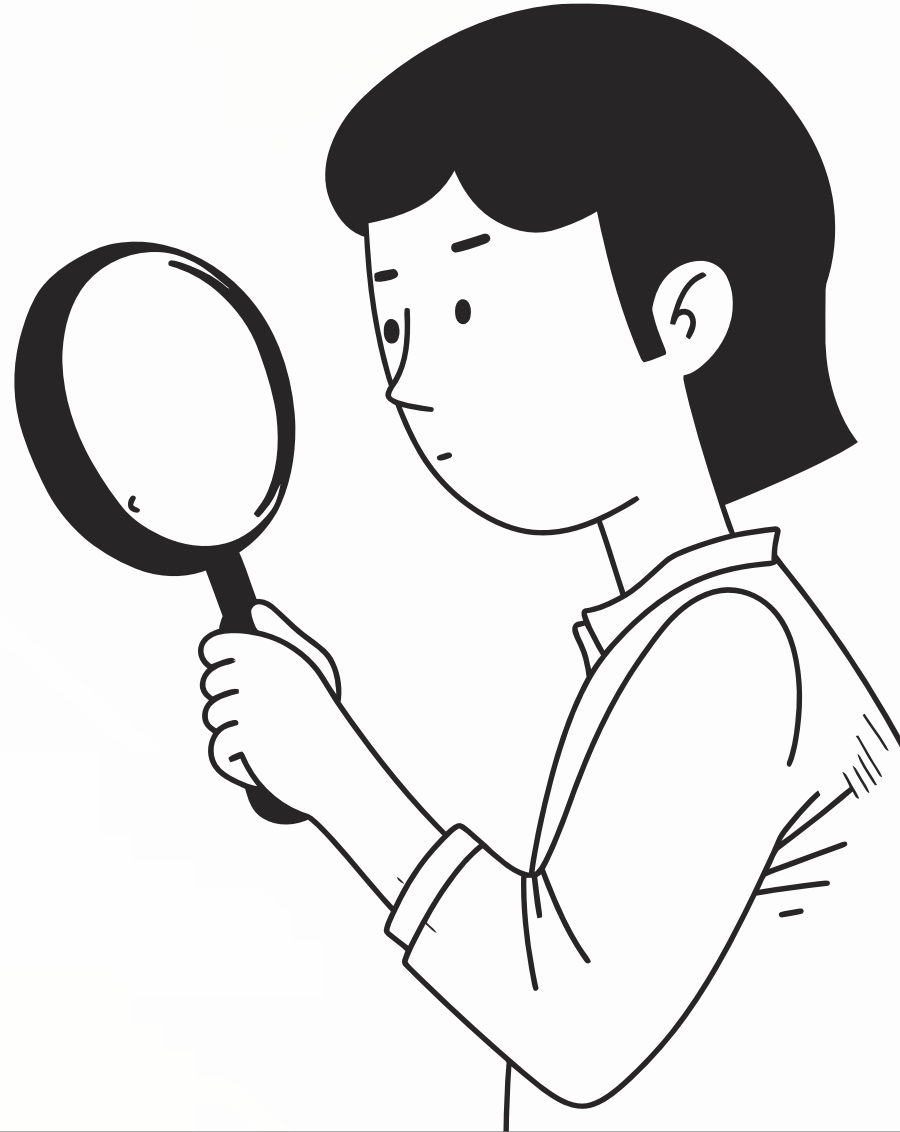


# ABDOS - AI-Based Dermatological Observation System

An Advanced Platform Using Deep Learning for Early Skin Cancer Detection  
and Patient Management By Emeka Adimora, 4th Year Software and  
Electronic Engineering Student



# Project Overview

## Purpose

Provide accessible tools for early skin cancer detection with medical guidance.

## Features

- Image-based skin lesion classification
- Risk assessment and recommendations
- User authentication and profile management
- API for frontend integration



