SKIN-CANCER DETECTION SYSTEM

Submitted in partial fulfilment of the requirements of the degree

BACHELOR OF ENGINEERING IN COMPUTER ENGINEERING

By

Suyog Bhoite 221041056

Aryan Chogale 221041037

Kaivalya Datar 221041002

Shubham Jadhav 221041

Guide

Prof. R. C. SURYAWANSHI

Department of Computer Engineering
A.C. Patil College of Engineering
Kharghar, Navi Mumbai
University of Mumbai

(AY 2025-26)

CERTIFICATE

This is to certify that the Mini Project entitled **“SKIN-Cancer Detection App**” is a bonafide work of **Suyog, Aryan, Kaivalya,Shubham** submitted to the University of Mumbai in partial fulfilment of the requirement for the award of the degree of “**Bachelor of Engineering**” in “**Computer Engineering**”.

Prof. R. C. Suryawanshi
(Project Guide)

Dr. M. M. Deshpande
(Head of Department)

Dr. V. N. Pawar
(Principal)

MINI PROJECT APPROVAL

This Mini Project entitled **“SKIN-CANCER DETECTION APP’’** Suyog Bhoite (03), Aryan Chogale (04), Kaivalya Datar (07),Shubham Jadhav(19) is approved for the degree of **Bachelor of Engineering in Computer Engineering.**

Examiners

1………………………………………
(Internal Examiner Name & Sign)

2………………………………………
(External Examiner name & Sign)

Date:

Place:

Contents

ABSTRACT

The Skin Cancer Detection App leverages Artificial Intelligence and Deep Learning to classify dermoscopic images of skin lesions into multiple categories such as Melanoma, Basal Cell Carcinoma, Benign Keratosis-like Lesions, and Nevi. The project uses a Convolutional Neural Network (CNN) trained on the HAM10000 dataset to identify early signs of skin cancer. The system integrates a Flask-based model API, a Node.js backend, and a React-based frontend to create a seamless web interface. The application enables users to upload a skin lesion image and instantly receive classification results with confidence scores. This project demonstrates the potential of AI-assisted healthcare solutions to support dermatologists in early diagnosis and reduce manual screening time.

ACKNOWLEDGEMENTS

If words are considered as a symbol of approval and token of appreciation then let the words play the heralding role expressing our gratitude. The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of people whose ceaseless cooperation made it possible, whose constant guidance and encouragement crowns all effort with success. We are grateful to our guide **Prof. R. C. Suryawanshi** for the guidance, inspiration and constructive suggestions that were helpful to us in the preparation of this project.

1.INTRODUCTION

1.1 Introduction

- Skin cancer is one of the most common types of cancer worldwide. Early detection plays a crucial role in effective treatment and recovery. However, diagnosis often requires expert dermatologists and specialized equipment. With advancements in Machine Learning and Artificial Intelligence, image-based diagnostic systems have emerged as a supportive tool for early identification of skin lesions. This project focuses on building a deep learning–based system that classifies skin lesion images and provides accurate predictions to aid medical professionals and patients.

1.2 Motivation

- The rising incidence of skin cancer, especially melanoma, has created a pressing need for early screening tools that are both reliable and affordable. Manual diagnosis is time-consuming, subjective, and heavily dependent on expert availability. This motivated us to develop an automated detection system that can provide quick and accurate classification results using AI techniques. Such systems can be integrated into mobile or web applications to reach wider populations and assist in preliminary screenings.

1.3 Problem Statement and Objectives

To design and develop a deep learning–based application that detects and classifies different types of skin lesions using dermoscopic images, enabling early intervention and diagnosis of skin cancer.

Objective:

1. To collect and preprocess the HAM10000 dataset for model training..
2. To design and implement a CNN-based classification model.
3. To integrate the trained model with a Flask API and a full-stack web interface.
4. To evaluate model performance using metrics like accuracy, precision, recall, and F1-score.
5. To provide a user-friendly interface for image upload and prediction visualization..

1.4 Organization of the Report

The report is structured as follows:

- **Chapter 1 introduces the background, motivation, and objectives of the project.**
- **Chapter 2 reviews related work and identifies the research gap.**
- **Chapter 3 details the system architecture, algorithms, implementation, results, and conclusion.**

2.LITERATURE SURVEY

2.1 Survey of existing system

- Existing systems for skin cancer detection rely on traditional image processing and manual feature extraction methods. Early models used color, texture, and edge detection techniques followed by classifiers such as SVM or Random Forest. These systems required domain expertise and were often limited by feature selection quality. Modern approaches utilize deep learning, especially CNNs, for automated feature extraction and classification, significantly improving accuracy.

2.2 Limitation of existing system

Despite their success, existing systems have several limitations:

- Manual systems lack consistency and require dermatologist expertise.
- Traditional machine learning models fail to generalize across diverse datasets.
- Many existing apps lack real-time feedback and accessibility for general users
- There is limited integration between deep learning models and user-friendly web platforms.
- The research gap addressed by this project is the development of a **complete, deployable system** — combining a CNN model with a responsive web interface for real-time predictions.

2.3 Mini Project Contribution

❑ **Design & Development**

❑ • Built and trained a CNN model using the HAM10000 dataset.

