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

Date	06 May 2023
Team ID	NM2023TMID05747
Project Name	Automated Weather Classification Using Transfer Learning

Technical Architecture:

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

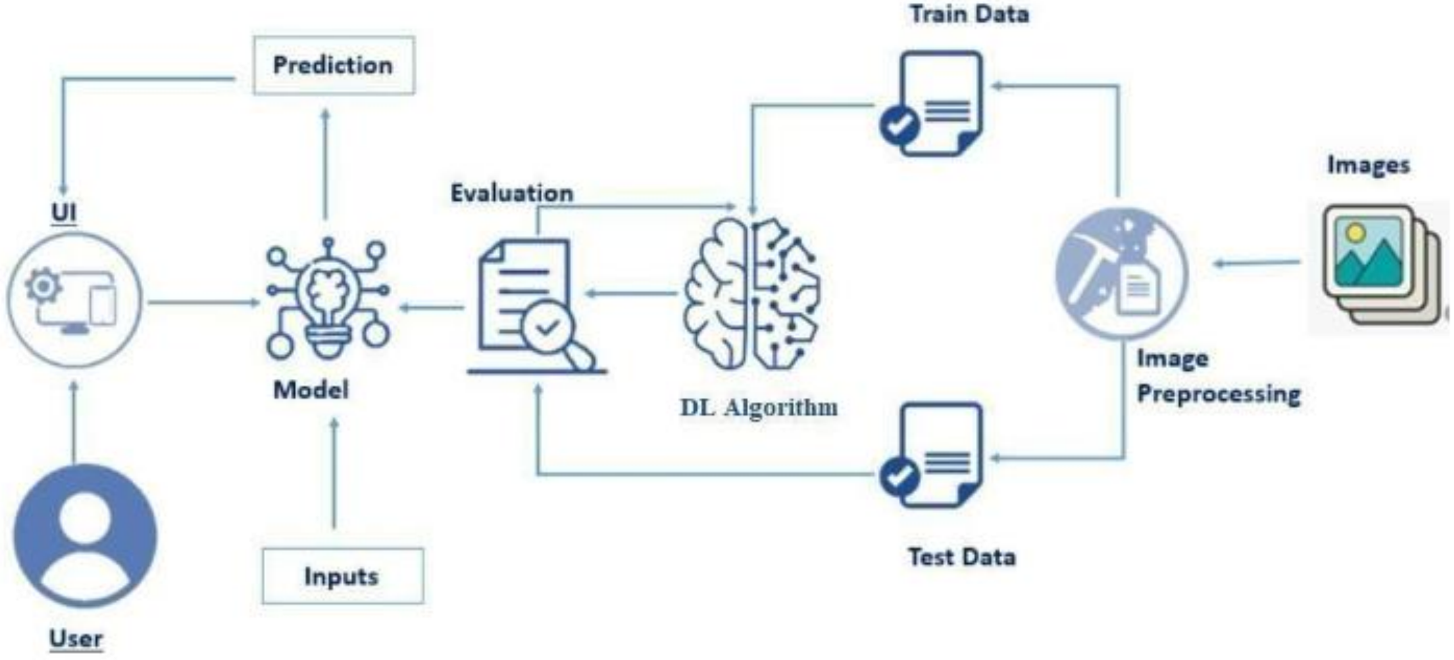

Table-1: Components & Technologies:

S No	Component	Description	Technology
1.	User Interface	It enables the users easily interact with our web application.	HTML, Css, JavaScript, Bootsrap
2.	Data Collection	The first step in building an automated weather classification system is to collect a large amount of weather data that is labelled and annotated. This data will be used to train the machine learning models that will classify weather conditions.	weather sensors, satellite imagery, and radar data
3.	Pre-processing and Feature Extraction	The raw data collected from various sources will need to be pre-processed and transformed into a format that can be used for training the machine learning models. This involves cleaning the data, extracting relevant features, and normalizing the data.	Python libraries such as NumPy, Pandas, and Scikit-learn
4.	Transfer Learning	Transfer learning is a technique in which a pre-trained machine learning model is used as a starting point for training a new model on a different dataset. This approach can be used to speed up the training process and improve the accuracy of the model.	deep learning frameworks such as TensorFlow and PyTorch
5.	Model Evaluation and Testing	Once the machine learning models have been trained, they need to be evaluated and tested to ensure that they are accurate and reliable. This involves using a separate test dataset to evaluate the performance of the models.	Python libraries such as Scikit-learn and TensorFlow
6.	Deployment	Once the machine learning models have been trained and tested, they can be deployed in a production environment. This involves integrating the models into a larger software system and making them accessible to end-users.	frameworks such as Flask and Django, cloud computing platforms such as Amazon Web Services and Microsoft Azure, and containerization technologies such as Docker and Kubernetes.

Table-2: Application Characteristics:

S No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Open-source frameworks are software frameworks that are publicly available and free to use. They provide a foundation for building software systems and can be customized to meet specific needs	Python, Flask or Django
2.	Security Implementations	Security implementations are measures taken to protect a software system from malicious attacks or unauthorized access. These can include encryption, authentication, and access control.	Cross-Site Request Forgery (CSRF) Protection, Password Hashing, HTTP Strict Transport Security (HSTS),Secure File Uploads
3.	Availability	Availability refers to the ability of a software system to remain operational and accessible even in the face of hardware or software failures or other disruptions	IBM Cloud offers load balancing services to distribute traffic across multiple servers, which can improve availability
4.	Performance	Performance refers to the speed and efficiency of a software system, including response time, throughput, and resource utilization..	IBM Cloud, Transfer Learning approach to improve performance