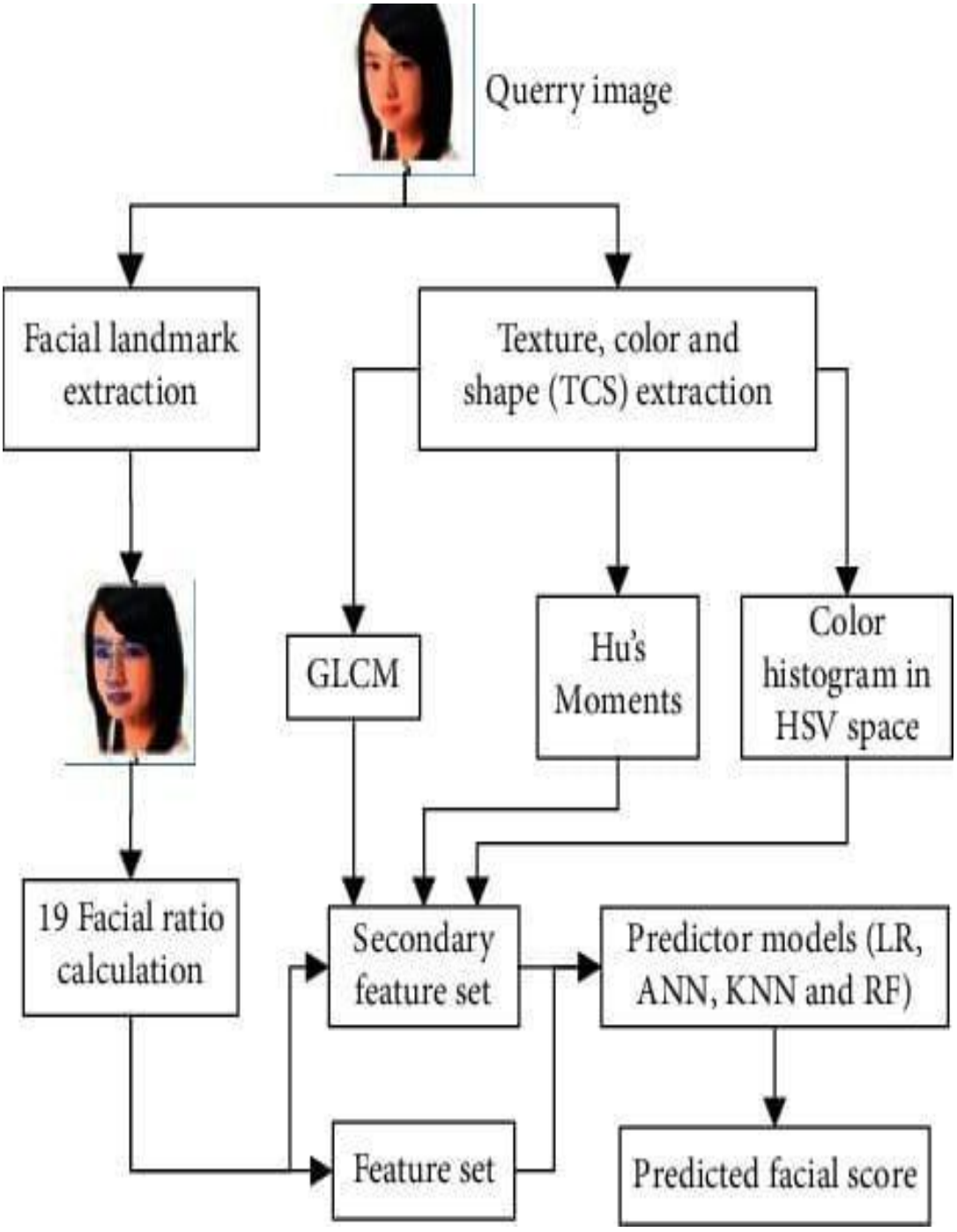
Deep Learning Approach:

- Utilizes data-driven models that learn patterns from large datasets.
- Adapts to diverse facial features and subtle variations effectively.
- Requires less frequent manual intervention as the model improves automatically with training.
- Provides greater accuracy and scalability for complex predictions.

Utilizes data-driven models that learn patterns from large datasets.
Adapts to diverse facial features and subtle variations effectively.
Requires less frequent manual intervention as the model improves automatically with training.
Provides greater accuracy and scalability for complex predictions.

Disadvantages of Machine Learning System:

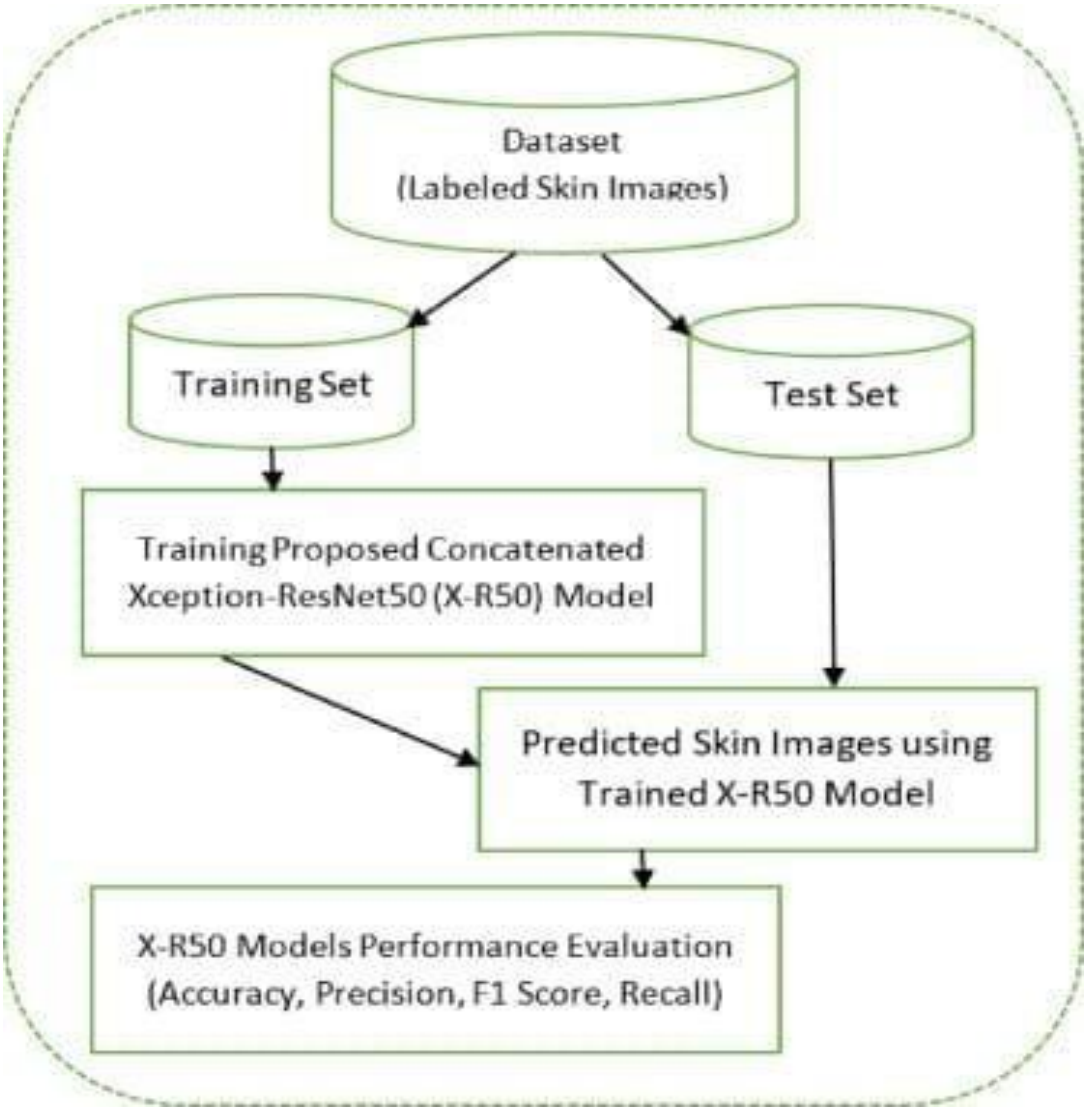
- Dependency on large and diverse datasets for effective training.
- Vulnerable to biases present in training data, impacting prediction accuracy.
- Lack of transparency in decision-making compared to rule-based systems.
- Continuous monitoring and updates required to maintain performance trends.



4.2 PROPOSED SYSTEM

The beauty and cosmetology industry is seeing significant advancements through the use of artificial intelligence, particularly in enhancing personal care.

A key challenge remains the accurate identification of Facial Beauty Attractivenesss and ensuring product safety to avoid allergic reactions. While numerous e-commerce platforms offer cosmetics and Facial care products, there is a gap in services that predict Facial conditions and recommend suitable products. The goal of this application is to leverage the capabilities of AI and machine learning, using digital Facial analysis to tailor product suggestions. By employing the EXCEPTION algorithm, the application aims to analyze Facial tones and conditions effectively, enabling users to choose cosmetics that are safe and appropriate for their Facial Beauty Attractiveness.



- The beauty and cosmetology industry is advancing through artificial intelligence, enhancing personal care.
- A key challenge remains the accurate identification of Facial Beauty Attractiveness and ensuring product safety.
- The application aims to leverage AI and machine learning, using digital Facial analysis to tailor product suggestions.
- By employing the XCEPTION algorithm, the application analyzes Facial tones and conditions effectively, enabling users to choose safe and suitable cosmetics.

This is a fascinating use case for AI in the beauty and cosmetology industry! Leveraging AI to bridge the gap in personalized skincare and cosmetics recommendations could greatly enhance user experience and safety.

The beauty and cosmetology industry is advancing through artificial intelligence, enhancing personal care.

A key challenge remains the accurate identification of Facial Beauty Attractiveness and ensuring product safety.

The application aims to leverage AI and machine learning, using digital Facial analysis to tailor product suggestions.

By employing the XCEPTION algorithm, the application analyzes Facial tones and conditions effectively, enabling users to choose safe and suitable cosmetics.

Actors:

User: Interacts with the application for personalized recommendations. AI System: Performs the digital analysis of facial tones and conditions.

Product Database: Provides the details of cosmetics and facial care products.

Use Cases:

Facial Analysis: The AI system examines the user's facial features using the EXCEPTION algorithm.

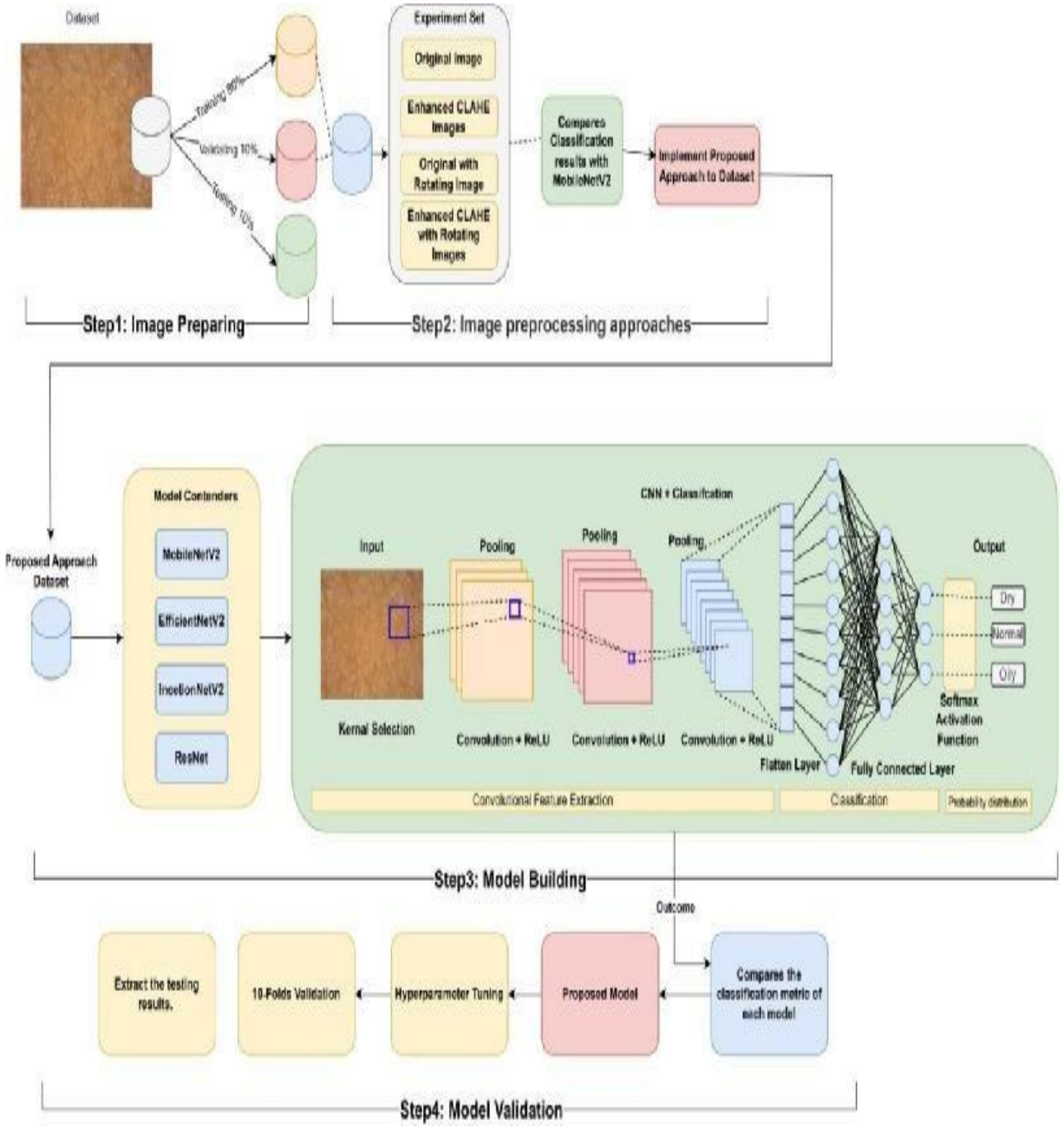
Product Matching: Based on analysis, the system retrieves safe and suitable products.

