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

Date	
Team ID	NM2023TMID03215
Project Name	Project - Unleashing The Potential Of Our Youth: A Student Performance Analysis
Maximum Marks	4 Marks

Technical Architecture:

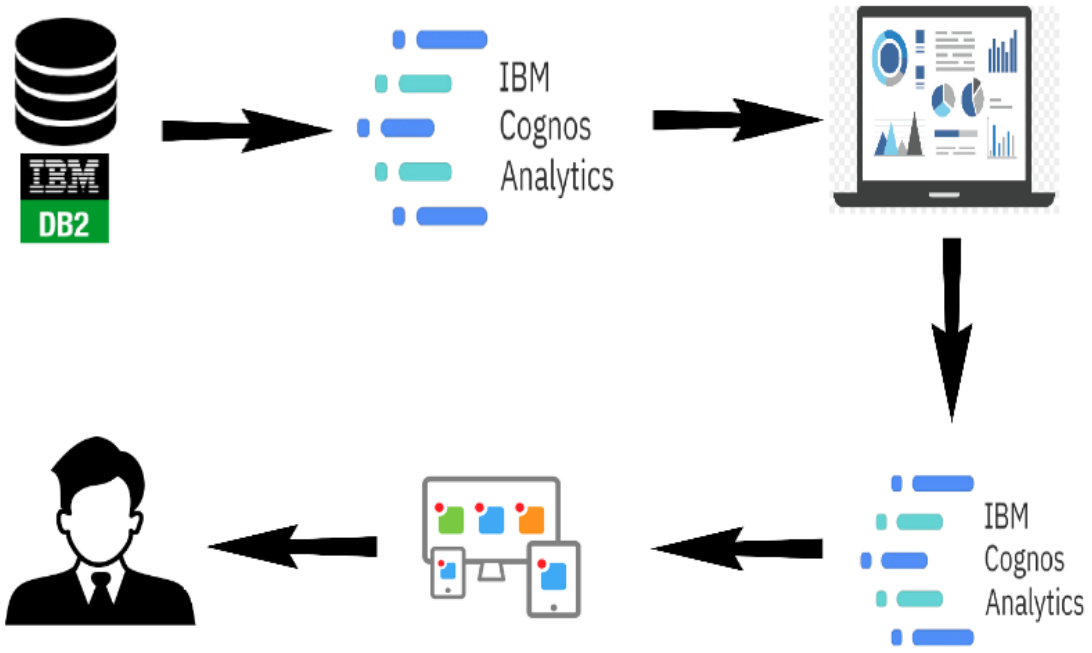

Table-1: Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	How user interacts with application e.g., Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript / Angular Js / React Js etc.
2.	Data Preprocessing and Cleaning	For accuracy and reliability	Java / Python
3.	Data analysis and Machine learning logic	To understand the distribution of data , trends , and patterns in student performance	IBM Watson STT service
4.	Report generation and Visualization logic	To effectively communicate insights and findings	IBM Watson Assistant
5.	Database	Data Type, Configurations etc.	MySQL, NoSQL, etc.
6.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
7.	File Storage	File storage requirements	IBM Block Storage or Other Storage Service or Local Filesystem
8.	Educational data APIs	Provide access to educational data, including student performance metrics, academic records, and demographic information	IBM Weather API, etc.
9.	Data storage APIs	For data storage and retrieval	Aadhar API, etc.
10.	Regression models	Used for predicting student outcomes, such as grades, test scores, or graduation rates	Object Recognition Model, etc.
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Data to be stored on premises Cloud Server Configuration: To leverage managed services for data analysis and machine learning	Local, Cloud Foundry, Kubernetes, etc.

Table-2: Application Characteristics

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	Technology of Opensource framework
2.	Security Implementations	List all the security / access controls implemented, use of firewalls etc.	e.g., SHA-256, Encryptions, IAM Controls, OWASP etc.
3.	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	Cloud platforms (Amazon Web Services (AWS), Microsoft Azure, or Google Cloud Platform (GCP))
S.No	**Characteristics**	**Description**	**Technology**
4.	Availability	Justify the availability of application (e.g., use of load balancers, distributed servers etc.)	Load Balancers (e.g., AWS Elastic Load Balancer, NGINX)
5.	Performance	Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN's) etc.	CDN (e.g., Cloudflare, AWS CloudFront)