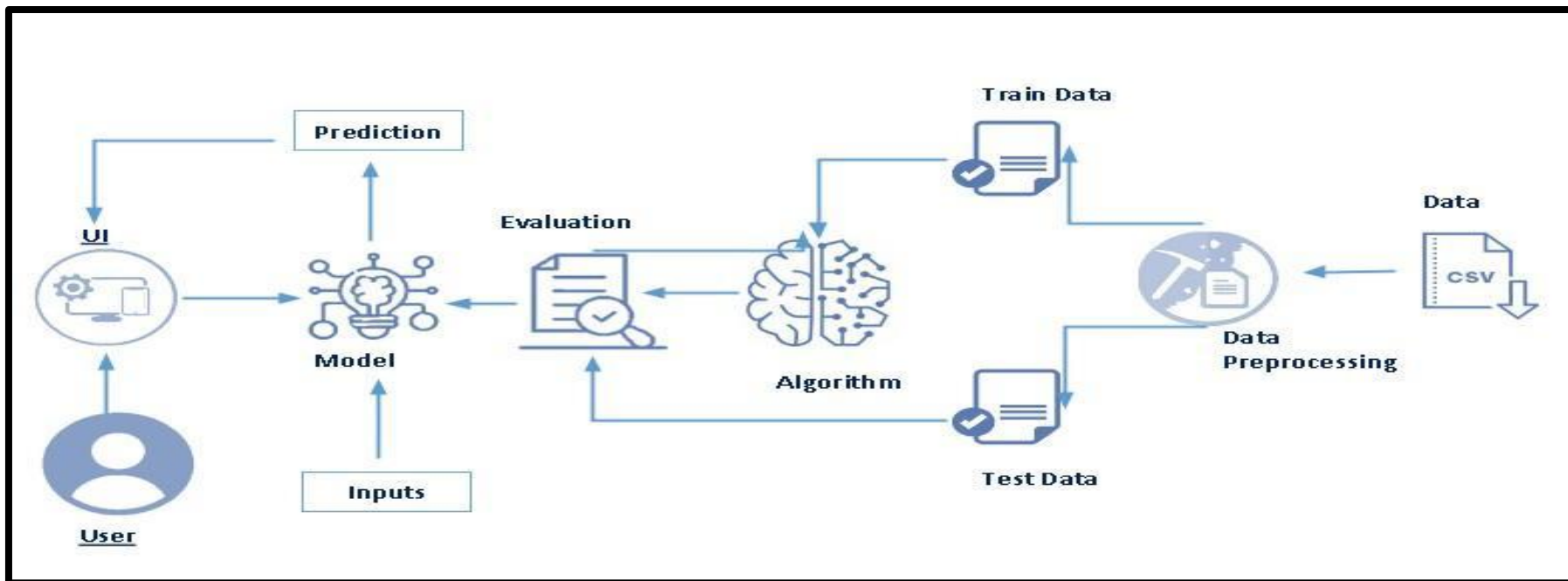


## Project Design Phase-II Technology Stack (Architecture & Stack)

Date	14 Feb 2026
Team ID	LTVIP2026TMIDS52481
Project Name	Dog Breed Identification Using Transfer Learning
Maximum Marks	4 Marks

### Technical Architecture:

The system is a web-based deep learning application that classifies dog breeds from uploaded images.



## Flow:

User → Web UI → Flask Backend → MobileNetV2 Model → Prediction Result → Display on UI

### Infrastructure

- Local training environment for model development
- Cloud deployment using Hugging Face Spaces for global access
- Image dataset stored locally during training and model weights stored for inference

**Table-1 : Components & Technologies:**

S.No	Component	Description	Technology
1.	User Interface	Allows users to upload dog images and view predicted breed.	HTML, CSS, JavaScript
2.	Application Logic-1	Handles HTTP requests, image upload, and prediction routing.	Python, Flask
3.	Application Logic-2	Preprocesses images and feeds them to the trained deep learning model.	TensorFlow, NumPy, PIL
4.	Database	Stores user login credentials for authentication.	SQLite
5.	File Storage	Stores trained model file and uploaded images temporarily.	Local file system / Hugging Face storage
6.	External API	Provides cloud hosting and model inference interface.	Hugging Face Spaces
7.	Machine Learning Model	Classifies dog breeds using transfer learning.	MobileNetV2 (TensorFlow/Keras)
8.	Infrastructure (Server / Cloud)	Training on local system and deployment on cloud.	Local: Python environment (VS Code / Jupyter) Cloud: Hugging Face Spaces

**Table-2: Application Characteristics:**

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Libraries used for ML training and web deployment.	TensorFlow, Keras, Flask, NumPy, Pandas
2.	Security Implementations	User authentication and password protection.	By using flask we have created login page
3.	Scalable Architecture	Separation of frontend, backend, and ML inference enables scalability.	Flask-based modular architecture, cloud deployment
4.	Availability	Cloud hosting ensures global access and continuous availability.	Hugging Face Spaces hosting
5.	Performance	Lightweight MobileNetV2 enables fast inference and low latency predictions.	TensorFlow optimized inference, image resizing preprocessing

**References:**

- ☐ <https://www.kaggle.com/competitions/dog-breed-identification/data>
- ☐ <https://www.tensorflow.org/>
- ☐ <https://keras.io/api/applications/mobilenet/>
- ☐ <https://flask.palletsprojects.com/>
- ☐ <https://huggingface.co/docs/hub/spaces>