Feedback and Review:

Users can provide feedback on recommendations to improve algorithms.

Advantages

1. Personalized Recommendations
2. product recommendations
3. Improved User Experience
4. Better Product Selection
5. Facial Health Monitoring



4.3 FUNCTIONAL REQUIREMENTS

The Functional requirements for a system describe the functionality or the services that the system is expected to provide. These are the statements of services the system should provide and how the system should react to particular inputs and how the system should behave in particular situation. User Registartion: User Register with their Registration details. User Login: User Login their account using password Live Inputs: Inputs Given By the User requirement. Load Model: Trained or Tested Model will be load. Predict Output: Output will be predict based on parameters.

4.4 NON-FUNCTIONAL REQUIREMENTS

The non-functional requirements describe the system constraints. Performance: The application should have better accuracy and should provide prediction in less time. Scalability: The system must have the potential to be enlarged to accommodate the growth. Capability: The capability of the storage should be high so the large amount of data can be stored in order to train the model.

4.5 FEASIBILITY ANALYSIS

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential. Three key considerations are involved in the feasibility analysis are:

- ➢ Economical feasibility
- ➢ Technical feasibility
- ➢ Social feasibility

4.5.1 Economical Feasibility

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

4.5.2 Technical Feasibility

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

4.5.3 Social Feasibility

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

CHAPTER-5

SYSTEM IMPLEMENTATION

5.1 DATASET GENERATION

It is necessary to compile a complete database for the cosmetic suggestion as it is important to analyse the different Facial Beauty Attractiveness. There is a multiple set of datasets that are given as a reference the comparison of the data to the captured image. The images are labelled and processed using the (PACS). The data distribution is of two phases the internal dataset and temporal dataset. Where the temporal dataset is used to train the data and the internal dataset is used to test the dataset by splitting it into three categories as training (70%), validation (15%) and test (15%) subsets. The distribution of dataset consists of 32% maxillary, 32% Front land34% Normal sinusitis.

5.1.1. Feature Extraction

The Feature Extraction process is used to reduce the amount of resources without losing the important data and remove irrelevant data. It can also reduce the presence of redundant data that is mostly used for analysis of Facial tones. The reduction of data can increase the effort of building variable features and facilitate the learning and machine generation speed in machine learning.

5.1.2. Dataset

One major advantage of using CNNs over NNs is that you do not need to flatten the input images to ID as they are capable of working with image data in 2D.This helps in retaining the "spatial" properties of images. We are using different type of Facial dataset which consist on four categories.

5.2 PREPROCESSING

The Pre-processing is the method of improving the image data and also in the process of enhancing the image features for further processes to be performed in the application. Here, the captured image or the inputted image has been pre- processed for the clear observation of the image data and the identification of the Facial Beauty Attractiveness for further analysis of products that match the Facial condition. The pre-processing methods we used are the gravy scaling, medium blur, image resizing thresholding.

5.2.1. Grayscale Conversion

Gray scale conversion is the process of converting the digitally inputted image to gray where some image pixels just show the images that are in darkest black to brightest white. In simple we just have converted the imahes to gray black and white that are green red and blue. The mathematical formula for the gray scale conversion

The gray scale conversion process provided us the image of 24bit into 8bit image.

5.2.2. Median Blur

Median blur is the process of removal of noise from the image during the classification phase and provide clear data input to the algorithm to proceed the further steps. The Median blur method during the process of noise removal provided us the smooth corner and clear image to identify the type of Facial and the Facial acnes and identify the products for the Facial.

5.2.3. Image Resizing

Image resizing is the method or technique used to increase or decrease the pixel of image and shorten the resolution of the image input provided. Resizing without resampling means to change the physical size of the image but not the data on the image. The method helped us remove the unwanted data and focus on the mostly required information for the Facial Beauty Attractiveness analysis. The process helped us resize the images given as input with the system required image pixel and data.

5.2.4. Thresholding

Thresholding is the simple process of segmentation. This includes the process of changing the pixel of image for easier analysis. Thresholding method helped us to convert the gray scale image and the coloured to binary image where the clear and perfect quality of the image obtained and the image analysis provided the classification of the Facial Beauty Attractivenesss and separated the image from its background.

5.3 TRAINING AND EVALUATION

To Minimize Unnecessary voice and convert our RGP input Images to Gray scale, and threshold to convert the 128bit images to binary input values for the analysis of Facial Beauty Attractiveness. The Inputs are given to prototype for the testing and training of the images after completion of the image pre-processing method. The important function is to choose the filter that can be used on photos to extract the proper features, for the training of images into the CNN model and evaluate the dataset. The Experiment

included various filters like binary image and thresholding and gray scale image scaling and then the median blur for the clear analysis and for the training process.

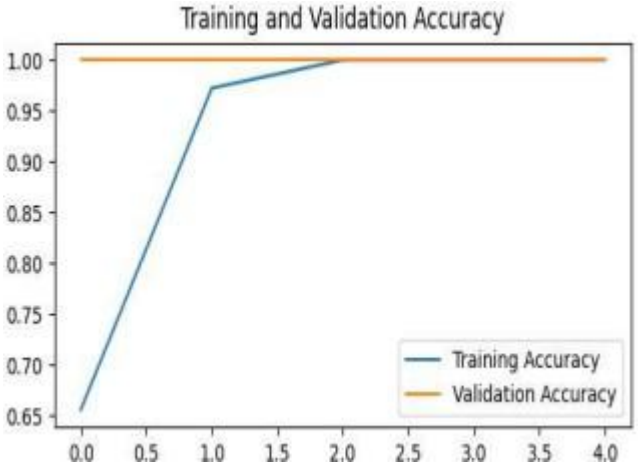


Fig.5.3.1 The Training and validation accuracy

The graph (fig 6.3.1) represents the Training and Validation Accuracy of the proposed system It clearly depicts the trained data and the accuracy of the classified input for the effective performance of the system. The chart provides the clarity on the training phase where the given input data and the trained set of data has been matched perfectly and the accuracy of validating the input comes under the provided category of datasets which can deliver a clarified detection of Facial Beauty Attractiveness and suitable suggestion for that Facial Beauty Attractiveness.

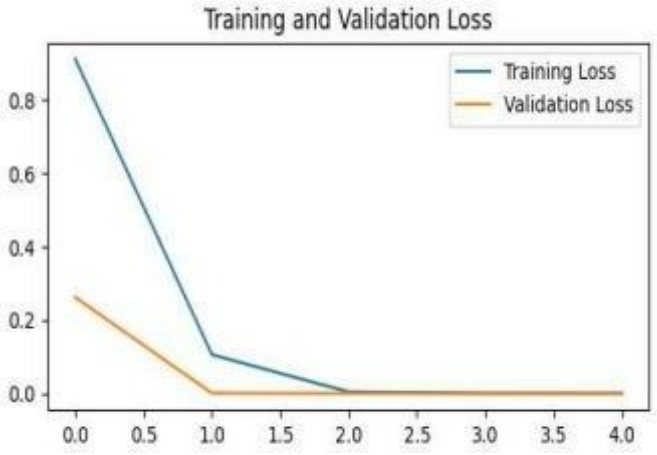


Fig.5.3.2 The Training and validation loss

The graph (fig 6.3.2) represents the Training and the validation loss of the proposed system and it does not include any of the gain value and accurate results. The dataset has been trained into five sets of testing phase where the accuracy been provided. The

chart provides the result of the five tested sets of trained data with an accurate loss and validation. The actual result is arrived after five frequent tests so that the accurate result be provided for the suggestion.

5.4 CLASSIFICATION

The Classification (in fig 6.4) of different types of Facial helped us find the products that match different Facial tones without causing any trouble to the Facial for the applicants.

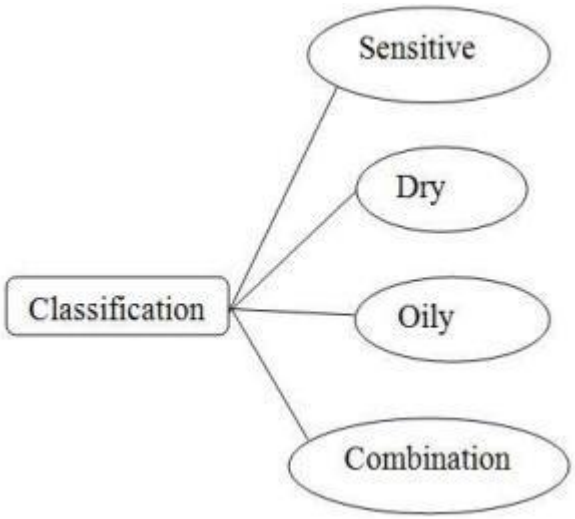


Fig 5.4: Types of Facial

Sensitive

The type of Facial that does not fit for any chemicals and also not suitable for some of the organic cosmetics. This type of Facial require more care and the application provide the type of cosmetic that does not give any kind of allergic symptoms to any type of Facial condition.

Fig 5.4.1: Facial classification

Dry

The main cause of dry Facial is the cold climatic condition and dehydration where the dryness cause irritation in Facial . The application provide the products like moisturizer and gels that cause hydration and creams for the climatic condition and Facial condition like extreme dry or minimum dry.

Oily

Oily Facial persons mostly get tan and black while coming out in sun and get hot fire during the winter. This causes trouble to persons mostly by causing acne and clogged pores. The Application provide the service of choosing the products according to the climatic condition like creams and sunscreen lotion for the summer and moisturizers for winter and snow.

Combination

Combination Facial is the type of Facial where everything matches the Facial of the person without causing any allergy and Facial problems in person with that Facial Beauty Attractiveness. The Application suggest the products that improve the Facial colour and texture and lotions for pores and acnes or black dots present in Facial.

5.5 FEATURES EXTRACTED

Feature Extraction is the process of taking the most important images that is the captured image and it enters into the image saturation process for the identification of Facial Beauty Attractiveness and product suggestion. In the proposed system we have implemented the best composition of products. The compact input got compressed and. gone through various attempts in CNN layers as the convolutional layer, max pooling layer and the flatten layer to get a clear and accurate visibility for the prediction of the Facial and suggest the products that match the people Facial tone. The input image extract the features that include a combination of product ingredients and product suitability for the Facial Beauty Attractiveness that are inputted into the various layers for the automatic feature learning. Finally the system provides a composition of cosmetic product on the output layer

5.6 MODULES

Load

Data

Data pre-cessing

Feature Selection

Feature

Extraction Deep

Learning

5.6.1 Load Data:

Pandas allows you to import data from a wide range of data sources directly into a dataframe. These can be static files, such as CSV, TSV, fixed width files, Microsoft Excel, JSON, SAS and SPSS files, as well as a range of popular databases, such as MySQL, PostgreSQL and Google Big Query. You can even scrape data directly from web pages into Pandas data frames.

5.6.2 Data Collection:

Data collection means pooling data by scraping, capturing, and loading it from multiple sources, including offline and online sources. High volumes of data collection or data creation can be the hardest part of a machine learning project, especially at scale.Data collection allows you to capture a record of past events so that we can use data analysis to find recurring patterns. From those patterns, you build predictive models using machine learning algorithms that look for trends and predict future changes.Predictive models are only as good as the data from which they are built, so good data collection practices are crucial to developing high- performing models. The data needs to be error-free and contain relevant information for the task at hand. For example, a loan default model would not benefit from tiger population sizes but could benefit from gas prices over time.

5.6.3 Data Pre-processing:

Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model. When creating a machine learning project, it is not always a case that we come across the clean and formatted data. And while doing any operation with data, it is mandatory to clean it and put in a formatted way. So for this, we use data preprocessing task.

A real-world data generally contains noises, missing values, and maybe in an unusable format which cannot be directly used for machine learning models.

It involves below steps:

- Getting the dataset
- Importing libraries
- Importing datasets
- Finding Missing Data
- Encoding Categorical Data
- Splitting dataset into training and test set
- Feature scaling

5.6.4 Feature Selection:

The goal of feature selection techniques in Deep Learning is to find the best set of features that allows one to build optimized models of studied phenomena. The techniques for feature selection in Deep Learning can be broadly classified into the following categories: Supervised Techniques: These techniques can be used for labeled data and to identify the relevant features for increasing the efficiency of supervised models like classification and regression. For Example- linear regression, decision tree, SVM, etc. Unsupervised Techniques: These techniques can be used for unlabeled data. For Example- K-Means Clustering, Principal Component Analysis, Hierarchical Clustering, etc. From a taxonomic point of view, these techniques are classified into filter, wrapper, embedded, and hybrid methods

5.6.5 Feature Extraction:

Feature extraction is a part of the dimensionality reduction process, in which, an initial set of the raw data is divided and reduced to more manageable groups. So when you want to process it will be easier. The most important characteristic of these large data sets is that they have a large number of variables. These variables require a lot of computing resources to process. So Feature extraction helps to get the best feature from those big data sets by selecting and combining variables into features, thus, effectively reducing the amount of data. These features are easy to process, but still able to describe the actual data set with accuracy and originality. Color features are obtained by extracting statistical features from image histograms. They are used to provide a general description of color statistics in the image.

