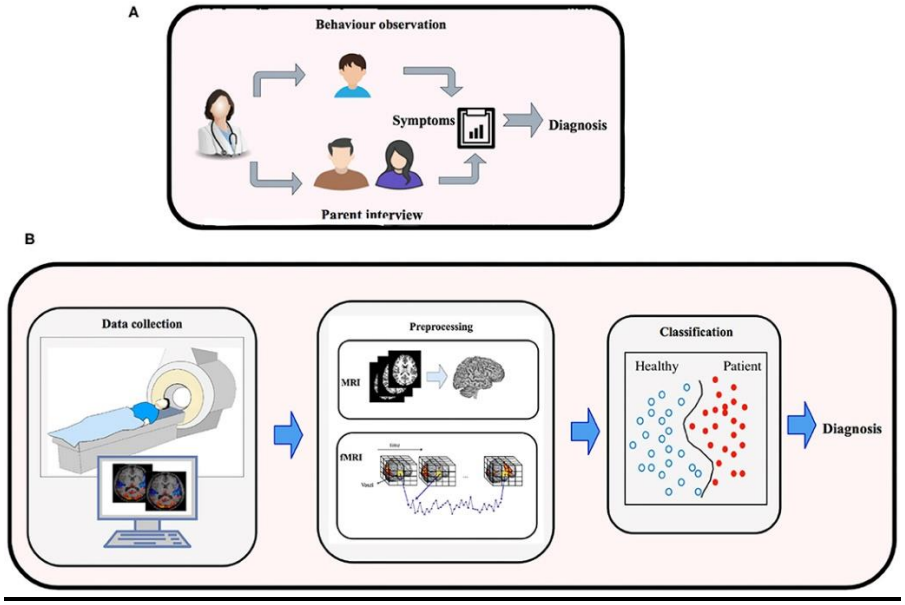
Technology Stack (Architecture & Stack)

Date	31 October 2023
Team ID	Team-592795
Project Name	Project - Predicting Mental Health Illness Of Working Professionals 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.	User Interface	Web/Mobile App for user interaction	HTML, CSS, JavaScript
2.	App Logic - Data Prep	Data preprocessing logic	Python
3.	App logic - STT	Speech-to-Text logic for audio data	IBM Watson STT service
4.	App logic - assistant	Conversational interface for user interaction	IBM Watson Assistant
5.	Database	Data storage,type,configurations	MySQL, NoSQL, etc.
6.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
7.	File Storage	File storage requirements	IBM Block Storage, Local FS
8.	External API-1	External API for health data	IBM health API, etc.
9.	External API-2	External API for additional data	Aadhar API, etc.
10.	Machine Learning Model	Machine Learning Model for predication	Object Recognition Model, etc.
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud	Local, Cloud Foundry, Kubernetes, etc.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Utilization of open-source frameworks for building the application	Scikit-learn, TensorFlow, PyTorch for machine learning models
2.	Security Implementations	Implementation of security measures and access controls to safeguard sensitive data	Encryption (e.g., SHA-256), Role-Based Access Control (RBAC), IAM Controls, Secure Communication Protocols
3.	Scalable Architecture	Justification of the architecture's scalability, whether it's a 3-tier architecture or micro-services	Microservices Architecture, Docker for containerization, Kubernetes for orchestration

S.No	Characteristics	Description	Technology
4.	Availability	Explanation of how availability is ensured, including the use of load balancers and distributed servers.	Load Balancers (e.g., NGINX, AWS Elastic Load Balancer), Distributed Server Configurations
5.	Performance	Design considerations to enhance application performance, including request handling, caching, and Content Delivery Networks (CDNs)	Caching (Redis, Memcached), CDN Integration (e.g., Cloudflare, Akamai), Optimized API Calls