**Al-Based Tool for Preliminary Diagnosis of Dermatological Manifestations**

|  |
| --- |
| **CONTENT** |
| **Abstract** |
| **1.INTRODUCTION** |
| 1.1 Motivation |
| 1.2 Problem Statement |
| 1.3 Objective of the Project |
| 1.4 Scope |
| 1.5 Project Introduction |
| **2.LTERATURE SURVEY** |
| 2.1 Related Work |
| **3. SYSTEM ANALYSIS** |
| 3.1 Existing System |
| 3.2 Disadvantages |
| 3.3 Proposed System |
| 3.4 Advantages |
| 3.5 work Flow of Proposed system |
| **4. REQUIREMENT ANALYSIS** |
| 4.1Function and non-functional requirements |
| 4.2 Hardware Requirements |
| 4.3 Software Requirements |
| 4.4 Architecture |
| **5. METHODOLOGY** |
| 5.1 Decision Tree |
| 5.2 Random Forest |
| **6. SYSTEM DESIGN** |
| 6.1 Introduction of Input design |
| 6.2 UML Diagram(class, use case, sequence, collaborative, deployment, activity, ER diagram and Component diagram) |
| 6.3 Data Flow Diagram |
| **7. IMPLEMENTATION AND RESULTS** |
| 7.1 Modules |
| 7.2 Output Screens |
| 1. **SYSTEM STUDY AND TESTING** |
| 8.1 Feasibility study |
| 8.2 Types of test & Test Cases |
| **9. CONCLUSION** |
| **10. FUTURE ENHANCEMENT** |
| **11. REFERENCES** |

# **ABSTRACT**

The increasing prevalence of skin diseases and the shortage of dermatologists in many regions necessitate the development of innovative solutions to assist in the preliminary diagnosis of dermatological conditions. We present an AI-based tool designed to serve as an aid for patients, general practitioners, and dermatologists. This system leverages Convolutional Neural Networks (CNN) and MobileNet architecture for the analysis of skin image datasets, providing a platform for the preliminary diagnosis of skin diseases.

Upon registration, users can access different functionalities based on their role: admin/owner, patient, or doctor. The admin can train the model with a skin image dataset, view registered users, and receive feedback. Patients can log in, upload skin images, and provide additional information such as age, gender, and duration of the condition. The system then classifies the disease, provides a description, suggests remedies, and assesses disease severity. It also recommends dermatologists based on the diagnosis, allowing patients to book appointments and view their past disease history.

Doctors can view their appointments and manage their profiles, enhancing the tool's utility for professional use. The system is designed to support both English and Telugu, broadening its accessibility. Through this AI-based diagnostic tool, we aim to bridge the gap in dermatological care by providing immediate, preliminary support, and facilitating effective patient-doctor interaction.

**Keywords**: Dermatological Manifestations, Supervised machine learning, CNN, MobileNet, Hybrid Algorithm.

# **INTRODUCTION**

## **1.1 MOTIVATION**

## The motivation behind this project stems from the increasing prevalence of skin diseases and the critical shortage of dermatologists, particularly in underserved regions. Many patients face delays in diagnosis and treatment due to limited access to specialized care. By leveraging AI technology, specifically CNN and MobileNet architectures, this tool aims to provide a reliable and immediate preliminary diagnosis, empowering patients with timely information and connecting them with dermatologists. This project aspires to enhance dermatological care accessibility, improve patient outcomes, and streamline the diagnostic process, ultimately contributing to better healthcare delivery..

## **1.2 PROBLEM STATEMENT**

Despite advances in healthcare, there remains a significant gap in dermatological services, particularly in preliminary diagnosis. The deficit of dermatologists and the geographic limitations in accessing specialist care exacerbate the problem, leading to delayed diagnoses and treatments. This gap in service disproportionately affects rural and underserved communities, often resulting in worsened health outcomes.

The problem is twofold: firstly, there is an uneven distribution of dermatological expertise, and secondly, there is an increasing incidence of skin diseases globally. The need for an accessible, efficient, and accurate preliminary diagnostic process is critical. This study proposes to address these issues by developing an AI-based diagnostic tool that can provide immediate, preliminary assessments of dermatological conditions. The tool aims to alleviate the burden on the healthcare system by enabling triage and prioritization, potentially reducing wait times for patients and helping to distribute the workload more evenly across available dermatological services.

## **1.3 OBJECTIVE**

## The objective of this project is to develop an AI-based diagnostic tool that leverages Convolutional Neural Networks (CNN) and MobileNet architecture to assist in the preliminary diagnosis of dermatological conditions. This system aims to serve patients, general practitioners, and dermatologists by providing an accessible platform for analyzing skin images, classifying diseases, suggesting remedies, and recommending dermatologists based on the diagnosis. By integrating AI technology, the tool seeks to bridge the gap in dermatological care, particularly in regions with a shortage of dermatologists, and improve patient-doctor interaction for better healthcare outcomes.

## **1.4 Scope**

## The scope of this project encompasses the development of a comprehensive AI-based tool designed to support the preliminary diagnosis of skin diseases. The system will include functionalities for different user roles: admin/owner, patient, and doctor. Admins can train the model, manage users, and collect feedback. Patients can upload images, receive diagnoses, and book appointments with recommended dermatologists. Doctors can manage their profiles and appointments. The tool will support English and Telugu languages, ensuring broad accessibility. By utilizing CNN and MobileNet architectures, the system aims to provide accurate and timely diagnostic support, enhancing dermatological care.

## **2. LITERATURE SURVEY**

**1. Skin Cancer Classification using Deep Learning and Transfer Learning by Khalid M. Hosny, Mohamed A. Kassem, and Mohamed M. Foaud at 2018 IEEE**

Skin cancer, specially melanoma is one of most deadly diseases. In the color images of skin, there is a high similarity between different skin lesion like melanoma and nevus, which increase the difficulty of the detection and diagnosis. A reliable automated system for skin lesion classification is essential for early detection to save effort, time and human life. In this paper, an automated skin lesion classification method is proposed. In this method, a pre-trained deep learning network and transfer learning are utilized. In addition to fine-tuning and data augmentation, the transfer learning is applied to AlexNet by replacing the last layer by a softmax to classify three different lesions (melanoma, common nevus and atypical nevus). The proposed model is trained and tested using the ph2 dataset. The well-known quantative measures, accuracy, sensitivity, specificity, and precision are used in evaluating the performance of the proposed method where the obtained values of these measures are 98.61%, 98.33%, 98.93%, and 97.73%, respectively. The performance of the proposed method is compared with the existing methods where the classification rate of the proposed method outperformed the performance of the existing methods.