5.6.6 Deep Learning

Deep learning is a subset of machine learning that focuses on the use of artificial neural networks to model and solve complex problems. It's inspired by the structure and function of the human brain, specifically the interconnected layers of neurons. Deep learning has gained significant popularity and success in various fields, thanks to its ability to automatically learn hierarchical representations from data.

Here are some key points about deep learning:

1. Neural Networks:At the core of deep learning are neural networks, which are composed of layers of interconnected nodes or artificial neurons. The "deep" in deep learning refers to the depth of these networks, meaning they have multiple layers (deep architectures).

2. Deep Neural Networks (DNNs): Deep learning often involves training deep neural networks, which can range from a few layers (shallow networks) to many layers (deep networks). The depth allows the network to learn intricate features and representations from the input data.

3. Representation Learning: Deep learning excels at automatic feature extraction and representation learning. Instead of manually engineering features, the model learns to extract relevant features from the data during the training process.

4. Training with Backpropagation: Deep neural networks are trained using a process called backpropagation. It involves iteratively adjusting the weights of connections between neurons to minimize the difference between the predicted output and the actual target. This is typically done using optimization algorithms like stochastic gradient descent.

5. Architectures: Various deep learning architectures exist, including Convolutional Neural Networks (CNNs) for image-related tasks, Recurrent Neural Networks (RNNs) for sequential data, and Transformers for natural language processing. These architectures are designed to handle specific types of data and tasks.

ResNet (Residual Network) and DenseNet (Densely Connected Convolutional Network) are two popular architectures in the field of deep learning, specifically designed to address challenges related to training very deep neural networks. Let's take a brief look at each:

ResNet (Residual Network):

1. Introduction: ResNet was introduced by Kaiming He et al. in 2015. The key innovation is the use of residual blocks, which contain shortcut connections that skip one or more layers. This helps in addressing the vanishing gradient problem and enables the training of very deep networks.

2. Shortcut Connections:The core idea of ResNet is the introduction of shortcut connections, or skip connections, that bypass one or more layers. These connections enable the gradient to flow more easily during backpropagation, facilitating the training of deeper networks.

3. Architecture: ResNet architectures typically consist of a series of residual blocks. Each block contains multiple convolutional layers, and the shortcut connections add the original input to the output of the block.

4. Benefits: ResNet has been successful in training extremely deep networks, reaching hundreds of layers. It has been widely used in image classification, object detection, and other computer vision tasks.

DenseNet (Densely Connected Convolutional Network):

1. Introduction: DenseNet, introduced by Gao Huang et al. in 2017, takes a different approach by promoting dense connectivity between layers. In a DenseNet, each layer receives input from all preceding layers and passes its output to all subsequent layers.

2. Dense Blocks: The fundamental building block in DenseNet is the dense block. It consists of multiple layers, and each layer receives the feature maps from all preceding layers as input. This dense connectivity enhances feature reuse and promotes gradient flow.

3. Transition Blocks: To manage the growth of parameters and computation in dense blocks, transition blocks are used to reduce the number of feature maps before passing them to the next dense block. This helps in maintaining computational efficiency.

4. Benefits: DenseNet often requires fewer parameters compared to traditional architectures, leading to more efficient models. The dense connectivity also helps in mitigating the vanishing gradient problem and encourages feature reuse.

5. Applications: DenseNet has been successfully applied to tasks such as image

classification and segmentation, achieving competitive performance with fewer parameters.

Training Dataset

The training data is the biggest (in -size) subset of the original dataset, which is used to train or fit the machine learning model. Firstly, the training data is fed to the ML algorithms, which lets them learn how to make predictions for the given task.

Test Dataset

Once we train the model with the training dataset, it's time to test the model with the test dataset. This dataset evaluates the performance of the model and ensures that the model can generalize well with the new or unseen dataset. **The test dataset is another subset of original data, which is independent of the training dataset.** However, it has some similar types of features and class probability distribution and uses it as a benchmark for model evaluation once the model training is completed. Test data is a well-organized dataset that contains data for each type of scenario for a given problem that the model would be facing when used in the real world. Usually, the test dataset is approximately 20-25% of the total original data for an ML project.

Gated Recurrent Unit (GRU) Algorithm

Gated Recurrent Unit (GRU) is a type of recurrent neural network (RNN) that was introduced by Cho et al. in 2014 as a simpler alternative to Long Short-Term Memory (LSTM) networks. Like LSTM, GRU can process sequential data such as text, speech, and time-series data.

The basic idea behind GRU is to use gating mechanisms to selectively update the hidden state of the network at each time step. The gating mechanisms are used to control the flow of information in and out of the network. The GRU has two gating mechanisms, called the reset gate and the update gate.

The reset gate determines how much of the previous hidden state should be forgotten, while the update gate determines how much of the new input should be used to update the hidden state. The output of the GRU is calculated based on the updated hidden state.

The equations used to calculate the reset gate, update gate, and hidden state of a GRU are as follows:

Reset gate: r_t = sigmoid(W_r * [h_{t-1}, x_t])

Update gate: z_t = sigmoid(W_z * [h_{t-1}, x_t])

Candidate hidden state: h_t' = tanh(W_h * [r_t * h_{t-1}, x_t])

Hidden state: h_t = (1 – z_t) * h_{t-1} + z_t * h_t'

where W_r, W_z, and W_h are learnable weight matrices, x_t is the input at time step t,

In summary, GRU networks are a type of RNN that use gating mechanisms to selectively update the hidden state at each time step, allowing them to effectively model sequential data. They have been shown to be effective in various natural language processing tasks, such as language modeling, machine translation, and speech recognition.

Prerequisites: Recurrent Neural Networks, Long Short Term Memory Networks

To solve the Vanishing-Exploding gradients problem often encountered during the operation of a basic Recurrent Neural Network, many variations were developed. One of the most famous variations is the **Long Short Term Memory Network(LSTM)**. One of the lesser- known but equally effective variations is the **Gated Recurrent Unit Network(GRU)**.

Unlike LSTM, it consists of only three gates and does not maintain an Internal Cell State. The information which is stored in the Internal Cell State in an LSTM recurrent unit is incorporated into the hidden state of the Gated Recurrent Unit. This collective information is passed onto the next Gated Recurrent Unit. The different gates of a GRU are as described below:-

1. **Update Gate(z):** It determines how much of the past knowledge needs to be passed along into the future. It is analogous to the Output Gate in an LSTM recurrent unit.
2. **Reset Gate(r):** It determines how much of the past knowledge to forget. It is analogous to the combination of the Input Gate and the Forget Gate in an LSTM recurrent unit.
3. **Current Memory Gate( ):** It is often overlooked during a typical discussion on Gated Recurrent Unit Network. It is incorporated into the Reset Gate just like the Input Modulation Gate is a sub-part of the Input Gate and is used to introduce some non-

linearity into the input and to also make the input Zero-mean. Another reason to make it a sub-part of the Reset gate is to reduce the effect that previous information has on the current information that is being passed into the future.

The basic work-flow of a Gated Recurrent Unit Network is similar to that of a basic Recurrent Neural Network when illustrated, the main difference between the two is in the internal working within each recurrent unit as Gated Recurrent Unit networks consist of gates which modulate the current input and the previous hidden state.

CONVOLUTIONAL NEURAL NETWORK :

Convolutional Neural Networks (CNNs) are a type of deep learning algorithm commonly used for image and video recognition. Here's a brief overview of the working process of a CNN:

1. **Input Layer:**

   Takes in the raw input data, which is usually an image in the case of computer vision tasks.

2. **Convolutional Layers:**

- Apply a set of filters (kernels) to the input data, performing convolution operations.
- Filters detect patterns and features in the input, such as edges, textures, or shapes.
- Convolution helps preserve spatial relationships within the data.

3. Activation Function:

- Introduces non-linearity to the system, typically using functions like ReLU (Rectified Linear Unit).
- Helps the network learn complex patterns and relationships.

4. Pooling (Subsampling) Layers:

- Reduce the spatial dimensions of the input volume.
- Common pooling methods include max pooling or average pooling.
- Helps decrease computational complexity and reduces the risk of overfitting.

5. Flattening:

- Transform the multi-dimensional data into a one-dimensional vector.
- Prepares the data for input into fully connected layers.

6. Fully Connected Layers:

- Neurons in a layer are connected to every neuron in the adjacent layer.
- Learn complex relationships in the data.
- Typically used in the final layers of the network.

7. Output Layer:

- Produces the final predictions based on the learned features and patterns.
- The activation function in the output layer depends on the task (e.g., softmax for classification).

8. Loss Function:

- Measures the difference between the predicted output and the actual target.
- The goal is to minimize this difference during training.

9. Backpropagation:

- Calculates the gradient of the loss function with respect to the weights.
- Adjusts the weights using optimization algorithms (e.g., stochastic gradient descent) to minimize the loss.

10. Training:

- Iteratively updates the network's parameters using backpropagation and optimization.
- Continues until the model performs well on the training data.

11. Testing/Prediction:

- The trained model is used to make predictions on new, unseen data.

In summary, CNNs are designed to automatically and adaptively learn spatial hierarchies of features from input data, making them well-suited for tasks like image recognition detection.

5.6.7 Model Selection In Deep Learning:

Model selection in Deep Learning is the process of selecting the best algorithm and model architecture for a specific job or dataset. It entails assessing and contrasting various models to identify the one that best fits the data & produces the best results. Model complexity, data handling capabilities, and generalizability to new examples are all taken into account while choosing a model. Models are evaluated and contrasted using methods like cross-validation, and grid search, as well as indicators like accuracy and mean squared error. Finding a model that balances complexity and performance to produce reliable predictions and strong generalization abilities is the aim of model selection.

5.7 TECHNOLOGIES

Python

Flask

5.7.1 Python

Python is a highly interpreted programming language Python provides man GUI development possibilities (Graphical User Interface). flask is, the most frequently used technique of all GUI methods. It's a standard Python interface to the Python Tk GUI toolkit.

Python is the quickest and simplest method for creating GUI apps using Flask outputs. It is a simple job to create a GUI using flask. Python is a common, flexible and popular language of programming.

It is excellent as a first language since it is succinet and simple to understand and also good to use in any programmer's pile because it can be utilized from development of the web to software. It's basic, easy-to-use grammar, making it the ideal language to first learn computer programming.

Most implementations of Python (including C and Python), include a read- eval-print (REPL) loop that enables the user to act as a command-line interpreter that results in sequence and instantaneous intake of instructions. Other shells like as IDLE and Python provide extra features such as auto-completion, session retention and highlighting of syntax.

Interactive mode programming

Invoking the interpreter without passing a script file as a parameter brings up the following prompt

– $ python

Python 2.4.3 (#1, Nov 11 2010, 13:34:43)

[GCC 4.1.2 20080704 (Red Hat 4.1.2-48)] on linux2

Type "help", "copyright", "credits" or "license" for more

information Type the following text at the Python prompt and

press the Enter –

>>> print "Hello, Python!" If you are running new version of Python, then you would need to use print statement with parenthesis as in print ("Hello, Python!");. However in Python version 2.4.3, this produces the following result

– Hello,

Script mode programming

Invoking the interpreter with a script parameter begins execution of the script and continues until the script is finished. When the script is finished, the interpreter is no longer active.

Let us write a simple Python program in a script. Python files have extension .py. Type the following source code in a test.py file –

Live Demo print "Hello, Python!" We assume that you have Python interpreter set in PATH variable. Now, try to run this program as follows –

$ python test.py This produces the following result –

Hello, Python! Let us try another way to execute a Python script. Here is the modified test.py file –

Live Demo
#!/usr/bin/python print "Hello, Python!"