❑ • Developed a Flask API to handle model inference.

❑ • Implemented a Node.js backend and React frontend for seamless user interaction.

❑ • Deployed the application using Docker for scalability.

❑ • Achieved a model accuracy of ~75.94% with efficient inference time.

3. PROPOSED SYSTEM

3.1 Introduction:

The proposed system automates the process of skin lesion classification. It takes an image as input, processes it through a trained CNN model, and returns a predicted class label along with confidence percentages. The system also provides a simple and intuitive web interface accessible to users and professionals alike.

3.2 Architecture/Framework

The system's architecture is designed to be modular and scalable. The workflow is as follows:

1. **Frontend: React + TailwindCSS — for user interface and image upload.**
2. **Backend: Node.js + Express — for handling requests and communicating with the ML model.**
3. **ML Model: Flask API hosting a CNN model trained using TensorFlow/Keras.**
4. **Database: MongoDB — for storing user uploads and logs.**

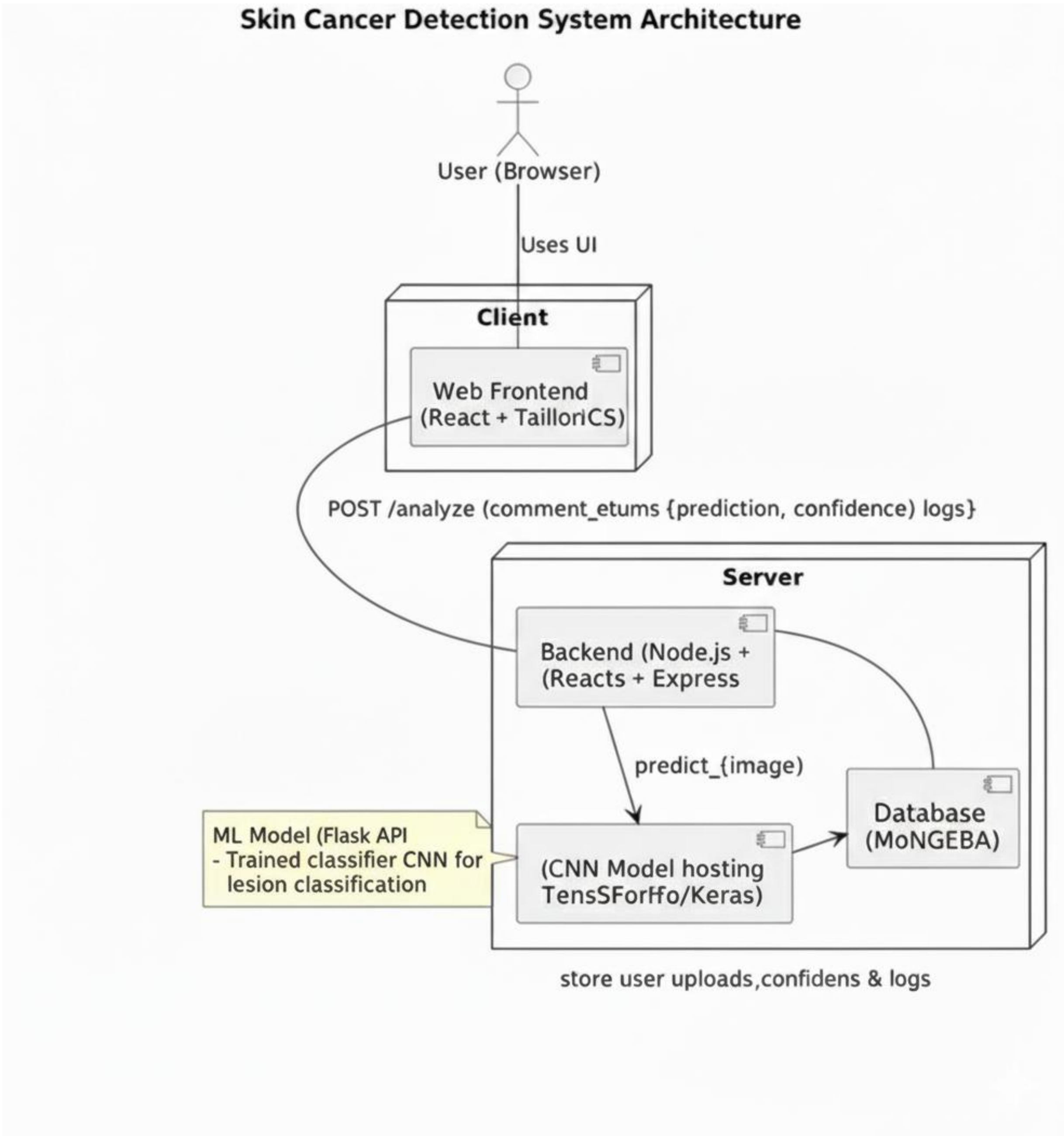


Fig. System Architecture

This minimal system architecture shows a simple, functional setup for a Toxic and Cyberbullying Comment Detection: a web frontend (JavaScript) sends a POST /analyze with the user's comment to a single backend (Flask app.py) which runs the local classifier and only when needed calls an external AI service (Google Gemini) to generate a polite rewrite; the backend returns a JSON report and suggestion to the frontend for display. Because there's no database, all processing is ephemeral (no persistence), which keeps the design lightweight and ideal.

