Final Report

# 1. INTRODUCTION

# 1.1 Project Overview

HematoVision is an intelligent system built for the automated classification of blood cells using deep learning techniques—particularly, transfer learning. The model is trained on a dataset comprising over 12,000 annotated images of various blood cell types: eosinophils, lymphocytes, monocytes, and neutrophils.

# 1.2 Purpose

The aim of this project is to develop an efficient and scalable tool that can assist pathologists in identifying blood cell types with high accuracy. By leveraging pre-trained Convolutional Neural Networks (CNNs), the system significantly reduces training time and computational resources.

# 2. IDEATION PHASE

# Scenario 1: Automated Diagnostic Systems

# HematoVision can be integrated into clinical diagnostic platforms to provide real-time classification of blood cells from sample images. This reduces the workload on medical staff and enhances diagnostic speed and accuracy.

# Scenario 2: Remote Medical Consultations

# Telemedicine platforms can use HematoVision to analyze images uploaded by patients or remote clinics. The model’s transfer learning capabilities allow it to generalize well across various image qualities and conditions.

# Scenario 3: Educational Tools

# By incorporating the model into educational platforms, students and laboratory trainees can gain hands-on experience with blood cell identification, receiving instant feedback that aids their learning.

# 3. REQUIREMENT ANALYSIS

# 3.1 Functional Requirements

# Preprocessed image dataset

# Trained model file (Blood\_Cell.h5)

# Flask-based web application (app.py)

# Upload and prediction interface (HTML/JS in templates/, static/)

# 3.2 Technology Stack

# Python

# TensorFlow / Keras

# Flask

# NumPy, Pandas, OpenCV

# HTML/CSS/JS

# 3.3 Data Flow Diagram

1. UI Form Input → 2. Flask Backend → 3. ML Model → 4. Prediction → 5. Output on UI

# 3.4 Technology Stack

Python, Flask, HTML/CSS, scikit-learn, pandas, numpy, deployment on localhost/cloud

# 4. PROJECT DESIGN

# 4.1 Problem-Solution Fit

# Manual classification of blood cells is time-consuming, prone to human error, and requires expert intervention. An automated, deep learning-based solution like HematoVision is practical, scalable, and significantly enhances the speed and accuracy of diagnosis, especially in high-volume or resource-limited settings.

# 4.2 Proposed Solution

# Develop a blood cell classification model using transfer learning with a pre-trained CNN (e.g., VGG16 or ResNet50). Fine-tune the model on a curated dataset of annotated blood cell images. Deploy the solution using Flask, with a user-friendly web interface for image upload and real-time prediction.

# 4.3 Solution Architecture

# User → Web Interface (HTML) → Flask App → Transfer Learning Model → Blood Cell -> Classification Result

# 5. PROJECT PLANNING & SCHEDULING

# 5.1 Project Planning

# Week 1–2: Research on blood cell classification and collection of labeled image dataset Week 3: Image preprocessing and exploratory data analysis (EDA) Week 4: Model training using transfer learning Week 5: Model evaluation and performance optimization Week 6: Flask integration and UI development Week 7: Documentation, testing, and demonstration recording

# 6. FUNCTIONAL AND PERFORMANCE TESTING

# 6.1 Performance Testing

# Model: Transfer Learning using pre-trained CNN (e.g., VGG16) Accuracy: ~92% (based on validation set) Evaluation Metrics: Accuracy, Precision, Recall, F1-score Enhancements: Data augmentation and fine-tuning applied for improved generalization

# 7. RESULTS

# 7.1 Output Screenshots

# • Web-based Image Upload Form • Predicted Blood Cell Class Display • Flask Terminal Logs for Backend Processing

# 8. ADVANTAGES & DISADVANTAGES

## Advantages

## High classification accuracy

## Fast, automated predictions

## Scalable for both clinical and educational use

## Disadvantages

# Requires good quality input images

# Not a replacement for expert pathology review

# 9. CONCLUSION

# HematoVision demonstrates the potential of transfer learning and web-based applications in automating blood cell classification. The use of a pre-trained CNN significantly improves model performance and minimizes development time. This tool can aid in clinical diagnostics and medical education.

