CineScope

An AI-Powered Movie Discovery and Collaboration Platform

Project Proposal

Course Code: CS4032 – Web Programming

Submitted By:

Name: Daud Tariq
Roll No: 21L-7746
Department: Computer Science
University: FAST National University of Computer and Emerging Sciences, Lahore Campus
Email: l217746@lhr.nu.edu.pk

1. Project Overview

CineScope is an AI-powered web application that helps users discover movies, track what they watch, and engage socially with friends through reviews and recommendations. It blends the personalized intelligence of AI with the social interactivity of a movie community. By combining real-time movie data, user analytics, and collaborative features, CineScope offers a complete entertainment companion experience.

The project aligns with three foundational pillars emphasized in this course:

a) Fundamental Business Logic: Core functionality like user management, CRUD operations, and data flow between backend and APIs.

b) Personal Productivity: Personalized dashboards, AI recommendations, and watch analytics.

c) Collaboration: Friend networks, discussion threads, and shared watchlists.

2. Objectives

- Build a dynamic, responsive web app that allows users to explore, review, and manage movie data.

• Integrate third-party APIs (TMDB, Google Maps) and AI-based recommendation systems.

• Develop user productivity tools (dashboards, analytics, and exportable

reports). • Enable collaborative engagement via shared lists, comments, and

notifications.

3. Technology Stack

• Frontend: React.js, Redux Toolkit, TailwindCSS

• Backend: Node.js, Express.js

• Database: MongoDB (Atlas)

• AI Integration: Content-based recommender system + optional OpenAI API •

Third-Party APIs: TMDB API for movie data, Google Maps API for location •

Hosting: Vercel (frontend), Render/Railway (backend)

• Version Control: Git + GitHub

4. Functional Requirements (Mapped to Project Guidelines)

1. Dashboard with Navbar and Sidebar: Interactive dashboard showing user stats, quick links to watchlist, and recommendations.

2. Login / Sign Up: Secure authentication using JWT tokens; registration via email.

3. Multi-Device Optimization: Responsive UI for desktop, tablet, and mobile de vices using TailwindCSS.

4. Contact and Google Location: Contact section with an embedded Google Map showing the project's development hub.

5. Photos and Biography: User profiles include photo, biography, and favorite movie genres.

6. Blogs: Blog section for movie reviews, trends, and entertainment news.

7. Social Media Buttons: Buttons for sharing reviews or watchlists to Instagram,

Facebook, or X (Twitter).

8. Well-Designed and Functional UI: Minimal and cinematic interface using Tail windCSS components.
9. Ease of Use: Clearly labeled sections, smooth navigation, and intuitive watchlist management.
10. High Security: User passwords hashed via bcrypt; secure API endpoints validated via JWT.
11. Redux: Used for global state management (user data, reviews, recommendations).
12. AI Recommendation System: Suggests movies based on user watch history and genre preferences using TMDB metadata.
13. CRUD Operations: For watchlists, reviews, blogs, and group discussions.
14. Cash Flow Statements (Debit + Credit): Optional premium plan mockup showcasing transaction logs and usage insights.
15. PDF / Certificate Generation: Allows exporting of viewing history and yearly analytics as PDF.
16. Search Bar: Dynamic search connected to TMDB API for movies and actors.
17. Good Error Handling: Custom error pages and backend middleware for failed API or DB operations.

18. Valid and Clean Code: Code formatted using ESLint and Prettier; modular and reusable components.
19. User Profiles: Each user can personalize their account with photo, bio, and pri vacy options.
20. Integration with Third-Party Services: TMDB for movie data, Google Maps for contact, OpenAI API for AI summaries.
21. Analytics Dashboard: Charts (via Recharts) for genre frequency, time spent watching, and review stats.
22. User Notifications: In-app alerts when friends review or add movies to shared lists.
23. Sub-domain: Deployed on a custom subdomain: cinescope.daudtariq.vercel.app.

5. Implementation Plan

- Phase 1: UI Design and Component Layout (Week 1–2)
- Phase 2: API Integration and CRUD Logic (Week 3–4)
- Phase 3: AI Recommendation and Analytics Modules (Week 5)
- Phase 4: Collaboration Features (Week 6)
- Phase 5: Testing, Security, and Deployment (Week 7)

6. Expected Outcome

CineScope will be a feature-rich, responsive, and AI-powered movie discovery web plat form that integrates personalization, productivity, and collaboration. It will demonstrate advanced use of APIs, modern web frameworks, and AI-assisted features—fulfilling all 23 project guidelines.

7. Group Member(s)

- Daud Tariq (21L-7746) – FAST NUCES Lahore, Department of Computer Science