3.3 Algorithm and Process Design

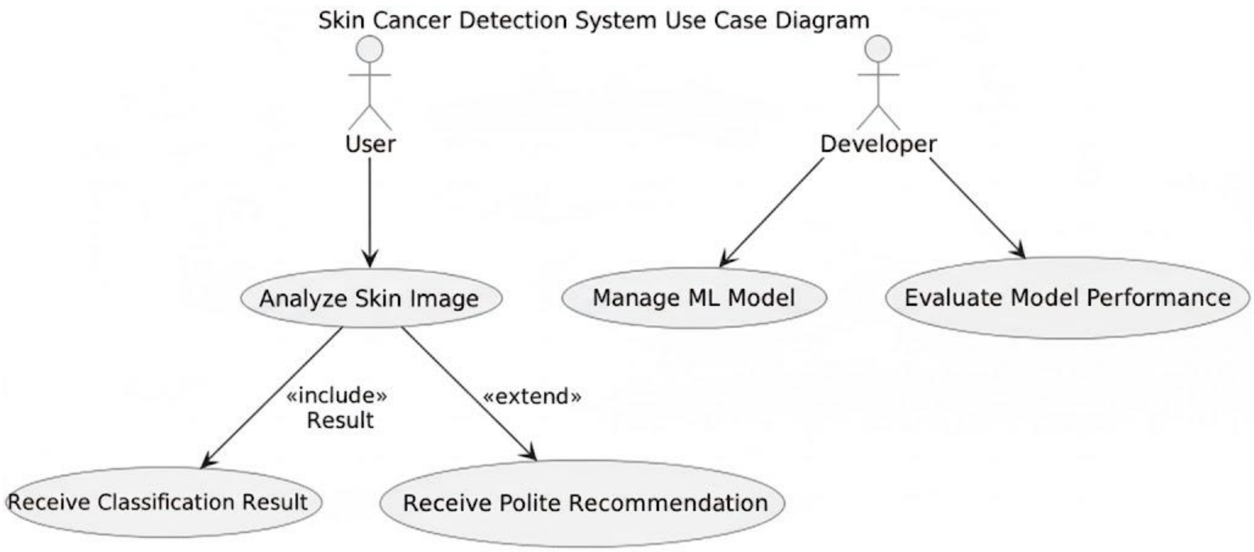
Algorithm Steps:

1. Image Acquisition (Upload by User).
2. Image Preprocessing — resizing, normalization, and augmentation.
3. Feature Extraction via CNN layers.
4. Classification using Softmax activation.
5. Result Interpretation and Display.

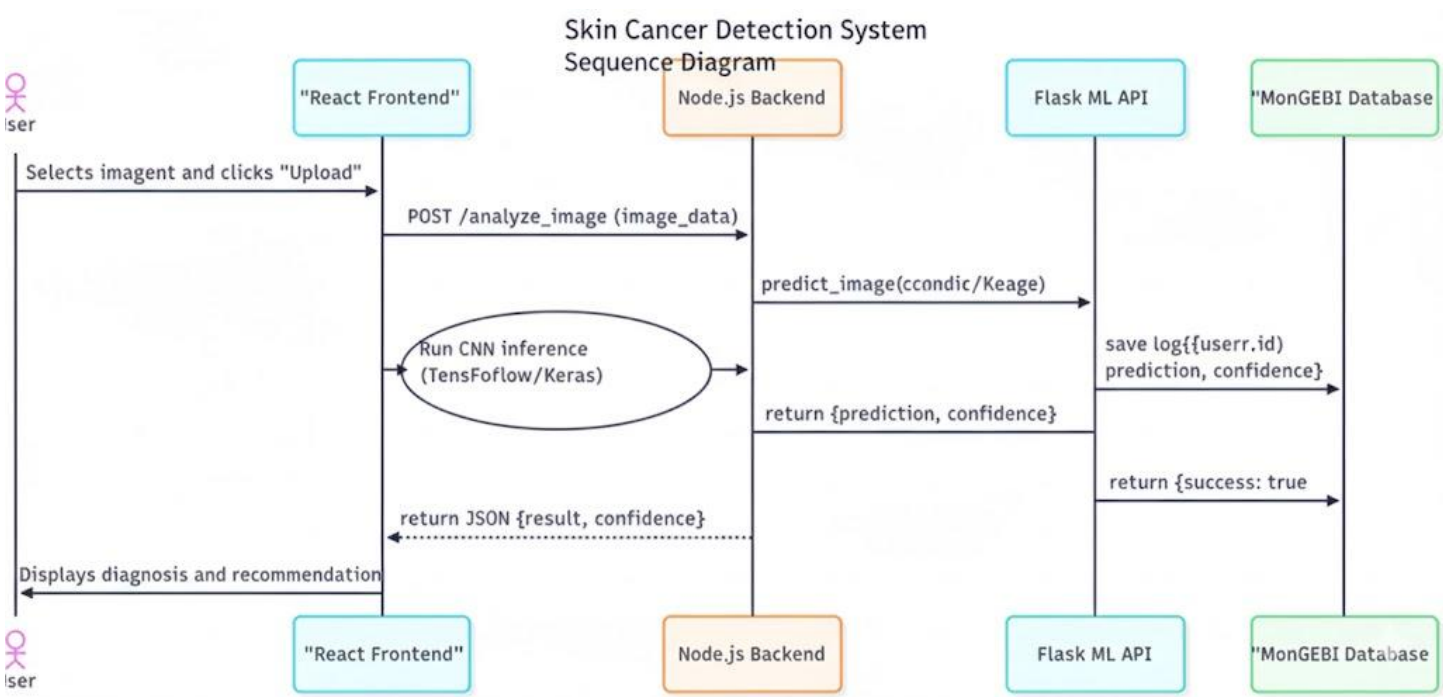
B. Model Summary:

1. Base Architecture: Custom CNN / Transfer Learning (ResNet50 or EfficientNet).
2. Activation Functions: ReLU and Softmax.
3. Loss Function: Categorical Cross-Entropy.
4. Optimizer: Adam.
5. Evaluation Metrics: Accuracy, Precision, Recall, F1-score.

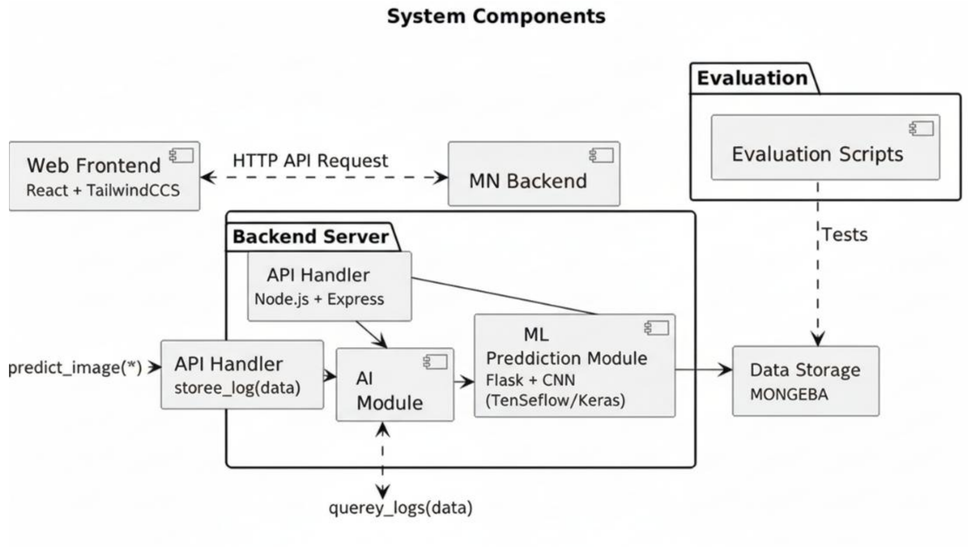
3.3.1 USE CASE DIAGRAM -



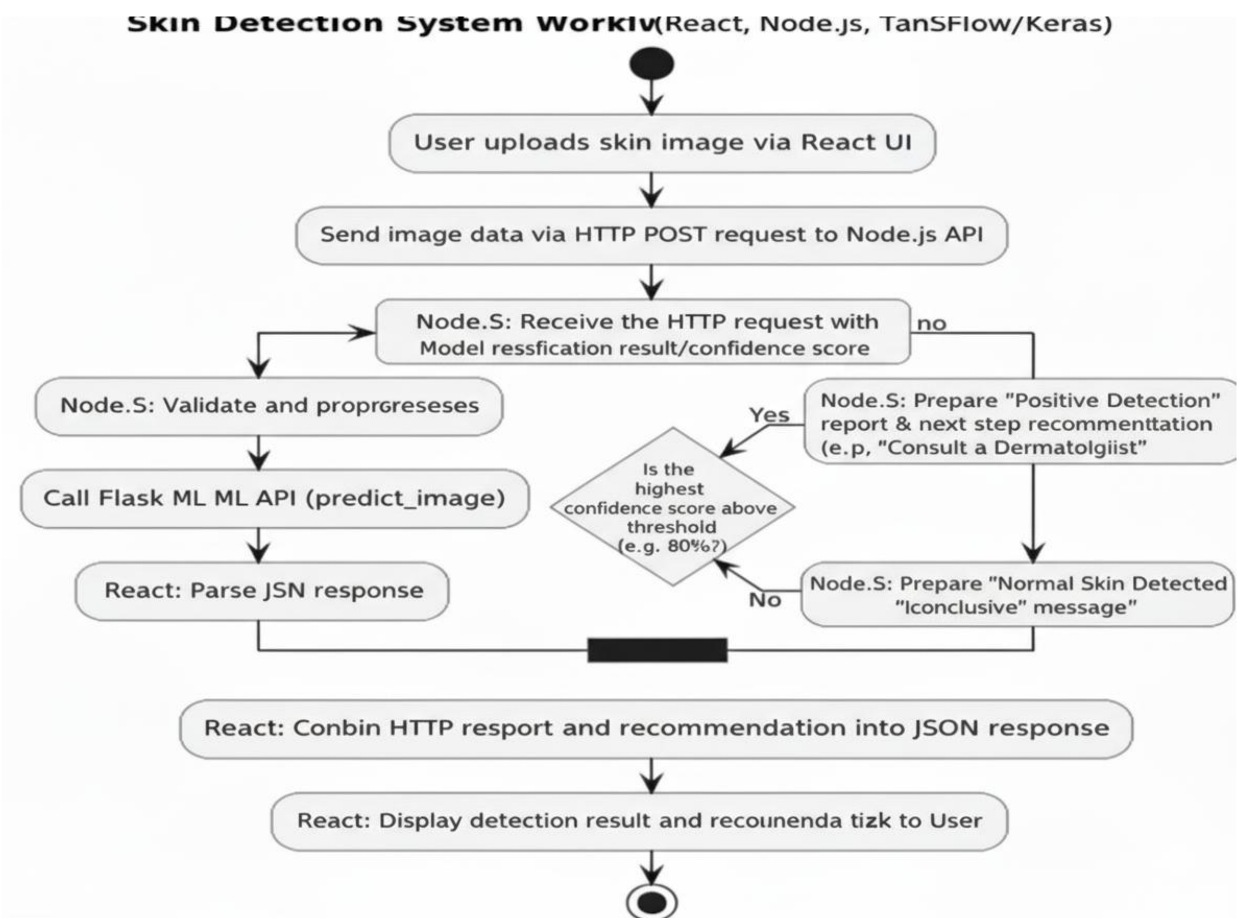
3.3.2 SEQUENCE DIAGRAM -



3.3.3 COMPONENT DIAGRAM -



3.3.4 ACTIVITY DIAGRAM -



3.4 Details of Hardware and Software

Software Requirements:

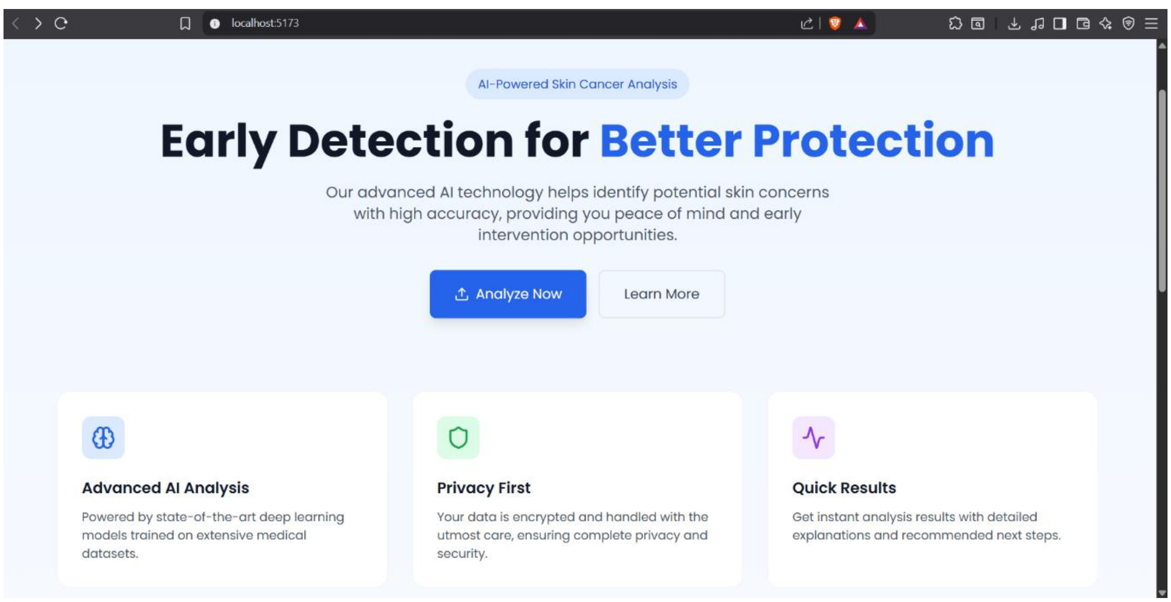
- Languages: Python, JavaScript
- Frameworks: Flask, TensorFlow/Keras, Node.js, React
- Tools: Docker, GitHub, Vercel
- Dataset: HAM10000 (Human Against Machine with 10000 training images