We assume that you have Python interpreter available in /usr/bin directory. Now, try to run this program as follows –
$ chmod +x test.py # This is to make file executable
$./test.py
This produces the following result

Flask web framework

Flask is a web application framework written in Python. Armin Ronacher, who leads an international group of Python enthusiasts named Pocco, develops it. Flask is based on Werkzeug WSGI toolkit and Jinja2 template engine. Both are Pocco projects.Unlike the Django framework, Flask is very Pythonic. It’s easy to get started with Flask, because it doesn’t have a huge learning curve. On top of that it’s very explicit, which increases readability. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre- existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

CHAPTER -6

CODING

6.1 : app.py

```
import os
from flask import Flask, redirect, render_template, request
from PIL import Image
import torchvision.transforms.functional as TF
import CNN
import numpy as np
import torch
import pandas as pd
import csv

supplement_info = pd.read_csv('supplement_info.csv',encoding='cp1252')
```

```
model = CNN.CNN(39)
# model.load_state_dict(torch.load("plant_disease_model_1_latest.pt"))
model.eval()
```

```
def recommend_cosmetics(skin_type):
    if skin_type == "Normal Skin":
        return """For normal skin, you're lucky to have a well-balanced
complexion. Your primary goal is to maintain your skin's health. Here's a
comprehensive skincare routine:
        1. Cleanser: Use a gentle, sulfate-free cleanser to remove impurities.
        2. Moisturizer: Opt for a lightweight, non-comedogenic moisturizer.
        3. Sunscreen: Apply a broad-spectrum sunscreen daily to protect your
skin.
        4. Optional: You can incorporate a mild exfoliant 1-2 times a week for
extra glow.
        """
```

```
    elif skin_type == "Sensitive Skin":
        return """Sensitive skin requires extra care to minimize irritation and
redness. Consider these steps:
        1. Cleanser: Use a fragrance-free, hypoallergenic cleanser.
        2. Moisturizer: Choose a product with soothing ingredients like aloe
vera or chamomile.
        3. Sunscreen: Use a physical sunscreen with zinc oxide or titanium
dioxide.
        4. Avoid harsh exfoliants and strong active ingredients, and patch-test
new products.
        """
```

```
    elif skin_type == "Dry Skin":
        return """Dry skin needs intense hydration and protection. Follow this
regimen:
```

```
        1. Cleanser: Use a hydrating, gentle cleanser.
        2. Moisturizer: Opt for a rich, creamy moisturizer with
ingredients like hyaluronic acid or ceramides.
        3. Sunscreen: Apply a broad-spectrum sunscreen daily to prevent
further dryness.
        4. Consider adding a hydrating serum or facial oil to your routine
for added moisture.
        """
```

```
    elif skin_type == "Oily Skin":
        return """To control excess oil and minimize breakouts, follow these
steps:
        1. Cleanser: Use a foaming, salicylic acid-based cleanser to control
oil.
        2. Moisturizer: Choose an oil-free, lightweight, and non-comedogenic
moisturizer.
        3. Sunscreen: Use an oil-free, mattifying sunscreen.
        4. Consider using products with ingredients like salicylic acid,
niacinamide, or witch hazel to manage oil and acne.
        """
```

```
    elif skin_type == "Scaly Skin":
        return """Scaly skin often results from dryness and flakiness. Try
these skincare steps:
        1. Cleanser: Use a gentle exfoliating cleanser to remove dead skin
cells.
        2. Moisturizer: Choose a rich, emollient moisturizer to lock in
moisture.
        3. Sunscreen: Protect your skin from further damage with a daily
sunscreen.
        4. Exfoliate with products containing glycolic acid or lactic acid to
improve texture.
        """
```

```
    elif skin_type == "Red_Spots_skin":
        return """Red spots can be due to various causes. Here's a general
approach:
        1. Cleanser: Use a gentle, fragrance-free cleanser to avoid irritation.
        2. Moisturizer: Select a calming and hydrating moisturizer.
        3. Sunscreen: Shield your skin from further damage with a broad-
spectrum sunscreen.
        4. Consult a dermatologist to identify the specific cause of redness
and receive tailored treatment.
        """
```

```
    elif skin_type == "Skin_moles":
        return """Moles are usually harmless but require care. Follow these
guidelines:
        1. Sunscreen: Protect your skin with a broad-spectrum sunscreen to
prevent sun damage.
```

```
        2. Regularly examine your moles for any changes in size, shape, or
color.
        3. If you notice changes in a mole, consult a dermatologist
for a thorough evaluation.
        4. Avoid sun exposure, and consider wearing protective clothing and
hats.
        """
```

```
    else:
        return "Please enter a valid skin type."
```

```
def prediction(image_path):
```

```
    import tensorflow as tf
    model = tf.keras.models.load_model('skin/skin/model.h5')
    import numpy as np
    import  matplotlib.pyplot  as  plt
    import warnings
    warnings.filterwarnings(action='once')
    from tensorflow.keras.preprocessing import image
    def prepare(img_path):
        img = image.load_img(img_path, target_size=(224,224))
        x = image.img_to_array(img)
        x = x/255
        return np.expand_dims(x, axis=0)
```

```
    img_path = image_path
    predictions = model.predict([prepare(img_path)])
    skin_types =['Red_Spots_skin', 'Dry Skin', 'Normal Skin', 'Oily Skin',
'Scaly Skin', 'Sensitive Skin', 'Skin_moles']
    predicted_skin_type = skin_types[np.argmax(predictions)]
    # print(f'Predicted Skin Type: {predicted_skin_type}')
```

```
    # # Generate skincare recommendations based on the predicted skin type
    # recommendations = recommend_cosmetics(predicted_skin_type)
    # print('Skincare Recommendations:')
    # print(recommendations)
```

```
    return predicted_skin_type
```

```
app = Flask(__name__)
```

```
@app.route('/')
def home_page():
    return render_template('home.html')
```

```
@app.route('/contact')
def contact():
    return render_template('contact-us.html')
```

```
@app.route('/index')
def ai_engine_page():
```

```
    return render_template('index.html')
```

```
@app.route('/mobile-device')
def mobile_device_detected_page():
    return render_template('mobile-device.html')
```

```
@app.route('/submit222', methods=['POST'])
def submit222():
    text = request.form['textfield']
    with open('data.csv', 'a', newline='') as csvfile:
        fieldnames = ['text']

        writer = csv.DictWriter(csvfile, fieldnames=fieldnames)
        writer.writerow({'text': text})
    return 'Review submitted successfully!'
```

```
  @app.route('/submit', methods=['GET',
  'POST']) def submit():
      if request.method == 'POST':
          image = request.files['image']
          filename = image.filename
          file_path = os.path.join('static/uploads',
          filename) image.save(file_path)
          print(file_path)
          pred = prediction(file_path)
          # title = disease_info['disease_name'][pred]
          #                               description
          =disease_info['description'][pred]        #
          prevent       =       disease_info['Possible
          Steps'][pred]      #      image_url      =
          disease_info['image_url'][pred]
          # supplement_name = supplement_info['supplement name'][pred]
          #    supplement_image_url    =    supplement_info['supplement
          image'][pred]  #  supplement_buy_link  =  supplement_info['buy
          link'][pred]
          # return render_template('submit.html' , title = title , desc =
  description , prevent = prevent ,
          #                         image_url = image_url , pred =
  pred ,sname = supplement_name , simage = supplement_image_url , buy_link
  = supplement_buy_link)

          # # Generate skincare recommendations based on the predicted skin
          type recommendations = recommend_cosmetics(pred)
          print('Skincare Recommendations:')
          print(recommendations)
```

```
        return render_template('submit.html' , pred =
pred,recommendations=recommendations)
```

```
@app.route('/market', methods=['GET', 'POST'])
def market():
    return render_template('market.html', supplement_image =
list(supplement_info['supplement image']),
```

```
                            supplement_name =
                            list(supplement_info['supplement
name']),disease_name= list(supplement_info['disease_name']), buy =
```

```
if __name__== '__main__':
    app.run(debug=True)
```

6.2 :CNN.py

```
import pandas as pd
import torch.nn as nn

class CNN(nn.Module):
    def __init__(self, K):
        super(CNN, self).__init__()
        self.conv_layers =
            nn.Sequential( # conv1
            nn.Conv2d(in_channels=3, out_channels=32,
                      kernel_size=3, padding=1),
            nn.ReLU(),
            nn.BatchNorm2d(32),
            nn.Conv2d(in_channels=32, out_channels=32,
                      kernel_size=3, padding=1),
            nn.ReLU(),
            nn.BatchNorm2d(32),
            nn.MaxPool2d(2),
            # conv2
            nn.Conv2d(in_channels=32, out_channels=64,
                      kernel_size=3, padding=1),
            nn.ReLU(),
            nn.BatchNorm2d(64),
            nn.Conv2d(in_channels=64, out_channels=64,
                      kernel_size=3, padding=1),
            nn.ReLU(),
            nn.BatchNorm2d(64),
            nn.MaxPool2d(2),
            # conv3
            nn.Conv2d(in_channels=64, out_channels=128,
                      kernel_size=3, padding=1),
            nn.ReLU(),
            nn.BatchNorm2d(128),
            nn.Conv2d(in_channels=128, out_channels=128,
                      kernel_size=3, padding=1),
            nn.ReLU(),
            nn.BatchNorm2d(128),
            nn.MaxPool2d(2),
            # conv4
            nn.Conv2d(in_channels=128, out_channels=256,
                      kernel_size=3, padding=1),
            nn.ReLU(),
            nn.BatchNorm2d(256),
```

```
            nn.Conv2d(in_channels=256, out_channels=256,
                      kernel_size=3, padding=1),
            nn.ReLU(),
            nn.BatchNorm2d(256),
            nn.MaxPool2d(2),
        )
```

```
        self.dense_layers =
            nn.Sequential( nn.Dropout(0.4)
            , nn.Linear(50176, 1024),
            nn.ReLU(),
            nn.Dropout(0.4),
            nn.Linear(1024, K),
        )
```

```
    def forward(self, X):
        out = self.conv_layers(X)
```

```
        # Flatten
        out = out.view(-1, 50176)
```

```
        # Fully connected
        out = self.dense_layers(out)
```

```
        return out
```

```
idx_to_classes = {0: 'Apple___Apple_scab',
                  1: 'Apple___Black_rot',
                  2: 'Apple___Cedar_apple_rust',
                  3: 'Apple___healthy',
                  4: 'Background_without_leaves',
                  5: 'Blueberry___healthy',
                  6: 'Cherry___Powdery_mildew',
                  7: 'Cherry___healthy',
                  8: 'Corn___Cercospora_leaf_spot Gray_leaf_spot',
                  9: 'Corn___Common_rust',
                  10: 'Corn___Northern_Leaf_Blight',
                  11: 'Corn___healthy',
                  12: 'Grape___Black_rot',
                  13: 'Grape___Esca_(Black_Measles)',
                  14: 'Grape___Leaf_blight_(Isariopsis_Leaf_Spot)',
                  15: 'Grape___healthy',
                  16: 'Orange___Haunglongbing_(Citrus_greening)',
                  17: 'Peach___Bacterial_spot',
                  18: 'Peach___healthy',
                  19: 'Pepper,_bell___Bacterial_spot',
                  20: 'Pepper,_bell___healthy',
                  21: 'Potato___Early_blight',
                  22: 'Potato___Late_blight',
                  23: 'Potato___healthy',
                  24: 'Raspberry___healthy',
```

```
                25: 'Soybean___healthy',
                26: 'Squash___Powdery_mildew',
                27: 'Strawberry___Leaf_scorch',
                28: 'Strawberry___healthy',
                29: 'Tomato___Bacterial_spot',
                30: 'Tomato___Early_blight',
                31: 'Tomato___Late_blight',
                32: 'Tomato___Leaf_Mold',
                33: 'Tomato___Septoria_leaf_spot',
                34: 'Tomato___Spider_mites Two-spotted_spider_mite',
                35: 'Tomato___Target_Spot',
                36: 'Tomato___Tomato_Yellow_Leaf_Curl_Virus',
                37: 'Tomato___Tomato_mosaic_virus',
                38: 'Tomato___healthy'}
```

6.3 : Predictions.ipynb:

```
def
    recommend_cosmetics(skin_ty
    pe): if skin_type ==
    "Normal Skin":
        return """For normal skin, you're lucky to have a well-balanced
complexion. Your primary goal is to maintain your skin's health. Here's a
comprehensive skincare routine:
        1. Cleanser: Use a gentle, sulfate-free cleanser to remove
           impurities.
        2. Moisturizer: Opt for a lightweight, non-comedogenic moisturizer.
        3. Sunscreen: Apply a broad-spectrum sunscreen daily to protect your
skin.
        4. Optional: You can incorporate a mild exfoliant 1-2 times a week
           for
extra glow.
    """

    elif skin_type == "Sensitive Skin":
        return """Sensitive skin requires extra care to minimize irritation
and redness. Consider these steps:
        1. Cleanser: Use a fragrance-free, hypoallergenic cleanser.
        2. Moisturizer: Choose a product with soothing ingredients like
aloe vera or chamomile.
        3. Sunscreen: Use a physical sunscreen with zinc oxide or
titanium dioxide.
        4. Avoid harsh exfoliants and strong active ingredients, and
patch-test new products.
        """

  elif skin_type == "Dry Skin":
      return """Dry skin needs intense hydration and protection. Follow this
regimen:
      1. Cleanser: Use a hydrating, gentle cleanser.
      2. Moisturizer: Opt for a rich, creamy moisturizer with ingredients
like hyaluronic acid or ceramides.
      3. Sunscreen: Apply a broad-spectrum sunscreen daily to prevent further
dryness.
```

```
        4. Consider adding a hydrating serum or facial oil to your routine
for added moisture.
        """
```

```
    elif skin_type == "Oily Skin":
        return """To control excess oil and minimize breakouts, follow these
steps:
        1. Cleanser: Use a foaming, salicylic acid-based cleanser to control
oil.
        2. Moisturizer: Choose an oil-free, lightweight, and non-comedogenic
moisturizer.
        3. Sunscreen: Use an oil-free, mattifying sunscreen.
        4. Consider using products with ingredients like salicylic acid,
niacinamide, or witch hazel to manage oil and acne.
        """
```

```
    elif skin_type == "Scaly Skin":
        return """Scaly skin often results from dryness and flakiness. Try
these skincare steps:
        1. Cleanser: Use a gentle exfoliating cleanser to remove dead skin
cells.
        2. Moisturizer: Choose a rich, emollient moisturizer to lock in
moisture.
        3. Sunscreen: Protect your skin from further damage with a daily
sunscreen.
        4. Exfoliate with products containing glycolic acid or lactic acid to
improve texture.
        """
```

```
    elif skin_type == "Red_Spots_skin":
        return """Red spots can be due to various causes. Here's a general
approach:
        1. Cleanser: Use a gentle, fragrance-free cleanser to avoid irritation.
        2. Moisturizer: Select a calming and hydrating moisturizer.
        3. Sunscreen: Shield your skin from further damage with a broad-
spectrum sunscreen.
        4. Consult a dermatologist to identify the specific cause of redness
and receive tailored treatment.
        """
```

