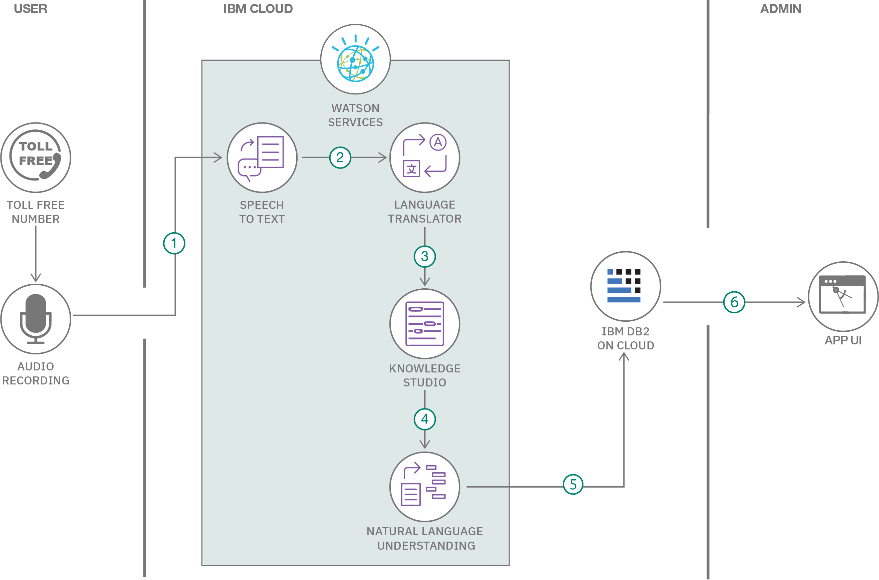
Project Design Phase-II Technology Stack (Architecture & Stack)

|  |  |
| --- | --- |
| Date | 17 October 2023 |
| Team ID | 90EDCD5A653F97E965540D7A2F454585 |
| Project Name | Quantitative Analysis Of Candidates In 2019 Lok Sabha Elections |
| 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: Order processing during pandemics for offline mode



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 | The User Interface component is responsible for providing an interactive and user-friendly interface for users to interact with the system. It allows users to input queries, view analysis results, and generate reports. | HTML, CSS, JavaScript / Angular Js / React Js etc. |
| 2. | Application Logic-1 | These components represent the application logic responsible for data processing, analysis, and business logic. | Java / Python |
| 3. | Application Logic-2 | These components represent the application logic responsible for data processing, analysis, and business logic. | Java / Python |
| 4. | Application Logic-3 | These components represent the application logic responsible for data processing, analysis, and business logic. | Java / Python |
| 5. | Database | The Database component is responsible for storing structured data related to candidates, constituencies, and election results. It allows for efficient data retrieval and management. | MySQL, PostgreSQL, or Oracle |
| 6. | Cloud Database | Cloud Database is a scalable and highly available data storage solution hosted on a cloud platform. It can be used to ensure data availability, redundancy, and scalability. | Amazon RDS, Google Cloud SQL, or Azure SQL Database. |
| 7. | File Storage | File Storage is used for storing unstructured or semi-structured data such as documents, images, and reports related to the election analysis. | Amazon S3, Google Cloud Storage, or Azure Blob Storage |
| 8. | External API-1 | These components are responsible for interacting with external data sources, such as election commission data or third-party APIs, to enrich the analysis with additional information. | RESTful APIs, GraphQL, or specific APIs |
| 9. | External API-2 | These components are responsible for interacting with external data sources, such as election commission data or third-party APIs, to enrich the analysis with additional information. | RESTful APIs, GraphQL, or specific APIs |
| 10. | Machine Learning Model | The Machine Learning Model component is responsible for predictive analysis, trend identification, or any machine learning-based tasks related to candidate analysis. | TensorFlow, or PyTorch. |
| 11. | Infrastructure (Server / Cloud) | Infrastructure represents the underlying hardware or cloud services on which the system is deployed. It ensures the availability, scalability, and reliability of the system. | AWS, Google Cloud, or Azure |

Table-2: Application Characteristics:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | Develop the user interface using open-source web frameworks | Technology of Opensource framework |
| 2. | Security Implementations | Implement security measures in the user interface | e.g. SHA-256, Encryptions, IAM Controls, OWASP etc. |
| 3. | Scalable Architecture | Design the UI to be scalable by following responsive web design principles and optimizing for various devices. | Technology used |
| 4. | Availability | Ensure high availability by using load balancers and failover mechanisms for the user interface servers. | Technology used |
| 5. | Performance | Optimize the user interface for fast loading and responsive user experiences, including caching static assets. | Technology used |