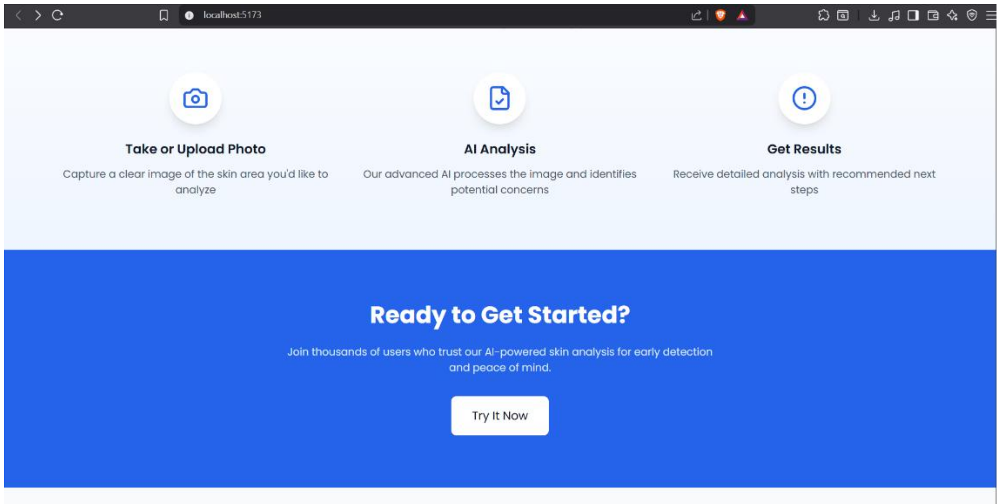
Hardware Requirements:

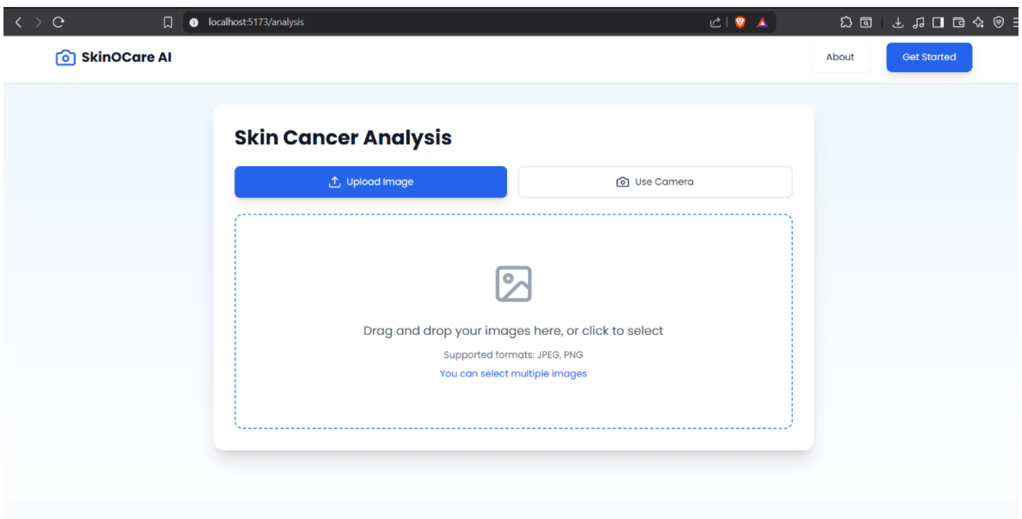
- Processor: Intel Core i5 or higher
- RAM: 8 GB minimum

GPU: NVIDIA CUDA-enabled (optional for faster training)