# System Architecture

## Frontend

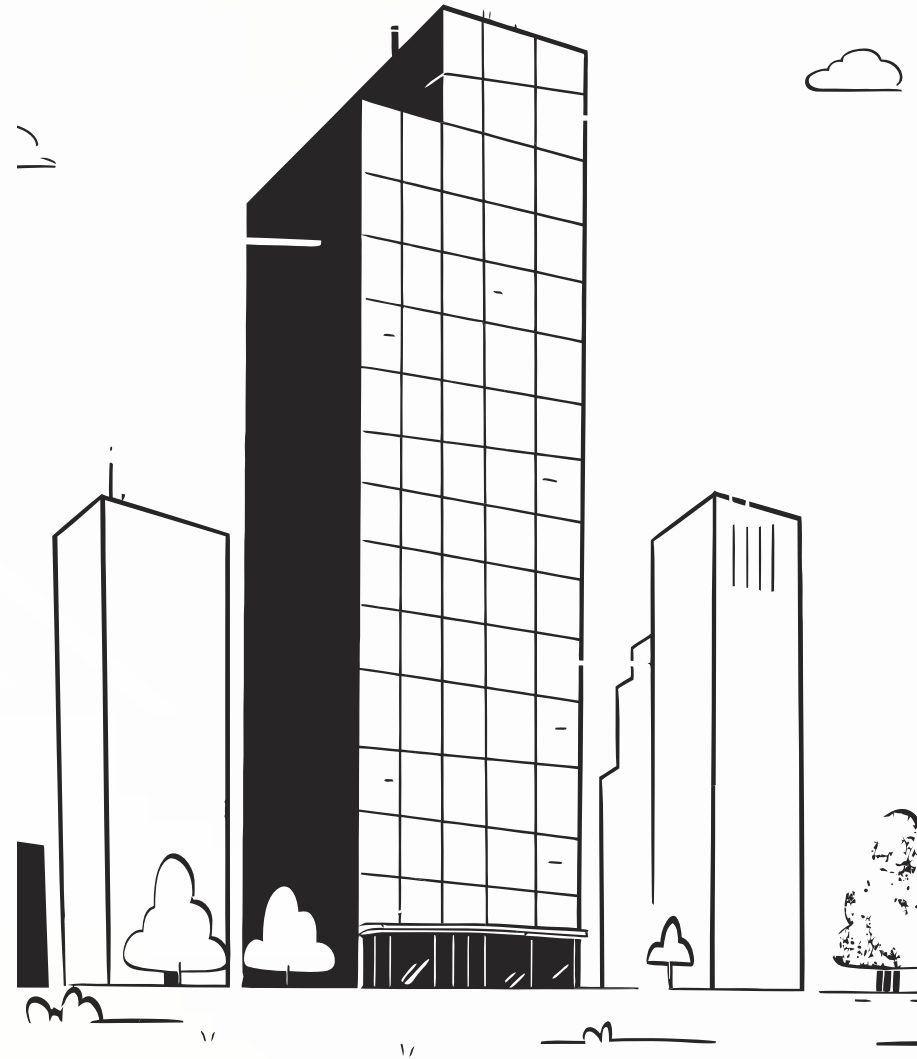
User interface built with React for uploading skin images, displaying classification results, and interacting with risk assessments and recommendations.

## Backend

RESTful API developed using Node.js and Express.js, handles user authentication, profile management, and serves machine learning model inference requests securely.

## Machine Learning Model

Deep convolutional neural network trained on diverse skin lesion datasets to classify multiple lesion types with high accuracy and reliability.



# User Authentication (Backend)



## Technologies

Node.js, Express, MongoDB, JWT for secure authentication.



## Features

- User registration and login
- JWT-based secure authentication
- Password hashing and validation
- Profile retrieval and cookie sessions



# Authentication API Endpoints

|                         |                                    |
|-------------------------|------------------------------------|
| POST /api/auth/register | Register a new user                |
| POST /api/auth/login    | User login and token generation    |
| POST /api/auth/logout   | User logout and token invalidation |
| GET /api/auth/profile   | Retrieve user profile information  |



# Slide 6: Machine Learning Model

1

## Model Architecture:

**MobileNetV2 backbone (transfer learning)**

Custom dense layers for classification

7 output classes (softmax)