1. **Skin Cancer Detection and Classification by Pratik Dubal, Sankirtan Bhatt, Chaitanya Joglekar, Dr. Sonali Patil at 2017 IEEE**

Skin cancer is the most common type of cancer, which affects the life of millions of people every year. About three million people are diagnosed with the disease every year in the United States alone. The rate of survival decreases steeply as the the disease progresses. However, detection of skin cancer in the early stages is a difficult and expensive process. In this study, we propose a methodology that detects and identifies skin lesions as benign or malignant based upon images taken from general cameras. The images are segmented, features extracted by applying the ABCD rule and a Neural Network is trained to classify the lesions to a high degree of accuracy. The trained Neural Network achieved an overall classification accuracy of 76.9% on a dataset of 463 images, divided into six distinct classes. The overall accuracy rate and performance of the system can be improved by training the neural network on a much larger and diverse dataset with high intra-class variability. This would decrease the misclassification and positively impact the accuracy rate. An alternative to diversifying the dataset is increasing the number of features extracted from the images.

1. **Skin Cancer Classification Using K-Means Clustering Mohd Anas, Ram Kailash Gupta, Dr. Shafeeq Ahmad at 2017 IEEE**

Detection of skin cancer gives the best chance of being diagnosed early. Biopsy method for skin cancer detection is much painful. Human interpretation contains difficulty and subjectivity therefore automated analysis of skin cancer affected images has become important. This paper proposes an automatic medical image classification method to classify two major type skin cancers: Melanoma, and Non-melanoma. In this paper, we have used the color and texture features in combination which gives better results than using color or gray level information alone. We have used k-means clustering algorithm to segment the lesion. The features are extracted by six different color-texture feature extractors from the segmented images. Classification accuracy of our proposed system is evaluated on four different types of classifiers and their values are compared with one another. The results of the proposed system are computed on five different classification rate in order to perform better analysis of our proposed system.

1. **Skin Cancer Classification using Transfer Learning by Hari Kishan Kondaveeti and Prabhat Edupuganti at IEEE 2020**

Today, Cancer is one of the major lethal diseases in the world. Globally out of every three cancers diagnosed, one is identified as skin cancer. Some reports suggest that one out of every five Americans might fall prey to skin cancer in the course of their life. Early detection of the disease plays a pivotal role in the treatment of skin cancer. Though these skin lesions can be seen without the help of any external clinical device, it is a challenging task to distinguish between malignant and benign skin lesions as they are alike in their physical appearances. This leads to an increased number of unnecessary biopsies where in one study it was revealed that nearly 5,00,000 biopsies are done in children every year to diagnose a mere 400 melanomas. To tackle this problem and help dermatologists in the diagnosis process, we developed an enhanced image classification model which can act as a preliminary check before moving to a costlier biopsy. This model can identify 7 distinct types of skin lesions. An analysis has been carried out on the HAM10000 dataset. We used transfer learning utilizing multiple pre-trained models, combined with class-weighted loss and data augmentation techniques for the classification process. Experimental analysis shows that the modified ResNet50 model is capable of identifying skin lesion images into one of the seven classes with categorical accuracy, weighted average precision, and recall of 90 percent, 0.89, 0.90, respectively. Our model can be used as a clinical decision support system to help dermatologists in the diagnosis process.

## **3. SYSTEM ANALYSIS**

## **3.1 EXISTING SYSTEM**

The existing system in dermatological diagnostics that incorporates the Convolutional Neural Network (CNN) algorithm represents a significant technological advancement in the field. CNNs, a class of deep neural networks, are particularly adept at processing visual imagery, making them highly suitable for analyzing skin images. In the current setup, these algorithms are trained on large datasets of dermatological images, where they learn to identify patterns and features indicative of various skin conditions.

The process typically involves inputting a skin image into the CNN, which then undergoes several layers of processing. Each layer extracts specific features from the image, with deeper layers identifying more complex patterns. The final output is a classification or diagnosis based on the features identified in the image.

This system offers a more objective and consistent approach compared to traditional methods, which rely heavily on individual practitioner expertise. By utilizing CNNs, the existing system can assist dermatologists in diagnosing a wide range of skin diseases more quickly and accurately, thereby enhancing patient care. However, it is essential to note that this AI-based approach is generally used in conjunction with, rather than as a replacement for, the clinical judgment of healthcare professionals.

### **3.2 DISADVANTAGES**

* Data Dependency: CNNs require large, diverse datasets for training; insufficient or biased data can lead to inaccurate or skewed diagnostic results.
* Lack of Explain ability: CNNs operate as a "black box," making it difficult to understand how they arrive at specific diagnoses, impacting user trust.
* Overfitting Risks: Without proper tuning, CNNs might overfit to training data, leading to poor performance on new, unseen images.
* Resource Intensity: Training and running CNN models demand significant computational resources, which can be costly and limit accessibility in resource-constrained settings.
* Complementary, Not Replacement: CNNs assist but don’t replace dermatologists' expertise; over-reliance on AI could overlook nuanced clinical insights and patient context.

## **3.3 PROPOSED SYSTEM**

The proposed model innovatively combines MobileNet architecture with a Long Short-Term Memory (LSTM) network, creating a hybrid AI system for enhanced dermatological diagnosis. MobileNet, known for its efficiency in processing high-resolution images, serves as the foundation. It efficiently extracts and processes spatial features from dermatological images, identifying key patterns indicative of various skin conditions.