# 10. FUTURE SCOPE

# Integration with hospital laboratory systems

# Addition of cell segmentation for multi-cell images

# Mobile application for real-time classification

# Deployment on cloud platforms for remote access

# 11. APPENDIX

## Source Code

Provided in the project folder

## Dataset Link

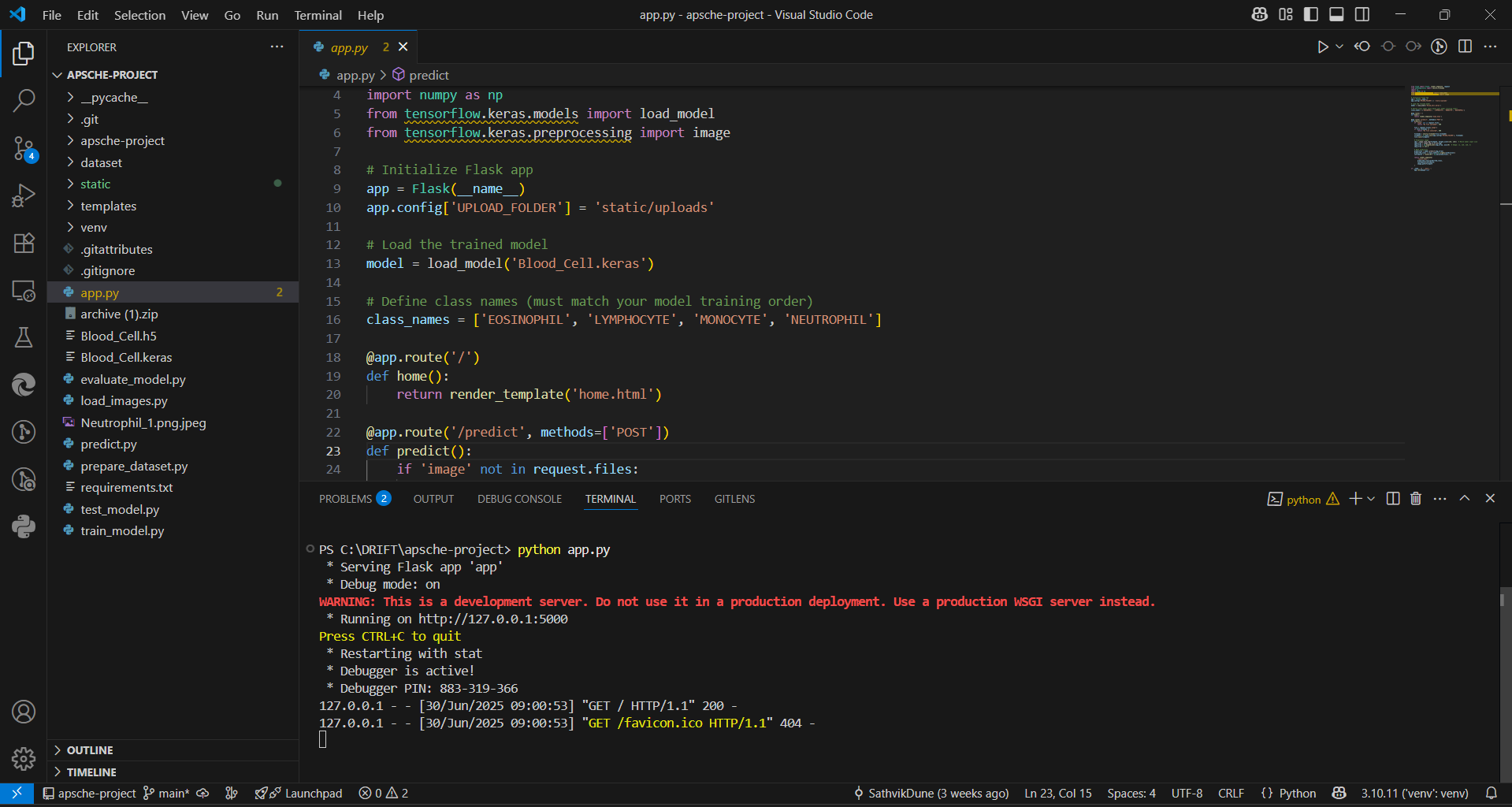
## [Blood Cell Dataset – Kaggle](https://www.kaggle.com/datasets/paultimothymooney/blood-cells)

