**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 31 January 2025 |
| Team ID | LTVIP2025TMID49051 |
| 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/**](https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/)

Guidelines:

Include all the processes (As an application logic / Technology Block)

Provide infrastructural demarcation (Local / Cloud)

Indicate external interfaces (third party API’s etc.)

Indicate Data Storage components / services

Indicate interface to machine learning models (if applicable)



**Table-1 : Components & Technologies:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
|  | User Interface | Dashboard for visual interaction | Tableau Public / Tableau Desktop |
|  | Application Logic-1 | Data preprocessing and filtering logic | Python (Pandas/Seaborn) |
|  | Application Logic-2 | Data connection to Tableau | Tableau Data Extract API / CSV Import |
|  | Application Logic-3 | Risk prediction based on conditions | IBM Watson Assistant |
|  | Database | Where raw site data is stored | MySQL, NoSQL, etc. |
|  | Cloud Database | Only if using cloud storage for CSV | e.g., Google Drive / Dropbox / AWS S3 |
|  | File Storage | Data files used in Tableau | Local File System / Cloud Storage |
|  | External API-1 | UNESCO API if using live data | UNESCO Open Data API |
|  | External API-2 | Country info from external API | RestCountries API |
|  | Machine Learning Model | Predictive model for inscription risk | Object Recognition Model, etc. |
|  | Infrastructure (Server / Cloud) | Where system is deployed | Local System / Tableau Online |

**Table-2: Application Characteristics:**

| **S.No** | **Characteristics** | **Description** | **Technology** |
| --- | --- | --- | --- |
|  | Open-Source Frameworks | Python data libraries used | Pandas, Seaborn, Matplotlib |
|  | Security Implementations | Data file access and file permissions | File-level access / Tableau Cloud IAM |
|  | Scalable Architecture | Can be extended to multiple dashboards or real-time data | Tableau Server, Python Scripts |
|  | Availability | High availability through Tableau Public or Cloud Storage | Tableau Online / Google Drive |
|  | Performance | Optimized by pre-cleaning data and publishing static dashboards | Tableau Engine, Cached Extracts |

**References:**

[**https://c4model.com/**](https://c4model.com/)

[**https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/**](https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/)

[**https://www.ibm.com/cloud/architecture**](https://www.ibm.com/cloud/architecture)

[**https://aws.amazon.com/architecture**](https://aws.amazon.com/architecture)

[**https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d**](https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d)