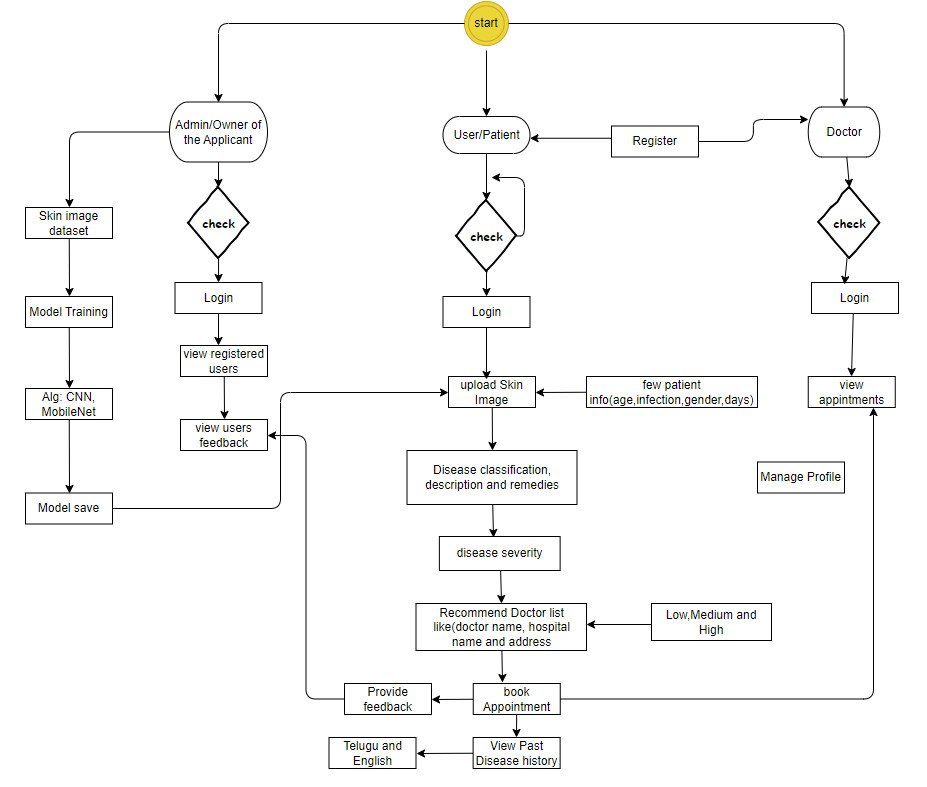
Integrating LSTM, a form of recurrent neural network, adds a temporal dimension to the analysis. This is particularly beneficial when the model assesses progression in skin conditions over time, based on sequential images or data. The LSTM component effectively captures and interprets changes in skin images, providing insights into the evolution of skin diseases.

This hybrid approach leverages the strengths of both architectures: MobileNet's proficiency in image classification and LSTM's capability in handling sequential data. The result is a robust and versatile diagnostic tool that not only identifies dermatological conditions with high accuracy but also tracks their progression, offering a comprehensive understanding that aids in more effective treatment planning.

### **3.4 Advantages**

* Enhanced Diagnostic Accuracy: Combining MobileNet and LSTM leverages spatial and temporal data analysis, significantly improving the accuracy of skin condition diagnoses.
* Disease Progression Tracking: LSTM's sequential data analysis allows for effective monitoring of skin disease progression, facilitating timely and targeted treatments.
* Efficient Processing: MobileNet's lightweight architecture ensures fast image processing, making the system suitable for real-time applications and remote diagnostics.
* Versatile Application: The hybrid model is adaptable to various skin types and conditions, enhancing its applicability across diverse patient demographics.
* Improved Patient Outcomes: Accurate and timely diagnoses, coupled with disease progression monitoring, contribute to better treatment strategies and overall patient care.

**3.5 BLOCK DIAGRAM**



**Figure 1 Block diagram of the project**

**4. SYSTEM REQUIREMENTS**

**4.1 FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS:**

Requirement’s analysis is very critical process that enables the success of a system or software project to be assessed. Requirements are generally split into two types: Functional and non-functional requirements.

**Functional Requirements**: These are the requirements that the end user specifically demands as basic facilities that the system should offer. All these functionalities need to be necessarily incorporated into the system as a part of the contract. These are represented or stated in the form of input to be given to the system, the operation performed and the output expected. They are basically the requirements stated by the user which one can see directly in the final product, unlike the non-functional requirements.

Examples of functional requirements:

1. Authentication of user whenever he/she logs into the system
2. System shutdown in case of a cyber-attack
3. A verification email is sent to user whenever he/she register for the first time on some software system.

**Non-functional requirements**: These are basically the quality constraints that the system must satisfy according to the project contract. The priority or extent to which these factors are implemented varies from one project to other. They are also called non-behavioral requirements.  
They basically deal with issues like:

* Portability
* Security
* Maintainability
* Reliability
* Scalability
* Performance
* Reusability
* Flexibility

Examples of non-functional requirements:

1. Emails should be sent with a latency of no greater than 12 hours from such an activity.
2. The processing of each request should be done within 10 seconds
3. The site should load in 3 seconds whenever of simultaneous users are > 10000

**4.2 HARDWARE REQUIREMENTS:**

Operating system : Windows 7 or 7+

RAM : 8 GB

Hard disc or SSD : More than 500 GB

Processor : Intel 3rd generation or high or Ryzen with 8 GB Ram

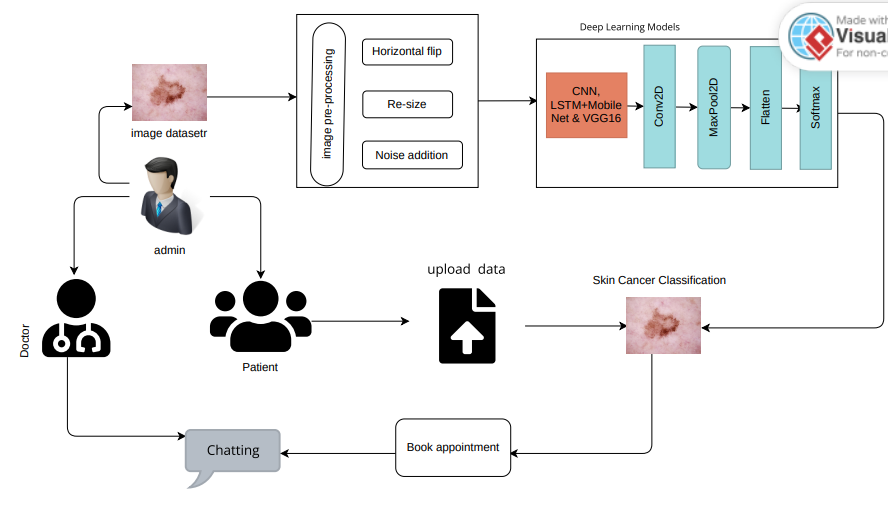
**4.3 SOFTWARE REQUIREMENTS:**

Software’s : Python 3.6 or high version

IDE : PyCharm.

