

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

Date	03 October 2022
Team ID	PNT2022TMIDxxxxxx
Project Name	Project - xxx:Crime Vision: Advanced Crime Classification With Deep Learning
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: Order processing during pandemics for offline mode

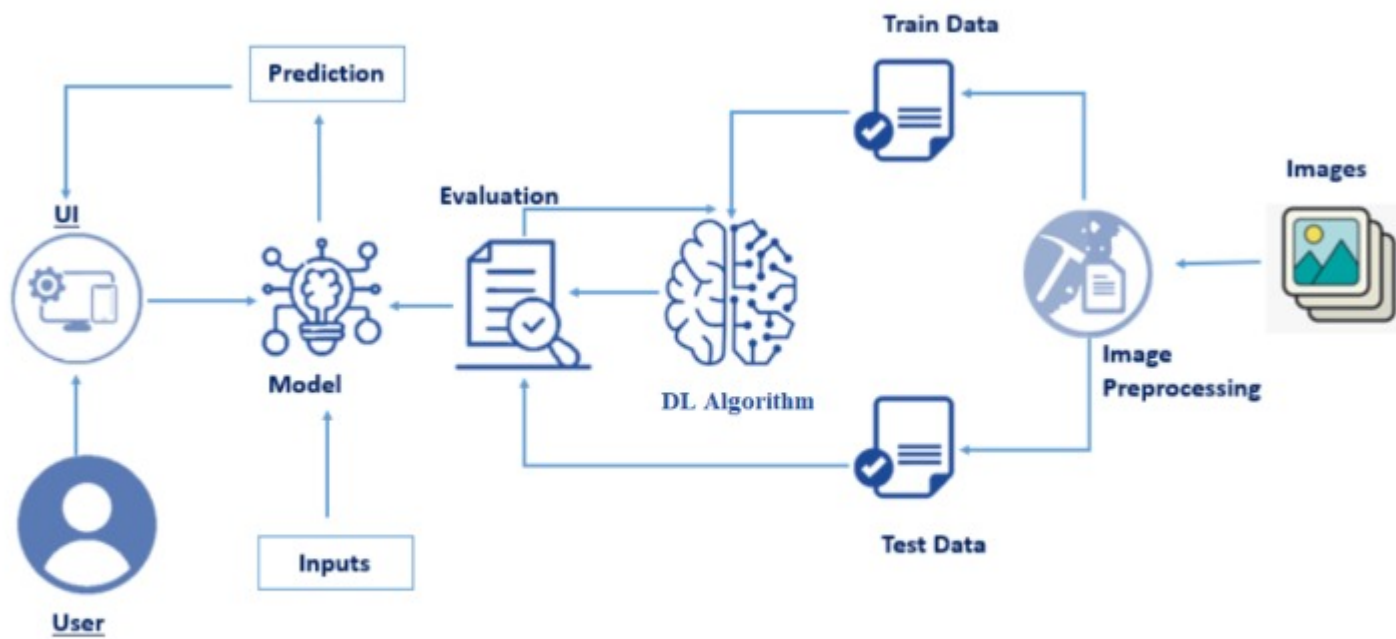


Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript / Angular Js / React Js etc.
2.	Application Logic-1	The logic of your application involves selecting a programming language compatible with your chosen deep learning framework.	Java / Python
3.	Application Logic-2	Consider the infrastructure needed for training your deep learning model.	IBM Watson STT service
4.	Application Logic-3	By incorporating IBM Watson Assistant into your project, you can enhance the user experience, improve accessibility, and streamline interactions with your crime detection system.	IBM Watson Assistant
5.	Database	Image Data;Text Data, Neural Network Architecture,Integration with IBM Watson Assistant,Continuous Integration/Continuous Deployment (CI/CD)	MySQL, NoSQL, etc.
6.	Cloud Database	A cloud-based database service provides scalable and managed database solutions, allowing users to store, retrieve, and manage data efficiently over the internet	IBM DB2, IBM Cloudant etc.
7.	File Storage	Establish a clear organizational structure for your files and consider metadata requirements for efficient file retrieval and	IBM Block Storage or Other Storage Service or Local Filesystem

		management	
8.	External API-1	Access to Third-Party Services,Social Media Integration,Machine Learning and AI Services	IBM Weather API, etc.
9.	External API-2	Data Aggregation,Data Retrieval,Customization and Extensibility	Aadhar API, etc.
10.	Machine Learning Model	Machine learning models are designed to generalize from existing data, allowing them to perform tasks without being explicitly programmed.	Object Recognition Model, etc.
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Install server software relevant to your needs, such as web servers (e.g., Apache, Nginx), database servers (e.g., MySQL, PostgreSQL), and application servers. Cloud Server Configuration :Select a cloud service provider (e.g., AWS, Azure, Google Cloud) based on your project requirements, budget, and preferences.	Local, Cloud Foundry, Kubernetes, etc.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	TensorFlow is an open-source machine learning framework developed by the Google Brain team. It is widely used for building and	Keras is an open-source high-level neural networks API written in Python. It is often used as a

		training deep learning models, including those for image recognition and classification. Pandas is an open-source data manipulation and analysis library for Python. It is commonly used for cleaning, transforming, and analyzing datasets.	frontend for TensorFlow and other deep learning frameworks, providing a simple and intuitive interface for building and training models.
2.	Security Implementations	Firewalls act as a barrier between a trusted internal network and untrusted external networks (like the internet)..	e.g. SHA-256, Encryptions, IAM Controls, OWASP etc.
3.	Scalable Architecture	The 3-tier architecture separates the application into three logical layers: presentation, application (business logic), and data. This separation allows each layer to scale independently based on specific requirements. Microservices architecture breaks down an application into small, independent services, each responsible for a specific business capability. This modularity allows for independent scaling of services based on their specific usage patterns and demands.	3-tier architecture

S.No	Characteristics	Description	Technology
4.	Availability	Ensuring high availability of an	Content Delivery Networks

		application is crucial to provide uninterrupted service to users. Various architectural patterns and technologies contribute to achieving high availability.	(CDNs)CDNs distribute static content across servers strategically located around the world.
5.	Performance	Design the application architecture to be horizontally scalable, allowing it to handle an increasing number of requests.	Leverage Content Delivery Networks to reduce latency and accelerate content delivery to users.

References:

<https://c4model.com/>

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

<https://www.ibm.com/cloud/architecture>

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

<https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d>