

Project Design Phase-II

Technology Stack (Architecture & Stack)

Date: April 14, 2025
Team ID: SWTID1742745633
Project Name: ViewVoyage
Maximum Marks: 4 Marks

Technical Architecture

ViewVoyage uses a modern **MERN stack architecture** to deliver a high-performing, scalable video-sharing platform. The architecture supports secure authentication, video playback, interactive features, and responsive design optimized for both desktop and mobile devices.

Table-1: Components & Technologies

S.No	Component Description	Technology Used
1	User Interface: Web UI to browse, search, and interact with videos. Includes theme switcher and animations.	React 18.x, Bootstrap 5, CSS3
2	Application Logic - Auth: Handles login, registration, JWT authentication, and route protection.	Node.js 20.x, Express.js 4.x, JWT, bcrypt
3	Application Logic - Video: Manages video upload, detail retrieval, and streaming logic.	Node.js, Express.js, Multer
4	Application Logic - User Interaction: Handles liking, commenting, and saving videos to lists.	Express.js, Mongoose 8.x
5	Database: Stores users, videos, comments, and likes as JSON documents.	MongoDB 7.x
6	File Storage: Local server stores uploaded videos and thumbnails.	Local Filesystem (Multer/File System)
7	Infrastructure: Local deployment with Node server; optionally cloud-deployable.	Vite, Node.js, MongoDB

Table-2: Application Characteristics

S.No	Characteristic	Description	Technology Used
1	Open-Source Frameworks	Fully built using open-source frontend, backend, and DB technologies.	React, Express, MongoDB, Mongoose, Bootstrap
2	Security Implementations	Uses token-based auth and secure password handling.	JWT, bcrypt, HTTPS
3	Scalable Architecture	3-tier architecture (UI, Logic, Data) allows for vertical and horizontal scaling.	MongoDB sharding (future), Express.js
4	Availability	MongoDB replication ensures data remains accessible even if one instance fails.	MongoDB Replica Sets
5	Performance Optimization	Indexing, caching, and optimized queries support high concurrent usage.	MongoDB Indexes, optional Redis