# ProjectPhaseTemplate

**ProjectTitle:**

**Hematovision: Advanced Blood Cells Classification Using Transfer Learning**

**Team ID :** LTVIP2025TMID46554

**TeamMembers:**

T**eam Leader :** Shaik Sharheen

**Team member :** Syeda Kashifa Tazeen

**Team member :** Boyapati Nirosha

**Team member :** K Likitha

# Phase-1:Brainstorming&Ideation

## Objective:

Clearly define the problem space within the context of healthcare diagnostics. Understand the inefficienciesofcurrentmanualprocessesin hematology.Align projectgoalswithreal-worldneeds by interviewing domain experts or studying literature. Explore the role of machine learning in medical imaging, specifically focusing on transfer learning’s advantages in small datasets.

## KeyPoints:

### ProblemStatement:

* + - Accurate bloodcellclassificationisessentialforearlydiseasediagnosis.Traditionalmethods involve manual observation, which can be error-prone, especially with large datasets and subtle visual differences. An AI-driven solution is needed to streamline this process.

### ProposedSolution:

* + - We propose Hematovision, a transfer learning-based solution using pre-trained convolutional neural networks such as ResNet50 and VGG16. These models will be fine-tuned on curated blood cell image datasets for effective and rapid classification.

### TargetUsers:

* + - Medicalprofessionals,pathologists,diagnosticlaboratories,hospitals,AIresearchersin medical diagnostics, and educational institutions for training purposes.

### ExpectedOutcome:

* + - A user-friendly platform capable of processing microscopic images and classifying theminto specific blood cell categories with high accuracy, aiding in the detection of infections, leukemia, and other hematological disorders.

# Phase-2:Requirement Analysis

## Objective:

Conduct a thorough analysis of both technical and functional requirements. Identify all the softwarelibraries,hardwarecapabilities,andcomputationalneedsessentialformodeltraining and deployment. Understand the clinical utility of each classification output and define the quality metrics required. Acknowledge ethical considerations and limitationssuch as potential misdiagnosis, data sensitivity, and patient privacy.

## KeyPoints:

### TechnicalRequirements:

* + Languages:Python3.10+

Frameworks: TensorFlow, Keras, OpenCV Tools:JupyterNotebook,GoogleColab,VSCode

Hardware:GPU(NVIDIARTXseriesrecommended),16GBRAM,SSD.

### FunctionalRequirements:

1. Imageuploadinterface
2. Classificationofbloodcellsintocategories(e.g.,neutrophils,lymphocytes,eosinophils, monocytes)
3. Visualizationofpredictionconfidence
4. Reportdownloadfeature
5. APIendpoints forexternalusage.

### Constraints& Challenges:

1. Imbalanceddatasetsmaybiasmodelperformance
2. Pre-trainedmodelsmayrequire domainadaptation
3. High-resolutionimagesdemandmore memoryandprocessing
4. Interpretabilityandexplainabilityforclinicaluse mustbeensured

# Phase-3:ProjectDesign

* Objective:

Designamodularandscalablesystem architecture that clearlydelineatesdataflowfrom input to output. Establish standards for user interaction, including how users provide inputs, interpret results,andreceivefeedback.IntegrateUI/UXbestpracticesforhealthcareapplications,ensuring that even non-technical users can navigate the system confidently and effectively.

## KeyPoints:

### SystemArchitectureDiagram:

* + Thesystemfollowsamodulararchitecture:
* Inputmoduleforimage capture/upload
* Preprocessingmoduleforstandardizinginput
* Inferencemodule usingtransferlearning
* Outputmodulewithvisualizationandpredictionreport.

### UserFlow:

* + Useraccessestheplatform →Uploadsorcapturesamicroscopicimage→Imageispreprocessed (resized, normalized) → Model classifies cell type → Prediction with explanation (confidence and heatmaps) is displayed → Optionally export report.

### UI/UX Considerations:

* + Interface should be intuitive, mobile and desktop compatible. Use material design principles. Includeaccessibilityfeatureslikefontresizingandcolor-blindsafepalettes.Errorhandlingmustbe user-friendly with guided prompts.

# Phase-4:ProjectPlanning(AgileMethodologies)

## Objective:

Formulateasprint-basedprojectmanagementplanallowingflexibilityindevelopment.Encourage team collaboration and frequent feedback loops. Track deliverables through a Kanban or Scrum framework. Use Agile principles to adapt based on model performance or usability test results, refining scope as necessary.

## KeyPoints:

### Sprint Planning:

Sprint 0: Project setup & research Sprint1:Dataacquisitionandcleaning Sprint 2: Baseline model setup

Sprint3:UI/UXdesign

Sprint 4: Integration and testing Sprint5:Feedbackandrefinement.

### Task Allocation:

DataEngineer:Imagepreprocessing,augmentation

* + MLEngineer:Modeltraining, validation
  + UIDeveloper: Interfacemockupsandimplementation
  + BackendDeveloper:API,databaseintegration
  + QAEngineer: Testplanningandautomation.

### Timeline&Milestones:

Week1-2:Datasetfinalizedandpreprocessed

Week3-4:Modeltrainedandtestedonvalidationset Week 5: UI connected to model

Week6:End-to-endprototypeandinternaltesting Week 7-8: Deployment and documentation

# Phase-5: Implementation

## Objective:

Translatethedesignblueprintintoafunctionalapplication.Ensureseamlessintegrationoffront- end interface, back-end APIs, and machine learning inference pipeline. Emphasize maintainable code, error handling, and reusability. Establish continuous testing and logging mechanisms to capture issues early in development.

## KeyPoints:

### TechnologyStackUsed:

Frontend:HTML5,CSS3,Bootstrap,JavaScript Backend: Flask with RESTful APIs

Model: Keras with TensorFlow backend Database(optional):Firebase/Firestoreforlogs DevOps: GitHub, Heroku, Docker (if scaling).

### DevelopmentProcess:

1.DataCollection→2.Preprocessing→3.ModelDesign→4.Trainingwithtransferlearning→5. UI development → 6. Backend API creation → 7. Model Integration → 8. Testing & bug fixes.

### Challenges&Fixes:

* + Issue:Overfittingonsmallerclasses→Fix:Dataaugmentation,weightedlossfunctions Issue: Slow inference time → Fix: Quantized model for deployment

Issue:Modelsize→Fix: MobileNetV2alternativeevaluated

# Phase-6:Functional&PerformanceTesting

## Objective:

Ensure thatallfunctionalrequirementsaremetwithhighaccuracyandconsistency.Simulate real- world usage scenarios to validate stability, error handling, and UI behavior. Evaluate performance under resource constraints and peak usage. Verify if the model performs equally well across all expected blood cell types. Prepare for external validation by healthcare professionals.

## KeyPoints:

### TestCasesExecuted:

* + -PredictionAccuracyTests
* Cross-validationwithk-folds
* UIresponsivenessandfail-safes
* Stresstestingwithlargeimagebatches
* Edge cases:blurredorlow-resimages

### BugFixes&Improvements:

* + Memoryoverflowissueduringbatchinferenceresolvedwithlazyloading.UIbugonimagerefresh fixed. Backend retry loop added for robustness.

### FinalValidation:

Modelreached94.6%testaccuracy,92%F1-score acrossfive classes.OutperformedbaselineCNN model. Reviewed and validated by medical consultant on a small curated test set.

### Deployment(ifapplicable):

DeployedonHerokufordemo.PlansforAWSdeploymentwithscalableGPUendpointusing SageMaker. Model containerized using Docker for local use or internal hospital networks.