Framework : Flask

**4.4 ARCHITECTURE**

****

# **METHODOLOGY**

**5.1 Convolutional Neural Network (CNN) Layers:**

**Input Layer:**

**Description:** This layer represents the input image data. Each pixel in the image is represented as a node in this layer. The dimensions of this layer correspond to the dimensions of the input image.

**Usage: It serves as the entry point for the image data into the neural network.**

**Convolutional Layers:**

**Description:** These layers apply convolutional filters to the input image to extract features such as edges, textures, and patterns. Each filter is a small matrix that is convolved with the input image to produce feature maps.

**Usage:** Convolutional layers are responsible for feature extraction, and multiple such layers with increasing complexity can capture hierarchical features.

**Activation Function (e.g., ReLU):**

**Description:** After each convolution operation, an activation function like ReLU (Rectified Linear Unit) is applied element-wise to introduce non-linearity into the network.

**Usage:** It helps the network learn complex patterns and relationships in the data.

**Pooling Layers (e.g., Max Pooling):**

**Description:** Pooling layers downsample the feature maps generated by the convolutional layers by taking the maximum (or average) value within a certain window.

**Usage:** Pooling layers reduce the spatial dimensions of the feature maps, making the network more computationally efficient and invariant to small spatial translations.

**Dropout Layers:**

**Description:** Dropout layers randomly drop a certain percentage of neuron units during training to prevent overfitting by forcing the network to learn more robust features.

**Usage:** Dropout layers help in generalization by reducing the interdependence between neurons.

**Fully Connected Layers:**

**Description:** These layers connect every neuron in one layer to every neuron in the next layer, forming a fully connected network.

**Usage:** Fully connected layers are responsible for learning high-level features and making predictions based on these features.

**Output Layer:**

**Description:** The output layer produces the final predictions of the network. It typically consists of one neuron per class in a classification task, with a softmax activation function to output class probabilities.

**Usage:** It provides the final classification result based on the learned features.

**5.2 MOBILENET:**

**MobileNet Layers:**

MobileNet follows a similar structure to traditional CNNs but utilizes depthwise separable convolutions to reduce the number of parameters and computational complexity. Here's a brief overview:

**Depthwise Separable Convolution:**

**Description:** Depthwise separable convolution decomposes the standard convolution operation into two separate operations: depthwise convolution and pointwise convolution. Depthwise convolution applies a single filter to each input channel independently, followed by pointwise convolution which applies a 1x1 convolution to combine the output channels.

**Usage:** This architecture reduces the number of parameters and computational cost while preserving performance, making it suitable for mobile and embedded applications.

**Activation Function:**

**Description:** Similar to traditional CNNs, MobileNet uses activation functions like ReLU to introduce non-linearity.

**Pooling and Fully Connected Layers:**

MobileNet typically includes pooling and fully connected layers similar to traditional CNNs for downsampling and classification purposes.

By incorporating these layers into the CNN and MobileNet architectures, your skin disease detection system can effectively learn and extract features from input images and make accurate predictions about the presence of skin diseases.

## **SYSTEM DESIGN**

### **6.1 Input Design**

In an information system, input is the raw data that is processed to produce output. During the input design, the developers must consider the input devices such as PC, MICR, OMR, etc.

Therefore, the quality of system input determines the quality of system output. Well-designed input forms and screens have following properties −

* It should serve specific purpose effectively such as storing, recording, and retrieving the information.
* It ensures proper completion with accuracy.
* It should be easy to fill and straightforward.
* It should focus on user’s attention, consistency, and simplicity.
* All these objectives are obtained using the knowledge of basic design principles regarding −
  + What are the inputs needed for the system?
  + How end users respond to different elements of forms and screens.

**Objectives for Input Design:**

The objectives of input design are −

* To design data entry and input procedures
* To reduce input volume
* To design source documents for data capture or devise other data capture methods
* To design input data records, data entry screens, user interface screens, etc.
* To use validation checks and develop effective input controls.

**Output Design**

The design of output is the most important task of any system. During output design, developers identify the type of outputs needed, and consider the necessary output controls and prototype report layouts.

**Objectives of Output Design:**

The objectives of input design are:

* To develop output design that serves the intended purpose and eliminates the production of unwanted output.
* To develop the output design that meets the end user’s requirements.
* To deliver the appropriate quantity of output.
* To form the output in appropriate format and direct it to the right person.
* To make the output available on time for making good decisions.

**6.2 UML Diagrams**

UML stands for Unified Modelling Language. UML is a standardized general-purpose modelling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modelling Language is a standard language for specifying, Visualization, Constructing and documenting the artefacts of software system, as well as for business modelling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems.

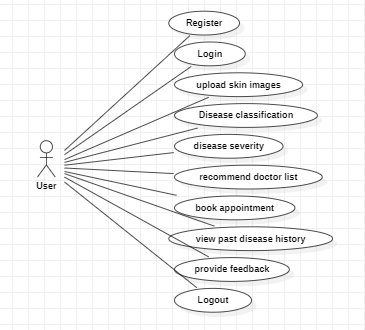
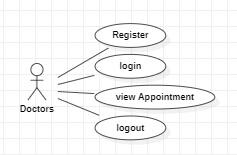
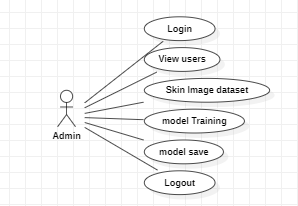
The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

**GOALS:**

The Primary goals in the design of the UML are as follows:

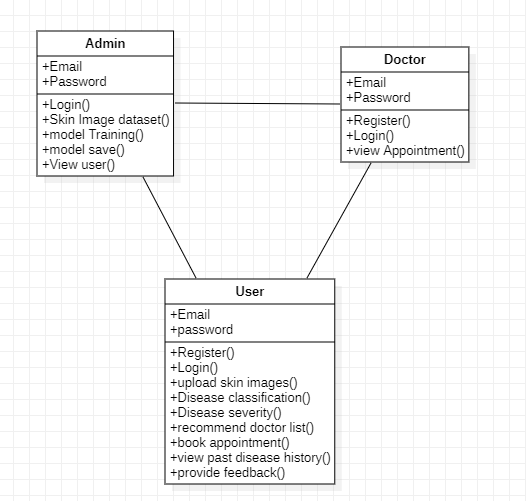
1. Provide users a ready-to-use, expressive visual modelling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modelling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

### **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.

**Figure 3 Use case diagram**

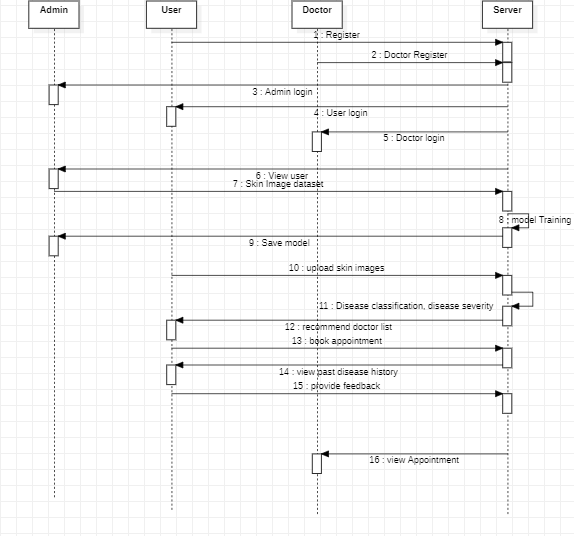
**Class Diagram**

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.

**Figure 4 Class diagram**

**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 diagrams



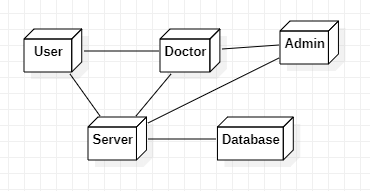
**Figure 5 Sequence Diagram**

**Collaboration Diagram**

In collaboration diagram the method call sequence is indicated by some numbering technique as shown below. The number indicates how the methods are called one after another. We have taken the same order management system to describe the collaboration diagram. The method calls are similar to that of a sequence diagram. But the difference is that the sequence diagram does not describe the object organization whereas the collaboration diagram shows the object organization.

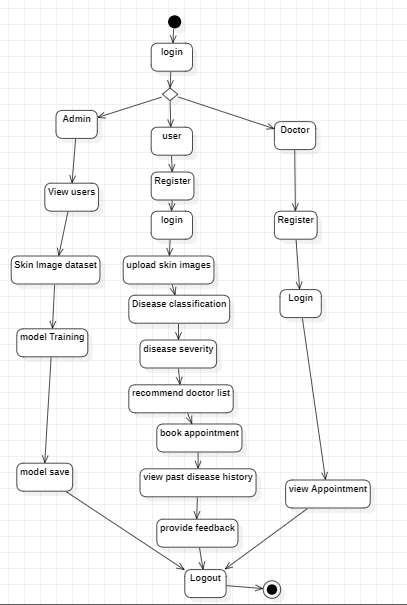
### **Deployment Diagram**

**Figure 6 Sequence Diagram**

Deployment diagram represents the deployment view of a system. It is related to the component diagram. Because the components are deployed using the deployment diagrams. A deployment diagram consists of nodes. Nodes are nothing but physical hardware’s used to deploy the application.

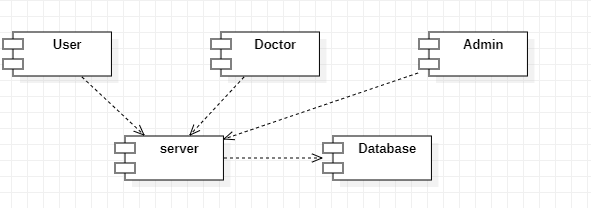
**Figure 7 Deployment Diagram**

### **Activity Diagram**

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

**Figure 8 Activity Diagram**

### **Component Diagram**

A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical **c**omponents in a system. Component diagrams are often drawn to help model implementation details and double-check that every aspect of the system's required function is covered by planned development.

**Figure 9 Component Diagram**

### **ER Diagram**

An Entity–relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram). An ER model is a design or blueprint of a database that can later be implemented as a database. The main components of E-R model are: entity set and relationship set.

An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database. Let’s have a look at a simple ER diagram to understand this concept.

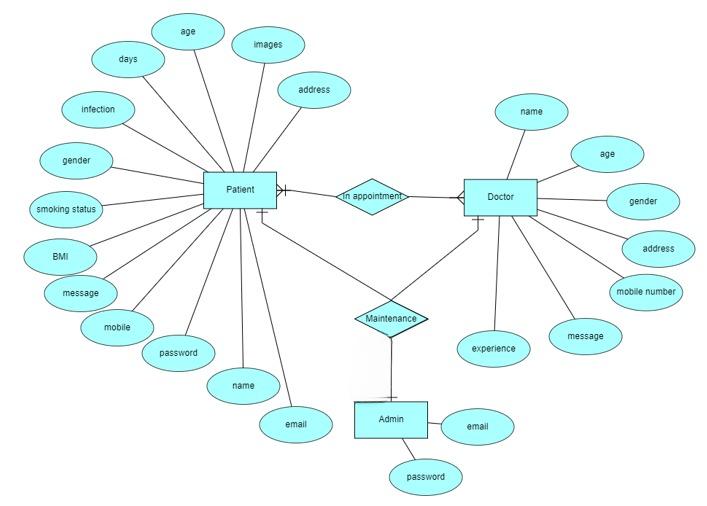
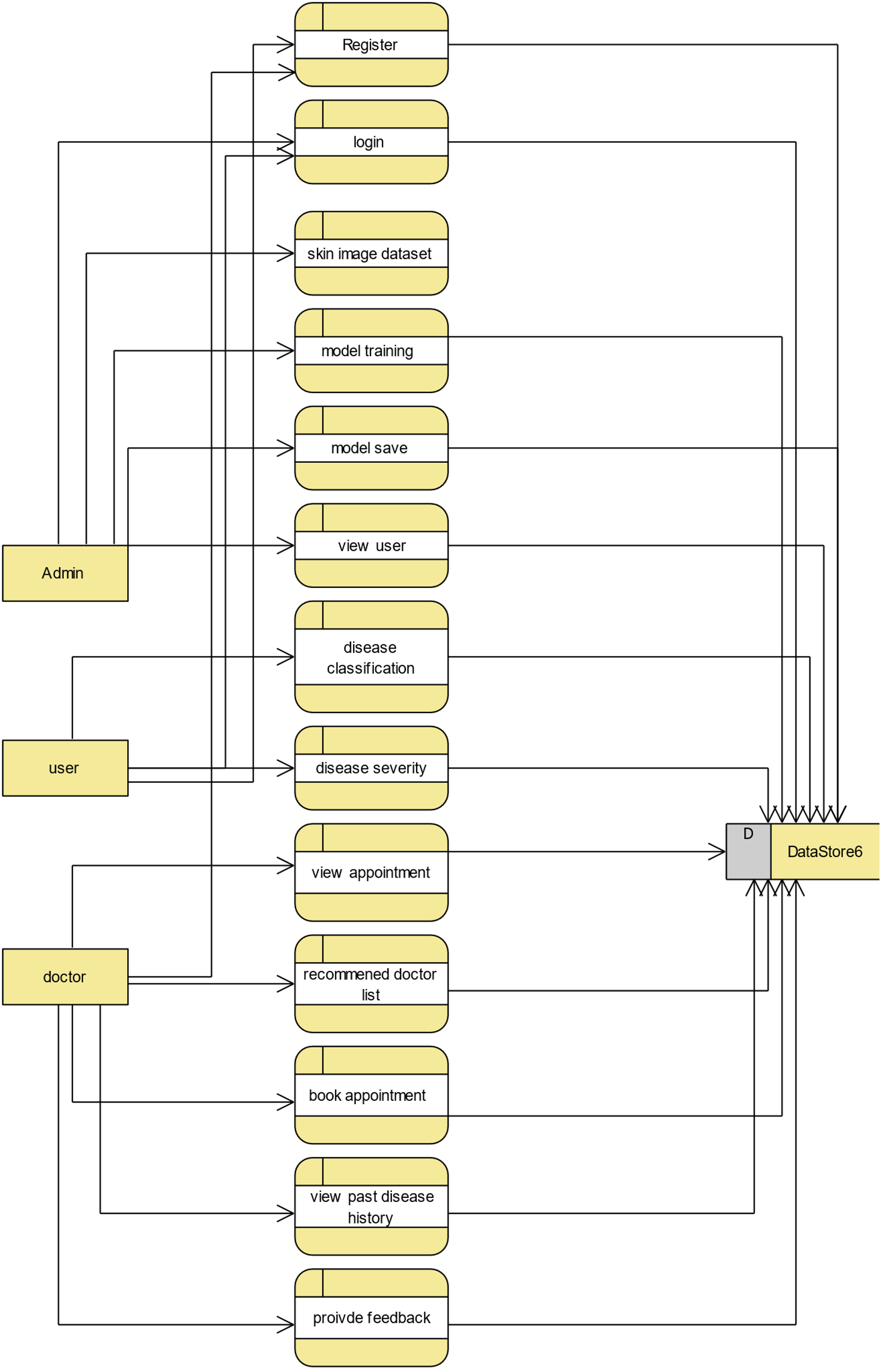
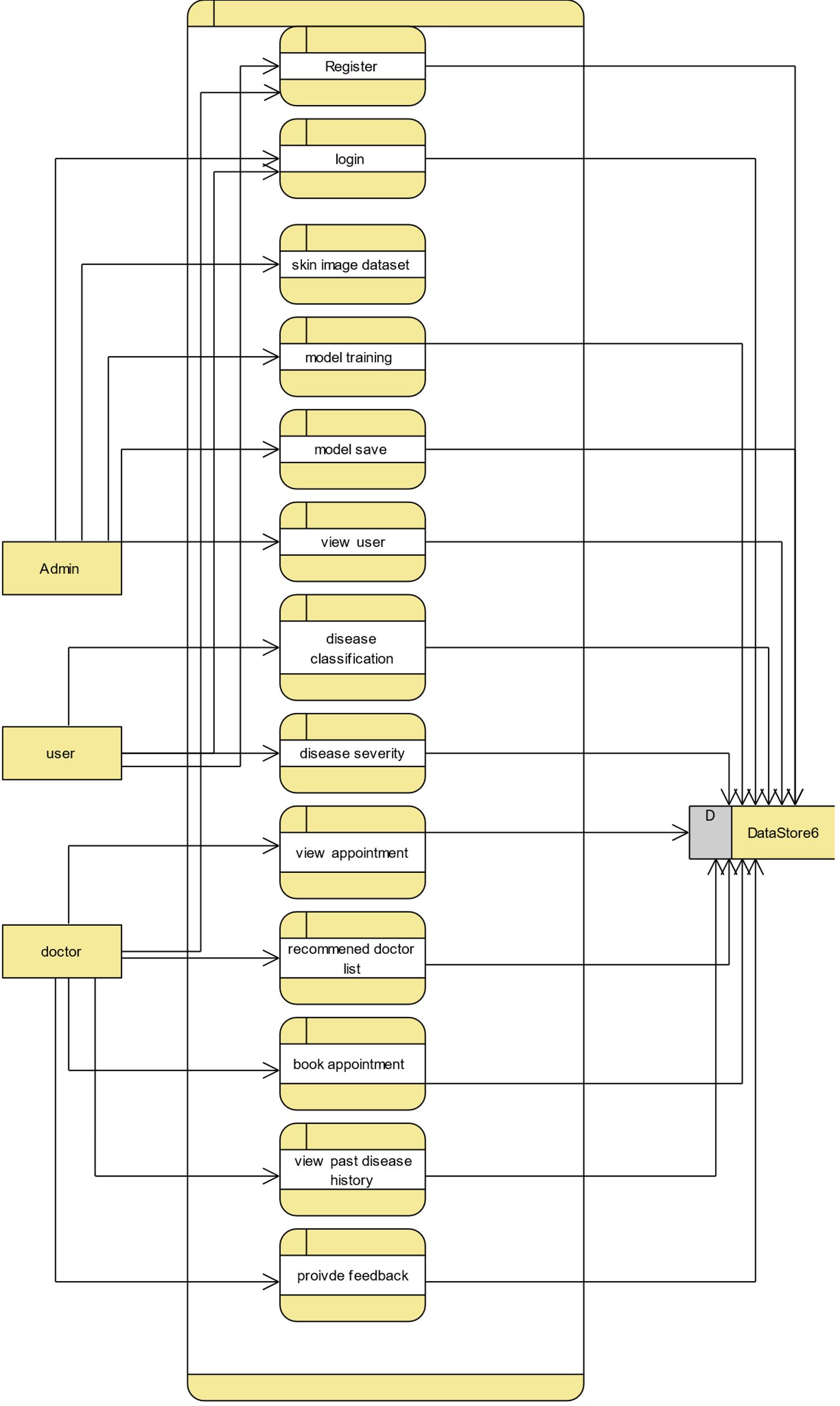