2

## Custom Preprocessing Layer:

Normalizes images to  $[-1, 1]$  range

3

## Training Data:

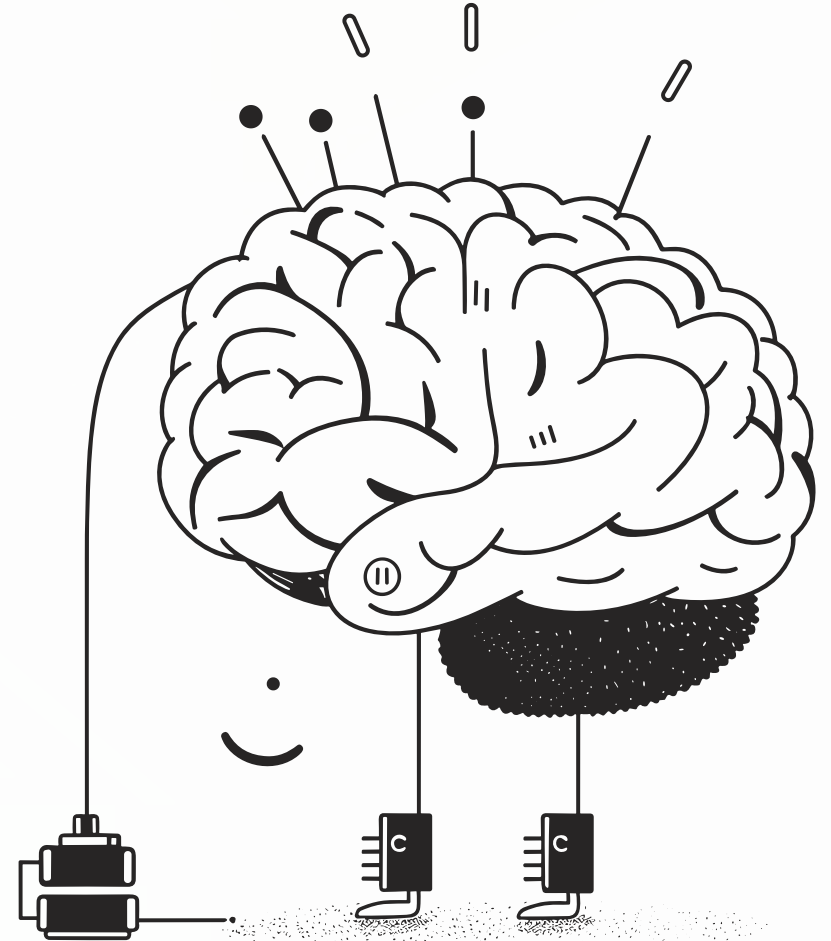
Ham10000

Data augmentation for robustness

4

## Model File:

Saved as .h5 file, loaded at runtime



# Skin Cancer Model Integration

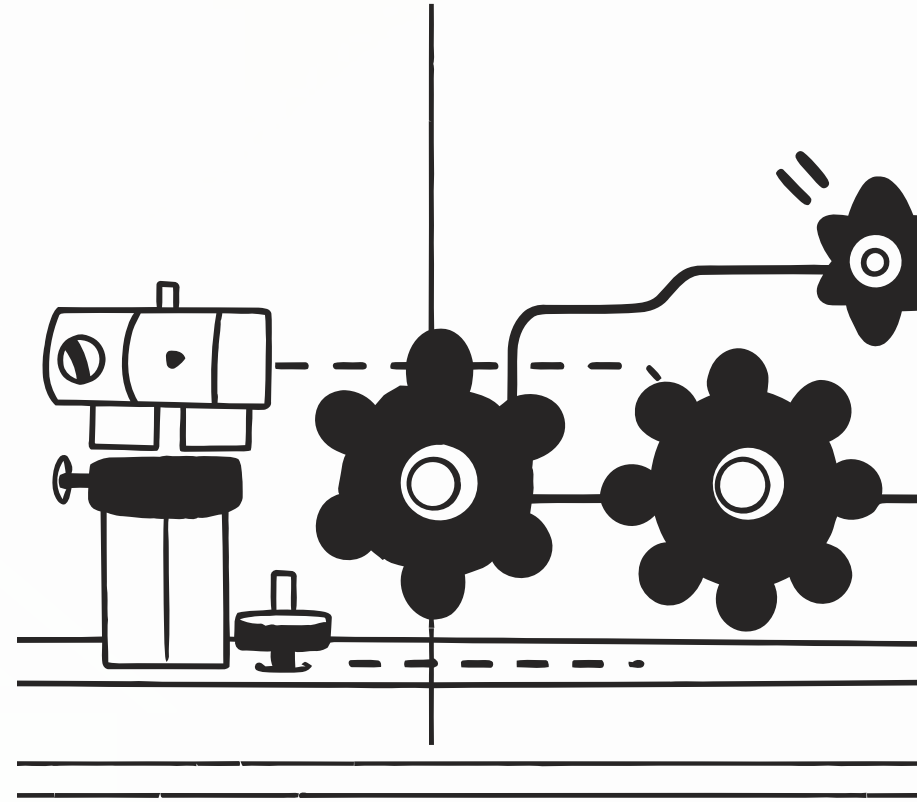
## Model

Keras/TensorFlow deep learning model with MobileNetV2 backbone.

Classifies 7 types of skin lesions.

