

Sound Wave: A Dual-Mode Secure Music Streaming Web Application

DR. A. Kovalan¹, S. Abhinivesh², S. Siva Manikanta³, S. Nandhishvhar⁴

¹Associate Professor, Department of Artificial Intelligence and Machine learning, Dhanalakshmi Srinivasa University, Tamil Nadu, India

^{2,3,4}Department of Artificial Intelligence and Data Science, Dhanalakshmi Srinivasan University, Tamil Nadu, India

Abstract - Music streaming applications have significantly transformed digital media consumption by enabling on-demand access to audio content. This paper presents Sound Wave, a full-stack music streaming web application designed to deliver secure authentication, efficient local audio streaming, and synchronized lyrical playback. The system is implemented using HTML, CSS, JavaScript, Node.js, Express.js, and MongoDB following the Model-View Controller (MVC) architectural pattern. A novel Dual Mode feature allows users to switch seamlessly between standard audio playback and synchronized lyric display without interrupting streaming. Performance optimization is achieved through HTTP range requests and partial content delivery mechanisms. Experimental evaluation confirms stable streaming performance, accurate lyric synchronization, and secure user authentication. The proposed system demonstrates scalable and modular design principles suitable for real-world deployment.

Keywords - Music Streaming, Full-Stack Development, Secure Authentication, Audio Streaming, Lyrics Synchronization.

I. INTRODUCTION

1.1 Emergence of Music Streaming Systems

The evolution of digital technology has significantly transformed the way people consume music. Earlier, users relied on physical media and offline downloads to access audio content. However, with the advancement of high-speed internet and web technologies, music streaming platforms have emerged as a dominant mode of media consumption. Modern music streaming systems provide users with instant access to large collections of songs without requiring local storage. These platforms offer features such as real-time playback, playlist creation, and

cross-device accessibility. Additionally, users increasingly expect interactive functionalities like synchronized lyrics and personalized recommendations.

As a result, the demand for scalable, efficient, and user-friendly streaming systems has grown rapidly. This has led to the development of advanced architectures and technologies that support seamless audio delivery and enhanced user experience.

1.2 Motivation and Problem Statement

Despite the popularity of existing music streaming platforms, several challenges remain unresolved. One of the primary concerns is secure user authentication, as applications must protect sensitive user data from unauthorized access.

Another major issue is efficient audio streaming. Traditional methods may lead to buffering delays and high bandwidth consumption, which negatively impact user experience. Therefore, optimized streaming techniques are required to ensure smooth and uninterrupted playback.

Furthermore, many recommendation systems fail to adapt to users' changing preferences over time. This problem, known as user interest drift, results in less accurate and repetitive recommendations. Existing systems often lack dynamic mechanisms to update recommendations in real time.

These limitations highlight the need for a system that combines secure authentication, efficient streaming, and intelligent recommendation techniques to enhance overall performance and user satisfaction.

II. LITERATURE SURVEY

2.1 Project Objectives

The main objectives of the proposed system are:

- To develop a secure music streaming web application
- To implement JWT-based authentication for user security
- To enable efficient audio streaming using HTTP Range Requests
- To provide synchronized lyric display for better user interaction
- To design a hybrid recommendation system for personalized music suggestions
- To address the issue of user interest drift using game theory concepts
- To build a scalable and modular system using MVC architecture

II. LITERATURE SURVEY

2.1 The growth of internet technologies and digital platforms has led to the rapid development of music streaming systems.

- Modern streaming applications focus on delivering audio efficiently while ensuring high performance and user satisfaction.

2.2 Music Streaming Technologies

- Use of Content Delivery Networks (CDNs) helps reduce latency and improve content delivery speed.
- HTTP-based streaming is widely used for efficient transmission of audio data.
- HTTP Range Requests enable partial content delivery instead of full file transfer.
- This approach reduces buffering time, optimizes bandwidth usage, and ensures smooth playback.

2.3 Recommendation Systems

- Recommendation systems enhance user experience by suggesting relevant music.
- Collaborative filtering is commonly used to analyze user behaviour and preferences.
- Challenges include: o Data sparsity o Cold start problem
- Hybrid recommendation models combine multiple techniques to improve accuracy and personalization.

