Project Design Phase-I
Solution Architecture

Date	30 October 2023
Team ID	NM2023TMID03139
Project Name	The Future of Work: Data Analysis Of Glassdoor Jobs
Maximum Marks	4 Marks

Solution Architecture:

Data Collection and Storage Layer:

Use Glassdoor API to collect data on data science job postings.
Store the collected data in the DB2 database for efficient management and retrieval.

Data Processing and Analysis Layer:

Implement IBM Cognos Analytics for data processing, analysis, and visualization.
Conduct in-depth analysis of the collected data to derive meaningful insights.

Backend Layer:

Use Flask to build the backend of the web application.
Implement necessary APIs and endpoints for communication with the frontend and data processing components.

Frontend Layer:

Develop the user interface using HTML, CSS, and JavaScript.
Integrate the frontend with the Flask backend to enable seamless data retrieval and visualization.

Security Layer:

Implement robust security measures to protect user data and prevent unauthorized access.
Use encryption techniques to secure data transmission and storage.

Scalability and Performance Layer:

Deploy the application on a scalable and reliable cloud infrastructure such as AWS or Azure to ensure high performance and availability.
Optimize the database queries and system architecture for efficient data processing and rendering.

Monitoring and Maintenance Layer:

Implement monitoring tools to track application performance and detect potential issues.
Establish a regular maintenance schedule to update and maintain the system components.

Integration and Deployment:

Integrate the different components of the solution architecture to ensure smooth communication and functionality.
Deploy the application on a production server for public access.
This solution architecture will help you build a robust and scalable data science job market analysis application, allowing users to explore and understand the evolving trends and requirements in the data science job market. It also ensures efficient data processing, security, and seamless integration between different layers of the application.

Example - Solution Architecture Diagram: