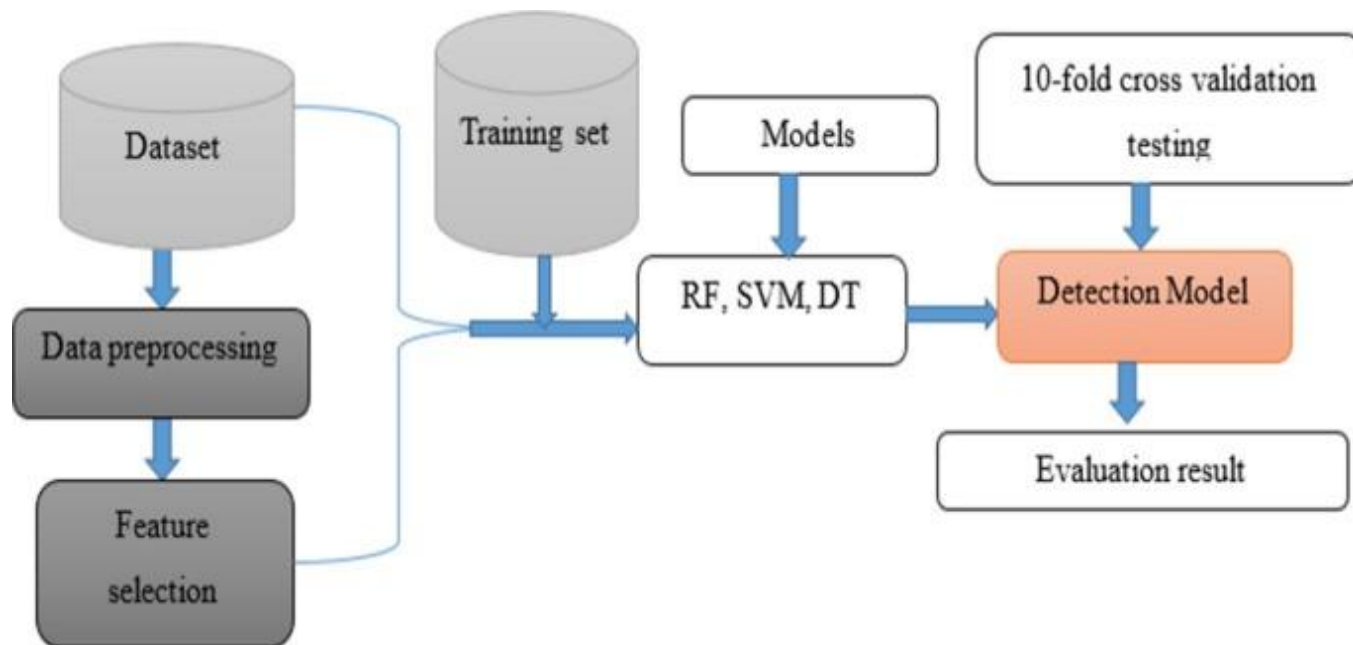


Project Design Phase-II Technology Stack (Architecture & Stack)

Date	8 November 2023
Team ID	592655
Project Name	Disease Prediction Using Machine 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



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.	Data Collection	Collects patient data for training the model	Healthcare databases, EHR systems, APIs
2.	Data Preprocessing	Cleans and prepares data for model input	Python (pandas, NumPy), scikit-learn
3.	Disease Classification ModulePython, TensorFlow, Keras	Module for classifying diseases	Python, TensorFlow, Keras
4.	Model Development	Develops the machine learning prediction model	TensorFlow, PyTorch, scikit-learn
5.	Model Deployment	Deploys the trained model for real-time predictions	AWS, Azure, Google Cloud, Docker
6.	Integration	Integration with external disease databases	RESTful APIs, Flask (Python), Django (Python)
7.	Frontend	User interface for input and displaying predictions	HTML, CSS, JavaScript, React, Angular, Vue
8.	Backend	Handles requests, communicates with the model	Node.js, Express (JavaScript), Flask (Python), Django (Python)
9.	Dashboard	Visualizes predictions and model metrics	Data visualization libraries (e.g., D3.js, Plotly), React, Angular, Vue
10.	User Authentication	Ensures secure user access to the system	

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	User Interface	Provides an interface for users to interact with the system and input data.	React, Angular, Vue, HTML, CSS, JavaScript
2.	Backend	Manages the business logic, handles requests, and communicates with the machine learning model.	Node.js, Express (JavaScript), Flask (Python), Django (Python)
3.	Data Storage	Stores and manages patient data and model-related information.	MySQL, PostgreSQL, MongoDB, Firebase
4.	Machine Learning Model	Develops, trains, and deploys the disease prediction model.	TensorFlow, PyTorch, scikit-learn
5.	API	Enables communication between different components of the system.	RESTful APIs, GraphQL

