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

Date	31 January 2026
Team ID	LTVIP2026TMIDS61478
Project Name	Gemini historical Artifact Description
Maximum Marks	4 Marks

Technical Architecture

Project: Gemini Historical Artifact Description

Architectural Diagram (3-Tier AI-Based Architecture)

Architecture Flow (Textual Representation)

User (Web / Mobile)
→ User Interface (React / HTML UI)
→ Backend API (Python Flask/Django)
→ AI Processing Layer (Gemini API / Image Recognition Model)
→ Database (User Data + Artifact Records)
→ Cloud Storage (Uploaded Images)
→ Response (Artifact Name + Historical Description)

Layer Explanation

1. Presentation Layer

- **Web UI for uploading artifact images**
- **Displays AI-generated historical description**

2. Application Layer

- **Authentication Module**

- **Image Processing Module**
- **AI Model Integration Module**
- **Description Generation Module**

3. Data Layer

- **User Database**
- **Artifact Metadata Database**
- **Cloud Image Storage**

Table-1: Components & Technologies

S.No	Component	Description	Technology
1	User Interface	Web interface for image upload and viewing description	HTML, CSS, JavaScript, React JS
2	Application Logic-1	User authentication and request handling	Python (Flask / Django)
3	Application Logic-2	Image preprocessing and AI request handling	Python
4	Application Logic-3	Artifact description generation using AI model	Gemini API / TensorFlow
5	Database	Stores user data and artifact records	MySQL / MongoDB
6	Cloud Database	Managed database service	AWS RDS / MongoDB Atlas
7	File Storage	Stores uploaded artifact images	AWS S3 / Local File System
8	External API-1	AI-based content generation	Gemini API
9	External API-2	Email verification service	SMTP / SendGrid API

S.No	Component	Description	Technology
10	Machine Learning Model	Image recognition & historical description generation	CNN Model / Generative AI Model
11	Infrastructure (Server / Cloud)	Application deployment	Local Server / AWS EC2 / Docker

Local Server Configuration:

- 8GB RAM
- Intel i5 / equivalent processor
- 256GB SSD

Cloud Server Configuration:

- AWS EC2
- Docker Container Deployment
- Kubernetes (for scaling)

Table-2: Application Characteristics

S.No	Characteristics	Description	Technology
1	Open-Source Frameworks	Frameworks used for development	Flask / Django / React JS
2	Security Implementations	Secure authentication and encrypted communication	SHA-256, JWT, HTTPS, IAM Controls, OWASP Guidelines
3	Scalable Architecture	3-Tier architecture with containerization	Docker, Kubernetes
4	Availability	Cloud hosting with distributed infrastructure	AWS EC2, Load Balancer

S.No	Characteristics	Description	Technology
5	Performance	Fast AI response and optimized request handling	Redis Cache, CDN