Project Design Phase-II

Technology Stack (Architecture & Stack)

Date	20 January 2026
Team ID	LTVIP2026TMIDS76912
Project Name	HematoVision: Advanced Blood Cell Classification Using Transfer Learning
Maximum Marks	4 Marks

Technical Architecture:

The technical architecture of HematoVision consists of a deep learning-based blood cell classification model built using transfer learning with a pre-trained CNN such as VGG16 or ResNet. The dataset is preprocessed and augmented before being split into training and testing sets for model development and evaluation. The trained model is saved and integrated into a Flask web application, where users can upload blood cell images through a simple HTML interface. The Flask backend processes the image, sends it to the model for prediction, and displays the classification result along with confidence scores on the user interface.
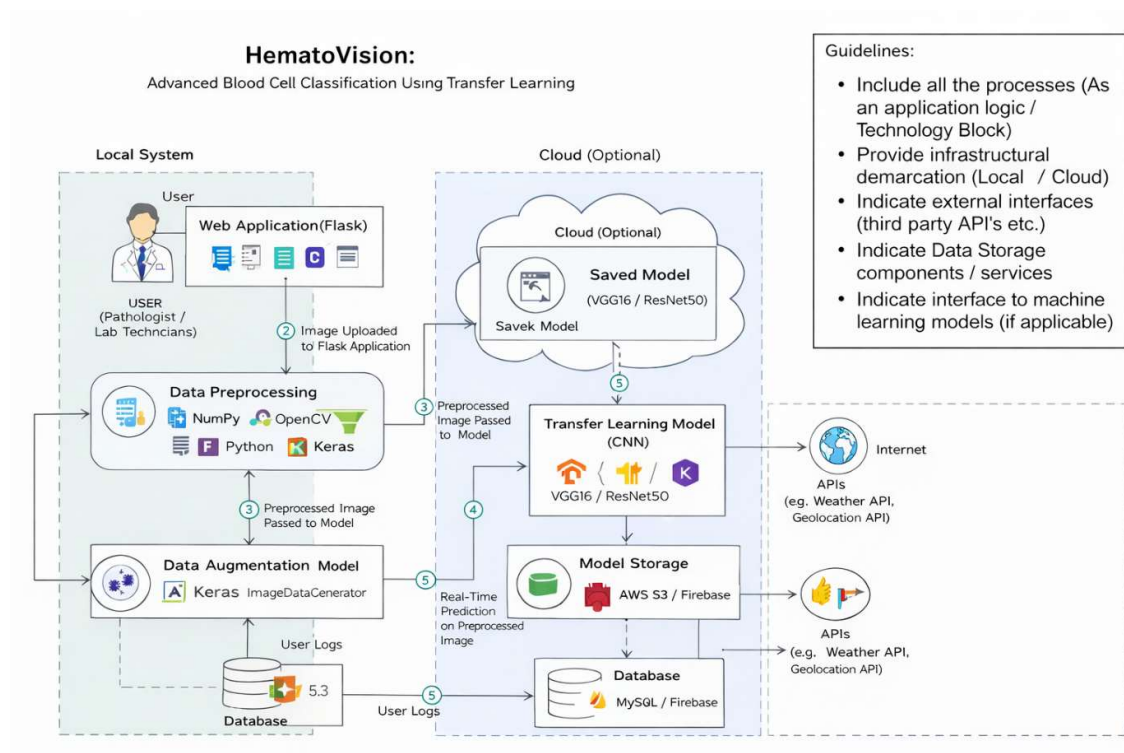
Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Allows users (pathologists / students) to upload blood cell images and view prediction results	HTML, CSS
2.	Web Application Framework	Handles routing, image upload, model invocation, and result display	Python Flask
3.	Data Preprocessing Module	Image resizing, normalization, noise removal, and transformation before model input	NumPy, OpenCV, TensorFlow/Keras preprocessing
4.	Data Augmentation Module	Enhances dataset diversity to reduce overfitting	Keras ImageDataGenerator
5.	Transfer Learning Model	Data Type, Pre-trained CNN adapted for blood cell classificationetc.	VGG16 / ResNet50 (TensorFlow / Keras / PyTorch)
6.	Model Training & Optimization	Training, fine-tuning layers, loss calculation, optimizer selection	Adam Optimizer, Categorical Crossentropy.
7.	Model Evaluation	Accuracy, confusion matrix, precision, recall, F1-score	Scikit-learn, Matplotlib
8.	Backend Integration	Loads saved model and performs real-time prediction	Flask + TensorFlow
9.	Model Storage	Saves trained model for deployment	H5 format / SavedModel format

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Deep learning framework for model development, transfer learning implementation, data preprocessing, visualization, and web deployment	TensorFlow / Keras, PyTorch (optional), Flask, NumPy, OpenCV, Scikit-learn, Matplotlib
2.	Security Implementations	Secure image upload validation, input sanitization, restricted file formats (JPG/PNG), prevention of code injection, secure model access, hashed credentials (if login added), HTTPS deployment	Flask Security Features, Werkzeug (Password Hashing – SHA-256), Input Validation, SSL/HTTPS, OWASP Security Practices