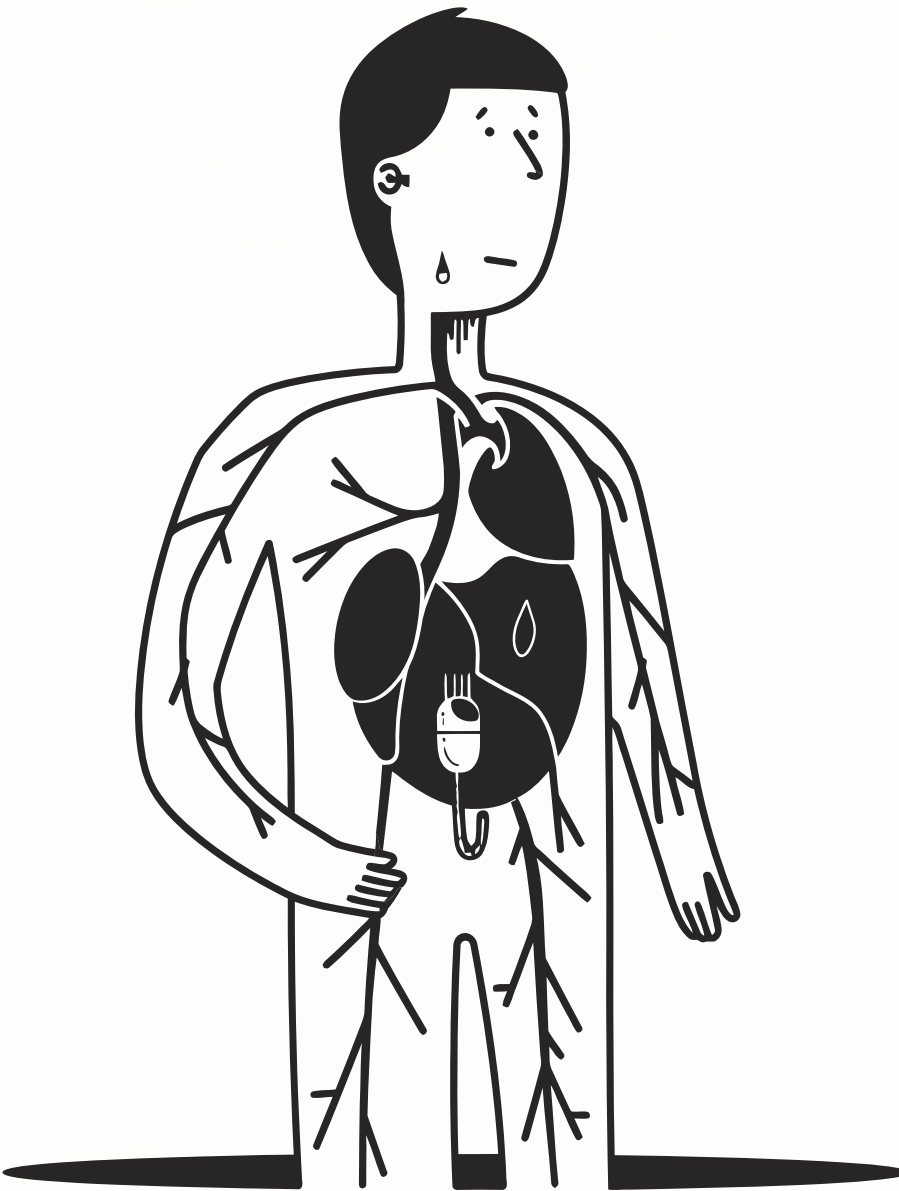
## Prediction Pipeline

- Image upload and preprocessing
- Model inference
- Outputs: predicted class, confidence, risk, action








# Skin Lesion Classes

| Index | Code  | Condition            | Risk Level |
|-------|-------|----------------------|------------|
| 0     | akiec | Actinic Keratoses    | Moderate   |
| 1     | bcc   | Basal Cell Carcinoma | High       |
| 2     | bkl   | Benign Keratosis     | Low        |
| 3     | df    | Dermatofibroma       | Low        |
| 4     | mel   | Melanoma             | Very High  |
| 5     | nv    | Melanocytic Nevus    | Low        |
| 6     | vasc  | Vascular Lesions     | Low        |





# Prediction Output Example

|   |  |  |
|---|--|--|
|  <div><div>Predicted Condition</div><div>Melanoma</div><div>This is a malignant tumor of melanocytes, the cells that produce pigment. Early detection is critical for effective treatment.</div></div> |  <div><div>Confidence</div><div>92.3%</div><div>The model is highly confident in this prediction, reflecting strong evidence from the input image features.</div></div>                                       |  <div><div>Risk Level</div><div>Very High</div><div>Melanoma is considered the most dangerous type of skin cancer due to its tendency to spread rapidly if untreated.</div></div> |
|  <div><div>Description</div><div>Most serious form of skin cancer.</div><div>It often appears as a new or changing mole and requires prompt diagnosis by a dermatologist.</div></div>                  |  <div><div>Recommended Action</div><div>Urgent medical attention required.</div><div>Immediate consultation with a healthcare professional is advised to confirm diagnosis and start treatment.</div></div> |  |

# Testing & Validation

## Model Loading Test

Verifies model loads and predicts correctly with sample input.

## API Test

Sends images to API, checks response accuracy and latency.

## Manual Testing

Uploads via frontend UI, reviews prediction results and UX.



# Security & Privacy



## Data Protection

User data is securely protected through JWT authentication and encrypted cookies.



## Password Security

Passwords are securely hashed and never stored in plain text.



## Privacy

Patient images and prediction data are never stored without explicit user consent.