2.4 User Interest Drift

- User preferences change over time, known as interest drift.
- Traditional systems fail to adapt dynamically to these changes.
- Existing solutions include:
 - o Time-window models (focus on recent activity)
 - o Long-short term models (combine past and recent behaviour)
- Limitations: high complexity and limited adaptability

2.5 Game Theory in Recommendation

- Game theory treats interaction between user and system as a dynamic process.
- The system updates recommendations based on user responses.
- Helps in:
 - o Handling interest drift
 - o Improving real-time recommendation accuracy

2.6 Security in Web Applications

- Security is essential to protect user data and privacy.
- Traditional session-based authentication has limitations.
- JWT (JSON Web Token) provides secure, stateless authentication.
- Password hashing ensures safe storage of user credentials.

III. PROPOSED METHODOLOGY

A System Architecture and Workflow

The system consists of:

1. User Authentication Process
2. Music Selection and Request Handling
3. Audio Streaming Mechanism
4. Lyrics Synchronization Process
5. Recommendation and Data Update Process

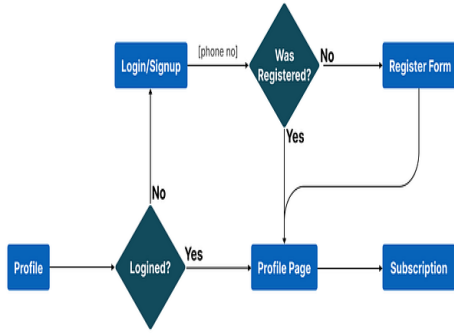


Fig. 1. System Architecture

1. System Design

- Follows MVC architecture for modular and scalable design.
- Divided into frontend, backend, and database layers.

2. Data Processing

- Stores user data and history in MongoDB.
- Processes data to identify user preferences.

3. Authentication Mechanism

- Uses JWT for secure login and session management.
- Prevents unauthorized access.

4. Audio Streaming Process

- Implements HTTP Range Requests for streaming.
- Ensures smooth playback with reduced buffering.

5. Lyrics Synchronization

- Stores lyrics with timestamps in JSON format.
- Displays synchronized lyrics during playback.

6. Recommendation System • Uses Hybrid Recommendation Model (HMRM).

- Combines offline filtering and online game theory.

7. Workflow of the System

- User login → select music → stream audio.
- Lyrics display and recommendation updates.

8. Performance and Scalability

- Provides efficient streaming and fast response.
- Supports scalable system design.

IV. MATHAMATICAL AND OPTIMIZATION FORMULATION

A. User Preference Model

- Represents how much a user likes a song.
- Calculated using similarity between songs:

$$P_{u,m} = \sum W_{m,i}$$
- Based on user listening history and item similarity.

B. Music Popularity Model

- Defines how popular a song is over time.

$$M_m = (A_m, K_m)$$
- A_m : Average listening frequency
- K_m : Popularity trend

C. User Interest Model (UPND)

- Captures user preference for new music.

$$M_u = (A_u, K_u)$$
- A_u : Average listening behavior
- K_u : Change in interest over time

D. Hybrid Recommendation Formula

- Combines user preference, interest, and popularity:

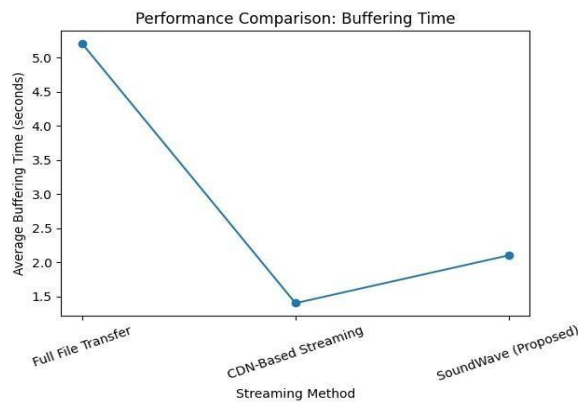
$$Score(u, m) = P_{u,m} \times e^{-t[bK_u + (1-b)A_u - a]}$$
- Adjusts recommendations dynamically for new and old songs.