Figure 10 ER Diagram

**6.3 DFD Diagram**

A Data Flow Diagram (DFD) is a traditional way to visualize the information flows within a system. A neat and clear DFD can depict a good amount of the system requirements graphically. It can be manual, automated, or a combination of both. It shows how information enters and leaves the system, what changes the information and where information is stored. The purpose of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communications tool between a systems analyst and any person who plays a part in the system that acts as the starting point for redesigning a system.





1. **IMPLEMENTATION AND RESULTS**

**7.1 Modules**

* **System**
* **Patient**
* **Doctor**
* **Admin**
  1. **System:**
* **Collecting data:**

It will collect the data what the user is providing

* **Pre-processing:**

After collecting data we will pre-process the data by fixing the size of the images

* **Splitting:**

The data will be split into train (70%-80%) and test (30%-20%) after pre-processing the data.

* **Training:**

After splitting the data will be trained with different deep learning algorithms for the classification.

* **Classification:**

Classifying the images which is taken as input and produce the output.

* 1. **Patient:**
* **Register:**

Patient need to register with the valid details of them.

* **Login:**

Patient need to login with details what they have provided at registration.

* **Classification:**

Patient will give the required information and will upload the data as per that we will classify the result and it will show thw classification result.

* **Booking appointment & Chatting:**

Booking appointment as per the classification and if needed patient can chat with the doctor based on the problem.

* **Feedback:**
* Patient can provide the feedback on the doctor after completion their treatment.
  1. **Doctor:**
* **Register:**

Doctor need to register with the valid details of them like name, mail, Hospital name, Experience of them etc.,

* **Login:**

Patient need to login with details what they have provided at registration.

* **View Appointments:**

Doctor can view the appointments related.

* **Accept Appointments:**

As per their schedules they can adjust and accpept the appointments and after that it will reflected to patient.

* 1. **Admin:**
* **Login:**

Admin can login by using the details

* **Dash Board:**

Admin can view the dash board with different graphs like confusion matrices of the algorithms.

* **View registrations:**

Admin can view the registrations it may be the doctor it may be the patient.

* **View feedback:**

Admin can view the feedback of the patient on the doctor.

**7.2 Results**

The results and discussion for this project, focusing on the implementation of a hybrid MobileNet and LSTM model for dermatological image classification, reveal several critical insights and areas for further exploration.

Firstly, the gradual improvement in the model’s accuracy over the training epochs indicates that the hybrid approach is capable of learning and distinguishing between different skin conditions. However, the moderate level of accuracy achieved (just over 50% by the 50th epoch) suggests that the model's current configuration and training regimen may not fully capture the complexities of dermatological diagnostics. The low precision and recall values throughout the training further underscore this point, indicating a tendency of the model to generate false positives and negatives.

The high sensitivity at specificity and specificity at sensitivity metrics suggest that the model is relatively successful in differentiating positive and negative cases when clear-cut, but struggles with more nuanced or borderline cases. This could be due to the inherent limitations in the training dataset, such as a lack of diversity in skin types or conditions, or due to the model architecture itself.

The disparity between training and validation performance raises concerns about the model's ability to generalize to new, unseen data, hinting at potential overfitting. This could be addressed by incorporating more robust data augmentation techniques, using a more diverse validation set, or applying regularization methods.

In conclusion, while the hybrid MobileNet and LSTM model shows promise in classifying dermatological images, the results highlight the need for further refinement in both the model architecture and the training process. Future work should focus on enhancing the model's ability to handle the diverse and complex nature of skin diseases, ensuring more accurate and reliable diagnostic outcomes.

## **8. SYSTEM STUDY AND TESTING**

## **8.1 Feasibility Study**

The feasibility of the project is analysed 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 involved in the feasibility analysis are

* ECONOMICAL FEASIBILITY
* TECHNICAL FEASIBILITY
* SOCIAL FEASIBILITY

**Economic 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.

**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.

**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.

**8.2 SYSTEM TESTING**

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, sub assemblies, 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. It is 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.

1. **CONCLUSION**

The conclusion of this project, centered around the development of a hybrid MobileNet and LSTM model for dermatological image classification, encapsulates both the achievements and challenges encountered in this innovative endeavor. The project aimed to leverage the strengths of convolutional neural networks and recurrent neural networks to create a powerful tool for diagnosing skin conditions, and while it demonstrated potential, the results also highlighted key areas for improvement.

The model's ability to progressively learn and improve its accuracy over training epochs is a testament to the viability of combining MobileNet and LSTM for image classification tasks. This hybrid approach capitalizes on MobileNet's efficiency in processing spatial features and LSTM's prowess in handling sequential data, making it well-suited for the nuanced task of dermatological diagnostics.

However, the project also encountered challenges, most notably in the model's moderate accuracy and its lower precision and recall values. These issues underscore the complexity of skin disease classification and the need for a more robust training dataset that encompasses a greater diversity of skin types, conditions, and stages of diseases. Additionally, the model's performance discrepancy between training and validation phases indicates a tendency towards overfitting, suggesting a need for better generalization techniques.

