**📄 Project Report: Hemo Classify - Blood Cell Classification using Deep Learning**

**Members:**

* BINDU NADAKUDITI  
  *(Add other team members if applicable)*

**1. INTRODUCTION**

**1.1 Project Overview**

Hemo Classify is an intelligent blood cell classification system that uses deep learning to analyze microscopic images of blood cells and identify four primary types:

* Eosinophil
* Lymphocyte
* Monocyte
* Neutrophil

This system enables fast, accurate identification to support medical diagnostics, particularly in hematology , reducing manual effort and potential misclassification.

**1.2 Purpose**

The purpose of this project is to assist healthcare professionals by providing a fast, reliable, and accessible AI-based diagnostic tool that can classify blood cells from images with high accuracy.

**2. IDEATION PHASE**

**2.1 Problem Statement**

Manual classification of blood cells is time-consuming, requires expert knowledge, and is prone to human error. Existing methods are either expensive or not easily accessible for rural healthcare centres.

**2.2 Empathy Map Canvas**

| **Thinks** | **Feels** | **Says** | **Does** |
| --- | --- | --- | --- |
| "Can I detect diseases quickly?" | Stressed about delays | "Is this accurate?" | Uploads images to the app |
| "Is my diagnosis correct?" | Anxious about results | "How does it work?" | Follows AI-based guidance |
| "Can I trust AI analysis?" | Curious but cautious | "Show me the result now." | Adopts new tools |

**2.3 Brainstorming**

* Manual classification: Slow, not scalable.
* Automated classification: High potential.
* Solution: Deep learning-based automated blood cell classifier.

**3. REQUIREMENT ANALYSIS**

**3.1 Customer Journey Map**

text

Copy Edit

Upload Blood Cell Image → Model Prediction → Classification Result Displayed → Diagnosis Support

**3.2 Solution Requirement**

* Input: Blood cell image (JPG/PNG)
* Output: Predicted cell type
* Model: Pre-trained MobileNetV2
* Framework: Flask, TensorFlow
* Hardware: Standard CPU/GPU

**3.3 Data Flow Diagram (Textual)**

text

Copy Edit

User Upload → Image Preprocessing → Model Prediction → Result Rendering (Web)

**3.4 Technology Stack**

* Frontend: HTML, CSS, Flask Templates
* Backend: Python, Flask
* Libraries: TensorFlow, Keras, OpenCV, NumPy
* Deployment: Localhost (for now)

**4. PROJECT DESIGN**

**4.1 Problem Solution Fit**

* Reduces manual effort
* Provides rapid analysis
* Accessible via web browser

**4.2 Proposed Solution**

The system takes an image input, processes it, uses a trained deep learning model to predict the cell type, and displays the result along with confidence score and additional cell type information.

**4.3 Solution Architecture**

text

Copy Edit

User Interface (Upload) → Flask Backend → MobileNetV2 Model → Prediction Result → Display via Flask

**5. PROJECT PLANNING & SCHEDULING**

| **Task** | **Responsible** | **Duration** |
| --- | --- | --- |
| Model Training | Bindu Nadakuditi | 3 Days |
| Flask Web App | Bindu Nadakuditi | 2 Days |
| UI Design | Bindu Nadakuditi | 2 Days |
| Testing & Debugging | Bindu Nadakuditi | 1 Day |
| Documentation | Bindu Nadakuditi | 1 Day |

**6. FUNCTIONAL & PERFORMANCE TESTING**

**6.1 Performance Testing**

* Accuracy: ~90% (based on MobileNetV2 transfer learning)
* Average Inference Time: ~1.2 seconds per image
* Model: MobileNetV2 (pre-trained, fine-tuned)

**7. RESULTS**

**7.1 Output Screenshots**

* Image Upload Screen
* Image Preview with “Analise Cell” Button
* Classification Result with Confidence Score
* Detailed Explanation about Cell Types

*(Attach screenshots in the final document)*

**8. ADVANTAGES & DISADVANTAGES**

**Advantages**

* Fast and accurate cell classification
* Easy to use web interface
* Supports major white blood cell types
* Scalable and can integrate more classes

**Disadvantages**

* Relies on good quality images
* Does not support real-time camera-based scanning
* Web-based demo; not yet mobile-optimized

**9. CONCLUSION**

The Hemo Classify system demonstrates the potential of AI in healthcare diagnostics by enabling quick blood cell analysis. It reduces manual errors, speeds up the diagnostic process, and offers a scalable solution for future development in medical AI.

**10. FUTURE SCOPE**

* Develop a mobile app for instant classification
* Add more blood cell classes
* Deploy using cloud platforms for wide accessibility
* Integrate real-time camera feeds for automated labs

**11. APPENDIX**

* Source Code: *Available in local files / GitHub repository*
* Dataset: Publicly available blood cell datasets
* GitHub Link: *(Provide the link)*
* Demo Video: *(Add YouTube/Drive link if available)*