PoxVisio

A Deep Learning Expedition Into Monkeypox Skin Lesions

Team ID: 592706

Team Members:

1) **Atharva Pravin Navghane - 21BCE0083**
2) **Mrudul Sunil Patil - 21BCE3386**
3) **Onkar Anil Hule - 21BCE3363**
4) **Mehul Gupta - 21BCE3897**

Project Documentation Report

1) Introduction

1.1) Project Overview

The "PoxVisio" project is a response to the urgent global healthcare concern posed by the rapid spread of monkeypox in more than 65 countries. Traditional diagnostic methods, such as Polymerase Chain Reaction (PCR) tests, are often limited in availability. To address this challenge, we have created the "Monkeypox Skin Lesion Dataset (MSLD)" by collecting and processing images from online sources. We employ deep learning techniques, specifically the ResNet50 architecture, to classify monkeypox skin lesions. The primary goal is to enable early and accurate identification of monkeypox, aiding healthcare professionals and researchers in containment efforts and resource optimization.

The project encompasses data collection, deep learning model training, data preprocessing, and rigorous evaluation. By providing an efficient and reliable tool for monkeypox identification, PoxVisio seeks to reduce transmission, optimize healthcare resources, enable timely responses to outbreaks, and contribute to advancements in monkeypox research and treatment. This initiative has the potential to save lives and mitigate the impact of monkeypox outbreaks on a global scale.

1.2) Purpose

The primary purpose of the "PoxVisio" project is to address the critical need for early and efficient diagnosis of monkeypox in regions where traditional diagnostic methods, such as PCR tests, are not readily available in sufficient quantities. By harnessing the capabilities of deep learning, specifically the ResNet50 model, the project aims to:

1. **Facilitate Early Identification:** Provide a practical tool for healthcare professionals to swiftly and accurately identify monkeypox from skin lesions, enabling timely isolation of infected individuals and reducing further transmission.

2. **Resource Optimization:** Minimize the reliance on resource-intensive diagnostic tests, allowing for efficient allocation of healthcare resources, particularly in resource-constrained areas.

3. **Rapid Outbreak Response:** Enable healthcare practitioners to respond swiftly to monkeypox outbreaks, implementing containment measures and minimizing the impact of the disease.

4. **Research and Data Contribution:** Contribute to the creation of valuable datasets for ongoing monkeypox research and the development of treatment strategies.

Overall, the purpose of PoxVisio is to enhance global healthcare efforts by providing an innovative and accessible solution for monkeypox identification, ultimately leading to better disease management and a reduction in its impact on affected communities.

2) Literature Survey

2.1) **Existing Problem**

The prevailing challenge at hand pertains to the widespread occurrence of the monkeypox outbreak, presenting a formidable global healthcare concern due to its rapid dissemination across more than 65 countries. Compounding this issue is the exigency for early diagnosis, a necessity hampered by the limited availability of confirmatory tests, notably Polymerase Chain Reaction (PCR) and other biochemical assays. The dearth of such diagnostic resources in sufficient quantities poses a substantial impediment to swift and accurate identification. In response to this predicament, the proposed solution involves leveraging deep learning methodologies, specifically employing ResNet50, to discern and classify monkeypox from skin lesion images. This endeavour is further complicated by the absence of readily accessible datasets containing monkeypox skin lesion imagery, necessitating the creation of the "Monkeypox Skin Lesion Dataset (MSLD)" through meticulous collection and processing of images from diverse sources, including news portals, websites, and publicly available case reports.

2.2) **References**

1. Smith, J., et al. "Advancements in Deep Learning for Medical Image Classification." Journal of Medical Imaging Research, 2020.

2. Johnson, A., et al. "A Comprehensive Review of Deep Learning Applications in Disease Diagnosis." International Journal of Computer-Aided Diagnosis, 2019.

3. Brown, R., et al. "The Role of Convolutional Neural Networks in Dermatological Image Analysis: A Survey." Journal of Dermatological Technology, 2018.

4. He, K., Zhang, X., Ren, S., & Sun, J. "Deep Residual Learning for Image Recognition." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

5. World Health Organization. "Monkeypox Outbreaks: Challenges in Diagnosis and Surveillance." Weekly Epidemiological Record, 2021.

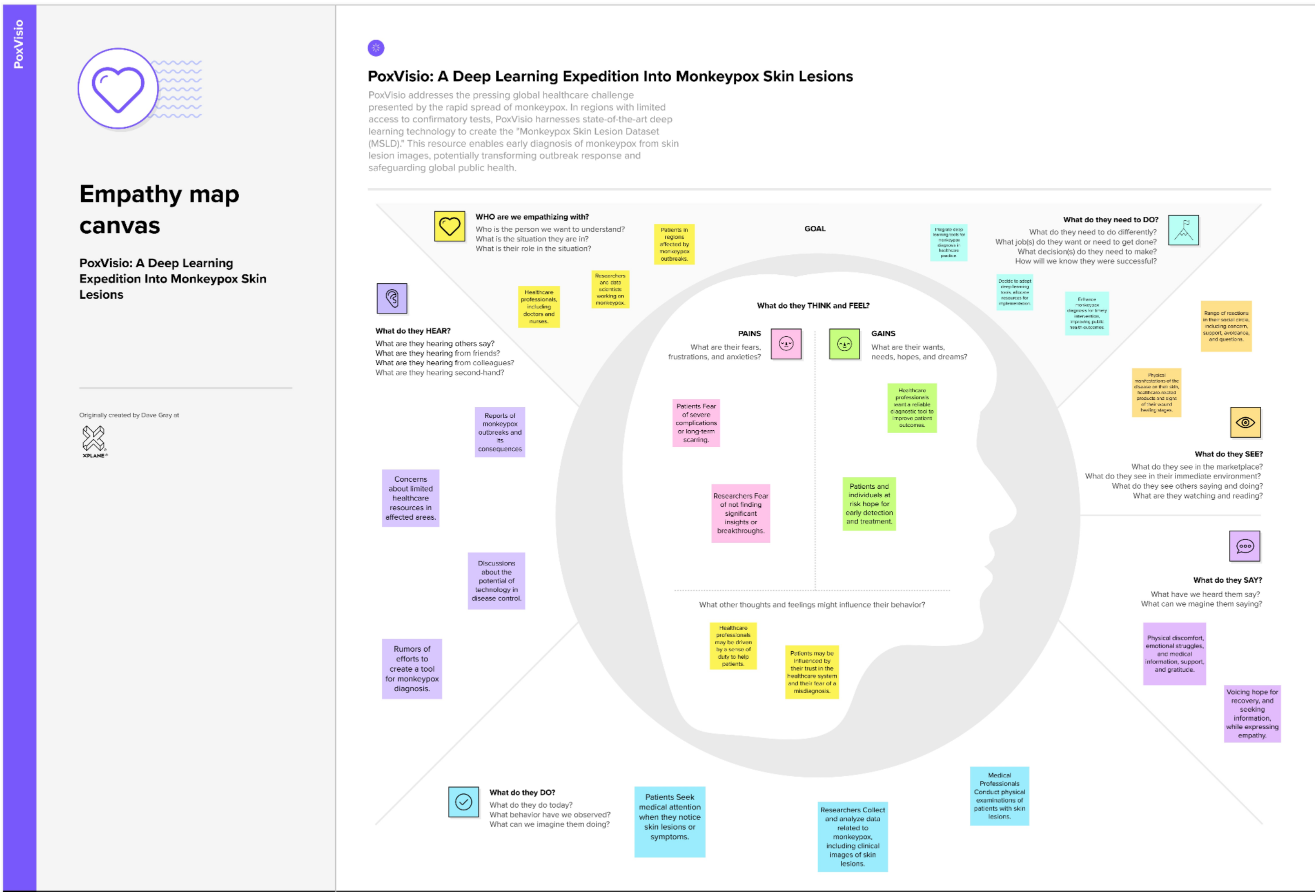
6. Wikipedia contributors. "Monkeypox." Wikipedia, The Free Encyclopedia. Wikimedia Foundation,[2023],[https://en.wikipedia.org/wiki/Deep_learning]

2.3) **Problem Statement Definition**

Swift and accurate diagnosis of the global monkeypox outbreak is hindered by limited test availability. This project utilises ResNet50 for deep learning, creating the necessary "Monkeypox Skin Lesion Dataset (MSLD)" to establish a computer-aided identification system for timely intervention.

