Project Report Format

1. INTRODUCTION

1.1. Project Overview

1.2. Purpose

2. IDEATION PHASE

2.1. Problem Statement

2.2. Empathy Map Canvas

2.3. Brainstorming

3. REQUIREMENT ANALYSIS

3.1. Customer Journey Map

3.2. Solution Requirement

3.3. Data Flow Diagram3.4. Technology Stack

4. PROJECT DESIGN

4.1. Problem Solution Fit

4.2. Proposed Solution

4.3. Solution Architecture

5. PROJECT PLANNING & SCHEDULING

5.1. Project Planning

6. FUNCTIONAL AND PERFORMANCE TESTING

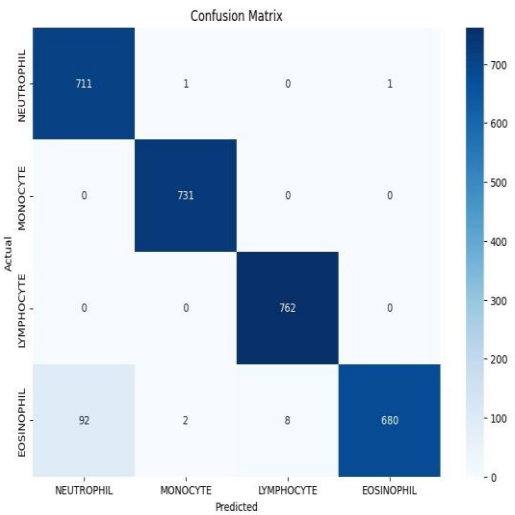
6.1. Performance Testing

7.RESULTS

```
  1/697 ━━━━━━━━━━━━━━━━━━━━ 5:49:19 30s/step - accuracy: 0.2500 - loss: 2.2116
I0000 00:00:1751080665.071249      98 device_compiler.h:188] Compiled cluster using XLA!  This line is logged at m
ost once for the lifetime of the process.
697/697 ━━━━━━━━━━━━━━━━━━━━ 197s 240ms/step - accuracy: 0.5235 - loss: 1.1117 - val_accuracy: 0.6808 - val_loss:
1.1162
Epoch 2/5
697/697 ━━━━━━━━━━━━━━━━━━━━ 90s 129ms/step - accuracy: 0.8617 - loss: 0.3655 - val_accuracy: 0.9319 - val_loss:
0.2177
Epoch 3/5
697/697 ━━━━━━━━━━━━━━━━━━━━ 86s 124ms/step - accuracy: 0.9129 - loss: 0.2253 - val_accuracy: 0.9577 - val_loss:
0.1130
Epoch 4/5
697/697 ━━━━━━━━━━━━━━━━━━━━ 86s 124ms/step - accuracy: 0.9375 - loss: 0.1696 - val_accuracy: 0.8924 - val_loss:
0.3352
Epoch 5/5
697/697 ━━━━━━━━━━━━━━━━━━━━ 84s 120ms/step - accuracy: 0.9458 - loss: 0.1577 - val_accuracy: 0.9756 - val_loss:
0.0644
```

	precision	recall	f1-score	support
EOSINOPHIL	0.89	1.00	0.94	713
LYMPHOCYTE	1.00	1.00	1.00	731
MONOCYTE	0.99	1.00	0.99	762
NEUTROPHIL	1.00	0.87	0.93	782
accuracy			0.97	2988
macro avg	0.97	0.97	0.97	2988
weighted avg	0.97	0.97	0.96	2988

Accuracy of the model 96.5%



8. ADVANTAGES & DISADVANTAGES

☑ **Advantages:**

1. **Accurate Predictions**: Uses MobileNetV2, a lightweight yet efficient deep learning model for blood cell classification.
2. **User-Friendly Interface**: Simple and intuitive web UI accessible by lab technicians and students without technical expertise.
3. **Fast Processing**: Predictions are generated within seconds on local systems.
4. **Cost-Effective**: No need for cloud infrastructure; runs on local machine.
5. **Offline Capability**: Once set up, can be run without internet connection.
6. **Extendable**: Easy to expand with additional classes or integrate cloud/database in future.

❌ **Disadvantages:**

1. **Local-Only Deployment**: The app is not yet deployed on cloud or accessible remotely.
2. **Inference-Only System**: Model training and fine-tuning are done separately and not integrated into the app.
3. **Dataset Limitations**: Accuracy may drop if the underlying dataset is small or unbalanced.
4. **No Real-Time Camera Input**: Requires manual file upload instead of live microscope or webcam feed.
5. **Not Yet Mobile-Friendly**: Current UI is not optimized for mobile screens.

---

9. CONCLUSION

The HematoVision project successfully demonstrates an AI-powered system for blood cell classification using transfer learning. A pre-trained MobileNetV2 model enables accurate classification of eosinophils, lymphocytes, monocytes, and neutrophils from uploaded images. The web app, developed using Flask, provides a simple interface for uploading images and viewing results in real time. The system is designed to be efficient, easy to use, and runs entirely on local machines.

---

10. FUTURE SCOPE

- **Model Enhancement**: Train with larger and more diverse blood cell datasets to increase classification accuracy.

- **Mobile App Version**: Develop a responsive Android/iOS app to allow broader field or clinical usage.
- **Live Microscope Feed Integration**: Enable live image analysis via webcam or digital microscope.
- **Report Generation**: Automatically create and download classification reports for analysis or sharing.
- **Multilingual Interface**: Add support for regional languages to improve accessibility for diverse user groups.

---

11. APPENDIX

◇ **Source Code**

The complete source code is organized into multiple files, including:

- Flask backend (app.py)
- Model prediction logic (utils.py)
- HTML templates (home.html, result.html)
- TensorFlow MobileNetV2 model (`Blood_Cell.h5`)

◇ **Dataset Link**

If using a public dataset, you may use or extend datasets such as:

- Blood Cell Dataset on Kaggle:https://www.kaggle.com/datasets/paultimothymooney/blood-cells

◇ **GitHub & Project Demo Link**

- GitHub Repository: https://github.com/boyarajesh18/Hematovision-Advanced-Blood-Cell-Classification-Using-Transfer-Learning.git
- Deployed Link: *(Currently not hosted online; runs locally at* `http://127.0.0.1:5000`*)*