E. Optimization Objective

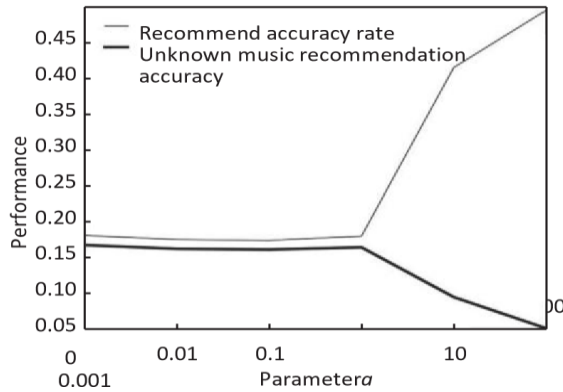
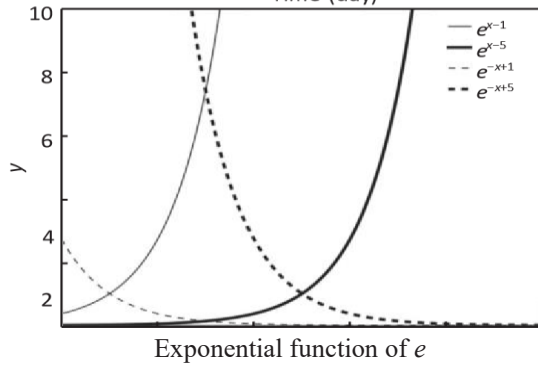
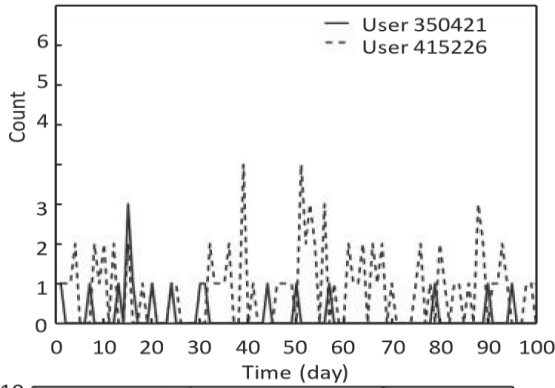
- Aim is to maximize recommendation accuracy:

$$\max_{\{u\}} \sum Score(u, m)$$
- Ensures relevant and personalized suggestions.

V. EXPERIMENTAL RESULTS AND ANALYSIS



The first 100-day music listening statistic



1 The recommendation performance change trend with parameter a

VI.DISCUSSION

The system achieves efficient and smooth audio streaming.

- Ensures secure authentication and data protection.
- Provides accurate lyrics synchronization.
- Improves recommendation accuracy and personalization.
- Handles user interest drift effectively.
- Offers scalable and high-performance system design.

Conclusion:

- The system provides secure, efficient, and scalable music streaming.
- Ensures smooth playback, lyrics synchronization, and personalized recommendations.
- Hybrid model and game theory improve accuracy and adaptability.

Future Scope:

- Can be enhanced using cloud deployment and mobile applications.
- Advanced AI-based recommendation systems can improve personalization.
- Additional features like voice control and social integration can be added.

REFERANCE

- [1] D. K. Krishnappa, M. Zink, and C. Griwodz, "Cache-centric video recommendation: An approach to improve the efficiency of video delivery in wireless networks," *IEEE Transactions on Multimedia*, vol. 15, no. 8, pp. 1802–1813, 2013.
- [2] X. Amatriain and J. M. Pujol, "Data mining methods for recommender systems," in *Recommender Systems Handbook*, 2nd ed., Springer, 2015, pp. 227–262.
- [3] R. Fielding, "Architectural styles and the design of network-based software architectures," Ph.D. dissertation, University of California, Irvine, 2000.
- [4] A. B. Johnston and J. C. Jones, "The evolution of HTTP: From HTTP/1.1 to HTTP/2 and beyond," *IEEE Internet Computing*, vol. 22, no. 3, pp. 7–15, 2018.
- [5] J. X. Li, "Music recommendation algorithm based on user portrait," in *Proc. IEEE Int. Conf. Computer Science, Electronic Information Engineering and Intelligent Control Technology*, 2021, pp. 10–13.
- [6] V. K. Adhikari et al., "Unreeling Netflix: Understanding and improving multi-CDN movie delivery," in *Proc. IEEE INFOCOM*, 2012, pp. 1620–1628.
- [7] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: Item-to-item collaborative filtering," *IEEE Internet Computing*, vol. 7, no. 1, pp. 76–80, 2003.

- [8] S. Narayanan and V. Shmatikov, “Fast dictionary attacks on passwords using timespace tradeoff,” in *Proc. ACM Conf. Computer and Communications Security*, 2005, pp. 364–372.
- [9] Y. Yu et al., “Deep attention based music genre classification,” *Neurocomputing*, vol. 372, pp. 84–91, 2020.
- [10] Y. C. Jin et al., “ContextPlay: Evaluating user control for context-aware music recommendation,” in *Proc. ACM Conf. User Modeling, Adaptation and Personalization*, 2019, pp. 294–302.