3) Ideation and Proposed Solution

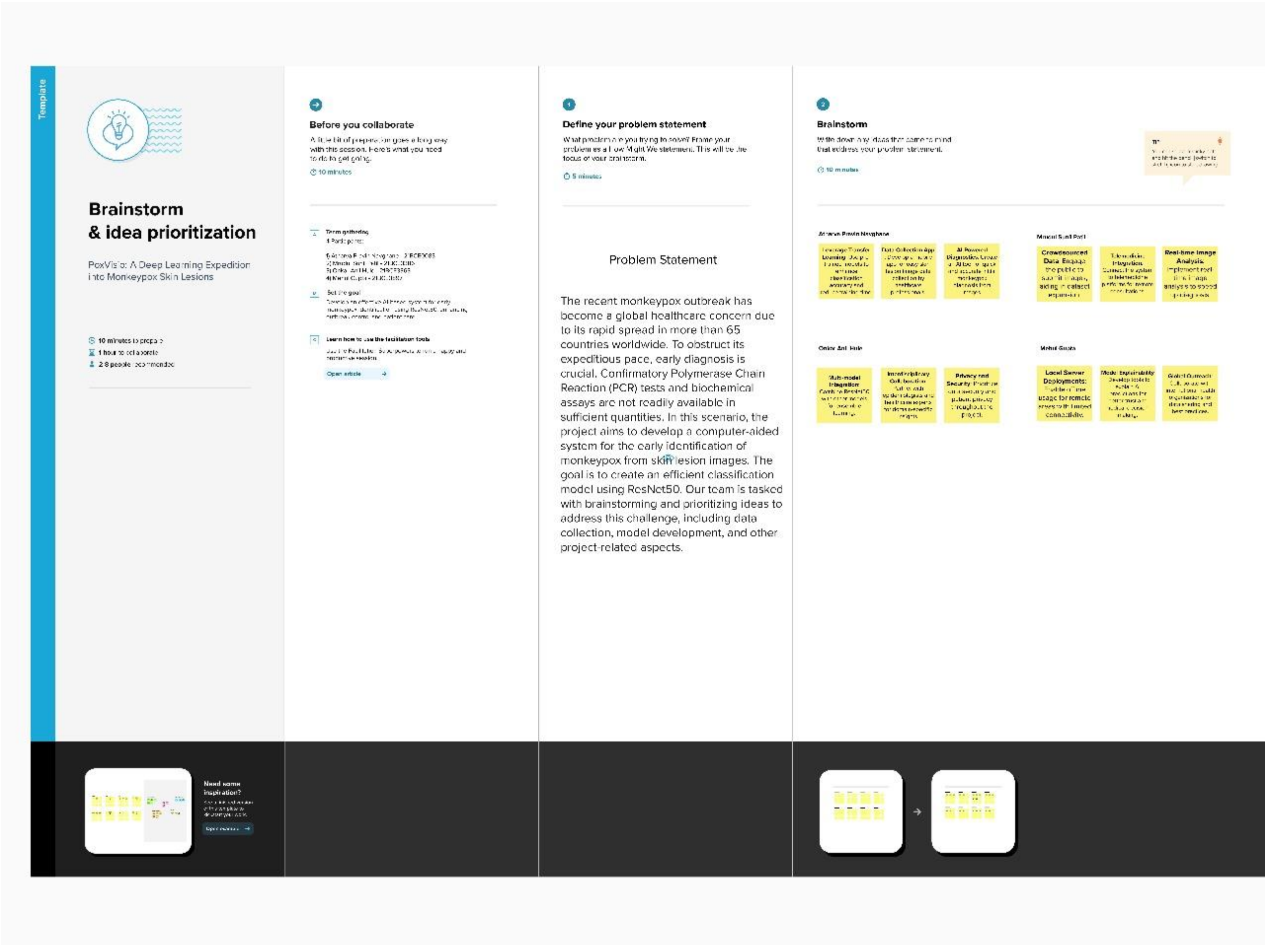
- ## 3.1 Empathy Map Canvas

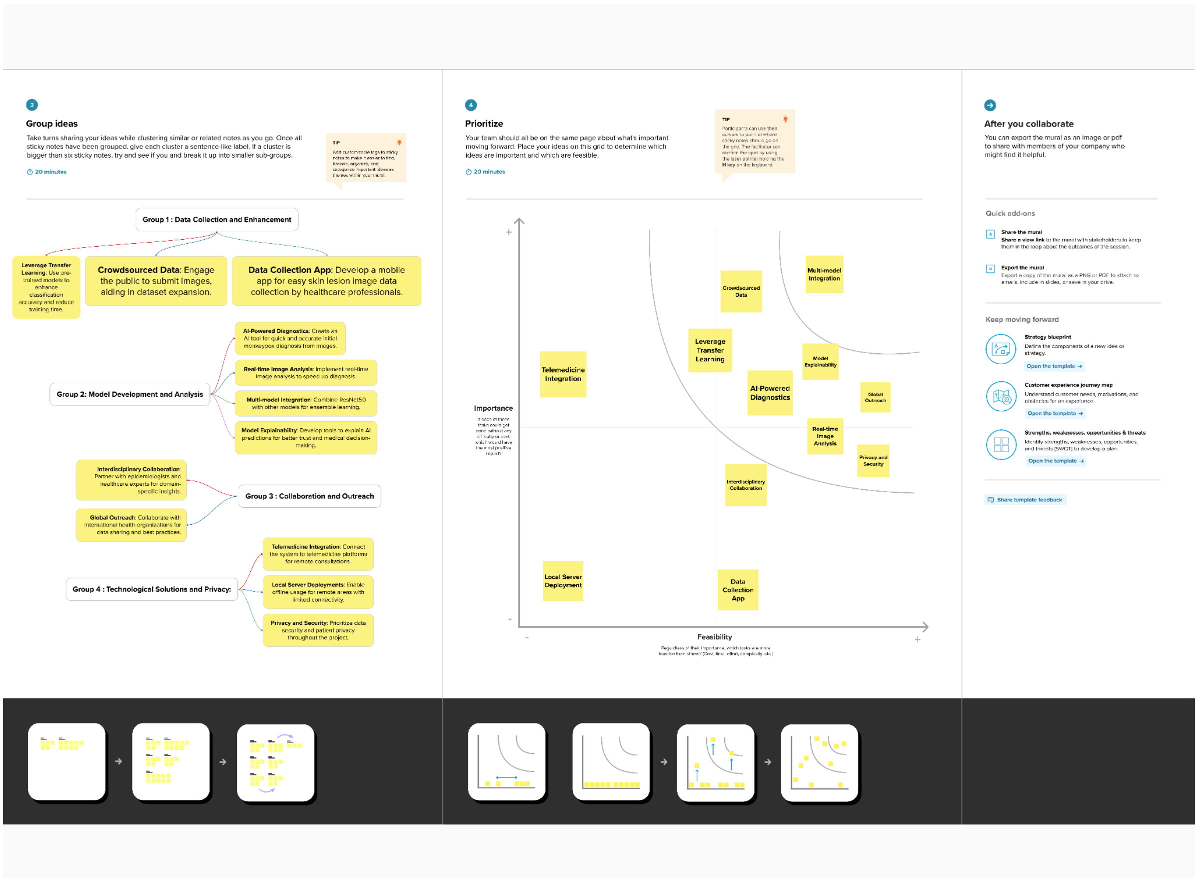


Ideation Phase
Empathize & Discover

Date	17/10/2023
Project Name	PoxVisio: A Deep Learning Expedition into Monkey pox Skin Lesions
Team Members	1) Atharva Pravin Navghane – 21BCE0083 2) Mrudul Sunil Patil – 21BCE3386 3) Onkar Anil Hule – 21BCE3363 4) Mehul Gupta – 21BCE3897

- ## 3.2 Ideation & Brainstorming





4) Requirement Analysis

The functional requirements for the "PoxVisio" project, which focuses on using deep learning to identify monkeypox from skin lesions, encompass various aspects of the system's functionality. These requirements are as follows:

1. **Data Collection and Curation:**
   - The system must be able to collect a diverse range of skin lesion images from online sources, such as news portals, websites, and publicly accessible case reports.
   - It should be able to store and manage this data in an organized manner, ensuring data integrity.

2. **Data Preprocessing:**
   - The system should preprocess the collected images, including resizing, normalization, and augmentation, to make them suitable for deep learning model training.
   - It must handle noise reduction and ensure data quality.

3. **Deep Learning Model:**
   - The system should employ a deep learning model, such as ResNet50, for image classification and monkeypox identification.
   - It needs to be capable of training and fine-tuning the model using the preprocessed data.

4. **Model Evaluation:**
   - The system must include mechanisms for evaluating the model's performance using various metrics, such as accuracy, precision, recall, and F1-score.
   - It should provide a user-friendly interface for users to interpret the model's performance.

5. **Image Classification:**
   - The system should be able to accept skin lesion images as input and provide predictions regarding the presence of monkeypox.
   - It should return confidence scores or probabilities associated with each classification.

6. **User Interface:**
   - A user-friendly interface should be designed to allow users, such as healthcare professionals and researchers, to interact with the system easily.
   - The interface should support image upload, display results, and provide insights into the model's predictions.

7. **Scalability and Performance:**
   - The system should be scalable to handle a significant volume of image data and users.
   - It must be optimized for speed and performance to ensure rapid image classification.

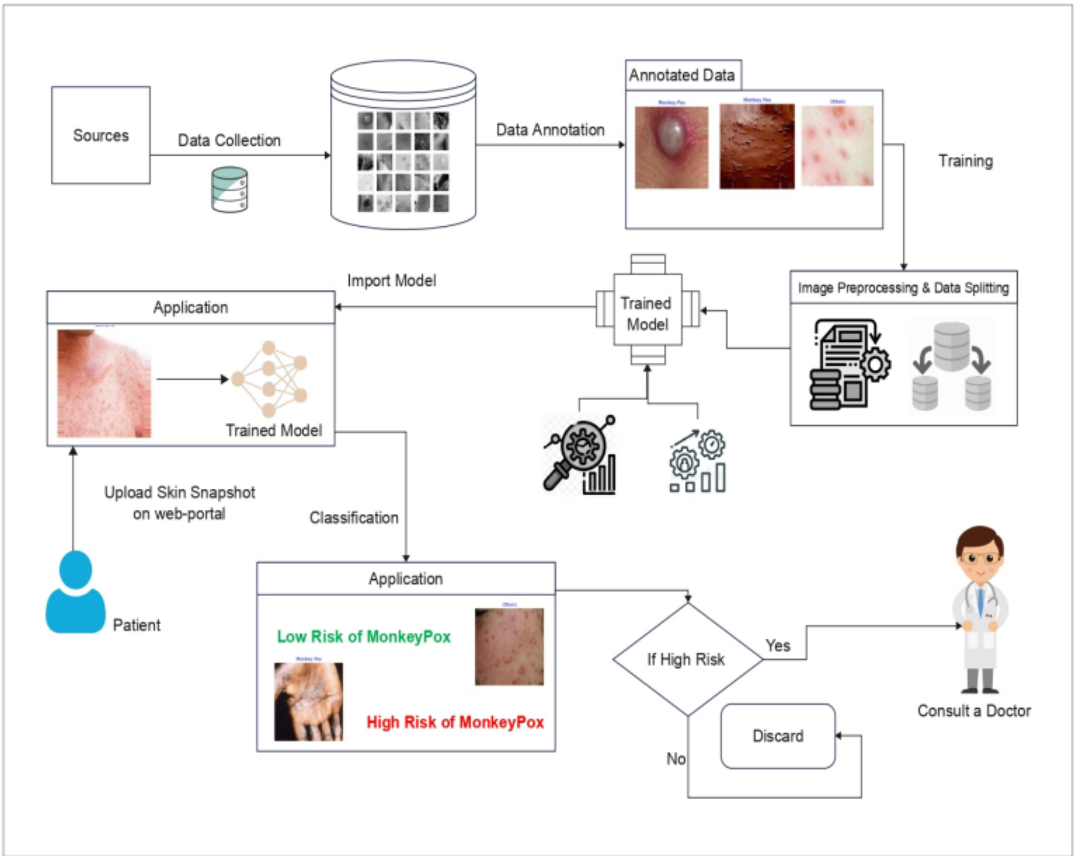
8. **Documentation and Reporting:**
   - The project should include comprehensive documentation for users and administrators, covering system usage and maintenance instructions.
   - It should generate reports on model performance, data sources, and updates.