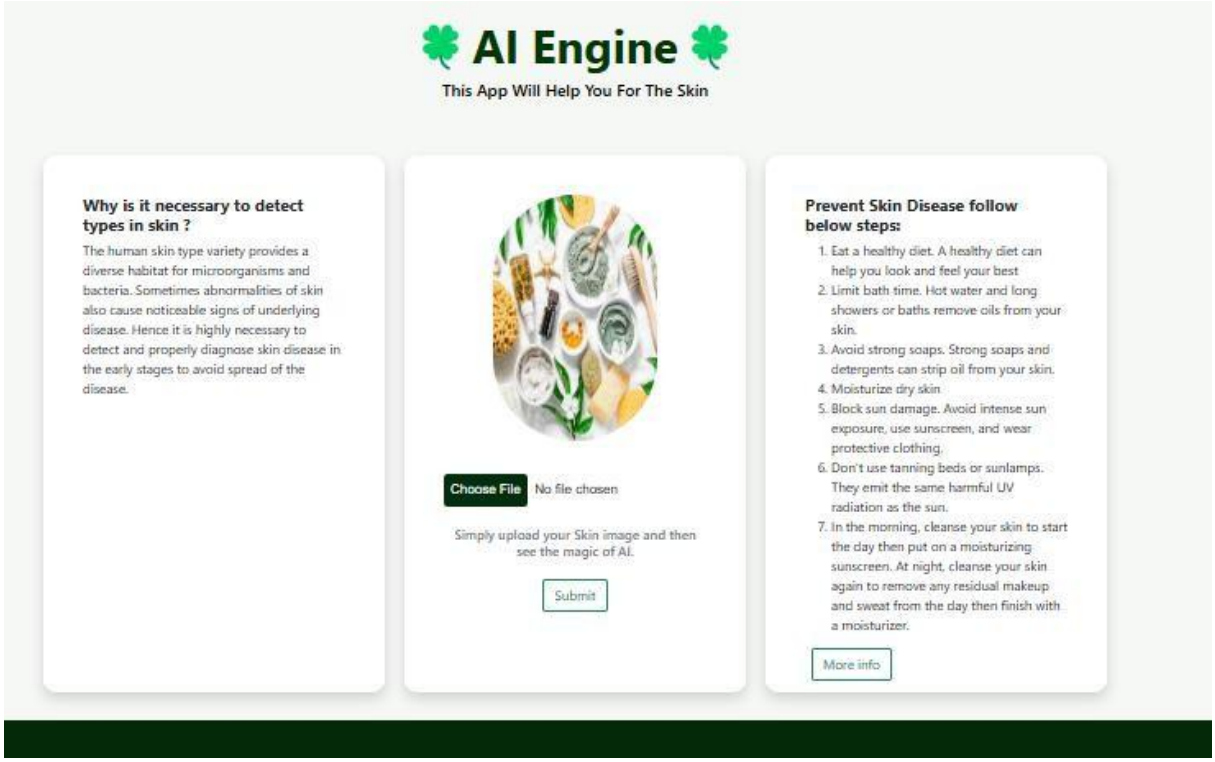
```
    elif skin_type == "Skin_moles":
        return """Moles are usually harmless but require care. Follow these
guidelines:
        1. Sunscreen: Protect your skin with a broad-spectrum sunscreen to
prevent sun damage.
        2. Regularly examine your moles for any changes in size, shape, or
color.
        3. If you notice changes in a mole, consult a dermatologist for a
thorough evaluation.
        4. Avoid sun exposure, and consider wearing protective clothing and
hats.
```

```
        """

    else:
        return "Please enter a valid skin type."
```

CHAPTER-7

OUTPUT SCREENS



Predicted Skin Type:

Normal Skin

SUGGESTION :

For normal skin, you're lucky to have a well-balanced complexion. Your primary goal is to maintain your skin's health. Here's a comprehensive skincare routine:

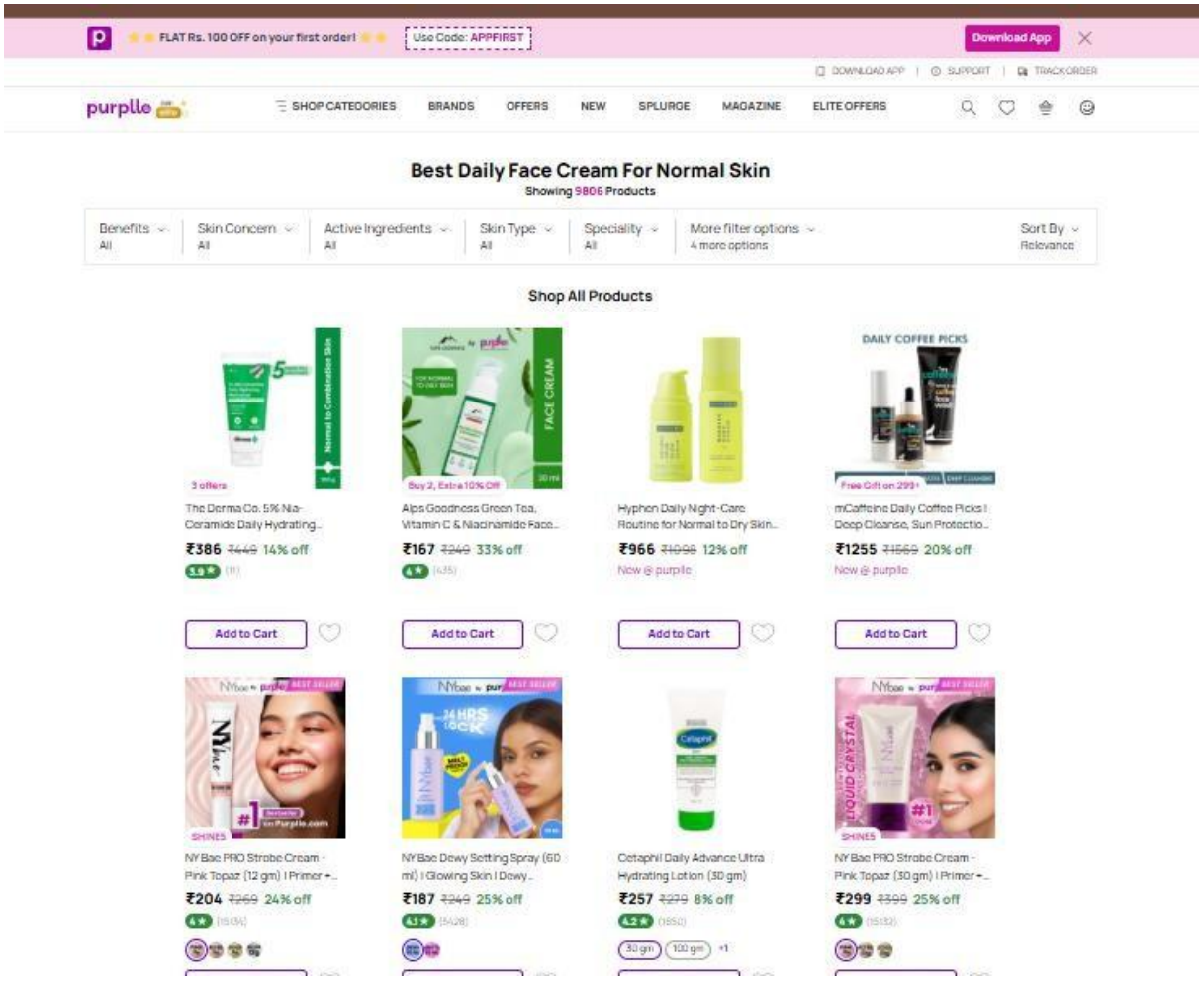
1. Cleanser: Use a gentle, sulfate-free cleanser to remove impurities.
2. Moisturizer: Opt for a lightweight, non-comedogenic moisturizer.
3. Sunscreen: Apply a broad-spectrum sunscreen daily to protect your skin.
4. Optional: You can incorporate a mild exfoliant 1-2 times a week for extra glow.

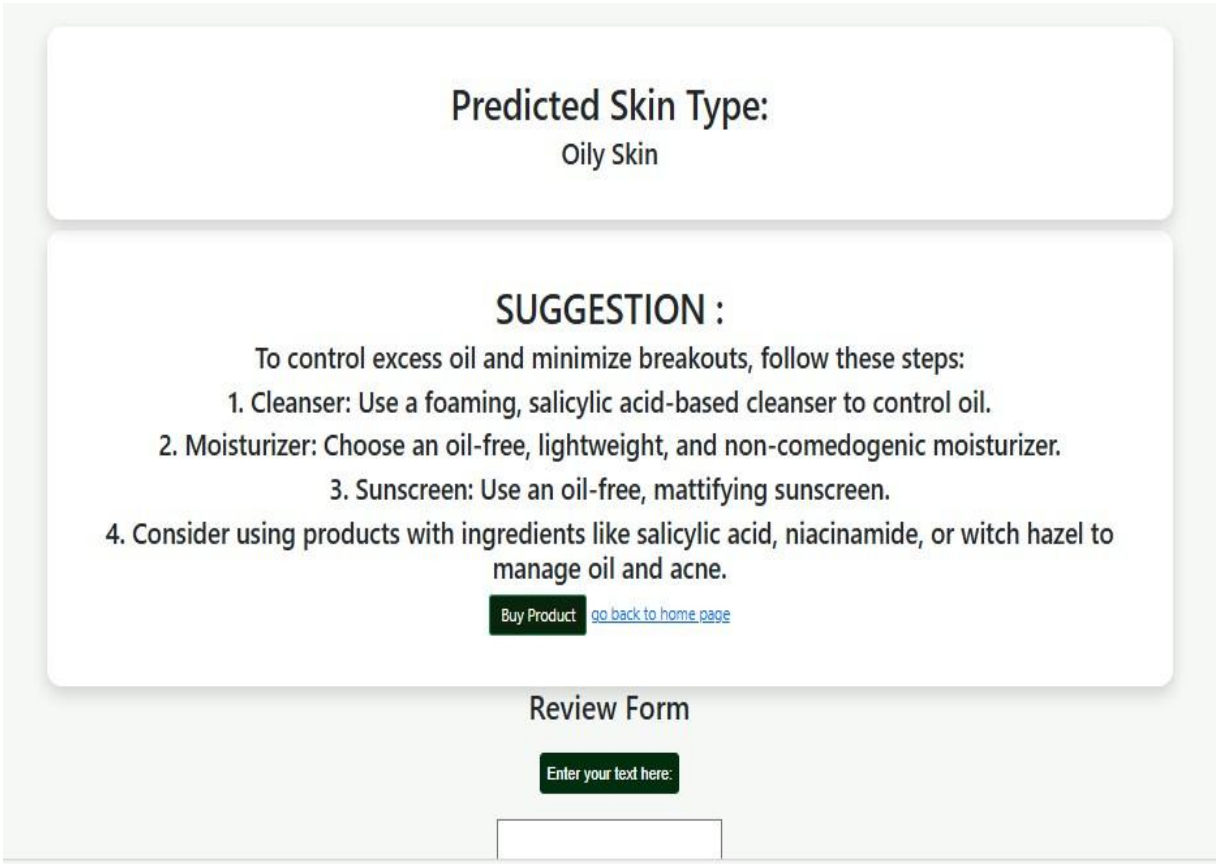
Buy Product online go back to home page

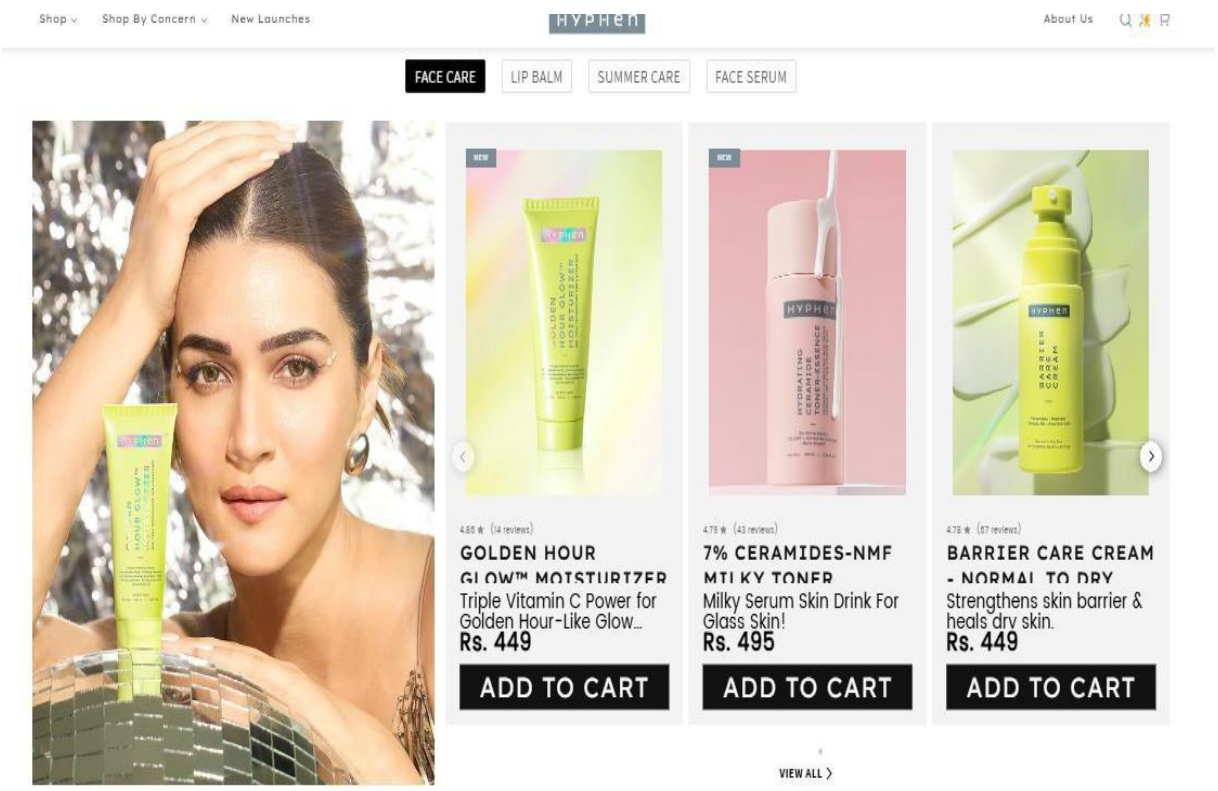
Review Form

Enter your text here:

