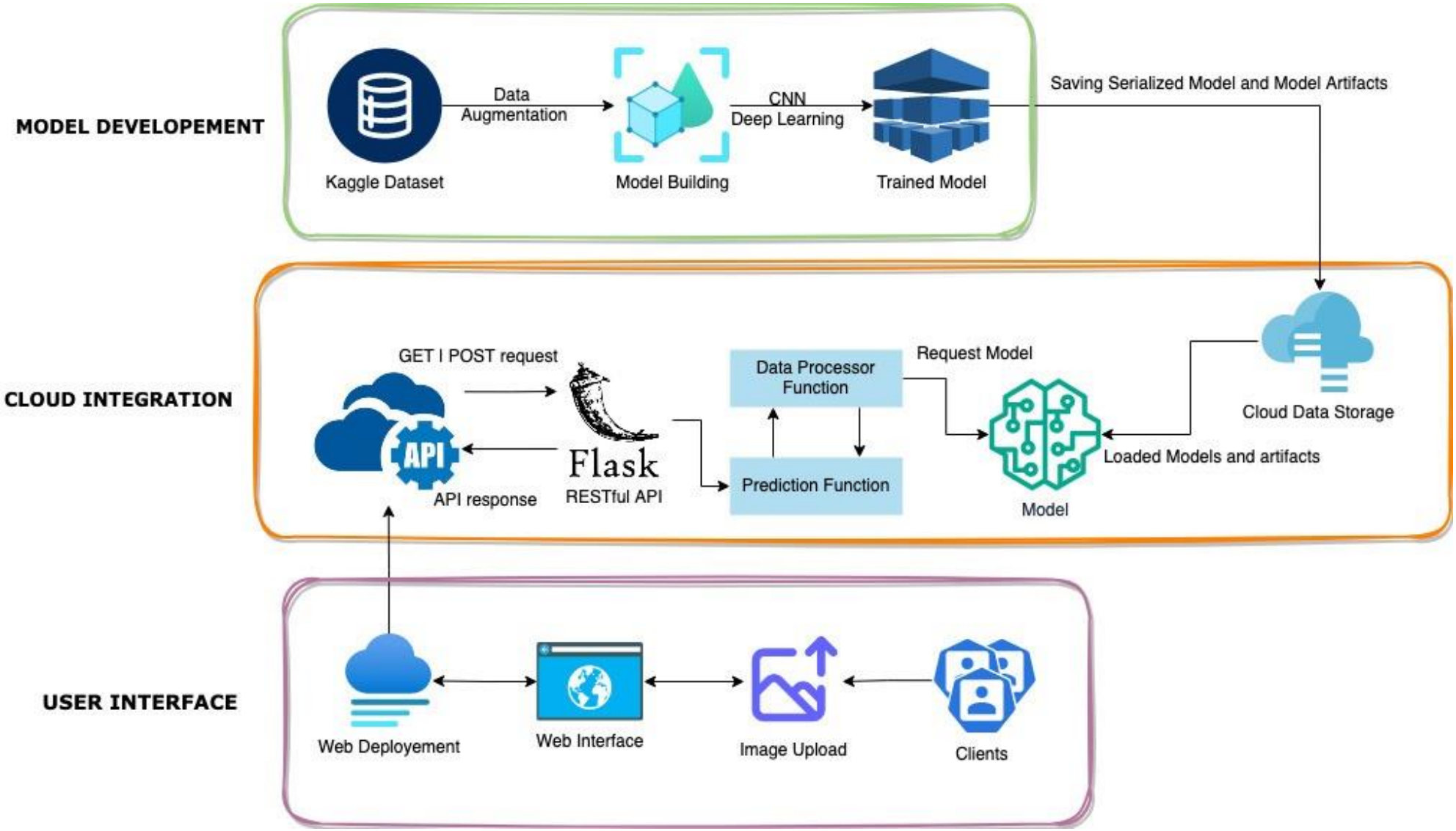
Project Design Phase-III
Technology Stack (Architecture & Stack)

Date	27 October 2023
Team ID	Team-593033
Project Name	**Fake/Real Logo Detection Using Deep Learning**
Team Members	Satyam Pundir, Ranpura Divy Dharmeshbhai, Samarth Patel, Utkarsh Gupta

Technical Architecture Diagram:



A) Technical Architecture:

The technical architecture for the Fake/Real Logo Detection Using Deep Learning would include the following components:

1. User Interface: Flask application with a simple web interface for user interaction.
2. Flask Application: This serves as the backend for the application and integrates the VGG19 model for logo image analysis.
3. VGG19 Model: Deep learning model implemented using TensorFlow and Python for image analysis and classification. It is responsible for distinguish between real and fake logos.
4. Kaggle API: Utilized to fetch the labelled logo dataset required for training and validation.
5. Cloud Computing: Infrastructure for scalable and reliable deployment of the Flask application and VGG19 model.
6. Web Server: Serving the Flask application and handling requests and acts as an entry point for user applications.

B) Open-Source Frameworks:

1. Flask: A micro web framework in Python for building the web application.
2. TensorFlow: Open-source deep learning framework for building and training neural networks, including the VGG19 model.
3. Keras: Open-source deep learning API written in Python, used as an interface for TensorFlow.
4. NumPy: A fundamental package for scientific computing with Python, used for numerical operations on images.
5. Pandas: A powerful data analysis and manipulation library for Python, utilized for handling datasets.

C) Third-party APIs:

1. Kaggle API: Used for accessing the labelled logo dataset from Kaggle.
2. Cloud API: potential for integration with cloud service APIs for storage and deployment

D) Cloud Deployment:

1. Amazon Web Services (AWS): Cloud platform for scalable, reliable, and secure deployment of the Flask application and VGG19 model. Services like Amazon S3 can be used for cloud storage.
2. Google Cloud Platform (GCP): An alternative cloud platform for deploying and managing the application and model. Google Cloud Storage can be used for storing datasets and models.
3. Microsoft Azure: Another option for deploying the application. Azure Blob Storage can be used for dataset and model storage.

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	User interaction with the application	HTML, CSS, JS, Flask
2.	Flask Application	Backend for the application	Python, Flask
3.	VGG19 Model	Deep learning Model for logo image analysis	TensorFlow, Keras, Python
4.	Kaggle API	Data retrieval for labelled logo dataset	Kaggle API
5.	Cloud Storage	Data and model storage	AWS S3 / Google Cloud Storage / Azure Blob Storage
6.	Cloud Computing	Infrastructure for scalable and reliable deployment	AWS, GCP, Azure
7.	Web Server	Serves the Flask application and handles requests	AWS, GCP, Azure

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Frameworks used for the project	Flask, TensorFlow, Keras, NumPy, Pandas
2.	Security Implementations	Security measures implemented	HTTPS, OAuth / AWS IAM / GCP IAM / Azure IAM
3.	Scalable Architecture	Architecture scalability justification	Load Balancers, Microservices
4.	Availability	Ensuring application availability	Failover systems, Disaster Recovery, AWS Availability Zones, GCP Regions, Azure Regions
5.	Performance	Design considerations for performance	Caching, CDNs, High-performance computing