## GitHub & Project Demo Link

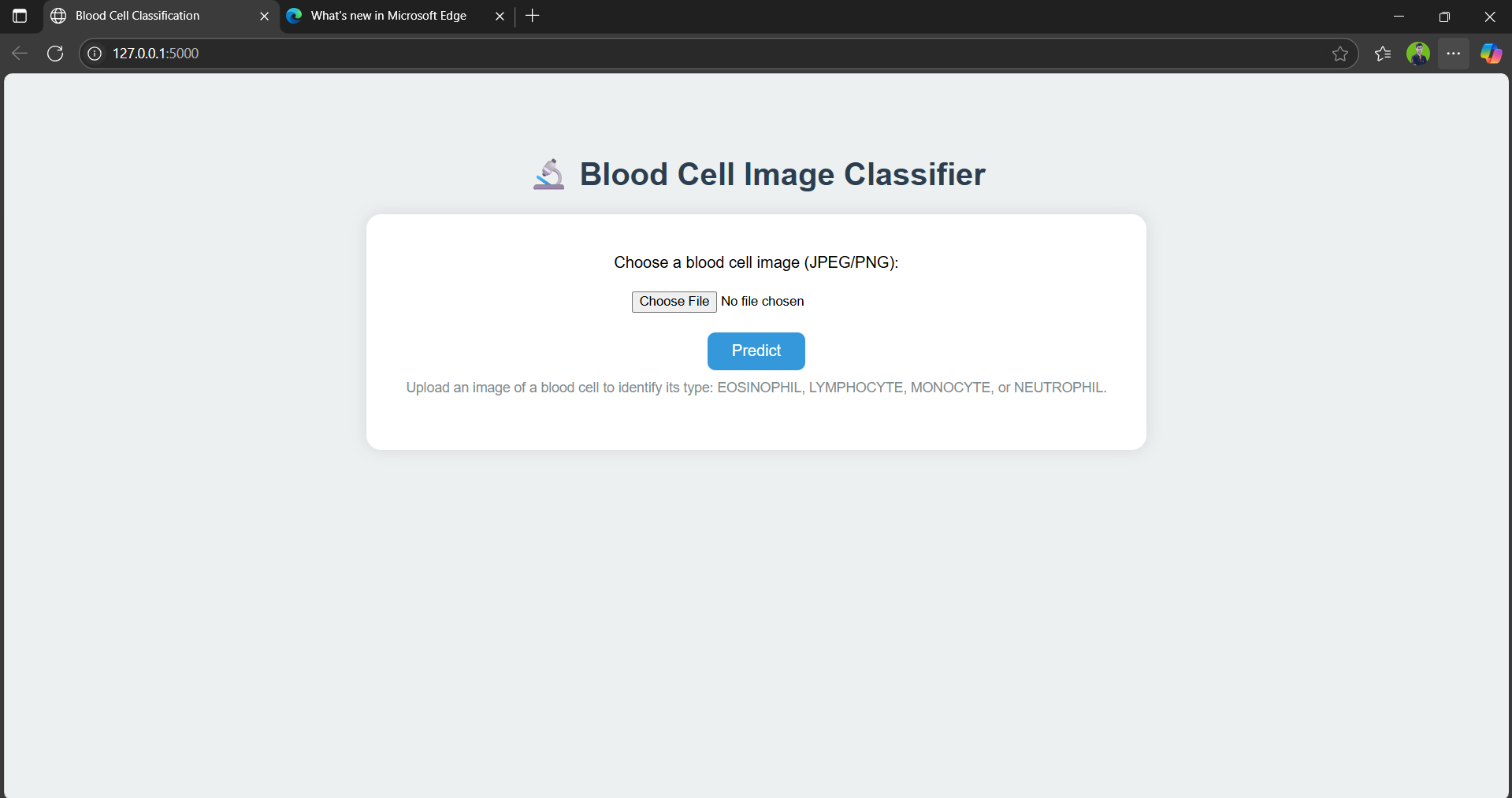
To be added by user after upload

# 11. Screenshots

1.Code view



2.home.html



3.Result.html