Submit




Predicted Skin Type:
Oily Skin
SUGGESTION :
To control excess oil and minimize breakouts, follow these steps:
1. Cleanser: Use a foaming, salicylic acid-based cleanser to control oil.
2. Moisturizer: Choose an oil-free, lightweight, and non-comedogenic moisturizer.
3. Sunscreen: Use an oil-free, mattifying sunscreen.
4. Consider using products with ingredients like salicylic acid, niacinamide, or witch hazel to manage oil and acne.
Buy Product
go back to home page
Review Form
Enter your text here:


Shop
Shop By Concern
New Launches
HYPHEN
About Us
FACE CARE
LIP BALM
SUMMER CARE
FACE SERUM
GOLDEN HOUR GLOW™ MOISTURIZER
Triple Vitamin C Power for Golden Hour-Like Glow...
Rs. 449
ADD TO CART
7% CERAMIDES-NMF MILKY TONER
Milky Serum Skin Drink For Glass Skin!
Rs. 495
ADD TO CART
BARRIER CARE CREAM - NORMAL TO DRY
Strengthens skin barrier & heals dry skin.
Rs. 449
ADD TO CART
VIEW ALL

Predicted Skin Type:

Dry Skin

SUGGESTION :

Dry skin needs intense hydration and protection. Follow this regimen:

1. Cleanser: Use a hydrating, gentle cleanser.
2. Moisturizer: Opt for a rich, creamy moisturizer with ingredients like hyaluronic acid or ceramides.
3. Sunscreen: Apply a broad-spectrum sunscreen daily to prevent further dryness.
4. Consider adding a hydrating serum or facial oil to your routine for added moisture.

Buy Product go back to home page

Review Form

Enter your text here:

DAILY DRY-SKIN CARE BUNDLE: FOR DRY TO NORMAL SKIN

A routine that heals skin dryness without clogging pores.

(16 reviews)

Rs. 1,469 ~~Rs. 1,547~~ 5% OFF

(Inclusive of all taxes)

ADD TO CART — 1 +

WHAT IT IS:

A hydrating yet lightweight skincare bundle that gives real results, especially curated for normal to dry skin

Read More...

WHAT IT DOES:

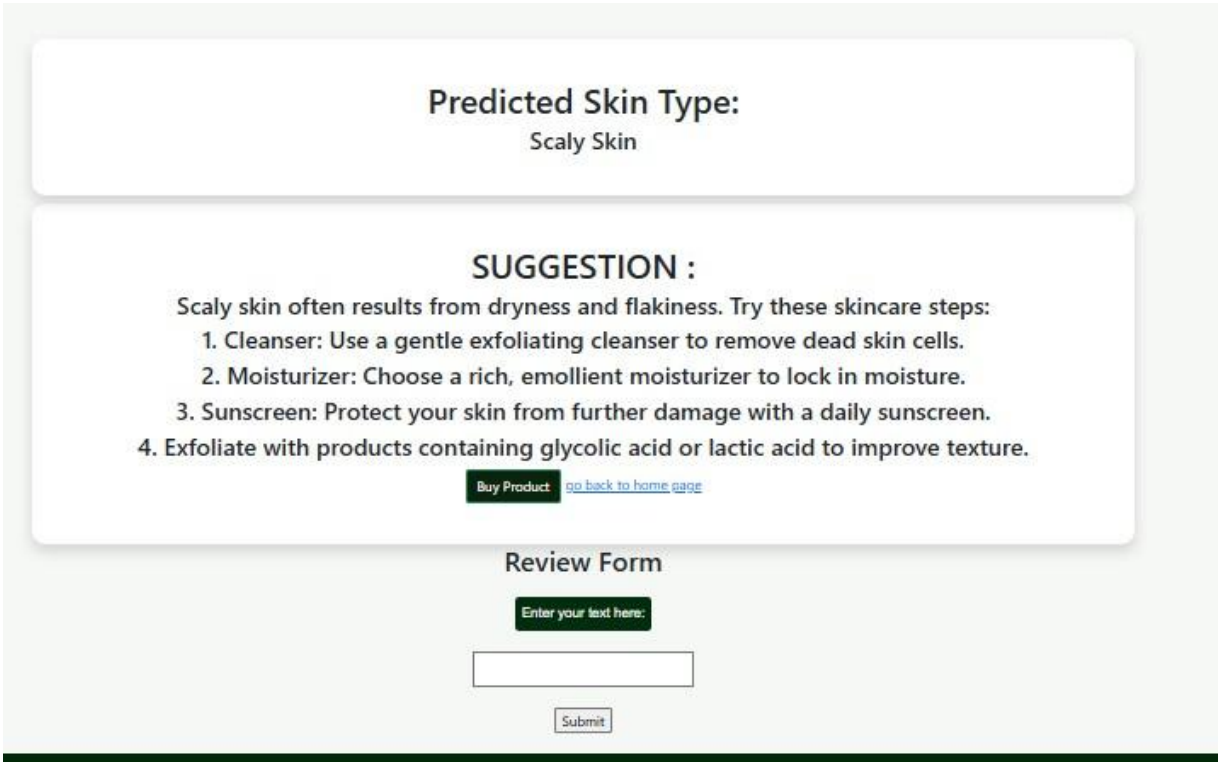
GLOW UP - MOISTURIZE - PROTECT

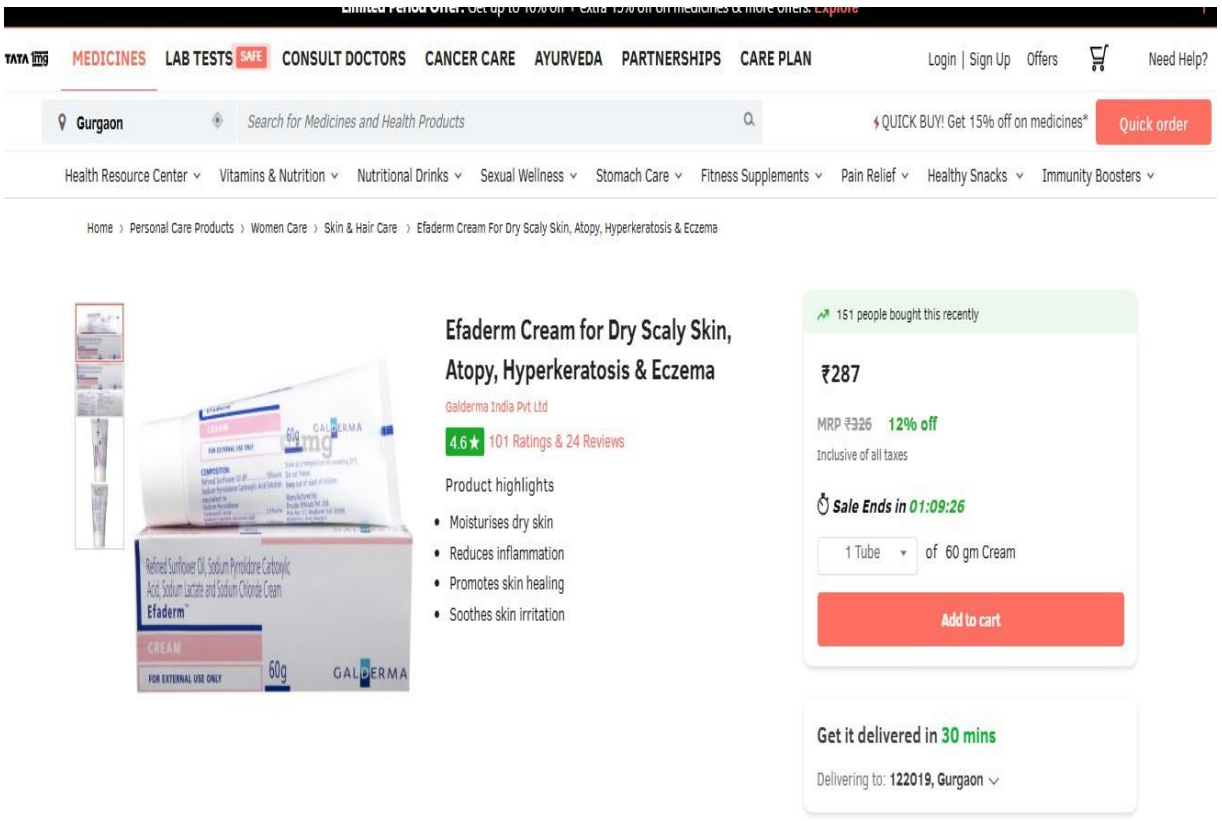
Read More...

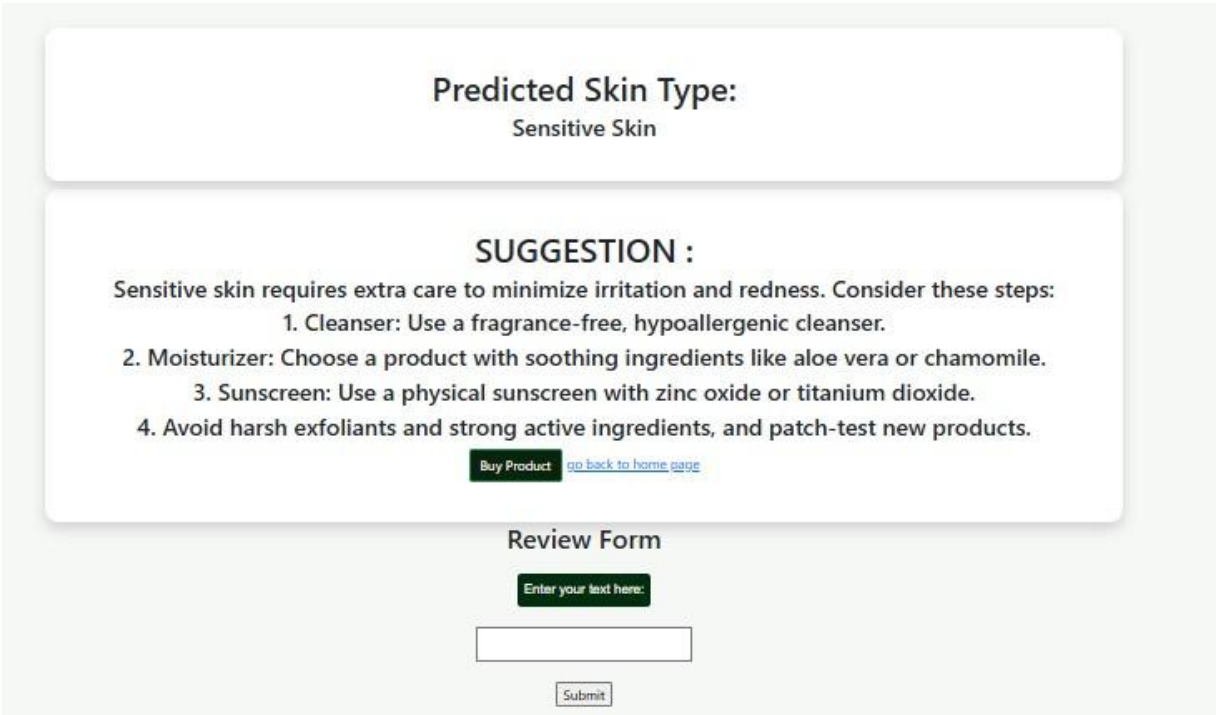
HOW IT DOES:

