

yMedia: A Lightweight Video Streaming and Sharing Platform

Mohammad Shifat-E-Rabbi

Department of Computer Science and Engineering

North South University

Dhaka, Bangladesh

rabbi.mohammad@northsouth.edu

Antara Labiba

Department of Computer Science and Engineering

North South University

Dhaka, Bangladesh

antara.labiba02@northsouth.edu

Md. Yeasin Hossain

Department of Computer Science and Engineering

North South University

Dhaka, Bangladesh

yeasin.hossain01@northsouth.edu

Pallab Biswas

Department of Computer Science and Engineering

North South University

Dhaka, Bangladesh

pallab.biswas@northsouth.edu

Abstract—Video streaming platforms have become essential tools for digital content consumption and creation; however, many existing solutions prioritize feature expansion over usability. This paper presents yMedia, a lightweight video streaming and sharing platform designed to emphasize clarity, performance, and essential functionality. The system focuses on organized content discovery, intuitive user interaction, and streamlined video management while avoiding unnecessary interface complexity. Built using the MERN stack and Firebase Cloud Storage, yMedia demonstrates a balanced approach to modern web application design with scalability and user experience as core objectives.

Index Terms—Video streaming, MERN stack, user experience, content sharing, web application

I. INTRODUCTION

Video streaming and sharing platforms have become an integral part of modern digital ecosystems, supporting entertainment, education, communication, and creative expression. As these platforms evolve, they often introduce increasingly complex features, dense interfaces, and algorithm-driven content delivery systems. While these additions benefit advanced users and advertisers, they can negatively affect usability for casual viewers and emerging creators.

The yMedia project aims to address this imbalance by developing a lightweight, intuitive, and user-focused video streaming platform. Instead of replicating the scale and complexity of existing services, yMedia emphasizes clarity, performance, and essential functionality. Users can watch, upload, and manage video content efficiently without unnecessary distractions.

This project was developed as part of the CSE299 (Junior Design) course and applies theoretical concepts of software engineering, full-stack development, and user experience design in a practical environment.

II. BACKGROUND AND LITERATURE REVIEW

A review of existing video-sharing platforms such as YouTube and Vimeo reveals a strong trend toward feature-heavy and algorithm-driven systems. While powerful, these

platforms often introduce usability challenges including cluttered interfaces, longer load times, and steep learning curves.

Studies on user engagement suggest that excessive recommendation mechanisms may limit visibility for new or smaller creators. Additionally, performance remains a concern for users accessing platforms through mobile devices or limited bandwidth networks. These findings highlight the need for a simpler, more accessible alternative.

III. PROJECT MOTIVATION

The motivation behind yMedia stems from the observation that many users do not require monetization tools or advanced analytics to enjoy or share content. A significant user group seeks a calm, organized environment for browsing and uploading videos.

yMedia adopts a functional minimalism approach, offering essential features in a visually clean and predictable layout. The goal is to provide familiarity and ease of use while maintaining technical scalability.

IV. PROJECT GOALS AND OBJECTIVES

The primary objective of yMedia is to design and implement a clean, scalable video streaming platform using modern web technologies. The system supports secure authentication, smooth video upload and playback, category-based content discovery, and meaningful interaction through likes, comments, and reposts.

Consistency in layout, responsive design, and intuitive navigation are emphasized to ensure a positive user experience across devices.

V. PROPOSED SYSTEM OVERVIEW

yMedia is developed as a full-stack web application based on the MERN stack. Users can create accounts, manage profiles, upload videos, and interact with content through a streamlined interface.

TABLE I
TECHNOLOGY STACK OVERVIEW

Layer	Technology Used
Frontend	React.js, HTML, CSS, JavaScript
Backend	Node.js, Express.js (REST API)
Database	MongoDB (NoSQL)
Video Storage	Firebase Cloud Storage
Authentication	JSON Web Token (JWT)
State Management	Redux Toolkit

Category-based browsing and repost functionality enhance content discoverability, while a structured settings page allows users to manage personal information and uploaded content efficiently.

VI. SYSTEM ARCHITECTURE AND TECHNOLOGY STACK

The system follows a modular, layered architecture to ensure maintainability and future extensibility. Each layer is designed to operate independently while communicating through well-defined interfaces.

The frontend is implemented using React.js, enabling component-based development, responsive layouts, and efficient state management. The backend is built with Node.js and Express.js, providing RESTful APIs for authentication, video management, and user interaction.

MongoDB is used as the primary database for storing user data, video metadata, comments, likes, and repost records. Firebase Cloud Storage is integrated for secure and scalable video file storage. Authentication and authorization are handled using JSON Web Tokens (JWT), ensuring secure access to protected resources.

This architecture allows new features to be introduced with minimal impact on existing components, making the system suitable for future growth.

VII. FEATURE DESIGN AND FUNCTIONAL SCOPE

yMedia supports both viewers and creators through well-defined features. User management enables secure registration, login, and profile control. Video management allows users to upload, categorize, and delete videos.

Interaction features such as likes, comments, and reposts encourage engagement without overwhelming users. Category-specific banners improve content discovery, and Progressive Web App support enhances accessibility.

VIII. IMPLEMENTATION PROGRESS

A stable project structure with separate frontend and backend directories has been established. Core functionalities including authentication, video upload, and category browsing have been implemented and tested.

Firebase integration and MongoDB connectivity have been successfully configured. UI consistency has been improved through iterative refinement of layouts and components.

IX. CHALLENGES FACED AND SOLUTIONS

Maintaining visual consistency across pages required multiple iterations and was resolved through reusable components and shared styles. Video storage challenges were addressed using Firebase's scalable cloud infrastructure.

State synchronization between frontend and backend was handled through structured API responses and controlled state updates.

X. FUTURE SCOPE AND ENHANCEMENTS

Future enhancements include advanced search and filtering, creator analytics dashboards, and real-time notifications. Additional personalization options may be introduced while preserving the platform's simplicity.

The modular system design allows these features to be added without significant architectural changes.

XI. CONCLUSION

yMedia demonstrates a balanced approach to video streaming platform design by prioritizing usability, clarity, and performance. Instead of competing with large-scale platforms, it focuses on delivering essential features in an accessible and organized manner.

The MERN stack and Firebase Cloud Storage provide a reliable technical foundation, while the system's modular architecture supports long-term scalability. Overall, yMedia reflects effective application of software engineering principles and serves as a strong foundation for future development.

REFERENCES

- [1] MongoDB Inc., "MongoDB Documentation." [Online]. Available: <https://www.mongodb.com/docs/>
- [2] OpenJS Foundation, "Node.js Documentation." [Online]. Available: <https://nodejs.org/en/docs>
- [3] Express.js, "Express Framework Documentation." [Online]. Available: <https://expressjs.com/>
- [4] Meta Platforms Inc., "React Documentation." [Online]. Available: <https://react.dev/>
- [5] Google LLC, "Firebase Cloud Storage Documentation." [Online]. Available: <https://firebase.google.com/docs>
- [6] North South University, "CSE299 Course Guidelines and Materials."