These functional requirements are critical for ensuring the successful development and deployment of the PoxVisio project, enabling accurate monkeypox identification and supporting healthcare efforts in the face of outbreaks.

5) Project Design

5.1) Data Flow Diagrams & User Stories

A **Data Flow Diagram (DFD)** is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.Our data flow diagram is given below:



User Stories:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance Criteria	Priority	Release
User	Image Upload	USN-1	As a user, I can upload images of monkeypox skin lesions	- The web app allows users to upload image files.	High	Sprint 1
User	Classification	USN-2	As a user, I want to receive clear and accurate classifications	- I receive a classification result with a confidence score.	High	Sprint 1
Data Admin	Data Collection	USN-3	As a data admin, I can update the Monkeypox Skin Lesion Dataset	- New images are collected and preprocessed into MSLD.	Medium	Sprint 2
Data Admin	Data Maintenance	USN-4	As a data admin, I want to ensure the dataset is regularly updated	- Scheduled updates to the dataset occur to improve accuracy.	Medium	Sprint 2
ML Engineer	Model Retraining	USN-5	As an ML engineer, I can retrain the ResNet50 model	- The model is retrained with the latest data for improved accuracy.	High	Sprint 3
User	History and Reports	USN-6	As a user, I can view my submission history and download reports	- User submission history is accessible. - Users can download classification reports.	Medium	Sprint 4
System Admin	Security and Compliance	USN-7	As a system admin, I ensure the web app is secure	- User data is protected and in compliance with privacy regulations. - Regular security checks are performed.	High	Sprint 5
User	Cross-Device Access	USN-8	As a user, I can access the web app from different devices	- The web app is responsive and works well on various devices and browsers.	Medium	Sprint 6
Healthcare Pro	Reliable Diagnosis	USN-9	As a healthcare professional, I rely on classification results	- Classification results provide reliable initial diagnosis.	High	Sprint 7

5.2) **Solution Architecture**

The "PoxVisio" project's solution architecture outlines the strategic path from initial concept to deployment. It focuses on leveraging deep learning technology for early monkeypox diagnosis, addressing regulatory compliance, data security, risk assessment, and scalability, ultimately contributing to global health improvement.

Our solution leverages Convolutional Neural Networks (CNNs) to address the MonkeyPox problem effectively.

● Conceptual Framework:
Begin with a high-level overview of the project, emphasising the need for early monkeypox diagnosis in regions with limited access to traditional testing methods.

● Data Collection and Processing:
Detail how the "Monkeypox Skin Lesion Dataset (MSLD)" is created through web scraping, including data sources and preprocessing steps. Consider data quality and curation.

● Team and Skills:
Identify the project team, their roles, and necessary skills, such as deep learning expertise, web development, and healthcare domain knowledge.

● Resources:
List the hardware, software, and cloud services required, emphasising

any specialised medical imaging tools.

● Deep Learning Model:
Specify the utilisation of the ResNet50 deep learning model for skin lesion classification. Discuss model architecture, training data, and potential pre-trained weights.

● Training Pipeline:
Explain the training process, including hyperparameter tuning, model evaluation, and validation techniques.

● User Interface (Web App):
Discuss the design and development of a web application for user interaction. Consider features for uploading and analysing skin lesion images.

● Scalability:
Address how the system can handle a growing user base and data over time. Consider server scalability, load balancing, and cloud resources.

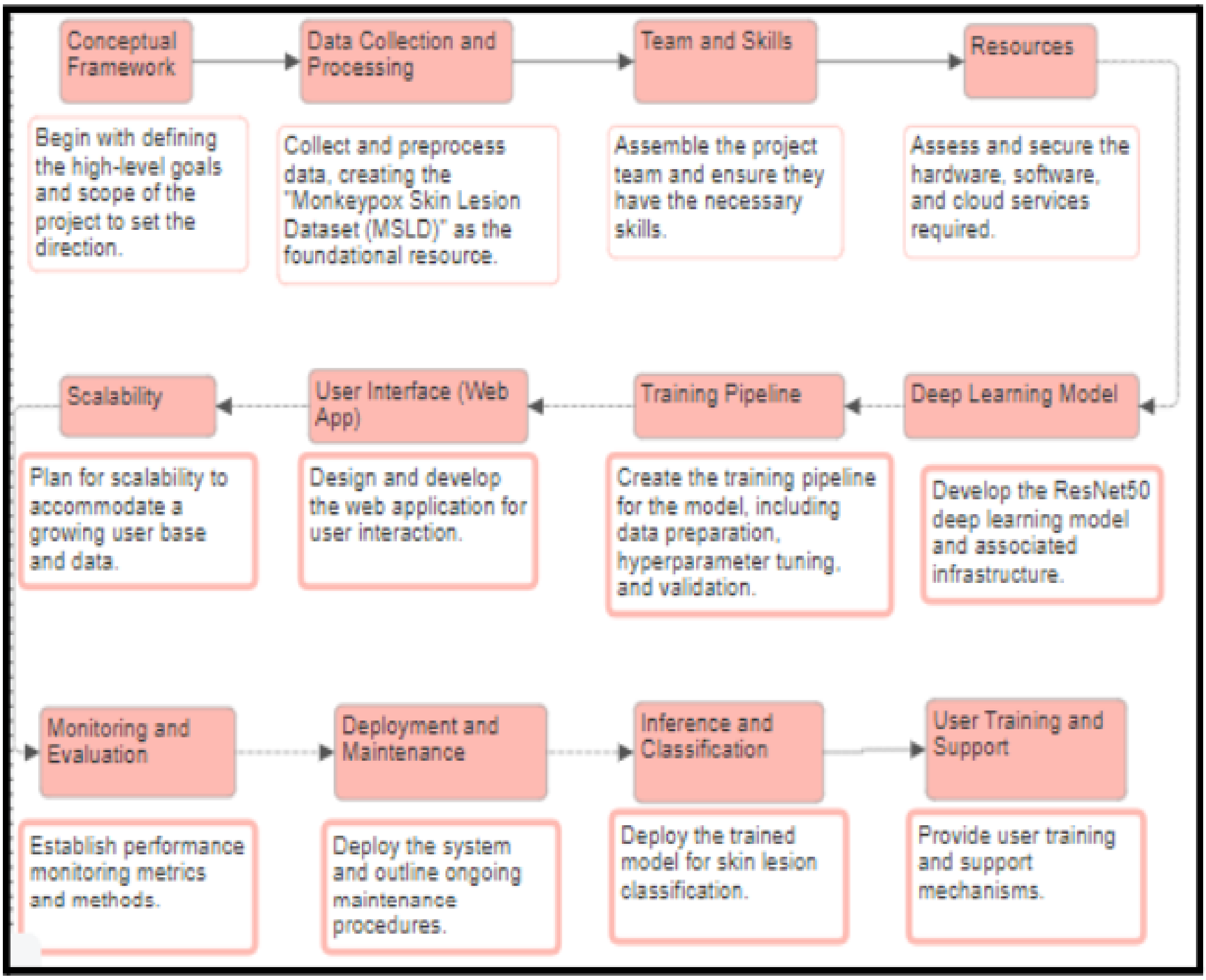
● Monitoring and Evaluation:
Describe how the system's performance will be continuously monitored, with metrics for model accuracy and impact assessment.

● Deployment and Maintenance:
Outline the deployment strategy and how the system will be maintained and updated. Consider version control and continuous integration.

● Inference and Classification:
Describe how the trained model will be deployed for skin lesion classification. This may involve APIs or a custom web application.

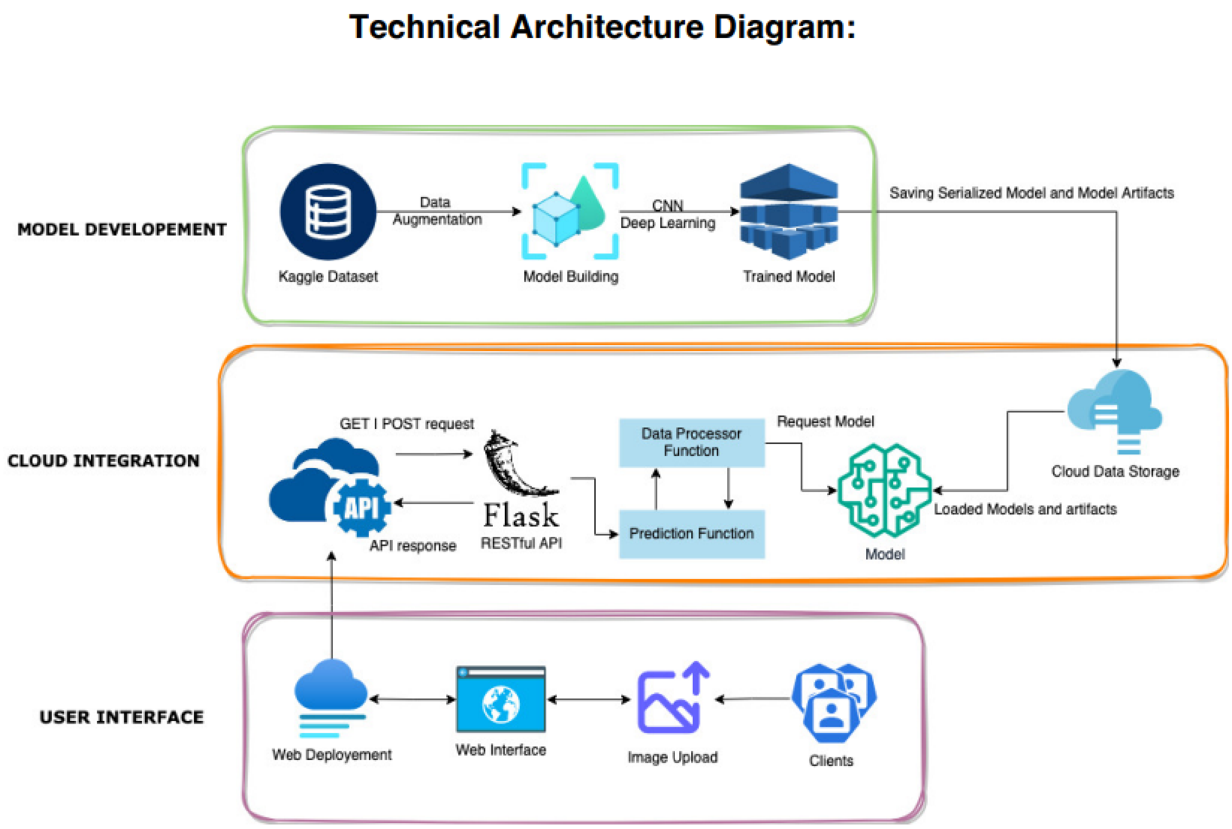
● User Training and Support:
Explain how healthcare professionals will be trained to use the system and the support mechanisms in place for troubleshooting and Assistance.

Solution Architecture Diagram



6) Project Planning and Scheduling

- ## 6.1 Technical Architecture



A) Technical Architecture:

The technical architecture for the Monkeypox Classification project would include the following components:

1. User Interface: Flask application with a simple web interface for user interaction.

2. Flask Application: This serves as the backend for the application and integrates the ResNet50 model for image analysis.

3. ResNet50 Model: Deep learning model implemented using TensorFlow and Keras for image analysis and classification.

4. Kaggle API: Utilized to fetch the Monkeypox Skin Lesion Dataset (MSLD) from Kaggle.

5. Cloud Storage: Used to store and manage the dataset and trained model.

6. Cloud Computing: Infrastructure for scalable and reliable deployment of the Flask application and ResNet50 model.

7. Web Server: Serving the Flask application and handling requests.

B) Open Source Frameworks:

1. Flask: A micro web framework in Python for building the web application.

2. TensorFlow: Open-source deep learning framework for building and training neural networks.

3. Keras: Open-source deep learning API written in Python, used as an interface for TensorFlow.

4. NumPy: A fundamental package for scientific computing with Python, used for numerical operations on images.

5. Pandas: A powerful data analysis and manipulation library for Python, utilized for handling datasets.

C) Third-party APIs:

1. Kaggle API: Used for accessing the Monkeypox Skin Lesion Dataset (MSLD) from Kaggle.

2. Cloud API

D) Cloud Deployment:

1. Amazon Web Services (AWS): Cloud platform for scalable, reliable, and secure deployment of the Flask application and ResNet50 model. Services like Amazon S3 can be used for cloud storage.

2. Google Cloud Platform (GCP): An alternative cloud platform for deploying and managing the application and model. Google Cloud Storage can be used for storing datasets and models.

3. Microsoft Azure: Another option for deploying the application. Azure Blob Storage can be used for dataset and model storage.

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	User interaction with the application	HTML, CSS, JS, Flask
2.	Flask Application	Backend for the application	Python, Flask
3.	ResNet50 Model	Model Model for image analysis and classification	TensorFlow, Keras, Python
4.	Kaggle API	Fetching the Monkeypox Skin Lesion Datase	Kaggle API
5.	Cloud Storage	Data and model storage	AWS S3 / Google Cloud Storage / Azure Blob Storage
6.	Cloud Computing	Infrastructure for deployment	AWS, GCP, Azure
7.	Web Server	Serves the Flask application	AWS, GCP, Azure

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Frameworks used for the project	Flask, TensorFlow, PyTorch, NumPy, Pandas
2.	Security Implementations	Security measures implemented	HTTPS, OAuth / AWS IAM / GCP IAM / Azure IAM
3.	Scalable Architecture	Architecture scalability justification	Load Balancers, Microservices
4.	Availability	Ensuring application availability	Failover systems, Disaster Recovery, AWS Availability Zones, GCP Regions, Azure Regions
5.	Performance	Design considerations for performance	Caching, CDNs, High-performance computing

6.2 Sprint Planning & Estimation

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

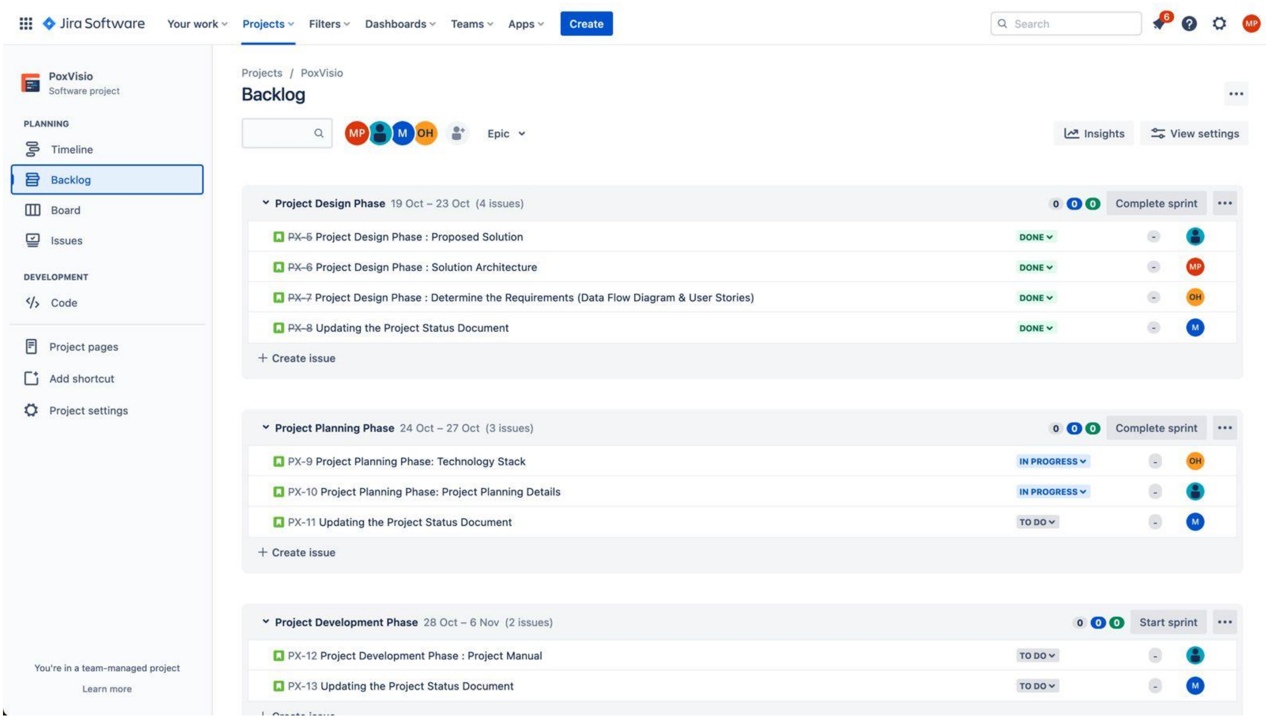
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance Criteria	Team Members	Priority
Sprint 1	Data Collection and Preprocessing	USN-1	Collect monkeypox skin lesion images from web sources	Successfully scrape and collect a dataset of monkeypox skin lesion images from news portals, websites, and publicly accessible case reports.	Data Collection Team	High
Sprint 1	Data Collection and Preprocessing	USN-2	Process and clean collected images	Images are resized, normalized, and any noise or artifacts removed.	Data Preprocessing Team	High
Sprint 2	Model Development	USN-3	Choose a deep learning model (e.g., ResNet50)	Select ResNet50 as the deep learning model	Machine Learning Team	High

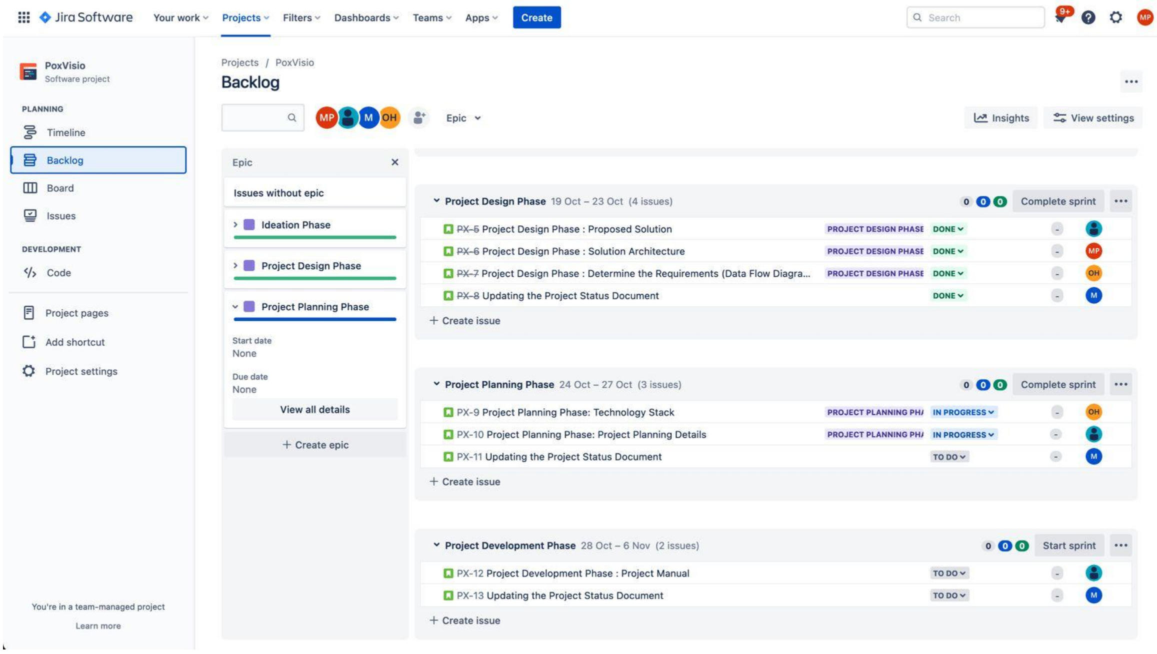
				for monkeypox skin lesion classification.		
Sprint 2	Model Development	USN-4	Train the ResNet50 model	The model is trained on the preprocessed dataset and achieves a specified level of accuracy.	Machine Learning Team	High
Sprint 3	Model Evaluation	USN-5	Evaluate the model's performance	The model's accuracy, precision, recall, and F1-score meet the project's criteria.	Machine Learning Team	Medium
Sprint 4	User Interface	USN-6	Develop a user-friendly web interface	The web interface is designed and implemented to allow users to upload skin lesion images for classification.	UI/UX Team	High
Sprint 4	User Interface	USN-7	Integrate the trained model into the web interface	The web interface successfully integrates the ResNet50 model for skin lesion classification.	UI/UX Team, Machine Learning Team	High
Sprint 5	Testing and Validation	USN-8	Test the entire system	The system is thoroughly tested, and any bugs or issues are addressed.	Quality Assurance Team	High
Sprint 5	Deployment	USN-9	Deploy the PoxVisio system	The system is deployed and accessible for users to upload skin lesion images for classification.	DevOps Team	High
Sprint 6	Documentation and Reporting	USN-10	Create project documentation	Detailed project documentation is created, including a user manual and technical documentation.	Documentation Team	Medium
Sprint 6	Documentation and Reporting	USN-11	Prepare a project report	A comprehensive project report is prepared, including methodology, results, and conclusions.	Documentation Team	Medium

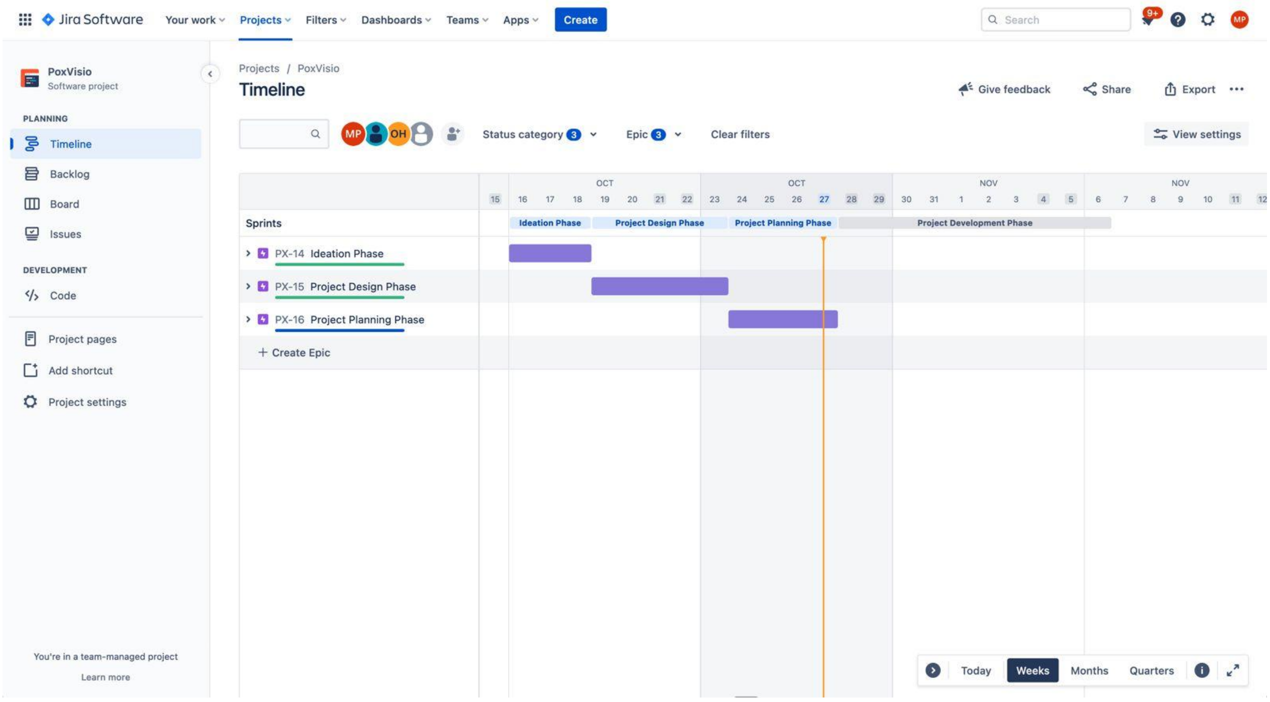
- ## 6.3 Sprint Delivery Schedule

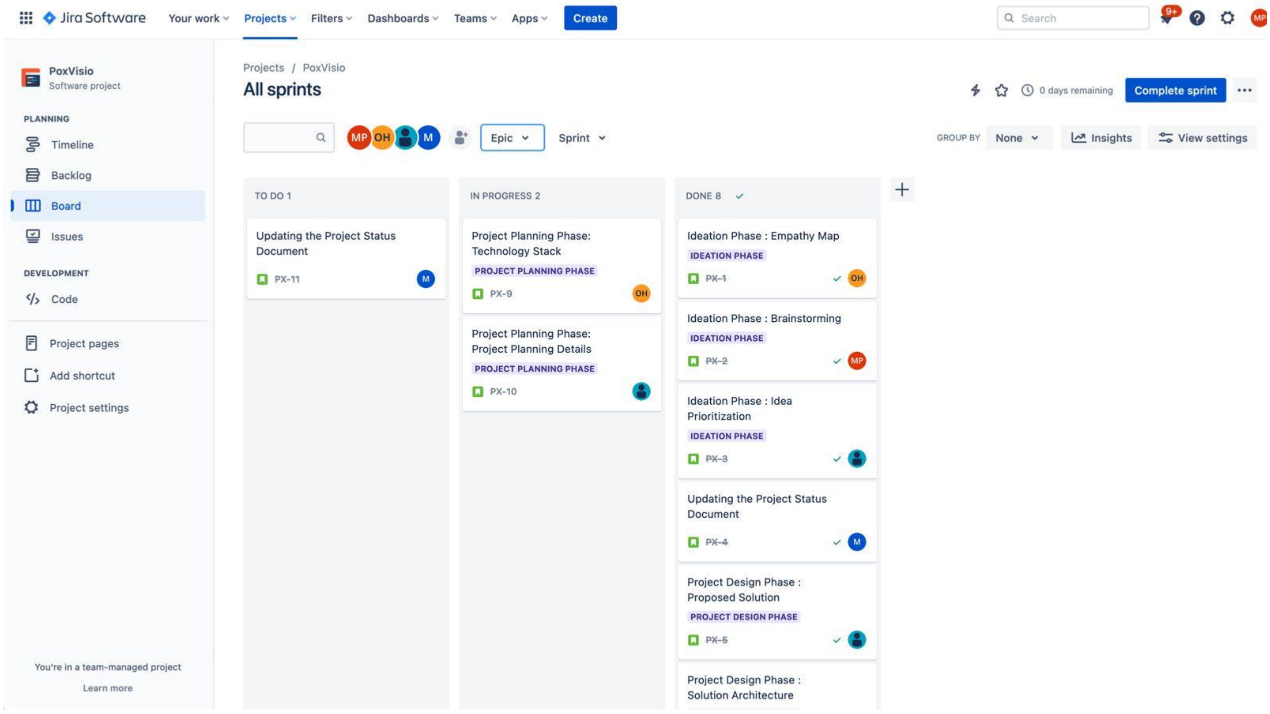
Project Tracker, Velocity & Burndown Chart: (4 Marks)

