Project Design Phase-II
Technology Stack (Architecture & Stack)

Date	30 June 2025
Team ID	LTVIP2025TMID41159
Project Name	Transfer Learning -Based Classification Of Poultry Diseases For Enhanced Health Management
Maximum Marks	4 Marks

Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

Example: Transfer Learning -Based Classification Of Poultry Diseases For Enhanced Health Management

Reference: https://thesai.org/Downloads/Volume12No7/Paper_49-Transfer_Learning_Approach_for_Poultry_Disease_Detection.pdf

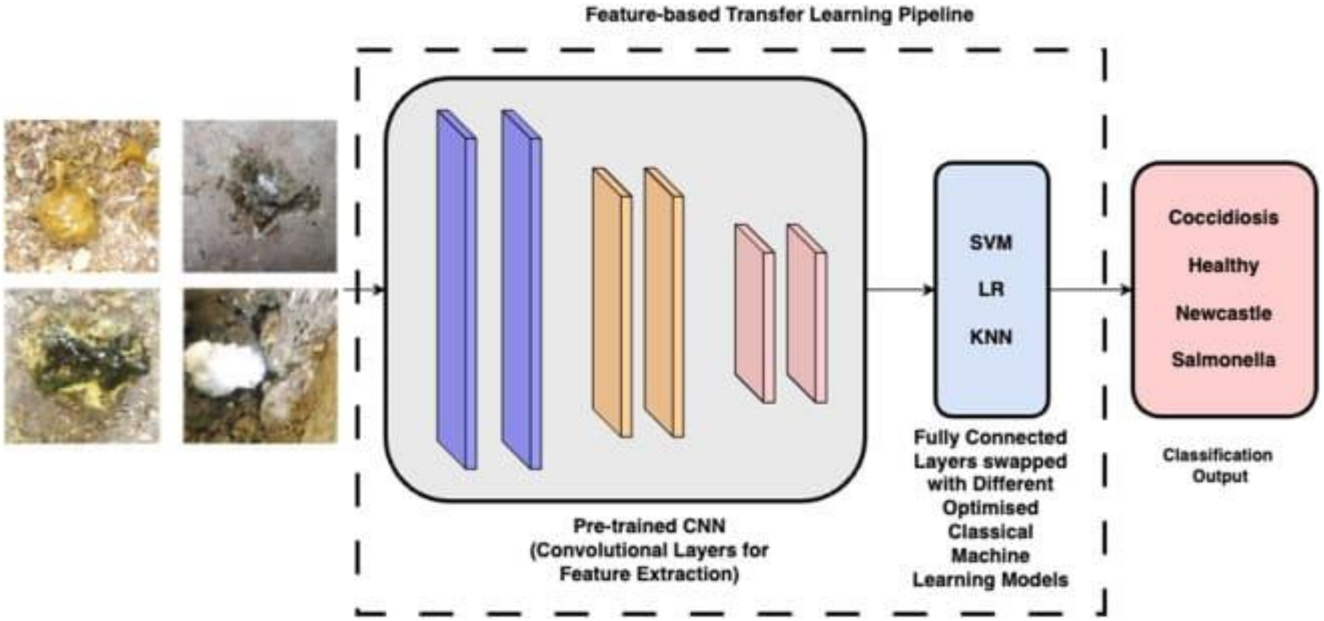
Feature-based Transfer Learning Pipeline
SVM
LR
KNN
Coccidiosis
Healthy
Newcastle
Salmonella
Fully Connected Layers swapped with Different Optimised Classical Machine Learning Models
Classification Output
Pre-trained CNN (Convolutional Layers for Feature Extraction)

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Mobile app for farmers to input symptoms or upload images	Android Studio (Java/Kotlin), XML for UI.
2.	Application Logic-1	Image/symptom data preprocessing before prediction	Python (OpenCV, Pandas), TensorFlow preprocessing utils
3.	Application Logic-2	Disease prediction using a Transfer Learning model	Python (TensorFlow/Keras or PyTorch
4.	Application Logic-3	Displaying results & suggested treatment to user	Java/Kotlin logic in mobile app
5.	Database	Storing user inputs, prediction results, treatment history	SQLite (Mobile) or MySQL
6.	Cloud Database	Centralized access to data and predictions from all users	Firebase Realtime DB / Google Firestore / AWS RDS
7.	File Storage	Storage of images used for prediction	Firebase Storage / AWS S3 / Local Storage
8.	External API-1	Weather conditions API to correlate with disease occurrences (optional)	OpenWeatherMap API / WeatherStack API
9.	External API-2	overnment poultry advisory API (if available)	Gov advisory APIs / WHO/FAO poultry datasets.
10.	Machine Learning Model	Purpose of Machine Learnin Classify input as Salmonella, Newcastle, Coccidiosis, or Healthy	ResNet50 / MobileNetV2 / EfficientNet with Transfer Learning g Model
11.	Infrastructure (Server / Cloud)	Model deployment and mobile-backend API handling	Firebase Functions / AWS Lambda / Flask or FastAPI on EC2 or Heroku.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	TensorFlow, Keras, PyTorch, OpenCV
2.	Security Implementations	List all the security / access controls implemented, use of firewalls etc.	TTPS, JWT Authentication, OAuth 2.0, IAM (Firebase IAM / AWS IAM), SHA-256, OWASP security guidelines
3.	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	Microservices, Docker Containers, Kubernetes, Serverless (AWS Lambda/GCP Functions)
4.	Availability	Justify the availability of application (e.g. use of load balancers, distributed servers etc.)	AWS Load Balancer, Multi-zone deployment, Failover mechanisms, Auto Scaling
5.	Performance	Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN's) etc.	Redis Cache, Content Delivery Network (Cloudflare/AWS CloudFront), Asynchronous queues (e.g., Celery, RabbitMQ)

References:

https://www.tensorflow.org/tutorials/images/transfer_learning

https://www.kaggle.com/datasets

https://aws.amazon.com/architecture/

https://docs.microsoft.com/en-us/azure/security/fundamentals/identity-management-overview

https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/