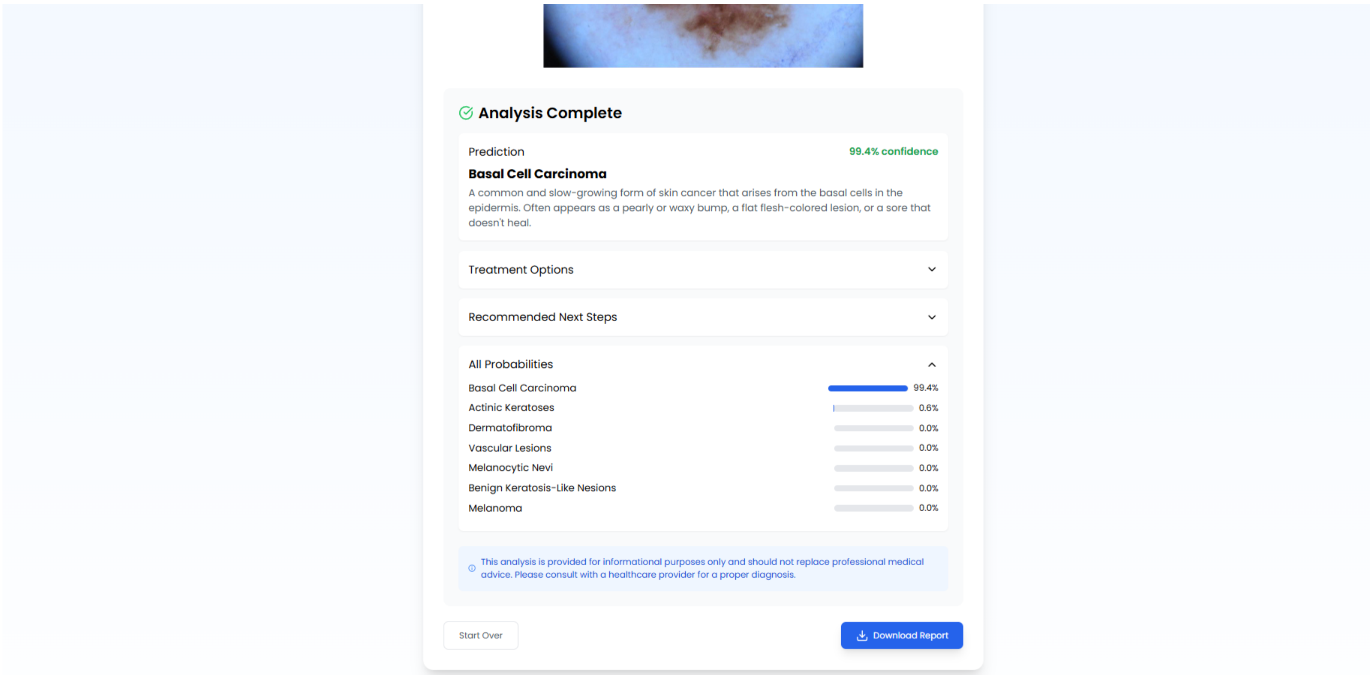
3.5 Experiments and Results



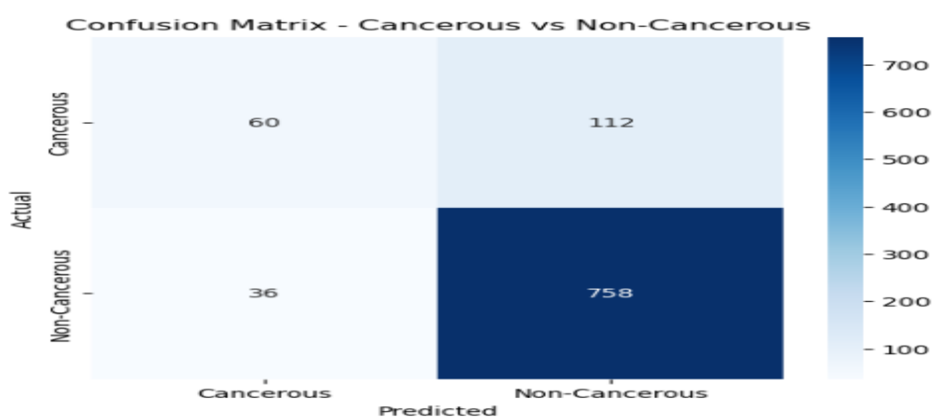

localhost:5173
Take or Upload Photo
Capture a clear image of the skin area you'd like to analyze
AI Analysis
Our advanced AI processes the image and identifies potential concerns
Get Results
Receive detailed analysis with recommended next steps
Ready to Get Started?
Join thousands of users who trust our AI-powered skin analysis for early detection and peace of mind.
Try It Now


localhost:5173/analysis
SkinOCare AI
About
Get Started
Skin Cancer Analysis
Upload Image
Use Camera
Drag and drop your images here, or click to select
Supported formats: JPEG, PNG
You can select multiple images