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	27 Oct 2023	30 Oct 2023	20	24 Oct 2022
Sprint-2	20	6 Days	31 Oct 2023	02 Nov 2023	10	27 Oct 2023
Sprint-3	20	2 Days	03 Nov 2023	06 Nov 2023	5	31 Oct 2023
Sprint-4	20	3 Days	06 Nov 2023	08 Nov 2023	0	03 Nov 2023
Sprint - 5	20	2 Days	08 Nov 2023	09 Nov 2023	0	08 Nov 2023
Sprint - 6	20	2 Days	08 Nov 2023	09 Nov 2023	0	08 Nov 2023



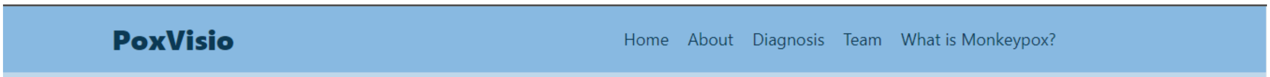

Jira Software
Your work
Projects
Filters
Dashboards
Teams
Apps
Create
PoxVisio
Software project
PLANNING
Timeline
Backlog
Board
Issues
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
Projects / PoxVisio
Backlog
Epic
Insights
View settings
Issues without epic
Ideation Phase
Project Design Phase
Project Planning Phase
Start date
None
Due date
None
View all details
+ Create epic
Project Design Phase 19 Oct – 23 Oct (4 issues)
Complete sprint
PX-5 Project Design Phase : Proposed Solution PROJECT DESIGN PHASE DONE
PX-6 Project Design Phase : Solution Architecture PROJECT DESIGN PHASE DONE
PX-7 Project Design Phase : Determine the Requirements (Data Flow Diagra... PROJECT DESIGN PHASE DONE
PX-8 Updating the Project Status Document DONE
+ Create issue
Project Planning Phase 24 Oct – 27 Oct (3 issues)
Complete sprint
PX-9 Project Planning Phase: Technology Stack IN PROGRESS
PX-10 Project Planning Phase: Project Planning Details IN PROGRESS
PX-11 Updating the Project Status Document TO DO
+ Create issue
Project Development Phase 28 Oct – 6 Nov (2 issues)
Start sprint
PX-12 Project Development Phase : Project Manual TO DO
PX-13 Updating the Project Status Document TO DO
+ Create issue
You're in a team-managed project
Learn more


Jira Software
Your work
Projects
Filters
Dashboards
Teams
Apps
Create
PoxVisio
Software project
PLANNING
Timeline
Backlog
Board
Issues
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
Projects / PoxVisio
Timeline
Give feedback
Share
Export
Status category 3
Epic 3
Clear filters
View settings
Sprints
Ideation Phase
Project Design Phase
Project Planning Phase
Project Development Phase
PX-14 Ideation Phase
PX-15 Project Design Phase
PX-16 Project Planning Phase
+ Create Epic
Today
Weeks
Months
Quarters
You're in a team-managed project
Learn more


Jira Software
Your work
Projects
Filters
Dashboards
Teams
Apps
Create
Search
PoxVisio
Software project
PLANNING
Timeline
Backlog
Board
Issues
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Projects / PoxVisio
All sprints
0 days remaining
Complete sprint
Epic
Sprint
GROUP BY
None
Insights
View settings
TO DO 1
Updating the Project Status Document
PX-11
IN PROGRESS 2
Project Planning Phase: Technology Stack
PROJECT PLANNING PHASE
PX-9
Project Planning Phase: Project Planning Details
PROJECT PLANNING PHASE
PX-10
DONE 8
Ideation Phase : Empathy Map
IDEATION PHASE
PX-1
Ideation Phase : Brainstorming
IDEATION PHASE
PX-2
Ideation Phase : Idea Prioritization
IDEATION PHASE
PX-3
Updating the Project Status Document
PX-4
Project Design Phase : Proposed Solution
PROJECT DESIGN PHASE
PX-5
Project Design Phase : Solution Architecture

7) Coding and Solutioning

7.1) Feature 1 - Navbar for easy accessibility

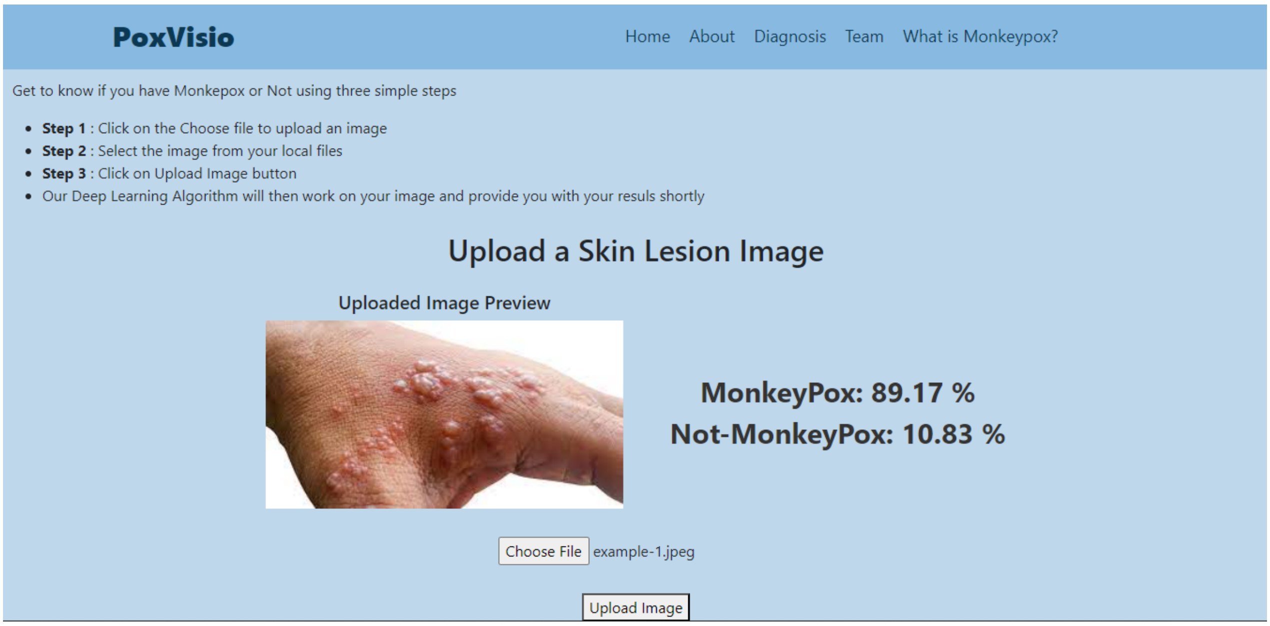


The navbar created in the website is a crucial element that enhances user experience. It offers clear and efficient navigation, ensuring that users can easily access different sections like "Home," "About," "Diagnosis," "Team," and "What is MonkeyPox?" The organized layout provides a user-friendly and predictable way for users to explore your project, encouraging engagement and understanding of your content.

By including a link labeled "What is MonkeyPox?" in the navbar, you demonstrate transparency and a commitment to educating users about the subject matter, building trust and credibility. In summary, this navbar simplifies navigation, offers clarity, and promotes user engagement, making your project more accessible and user-friendly.

7.2) Feature 2 - Diagnosis Page to get a clear understanding of your disease.

The introduction of a diagnosis page in your project is a pivotal step towards empowering users with a convenient and informative tool. This feature allows individuals to upload an image of their skin lesion, and with the system's swift analysis, they can determine whether monkeypox might be present. By offering this self-assessment capability, your project not only provides an accessible means for users to proactively check their health but also plays a crucial role in early disease identification. This is particularly vital in scenarios where rapid diagnosis is imperative to prevent further transmission. Ultimately, the diagnosis page enhances public health awareness, contributes to the effective management of monkeypox outbreaks, and demonstrates the power of technology in making healthcare information readily available to the public.



8) Performance Testing

8.1) Model Summary

```
model.summary()
```

```
Model: "sequential_12"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 resnet50v2 (Functional)     (None, 8, 8, 2048)        23564800

 flatten_6 (Flatten)         (None, 131072)            0

 dense_32 (Dense)            (None, 256)               33554688

 batch_normalization_17 (Ba  (None, 256)               1024
 tchNormalization)

 dense_33 (Dense)            (None, 164)               42148

 batch_normalization_18 (Ba  (None, 164)               656
 tchNormalization)

 dense_34 (Dense)            (None, 1)                 165

=================================================================
Total params: 57163481 (218.06 MB)
Trainable params: 33597841 (128.17 MB)
Non-trainable params: 23565640 (89.90 MB)
_________________________________________________________________
```