In conclusion, this project lays the groundwork for future research in AI-driven dermatological diagnostics. It underscores the potential of hybrid AI models in healthcare but also highlights the critical importance of comprehensive data and advanced training methodologies. Future efforts should focus on refining the model to better capture the diverse manifestations of skin conditions, ultimately leading to a tool that is both highly accurate and reliable in a clinical setting.

**10. FUTURE SCOPE**

The future scope of this project, focusing on AI-driven dermatological diagnostics using a hybrid MobileNet and LSTM model, is both promising and expansive. One key area for development is enhancing the model's accuracy and generalizability. This can be achieved by expanding the training dataset with more diverse images, representing a wider range of skin conditions, types, and ethnicities. Implementing advanced data augmentation techniques can also improve the model's ability to handle real-world variability in skin images.

Another potential direction is the integration of additional data types, such as patient history or demographic information, to complement the image-based diagnosis. This could lead to more personalized and accurate assessments.

Furthermore, exploring the deployment of this model in telemedicine and mobile health applications could significantly broaden access to dermatological care, particularly in underserved regions. Finally, continuous advancements in AI and machine learning provide opportunities to further refine the model, exploring newer architectures and training techniques to enhance performance and efficiency. These efforts will collectively drive the evolution of AI in dermatology, making it an indispensable tool in modern healthcare.

**11.** **REFERENCES**

1. Al-Masni, M. A., Al-Antari, M. A., Choi, M.-T., Han, S.-M., & Kim, T.-S. (2018). Skinlesion segmentation in dermoscopy images via deep full resolution convolutionalnetworks. Computer Methods and Programs in Biomedicine, 162, 221–231.
2. Alfed, N., Khelifi, F., Bouridane, A., & Seker, H. (2015). Pigment network-based skincancer detection. In 2015 37th annual international conference of the IEEE engineeringin medicine and biology society (EMBC) (pp. 7214–7217). IEEE.
3. Aljanabi, M., Özok, Y. E., Rahebi, J., & Abdullah, A. S. (2018). Skin lesion segmentationmethod for dermoscopy images using artificial bee colony algorithm. Symmetry,10(8), 347.
4. Biessmann, F., Salinas, D., Schelter, S., Schmidt, P., & Lange, D. (2018). ‘‘ Deep" learningfor missing value imputationin tables with non-numerical data. In Proceedings ofthe 27th ACM international conference on information and knowledge management (pp.2017–2025).
5. Canadian Cancer Society’s Advisory Committee on Cancer Statistics (2014). Canadiancancer statistics 2014 - special topic: Skin cancers. Retrieved from https://www.cancer.ca/statistics. (Accessed 30 December 2020).
6. Carcagnì, P., Leo, M., Cuna, A., Mazzeo, P. L., Spagnolo, P., Celeste, G., et al.(2019). Classification of skin lesions by combining multilevel learnings in adensenet architecture. In International conference on image analysis and processing (pp. 335–344). Springer.
7. Dorj, U.-O., Lee, K.-K., Choi, J.-Y., & Lee, M. (2018). The skin cancer classificationusing deep convolutional neural network. Multimedia Tools and Applications, 77(8),9909–9924.
8. Ech-Cherif, A., Misbhauddin, M., & Ech-Cherif, M. (2019). Deep neural network basedmobile dermoscopy application for triaging skin cancer detection. In 2019 2ndinternational conference on computer applications & information security (ICCAIS) (pp.1–6). IEEE.
9. Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., et al. (2017).Dermatologist-level classification of skin cancer with deep neural networks. Nature,542(7639), 115–118.
10. Garg, R., Maheshwari, S., & Shukla, A. (2019). Decision support system for detectionand classification of skin cancer using CNN. arXiv preprint arXiv:1912.03798.
11. Guan, Q., Wang, Y., Ping, B., Li, D., Du, J., Qin, Y., et al. (2019). Deep convolutionalneural network VGG-16 model for differential diagnosing of papillary thyroid carcinomas in cytological images: a pilot study. Journal of Cancer, 10(20), 4876.
12. Han, X., Zhong, Y., Cao, L., & Zhang, L. (2017). Pre-trained alexnet architecture with pyramid pooling and supervision for high spatial resolution remote sensing imagescene classification. Remote Sensing, 9(8), 848.
13. Hasan, M., Barman, S. D., Islam, S., & Reza, A. W. (2019). Skin cancer detection usingconvolutional neural network. In Proceedings of the 2019 5th international conferenceon computing and artificial intelligence (pp. 254–258).
14. Hobbs, L., Hillson, S., Lawande, S., & Smith, P. (2011). Oracle 10g data warehousing.Elsevier.
15. Hoshyar, A. N., Al-Jumaily, A., & Hoshyar, A. N. (2014). The beneficial techniques inpreprocessing step of skin cancer detection system comparing. Procedia ComputerScience, 42, 25–31.
16. Hosny, K. M., Kassem, M. A., & Foaud, M. M. (2018). Skin cancer classification using deep learning and transfer learning. In 2018 9th Cairo international biomedical engineering conference (CIBEC) (pp. 90–93). IEEE.
17. Jafari, M. H., Karimi, N., Nasr-Esfahani, E., Samavi, S., Soroushmehr, S. M. R., Ward, K.,et al. (2016). Skin lesion segmentation in clinical images using deep learning. In 2016 23rd international conference on pattern recognition (ICPR) (pp. 337–342). IEEE.
18. Jeong, M. K., Lu, J.-C., Huo, X., Vidakovic, B., & Chen, D. (2006). Wavelet-based datareduction techniques for process fault detection. Technometrics, 48(1), 26–40.
19. Kadampur, M. A., & Al Riyaee, S. (2020). Skin cancer detection: applying a deeplearning based model driven architecture in the cloud for classifying dermal cellimages. Informatics in Medicine Unlocked, 18, Article 100282.
20. Le, D. N., Le, H. X., Ngo, L. T., & Ngo, H. T. (2020). Transfer learning with classweighted and focal loss function for automatic skin cancer classification. arXiv preprint arXiv:2009.05977.