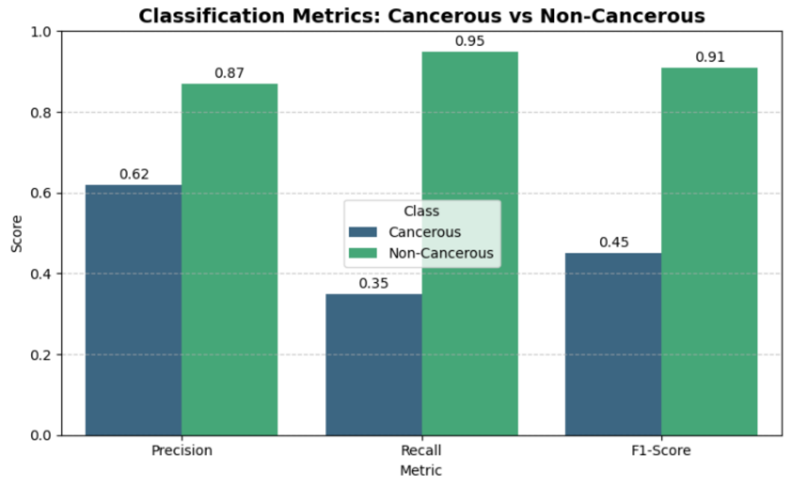

Skin Cancer Analysis
Analyzed
Analysis Complete

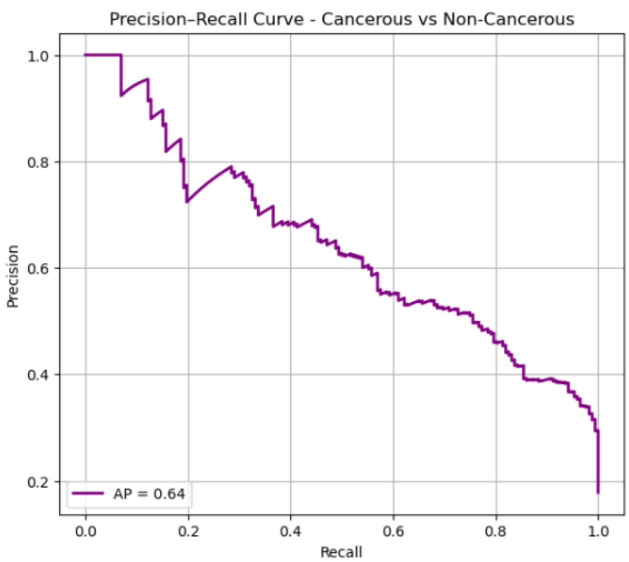


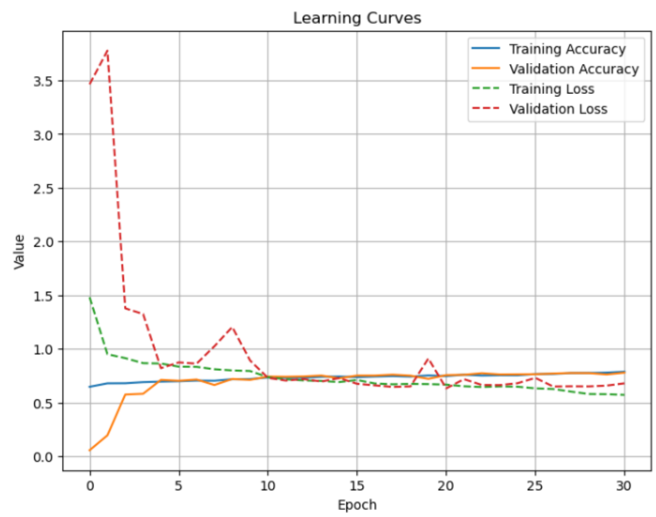
MODEL VISUALIZATION:

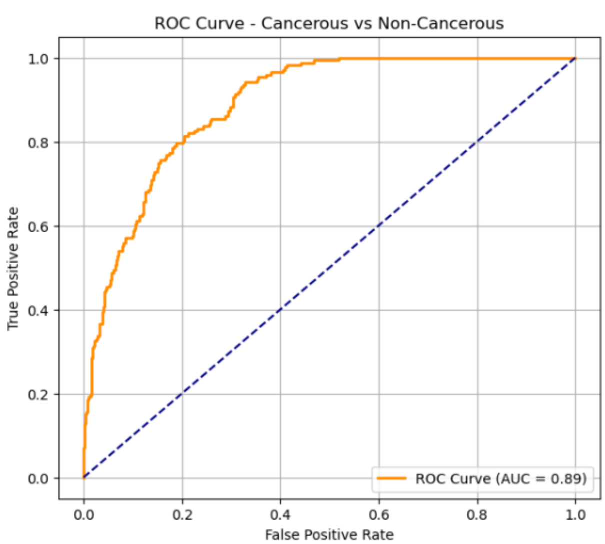


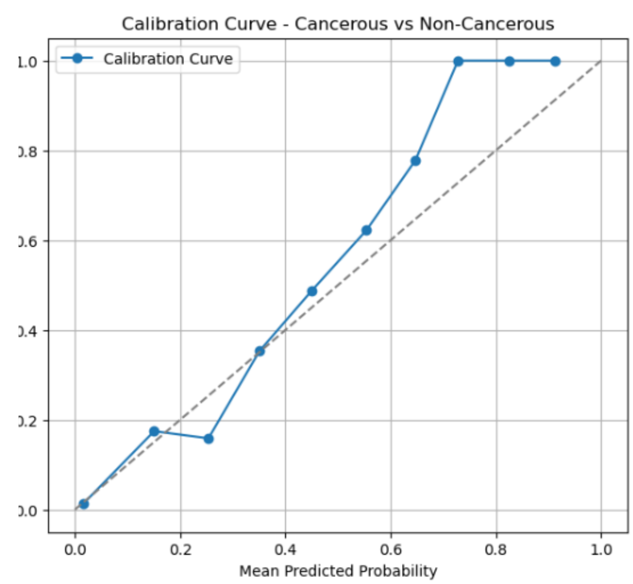

```
Classification Report (Cancerous vs Non-Cancerous):
              precision    recall  f1-score   support

    Cancerous       0.62      0.35      0.45       172
Non-Cancerous       0.87      0.95      0.91       794

     accuracy                           0.85       966
    macro avg       0.75      0.65      0.68       966
 weighted avg       0.83      0.85      0.83       966
```


Classification Metrics: Cancerous vs Non-Cancerous
Score
Metric
Precision
Recall
F1-Score
Class
Cancerous
Non-Cancerous
0.62
0.87
0.35
0.95
0.45
0.91


Precision–Recall Curve - Cancerous vs Non-Cancerous
Precision
Recall
AP = 0.64


Learning Curves
Value
Epoch
Training Accuracy
Validation Accuracy
Training Loss
Validation Loss


ROC Curve - Cancerous vs Non-Cancerous
True Positive Rate
False Positive Rate
ROC Curve (AUC = 0.89)


Calibration Curve - Cancerous vs Non-Cancerous
Calibration Curve
Mean Predicted Probability

3.6 Conclusion and Future Work:

This project successfully developed an integrated system for skin cancer detection app .The skin Cancer detection app successfully demonstrates how deep learning can be applied in healthcare diagnostics.

It provides a scalable,efficient and easy-to-use platform for early screening of skin cancer.

Future Work: There are several avenues for future improvement:

1. Incorporate advanced models like EfficientNetV2 or Vision Transformers.
2. Improve dataset diversity and augmentations for higher accuracy.
3. Integrate Grad-CAM for model explainability.
4. Extend platform to mobile devices for broader accessibility