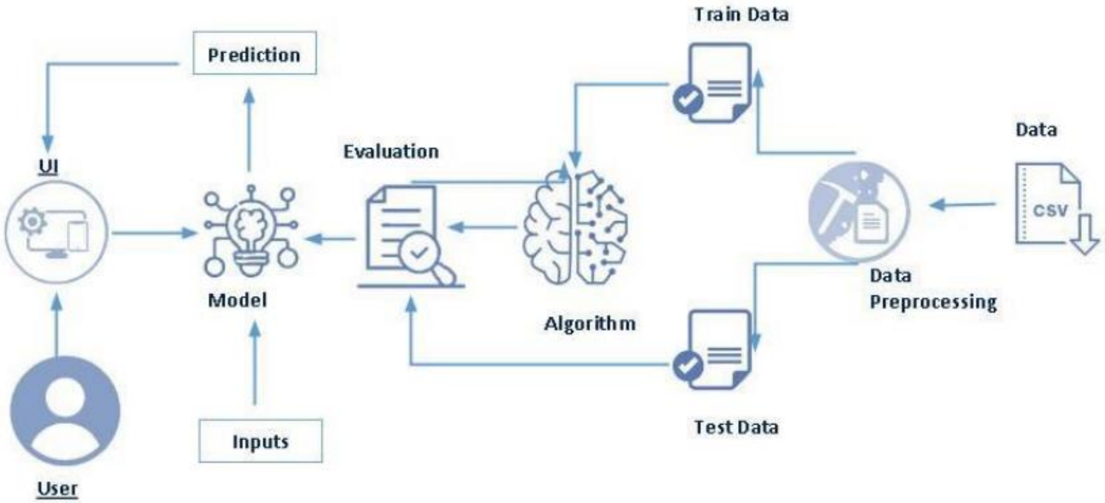
Project Design Phase-II
Technology Stack (Architecture & Stack)

Date	27 October 2023
Team ID	Team-593386
Project Name	Machine Learning Approach For Employee Performance Prediction
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: Machine Learning Approach For Employee Performance Prediction



Guidelines:

1. Include all the processes (As an application logic / Technology Block)
2. Provide infrastructural demarcation (Local / Cloud)
3. Indicate external interfaces (third party API's etc.)
4. Indicate Data Storage components / services
5. Indicate interface to machine learning models (if applicable)

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1	User Interface	User interaction with the application, e.g., Web UI, Mobile App, Chatbot, etc.	HTML, CSS, JavaScript, React JS, etc.
2	Data Collection	Gathering relevant employee data including demographics and performance metrics.	HR systems APIs, Database management systems (e.g., SQL)
3	Data Preprocessing	Cleaning, transforming, and preparing collected data for analysis.	Python (Pandas), Numpy, Data wrangling tools
4	Feature Selection	Identifying important features for prediction.	Feature importance analysis, Scikit-learn
5	Model Development	Choosing, training, and evaluating machine learning models.	Python (Scikit-learn, TensorFlow, PyTorch), Hyperparameter tuning, Cross-validation
6	Model Evaluation	Assessing model performance and accuracy.	Evaluation metrics (e.g., RMSE, MAE, R-squared), Visualization tools (e.g., Matplotlib, Seaborn)
7	Model Deployment	Implementing the model for real-world use, including user interface development.	Web application frameworks (e.g., Flask, Django), Cloud platforms (e.g., AWS, Azure), Containerization (e.g., Docker)
8	Documentation	Documenting the project and its processes, including code documentation and reporting.	Code documentation (e.g., Jupyter Notebooks), Documentation tools (e.g., Confluence, Git)
9	Testing and QA	Ensuring the model's reliability and accuracy, quality assurance, and bug tracking.	Testing frameworks (e.g., pytest), CI/CD pipelines (e.g., Jenkins)

Table-2: Application Characteristics:

S.No	Characteristic	Description	Technology
1	Scalability	The ability of the system to scale with increased data and usage.	Cloud computing platforms, load balancing, containerization
2	Real-time Processing	The capability to process data and provide predictions in real-time.	Stream processing frameworks (e.g., Apache Kafka, Apache Flink)
3	Data Security	Ensuring the protection and privacy of sensitive employee data.	Encryption, access controls, compliance with data protection regulations
4	Interpretability	Making machine learning models explainable and understandable.	Interpretability libraries (e.g., SHAP, Lime)
5	Model Monitoring	Continuously monitoring the model's performance and health.	Model monitoring tools, alerting systems
6	Data Quality	Ensuring the quality and accuracy of input data.	Data validation, data cleaning, data governance