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

|  |  |
| --- | --- |
| Date | 31 January 3035 |
| Team ID | LTVIP2025TMID50416 |
| Project Name | Heritage Treasures: An In-Depth Analysis of UNESCO World Heritage Sites in Tableau |
| Maximum Marks | 4 Marks |

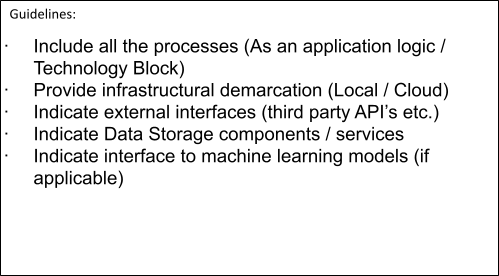
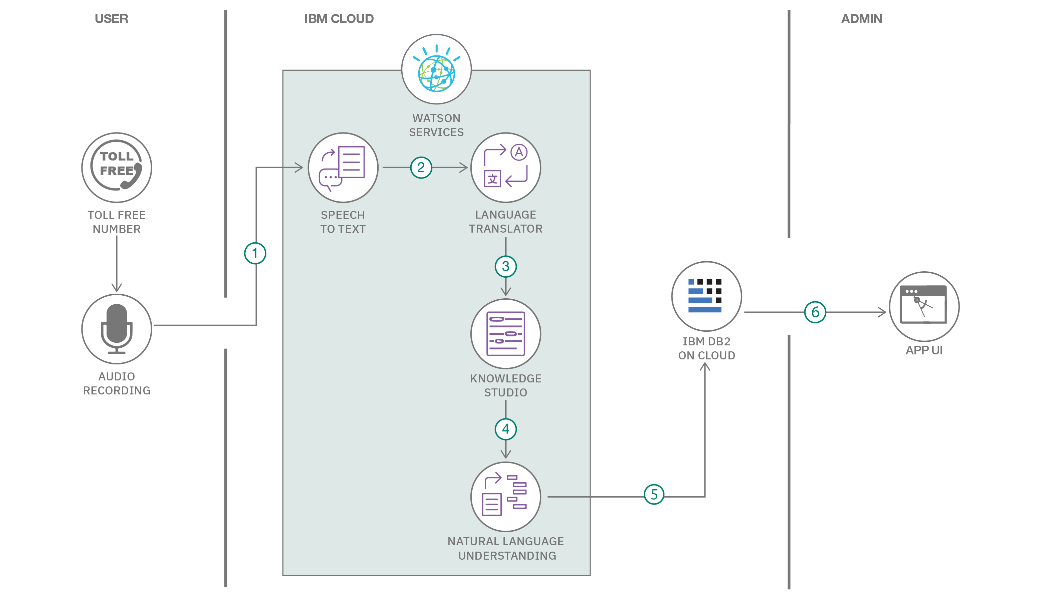
Technical Architecture:

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

Reference: <https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/>

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. |

|  |  |  |  |
| --- | --- | --- | --- |
| 2. | Data Ingestion & ETL | Extract, transform, and load heritage site data from multiple sources | Python (Pandas, Apache Airflow), Apache NiFi (optional) |
| 3. | Data Enrichment | Integration with external APIs for geographic and heritage metadata | Python REST clients, GeoNames API, OpenStreetMap API, UNESCO Metadata API |
| 4. | Analytics & Visualization | Business logic for metrics, KPIs, and visual calculations in dashboards | Tableau Calculated Fields, Tableau Parameters |
| 5. | Relational Database | Structured storage of master and reference data | MySQL, PostgreSQL. |
| 6. | NoSQL Database | Flexible storage for semi-structured heritage data and logs | MongoDB, Amazon DynamoDB |
| 7. | Cloud Data Warehouse | Scalable analytics-ready storage for large datasets | Google BigQuery, IBM Db2 Warehouse |
| 8. | File Storage | Storage for raw data files, images, and backups | Google Cloud Storage, IBM Cloud Object Storage |
| 9. | Infrastructure & Deployment | Cloud infrastructure and dashboard hosting | IBM Cloud, Tableau Server |

Table-2: Application Characteristics:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | Open-source tools and frameworks for data processing, and backend services | Python (Pandas, Flask, Requests) |
| 2. | Security Implementations | Data encryption, secure authentication, and fine-grained access control | TLS/SSL, Network Firewalls,VPN |
| 3. | Scalable Architecture | Modular, scalable architecture supporting increased data volumes and user load | Microservices architecture. |
| 4. | High Availability | Ensuring minimal downtime and fault tolerance | Load balancers, Multi-region deployments, Tableau Server clustering |

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 5. | Performance Optimization | Fast dashboard load times through caching, query optimization. | Tableau Data Extracts. |

References:

<https://c4model.com/>

<https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/> <https://www.ibm.com/cloud/architecture>

<https://aws.amazon.com/architecture>

<https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d>