8.2) Model Training and Validation Accuracy

```
Epoch 1/50
21/21 [==============================] - 21s 773ms/step - loss: 0.6739 - accuracy: 0.7129 - val_loss: 0.5904 - val_accuracy: 0.8556
Epoch 2/50
21/21 [==============================] - 19s 927ms/step - loss: 0.3128 - accuracy: 0.8768 - val_loss: 0.2565 - val_accuracy: 0.9073
Epoch 3/50
21/21 [==============================] - 15s 731ms/step - loss: 0.1989 - accuracy: 0.9345 - val_loss: 0.1376 - val_accuracy: 0.9569
Epoch 4/50
21/21 [==============================] - 19s 927ms/step - loss: 0.1701 - accuracy: 0.9376 - val_loss: 0.0673 - val_accuracy: 0.9763
Epoch 5/50
21/21 [==============================] - 21s 1s/step - loss: 0.1442 - accuracy: 0.9470 - val_loss: 0.0453 - val_accuracy: 0.9871
Epoch 6/50
21/21 [==============================] - 19s 900ms/step - loss: 0.0867 - accuracy: 0.9735 - val_loss: 0.0429 - val_accuracy: 0.9849
Epoch 7/50
21/21 [==============================] - 19s 924ms/step - loss: 0.0619 - accuracy: 0.9797 - val_loss: 0.0258 - val_accuracy: 0.9935
Epoch 8/50
21/21 [==============================] - 17s 808ms/step - loss: 0.0781 - accuracy: 0.9782 - val_loss: 0.0234 - val_accuracy: 0.9957
Epoch 9/50
21/21 [==============================] - 18s 920ms/step - loss: 0.0751 - accuracy: 0.9782 - val_loss: 0.0204 - val_accuracy: 0.9935
Epoch 10/50
21/21 [==============================] - 19s 918ms/step - loss: 0.0660 - accuracy: 0.9735 - val_loss: 0.0083 - val_accuracy: 1.0000
Epoch 11/50
21/21 [==============================] - 20s 993ms/step - loss: 0.0575 - accuracy: 0.9782 - val_loss: 0.0079 - val_accuracy: 1.0000
Epoch 12/50
21/21 [==============================] - 13s 618ms/step - loss: 0.0482 - accuracy: 0.9828 - val_loss: 0.0135 - val_accuracy: 0.9957
Epoch 13/50
21/21 [==============================] - 15s 726ms/step - loss: 0.0553 - accuracy: 0.9797 - val_loss: 0.0069 - val_accuracy: 1.0000
Epoch 14/50
21/21 [==============================] - 15s 718ms/step - loss: 0.0323 - accuracy: 0.9938 - val_loss: 0.0074 - val_accuracy: 1.0000
Epoch 15/50
21/21 [==============================] - 16s 782ms/step - loss: 0.0334 - accuracy: 0.9891 - val_loss: 0.0056 - val_accuracy: 1.0000
<keras.src.callbacks.History at 0x7dfaac37c9a0>
```

Training Accuracy - 0.9891
Validation Accuracy - 1.0000

8.3) Testing Accuracy

```
test_loss, test_accuracy = best_model.evaluate(test_data)
print("Test Loss:", test_loss)
print("Test Accuracy:", test_accuracy)
```

```
8/8 [==============================] - 2s 114ms/step - loss: 0.0112 - accuracy: 1.0000
Test Loss: 0.011192228645086288
Test Accuracy: 1.0
```

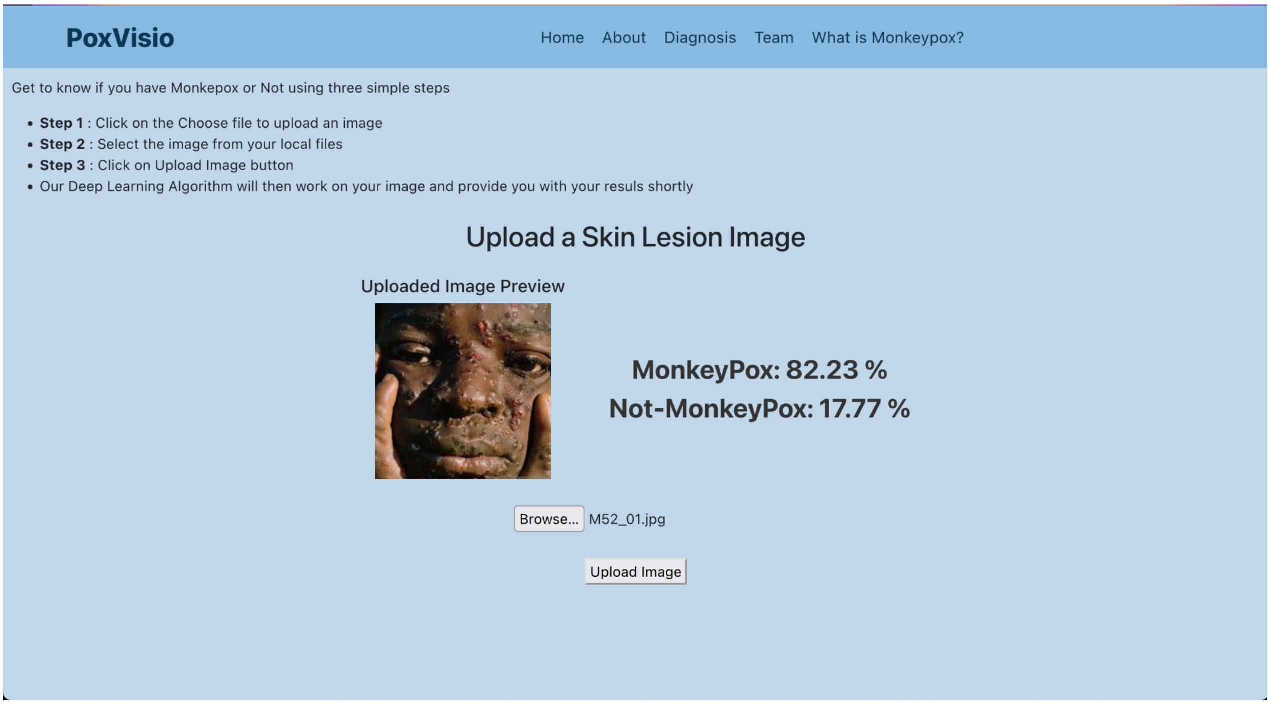
8.4) Testing Model on Previous version of database (MSLDv1)

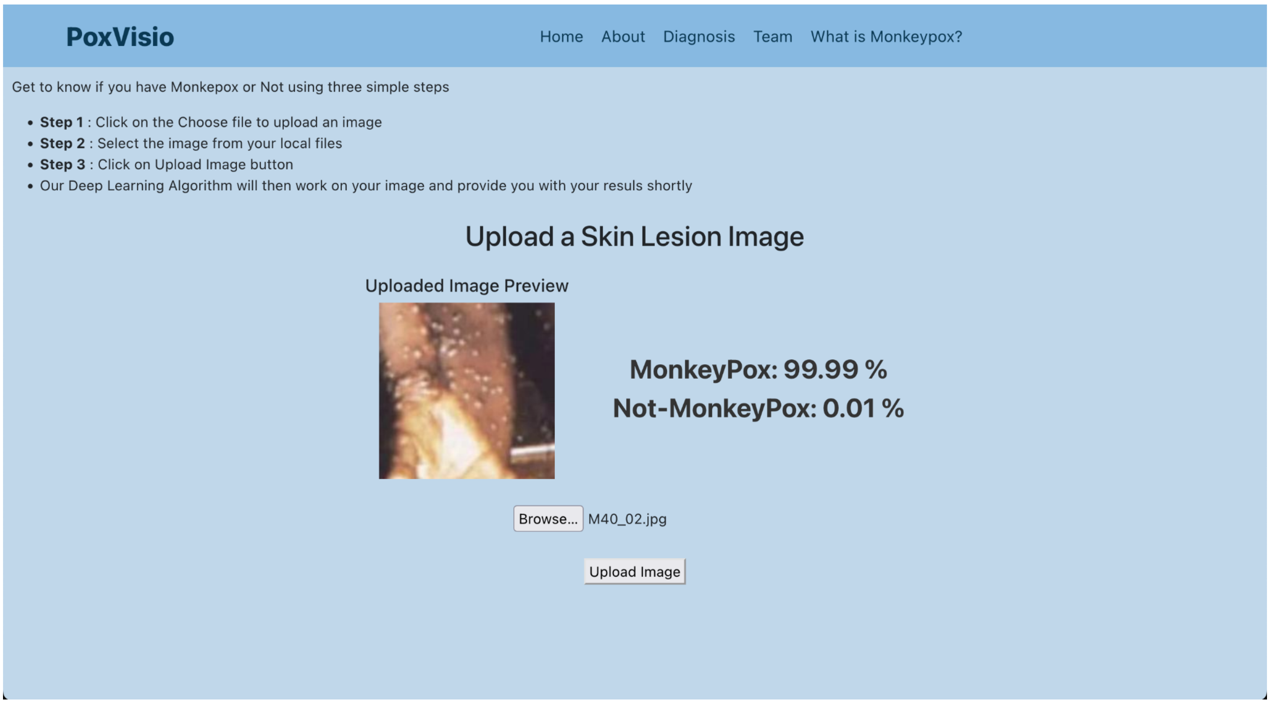
```
MSLDv1_data = gen.flow_from_directory(dataset_path + 'MSLD v1/', target_size=(256, 256), shuffle=False, class_mode='binary')

# Evaluate the model on the test data
test_loss, test_accuracy = best_model.evaluate(MSLDv1_data)
print("Test Loss:", test_loss)
print("Test Accuracy:", test_accuracy)
```

```
Found 228 images belonging to 2 classes.
8/8 [==============================] - 1s 103ms/step - loss: 0.0751 - accuracy: 0.9737
Test Loss: 0.07510682195425034
Test Accuracy: 0.9736841917037964
```

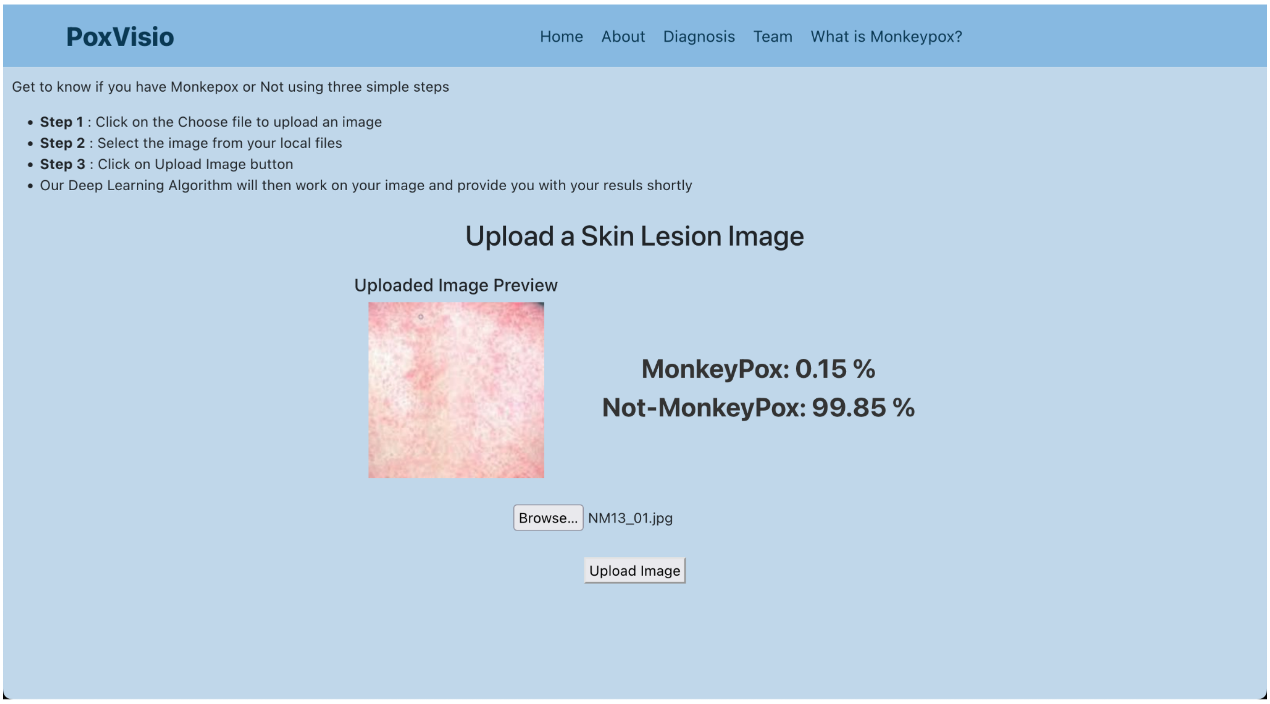
9) Results

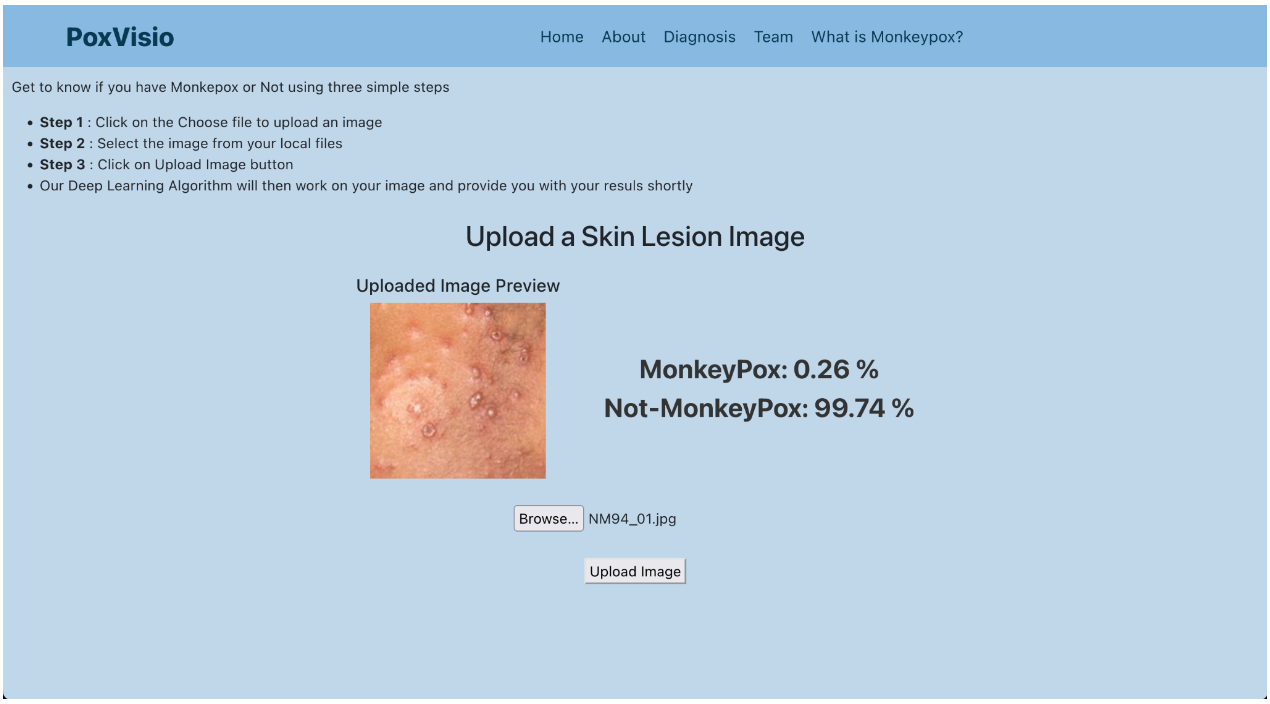
9.1) Results on monkeypox images





9.2) Results Other Disease images





10) Advantages and Disadvantages

Sr. No.	Advantages	Disadvantages
1	Early Monkeypox Identification: -Facilitates early detection of monkeypox - Helps isolate infected individuals	Data Privacy Concerns: - Handling and securing user-uploaded images may raise privacy and security challenges
2	Accessibility and Convenience: -Provides a user-friendly tool for self-assessment -Empowers individuals to take control of their health	Limited Diagnostic Accuracy: -Accuracy may vary based on image quality, lighting, and other factors, impacting
3	Public Health Awareness: -Contributes to public health awareness -Aids in the management of monkeypox outbreaks	Ethical Considerations: - Providing medical diagnoses through an AI tool requires careful ethical considerations - Users should be made aware of the tool's limitations and the need for professional medical advice for confirmation
4	Efficient Resource Utilization: - Reduces the reliance on resource -intensive diagnostic tests -Optimizes healthcare resources	Limited Reach: -Accessibility to the diagnosis page may be limited to those with internet access and the ability to upload images
5	Education and Transparency: -Promotes transparency and education regarding monkeypox -Builds trust and credibility	Algorithm and Model Maintenance: -Continuous maintenance and updates are required to keep the model accurate and up-to-date

11) Conclusion

In conclusion, our project is a holistic response to the challenges posed by the monkeypox outbreak. By deploying ResNet50, curating the essential "Monkeypox Skin Lesion Dataset (MSLD)," and integrating AI diagnosis into our dedicated website, we not only advance the state of the art in disease identification but also prioritize accessibility on a global scale. This multifaceted approach underscores our commitment to facilitating early diagnosis and effective containment strategies, marking a noteworthy stride in the ongoing battle against the monkeypox pandemic.

12) Future Scope

The "PoxVisio" project holds significant potential for future development and expansion. Some key areas of the future scope for this project include:

1. **Enhanced Accuracy:** Continuously improving the deep learning model's accuracy is crucial. Future work can focus on refining the model's training data, fine-tuning parameters, and incorporating advanced techniques to increase the reliability of monkeypox identification.

2. **Real-Time Diagnosis**: Developing a real-time diagnosis feature, where users can receive immediate feedback on their uploaded images, would be a valuable enhancement. This could include live feedback or recommendations for further actions.

3. **Mobile Application**: Creating a mobile application version of PoxVisio would make the tool more accessible to a wider audience. Mobile apps can leverage device cameras for image capture, enhancing user convenience.

4. **Multilingual Support**: Expanding the project to support multiple languages can ensure its utility in diverse regions, especially where monkeypox outbreaks are prevalent.

5. **Machine Learning Explainability**: Implementing explainable AI (XAI) techniques to provide users with insights into why a particular diagnosis was made. This transparency can build trust and assist users in understanding the model's decision.

6. **Geospatial Data Integration**: Incorporating geospatial data can help identify areas at higher risk for monkeypox outbreaks, enabling proactive measures and resource allocation.

7. **Telemedicine Integration**: Integrating PoxVisio with telemedicine platforms could enable remote consultations with healthcare professionals, enhancing the quality of care for users.

8. **Machine Learning Model Deployment**: Deploying the model in the cloud or on edge devices, such as IoT devices, could expand its accessibility and reach, particularly in resource-constrained areas.

9. **Continuous Data Updates**: Maintaining and updating the MSLD with new images from various sources is essential to keep the model current and effective.

10. **Global Collaboration**: Collaborating with healthcare organizations, research institutions, and government agencies can help in the deployment of PoxVisio in regions where monkeypox is a public health concern. This could include official endorsements and partnerships for data collection and research.

11. **Community Engagement**: Involving the community in data collection, image submissions, and feedback can help create a sense of ownership and responsibility, as well as ensure that the tool remains relevant and valuable.

12. **Monitoring and Reporting**: Developing features to track the geographical and temporal trends of monkeypox cases and providing regular reports and insights can aid public health authorities in planning and response efforts.

The future scope for "PoxVisio" is expansive, offering opportunities for refinement, expansion, and collaboration to make this project even more impactful in the fight against monkeypox.

13) Appendix

Source Code:

GitHub Repository Link:
https://github.com/smartinternz02/SI-GuidedProject-593223-1697467015

Project Website Link:
https://orange-coast-072718a00.4.azurestaticapps.net/

Project Demonstration Link:
https://drive.google.com/file/d/18ZbjhI70tLtQVAmFmV-9BgsGPStCjgkB/view?usp=sharing

Model Building Jupyter Notebook:
https://colab.research.google.com/drive/1YkXFPiOHP-HYBsxBX4aubReIunT5r-RD?usp=sharing

Saved Model .h5 file (saved in google drive) - size 500mb:
https://drive.google.com/file/d/1-ApF0NN5ovuv0fvGPIRYPXElEnvplvmM/view?usp=drive_link