We've hyphened natural and active ingredients like Tamanu & Chia Seed Oil, Cica Extracts etc. that nourish ski

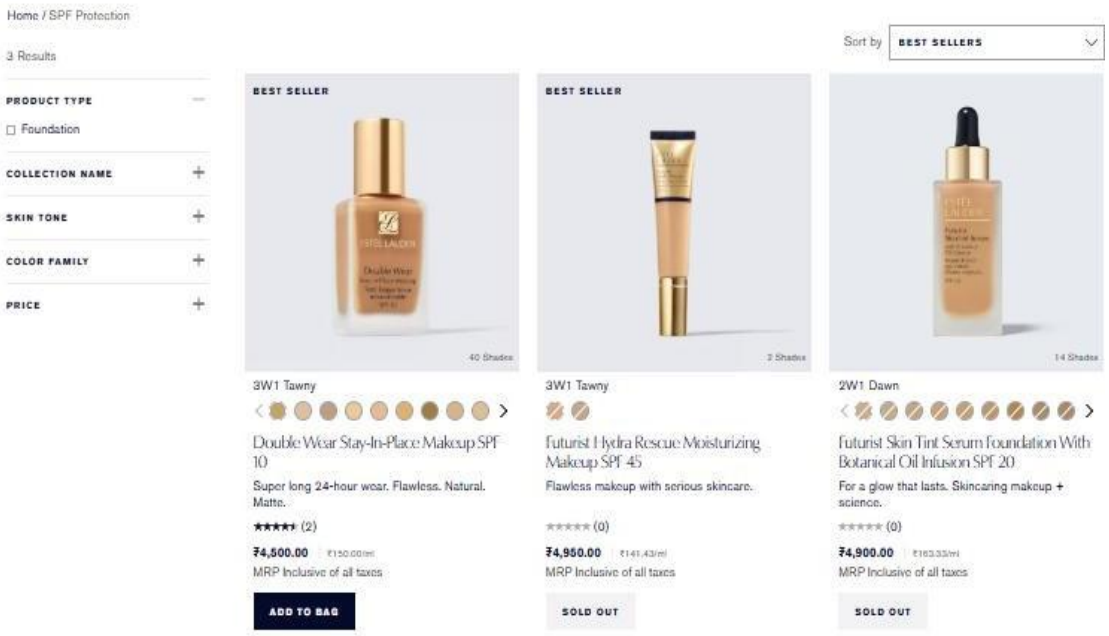
Read More...

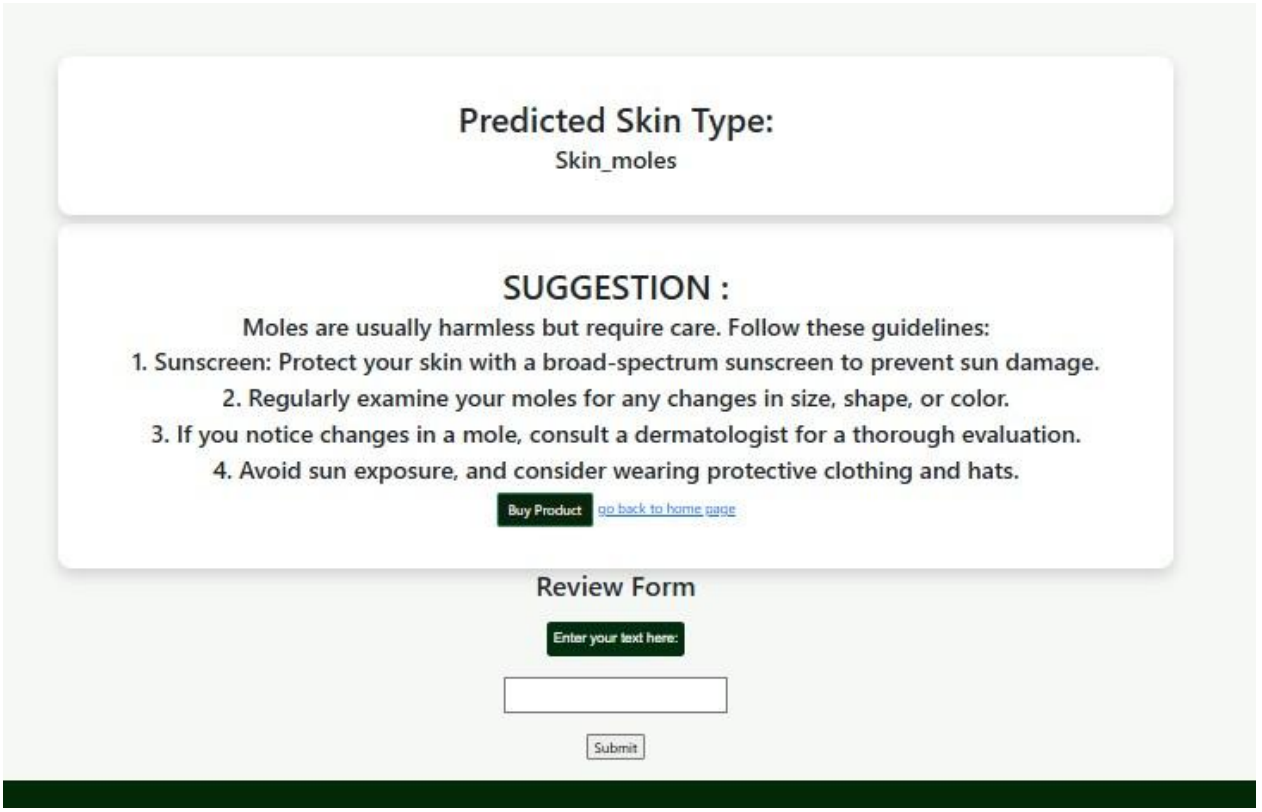

Predicted Skin Type:
Scaly Skin
SUGGESTION :
Scaly skin often results from dryness and flakiness. Try these skincare steps:
1. Cleanser: Use a gentle exfoliating cleanser to remove dead skin cells.
2. Moisturizer: Choose a rich, emollient moisturizer to lock in moisture.
3. Sunscreen: Protect your skin from further damage with a daily sunscreen.
4. Exfoliate with products containing glycolic acid or lactic acid to improve texture.
Buy Product
go back to home page
Review Form
Enter your text here:
Submit

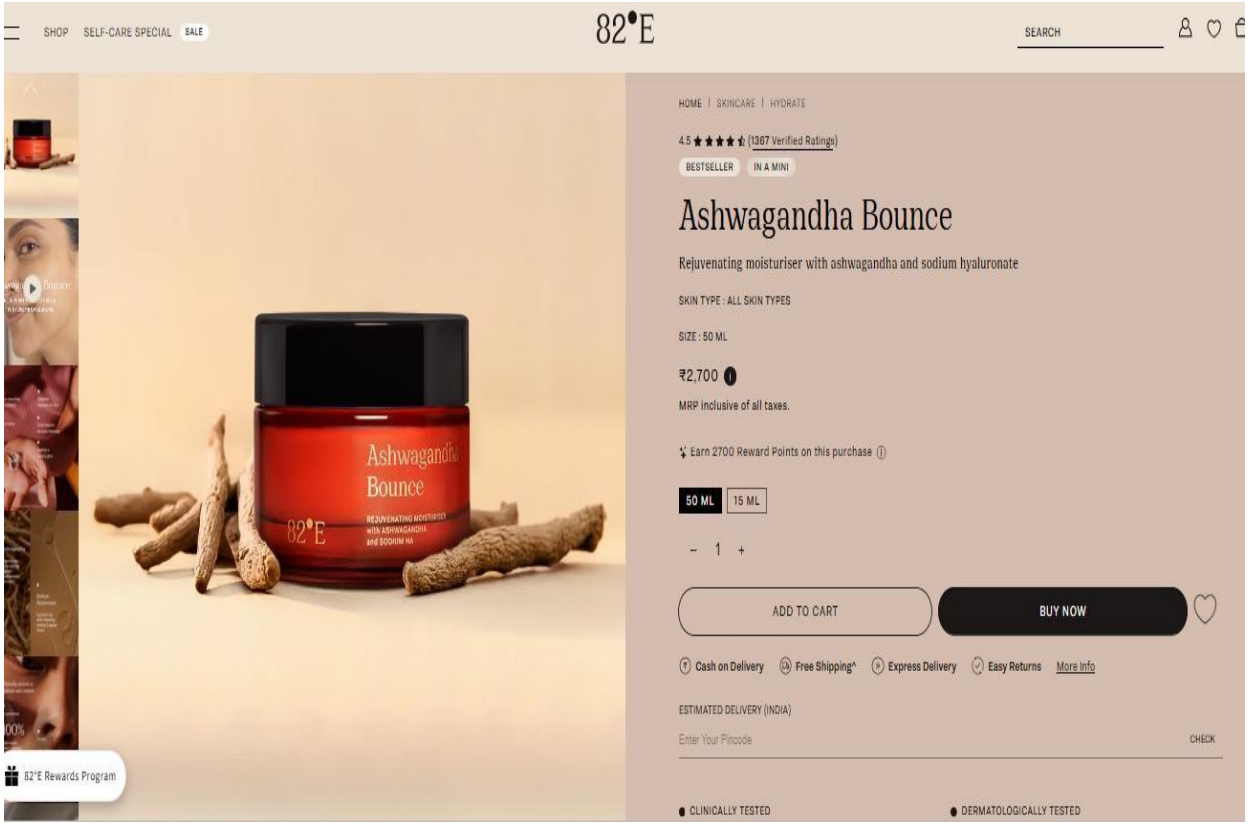

MEDICINES
LAB TESTS
CONSULT DOCTORS
CANCER CARE
AYURVEDA
PARTNERSHIPS
CARE PLAN
Login | Sign Up
Offers
Need Help?
Gurgaon
Search for Medicines and Health Products
QUICK BUY! Get 15% off on medicines*
Quick order
Health Resource Center
Vitamins & Nutrition
Nutritional Drinks
Sexual Wellness
Stomach Care
Fitness Supplements
Pain Relief
Healthy Snacks
Immunity Boosters
Home › Personal Care Products › Women Care › Skin & Hair Care › Efaderm Cream For Dry Scaly Skin, Atopy, Hyperkeratosis & Eczema
Efaderm Cream for Dry Scaly Skin, Atopy, Hyperkeratosis & Eczema
Galderma India Pvt Ltd
4.6★ 101 Ratings & 24 Reviews
Product highlights
Moisturises dry skin
Reduces inflammation
Promotes skin healing
Soothes skin irritation
151 people bought this recently
₹287
MRP ₹326 12% off
Inclusive of all taxes
Sale Ends in 01:09:26
1 Tube
of 60 gm Cream
Add to cart
Get it delivered in 30 mins
Delivering to: 122019, Gurgaon


Predicted Skin Type:
Sensitive Skin
SUGGESTION :
Sensitive skin requires extra care to minimize irritation and redness. Consider these steps:
1. Cleanser: Use a fragrance-free, hypoallergenic cleanser.
2. Moisturizer: Choose a product with soothing ingredients like aloe vera or chamomile.
3. Sunscreen: Use a physical sunscreen with zinc oxide or titanium dioxide.
4. Avoid harsh exfoliants and strong active ingredients, and patch-test new products.
Buy Product
go back to home page
Review Form
Enter your text here:
Submit


Sunscreen
Essentials


Home / SPF Protection
3 Results
PRODUCT TYPE
Foundation
COLLECTION NAME
SKIN TONE
COLOR FAMILY
PRICE
Sort by
BEST SELLERS
BEST SELLER
3W1 Tawny
Double Wear Stay-In-Place Makeup SPF 10
Super long 24-hour wear. Flawless. Natural. Matte.
(2)
₹4,500.00
MRP Inclusive of all taxes
ADD TO BAG
BEST SELLER
3W1 Tawny
Futurist Hydra Rescue Moisturizing Makeup SPF 45
Flawless makeup with serious skincare.
(0)
₹4,950.00
MRP Inclusive of all taxes
SOLD OUT
2W1 Dawn
Futurist Skin Tint Serum Foundation With Botanical Oil Infusion SPF 20
For a glow that lasts. Skincaring makeup + science.
(0)
₹4,900.00
MRP Inclusive of all taxes
SOLD OUT


Predicted Skin Type:
Skin_moles
SUGGESTION :
Moles are usually harmless but require care. Follow these guidelines:
1. Sunscreen: Protect your skin with a broad-spectrum sunscreen to prevent sun damage.
2. Regularly examine your moles for any changes in size, shape, or color.
3. If you notice changes in a mole, consult a dermatologist for a thorough evaluation.
4. Avoid sun exposure, and consider wearing protective clothing and hats.
Buy Product
go back to home page
Review Form
Enter your text here:
Submit


SHOP
SELF-CARE SPECIAL
SALE
82°E
SEARCH
HOME | SKINCARE | HYDRATE
4.5 (1367 Verified Ratings)
BESTSELLER
IN A MINI
Ashwagandha Bounce
Rejuvenating moisturiser with ashwagandha and sodium hyaluronate
SKIN TYPE : ALL SKIN TYPES
SIZE : 50 ML
₹2,700
MRP inclusive of all taxes.
Earn 2700 Reward Points on this purchase
50 ML
15 ML
- 1 +
ADD TO CART
BUY NOW
Cash on Delivery
Free Shipping^
Express Delivery
Easy Returns
More Info
ESTIMATED DELIVERY (INDIA)
Enter Your Pincode
CHECK
CLINICALLY TESTED
DERMATOLOGICALLY TESTED
82°E Rewards Program

CHAPTER -8

TESTING AND CONCLUSION

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

TYPES OF TESTS

Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals. Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output: Identified classes of application outputs must be exercised.

Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined. System Test System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. Itis used to test areas that cannot be reached from a black box level.

Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works. Unit Testing Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results:

All the test cases mentioned above passed successfully. No defects encountered.

Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results:

All the test cases mentioned above passed successfully. No defects encountered.

ADVANTAGES AND APPLICATIONS

- Personalized Recommendations.
- product recommendations.
- Improved User Experience.
- Better Product Selection.
- Facial Health Monitoring.

8.1 Conclusion

Nowadays, consumers have to choose from a greater variety of options across several brands and product categories, which makes decision-making more challenging than it was in the past. This approach gives the composition based on several Facial Beauty Attractivenesss (i.e. dry, oily or natural). Our main objective is to improve the makeup product composition. Using the suggested technique will surely lead to a superior makeup product formulation.Some areas where this system can be improved are adding more templates and colors to the makeup synthesis library to recommend more styles, expanding the dataset, expanding the system to recommend for both men and women, and learning the knowledge base rules directly from the labeled data.

The Facial Skincare Recommendation System represents a comprehensive and intelligent solution for personalized skincare guidance. Leveraging Convolutional Neural Network (CNN) models, including Efficient Net B0, the system excels in facial skincare recommendations system and skin type classification. By extracting features from diverse facial images, it provides accurate insights into individual skincare needs, allowing for the formulation of tailored product recommendations with accuracy of 92.34%. The system's success lies in its ability to handle various scenarios, including different skin types, lighting conditions and the presence of makeup or accessories. The inclusion of test cases ensures the robustness and reliability of the model across diverse real-world situations. It emerges as a valuable tool for users seeking personalized and effective skincare routines.

The Facial Skincare Recommendation System offers promising avenues for future development and enhancement. Some potential directions for future research and expansion include Implementation of features that allow users to provide real-time feedback on recommended products, creating a dynamic system that adapts to individual preferences over time and exploration of possibilities for integrating the system with wearable devices that continuously monitor skin health metrics, providing a continuous and proactive skincare advisory service.

Our main objective is to improve the makeup product composition. Using the suggested technique will surely lead to a superior makeup product formulation. Some areas where this system can be improved are adding more templates and colors to the makeup synthesis library to recommend more styles, expanding the dataset, expanding the system to recommend for both men and women, and learning the knowledge base rules directly from the labeled data.

8.2 Future Scope

This project demonstrates the powerful intersection of technology and self-care. Its ability to blend deep learning with practical, real-world applications makes it not just innovative, but also socially impactful

1. Enhanced Facial Feature Recognition and Analysis

In the future, the deep learning model (currently using CNN and Xception) could be further refined to analyze a wider range of facial features, including finer details like wrinkles, pores, and pigmentation variations. This would lead to more precise skincare recommendations, tailored not only to skin type but also to specific conditions like acne scars, fine lines, or sun spots. Leveraging advanced AI techniques, the app could continuously improve by learning from users' feedback and updating its analysis capabilities.

2. Real-Time Skin Analysis & Feedback

Incorporating real-time image processing into the app could allow users to take live photos of their face and receive immediate skincare recommendations. This feature would be especially useful for daily skincare routines or before/after treatment comparisons. Over time, the app could also track changes in the user’s skin and adjust product suggestions accordingly, offering dynamic updates as users' skin conditions evolve.

REFERENCE

[1]. Hsia, C.-H.; Li, H.-T. Real-time wavelet face detection system with occlusion condition. J. Technol. 2015, 30, 343–350.

[2]. The Power of Artificial Intelligence for Cosmetics Brands. Available online: https://www.launchmetrics.com/resources/blog/artificial-intelligence-beauty-industry (accessed on 20 October 2021).

[3]. Junayed, M.S.; Jeny, A.A.; Atik, S.T.; Neehal, N.; Karim, A.; Azam, S.; Shanmugam, B. AcneNet—A deep CNN based classification approach for acne classes. In Proceedings of the 2019 12th International Conference on Information & Communication Technology and System (ICTS), Surabaya, Indonesia, 18 July 2019; pp. 203–208.

[4]. Vesal, S.; Ravikumar, N.; Maier, A. Facial Net: A deep learning framework for Facial lesion segmentation. In Proceedings of the 2018 IEEE Nuclear Science Symposium and Medical Imaging Conference Proceedings (NSS/MIC), Sydney, NSW, Australia, 10–17 November 2018; pp. 1–3.

[5]. Ronneberger, O.; Philipp, F.; Thomas, B. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention; Medical Image Computing and Computer-Assisted Intervention-MICCAI, Munich, Germany, 5–9 October 2015; pp. 234–241.

[6]. Hameed, N.; Shabut, A.M.; Hossain, M.A. Multi-class Facial diseases classification using deep convolutional neural network and support vector machine. In Proceedings of the 2018 12th International Conference on Software, Knowledge, Information Management & Applications (SKIMA), Phnom Penh, Cambodia, 3–5 December 2018; pp. 1–7.

[7]. Cortes, C.; Vapnik, V. Support-vector networks. Mach. Learn. 1995, 20, 273–297. [CrossRef] [8]. Goyal, M.; Oakley, A.; Bansal, P.; Dancey, D.; Yap, M.H. Facial lesion segmentation in

dermoscopic images with ensemble deep learning methods. IEEE Access. 2020, 8, 4171– 4181. [CrossRef]

[9]. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580– 587.

[10]. Adegun, A.A.; Viriri, S. Deep learning-based system for automatic melanoma

detection.

IEEE Access. 2020, 8, 7160–7172. [CrossRef]

[11]. Minaee, S.; Abdolrashidi, A.; Su, H.; Bennamoun, M.; Zhang, D. Biometrics recognition using deep learning: A Survey. arXiv 2021, arXiv:1912.00271.

[12]. Hsia, C.-H.; Chiang, J.-S.; Lin, C.-Y. A fast face detection method for illumination variant condition. Scientia Iranica 2015, 22, 2081–2091. 1

[13]. Hsia, C.-H.; Li, H.-T. Real-time wavelet face detection system with occlusion condition. J. Technol. 2015, 30, 343–350.

[14]. Prasad, P.S.; Devi, B.S.; Reddy, M.J.; Gunjan, V.K. A survey of fingerprint recognition systems and their applications. In Proceedings of the International Conference on Communications and Cyber Physical Engineering, Hyderabad, India, 24–25 January 2018;

pp. 513–520.

[15]. Guo, J.-M.; Hsia, C.-H.; Liu, Y.-F.; Yu, J.-C.; Chu, M.-H.; Le, T.-N. Contact-free hand geometry-based identification system. Exp. Syst. App. 2012, 39, 11728–11736. [CrossRef]

[16]. Trokielewicz, M.; Czajka, A.; Maciejewicz, P. Iris recognition after death. IEEE Trans. Inf.

Forensics Secur. 2018, 14, 1501–1514. [CrossRef]

[17]. Hsia, C.-H. New verification strategy for finger-vein recognition system. IEEE Sens. J. 2017, 18, 790–797. [CrossRef]

[18]. Gnavi, S.; Penna, B.; Grangetto, M.; Magli, E.; Olmo, G. Wavelet kernels on a DSP: A comparison between lifting and filter banks for image coding. EURASIP J. Adv. Signal Process. 2002, 2002, 981–989. [CrossRef]

[19]. Liu, Y.; Guo, J.; Lai, B.; Lee, J. High efficient contrast enhancement using parametric approximation. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013; pp. 2444–2448. 20. Wang, K.; Liu, J.; Oluwatoyin, P.P.; Feng, W. Finger vein identification based on 2-D Gabor filter. In Proceedings of the 2010 the 2nd International Conference on Industrial Mechatronics and Automation, Wuhan, China, 30–31 May 2010; pp. 10–13.

[21]. Kocakulak, M.; Acir, N. Enhancement of finger vein images using Gabor filter. In Proceedings of the Signal Processing and Communications Applications Conference, Izmir, Turkey, 2–5 May 2018; pp. 1–4.

[22]. Hsia, C.-H. Improved finger-vein pattern method using wavelet-based for real-time personal identification system. J. Imaging Sci. Technol. 2018, 62, 30402-1–30402-8.

[CrossRef]

[23]. Hsia, C.-H. A new VLSI architecture for symmetric mask-based discrete wavelet transform. J. Internet Technol. 2014, 15, 1083–1090.

[24]. Bracewell, R.N.; Bracewell, R.N. The Fourier Transform and Its Applications; McGraw-Hill: New York, NY, USA, 1986; Volume 31999.

[25]. Daugman, J.G. High confidence visual recognition of persons by a test of statistical independence. IEEE Trans. Pattern Anal. Mach. Intell. 1993, 15, 1148–1161. [CrossRef]

[26]. Otsu, N. A threshold selection method from gray-level histograms. IEEE Trans. Sys. ManCybern. 1979, 9, 62–66. [CrossRef]

[27]. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.

[28]. Zhou, E.; Fan, H.; Cao, Z.; Jiang, Y.; Yin, Q. Extensive facial landmark localization with coarse-to-fine convolutional network cascade. In Proceedings of the 2013 IEEE International Conference on Computer Vision Workshops, Sydney, NSW, Australia, 2–8 December 2013;

pp. 386–391.

[29]. Stankovi´c, R.S.; Falkowski, B.J. The Haar wavelet transform: Its status and achievements.

Comput. Electr. Eng. 2003, 29, 25–44. [CrossRef]

[30]. Hsia, C.-H.; Chiang, J.-S. Real-time moving objects detection and tracking with direct LL- mask band scheme. Int. J. Innov. Comput. Inf. Control. 2012, 8, 4451–4468.

[31]. Hsia, C.-H.; Guo, J.-M.; Chiang, J.-S. Improved low-complexity algorithm for 2-D integer lifting-based discrete wavelet transform using symmetric mask-based scheme. IEEE Trans. Circuits Syst. Video Technol. 2009, 19, 1201–1208.

[32]. Lee, J.; Chen, Y.; Liu, L.-C.; Huang, C.-H. A facial-Facial condition classification system in wavelet domain. In Proceedings of the IECON 2007–33rd Annual Conference of the IEEE Industrial Electronics Society, Taipei, Taiwan, 5–8 November 2007; pp. 2419–2422.

[33]. Chen, L.-C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 833–851.

[34]. Piciucco, E.; Kuzu, R.S.; Maiorana, E.; Campisi, P. On the cross-finger similarity of vein patterns. In Proceedings of the International Conference on Image Analysis and

Processing, Trento, Italy, 9–13 September 2019; pp. 12–20.

[35]. Miura, N.; Nagasaka, A.; Miyatake, T. Feature extraction of finger-vein patterns based on repeated line tracking and its application to personal identification. Mach. Vis. Appl. 2004, 15, 194–203. [CrossRef]

[36]. Lee, E.C.; Jung, H.; Kim, D. New finger biometric method using near infrared imaging.

Sensors 2011, 11, 2319–2333. [CrossRef]

[37]. Liu, Z.; Song, S. An embedded real-time finger-vein recognition system for mobile devices.

IEEE Trans. Consum. Electron. 2012, 58, 522–527. [CrossRef]

[38]. Wang, D.; Li, J.; Memik, G. User identification based on finger-vein patterns for consumer electronics devices. IEEE Trans. Consum. Electron. 2010, 56, 799–804. [CrossRef]

APPENDIX

Appendix-1: System Architecture Overview

Frontend:

Built using HTML, CSS, and JavaScript for a responsive, user-friendly interface. Users can upload facial images and view personalized recommendations.

Backend:

Developed with Python, utilizing Flask and Django frameworks for robust server- side processing.

Handles image processing, model inference, and database interactions.

Database:

MySQL is used for storing user data, skin type classifications, beauty scores, and recommended products.

Machine Learning Models:

Convolutional Neural Networks (CNNs) for feature extraction.

Xception Model for high-accuracy facial analysis and beauty prediction. **E-commerce Integration:**

Embedded marketplace for users to browse and purchase recommended skincare products.

Appendix-2: Dataset Details

Sources:

Public facial datasets like CelebA and UTKFace for diverse skin tones and facial features.

Custom datasets for skin conditions (oily, dry, combination, normal).

Preprocessing:

Image resizing (224×224 pixels), normalization, and data augmentation (rotation, flipping, brightness adjustments).

Skin type annotations based on dermatological standards**.**

Appendix-3 Deep Learning Model Architecture

Facial Feature Analysis:

Base Model: Xception (for efficient facial feature extraction) Architecture:

Convolutional layers for extracting facial features

Global Average Pooling for dimensionality reduction

Fully Connected Layers for classification (attractiveness score and skin type)

Skin Type Classification:

Output Layer: Softmax activation for multi-class classification (oily, dry, combination, normal)

Loss Function: Categorical Cross-Entropy

Attractiveness Prediction:

Output Layer: Sigmoid activation for binary classification or regression for attractiveness scoring

Loss Function: Binary Cross-Entropy or Mean Squared Error (MSE)

Appendix-4: Implementation.

Appendix-5: Code Snippet.

Appendix-6: Evaluation Metrics.