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

Date	31 January 2026
Team ID	LTVIP2026TMIDS24242
Project Name	Gemini Historical Artifact Description
Maximum Marks	4 Marks

Technical Architecture:

Table-1: Components & Technologies

S.No	Component	Description	Technology
1.	**User Interface**	Museum-themed web interface for users to input text or upload images.	Start: **Streamlit** (Python-based UI framework), HTML, CSS (Custom Styling)
2.	**Application Logic**	Core logic for handling user inputs, API communication, and state management.	**Python 3.8+**
3.	**Generative AI Service**	AI Engine for generating historical descriptions and analyzing images.	**Google Gemini 2.5 Flash**
4.	**Image Processing**	Logic for handling and encoding uploaded images before API transmission.	**Pillow (PIL)** Library
5.	**Environment Config**	Management of sensitive credentials like API keys.	**python-dotenv**
6.	**File Storage**	Temporary in-memory handling for uploads; local storage for downloads.	**Python** io **module**, Local Filesystem

7.	**External API**	The primary AI service used for content generation.	**Google Generative AI API**
8.	**Infrastructure**	Deployment environment for running the Python application.	**Local Machine** (Development) / **Streamlit Community Cloud** (Production)

Table-2: Application Characteristics

S.No	Characteristics	Description	Technology
1.	**Open-Source Frameworks**	The core framework used to build and deploy the web application quickly.	**Streamlit**
2.	**Security Implementations**	Secure handling of API keys to prevent exposure.	**Environment Variables (.env)** / **Streamlit Secrets**
3.	**Scalable Architecture**	The app is stateless, allowing multiple instances to run independently behind a load balancer.	**Cloud-Native / Serverless** (when deployed on Cloud)
4.	**Availability**	High availability is managed by the cloud platform hosting the app.	**Streamlit Cloud** / **Docker Container** orchestration
5.	**Performance**	Basic caching of static resources and optimized API calls for faster response.	**Streamlit Caching (** @st.cache_data **)**, **Gemini Flash Model** (Low Latency)