

**Tribhuvan University**

**Faculty of Humanities and Social Science**

**MusicCloud**

**A Proposal Report**

**Submitted to**

**Department of Computer Application**

**Swoyambhu Int’l College**

***In partial fulfillment of the requirements for the Bachelors in Computer* Application**

Submitted by

Suresh Tamang (6-2-927-733-2021)

Feb-26-2024

Under the Supervision of

**Mr. Samir Thapaliya**

Table of Contents

Contents

[Chapter 1: Introduction 1](#_Toc193065566)

[Chapter 2: Problem Statement : 2](#_Toc193065567)

[Chapter 3: Objective and Aim : 3](#_Toc193065568)

[Chapter 4: Methodology 4](#_Toc193065569)

[4.1 Requirement Identification 4](#_Toc193065570)

[4.1.1 Study of existing system 4](#_Toc193065571)

[4.1.2 Literature Review 5](#_Toc193065572)

[4.1.3 Requirement Collection 7](#_Toc193065573)

[4.2 feasibility study 7](#_Toc193065574)

[4.2.1 Technical feasibility 7](#_Toc193065575)

[4.2.2 Operational feasibility 7](#_Toc193065576)

[4.2.3 Economical feasibility 8](#_Toc193065577)

[4.3 High Level Design of System 8](#_Toc193065578)

[4.3.1 System Flowchart 9](#_Toc193065579)

[4.3.2 Object Diagram 10](#_Toc193065580)

[4.3.3 Description of Algorithm used 10](#_Toc193065581)

[Project timeline 12](#_Toc193065582)

[Conclusion 13](#_Toc193065583)

**Table of Figures**

Figure 1: Waterfall Methodology [3](#_heading=h.lnxbz9)

Figure 2: Use case diagram [4](#_heading=h.1ksv4uv)

Figure 3: Entity Relation Diagram [5](#_heading=h.3fwokq0)

Figure 4 : System Flow Chart [6](#_heading=h.1v1yuxt)

Figure 5 : Context Diagram 7

# Chapter 1: Introduction

Music cloud is a revolutionary web-based application designed to transform the way users experience, organize, and share music across the globe. In a world where music is an integral part of our lives, MusicCloud offers a seamless platform for music lovers, creators, and listeners to connect, explore, and enjoy their favorite tunes anytime, anywhere.

MusicCloud is more than just a music storage solution it’s a personal music universe. Wheather users are avid listener, a butting artist, or a seasoned musician, our platform empowers users to upload, stream, and manage their music library with ease. Say goodbye to the limitations of physical storage and hello to a dynamic, cloud-based ecosystem that adapts to users need.

With MusicCloud user can:

* **Upload and Store:** Safety store entire music collection in the cloud, accessible from any device with an internet connection.
* **Stream Anytime, Anywhere:** Enjoy uninterrupted streaming of you favorite tracks, playlists, and albums on-the- go.
* **Create and Share Playlists:** Curate Personalized playlists from every mood and occasion, and Share them with friends or the musicCloud community.
* **Discover New Music:** Explore a vast library of songs from emerging artists and established musicians, all in one place.
* **Collaborate and Connect:** Collaborate with other artists, share your creations, and build fanbase within vibrant Community.

MusicCloud is built with cutting-edge technology to ensure a smooth, secure, and immersive music experience. Our intuitive interface makes it easy for users of all levels to navigate, while advanced features like smart recommendations and offline listening cater to users of all levels to navigate, while advanced features like smart recommendations and offline listening cater to users’s unique preferencess.

# Chapter 2: Problem Statement :

Modern music streaming platforms have revolutionized the way we consume music, but they are not without their challenges. here are some common problems faced by users and creators, along with potential solutions to address these issues:

While algorithms recommend music, they often fail to capture the nuances of a user’s taste, leading to repetitive or irrelevant suggestions. The sheer volume of music available on streaming platforms can make it difficult for users to discover new songs or artists they genuinely enjoy. Free tier-users often experience lower audio quality, which can detract from the listening experience. Many platforms lack of social features, making it difficult for users to connect with friends or share their music experiences.

# 

# 

# Chapter 3: Objective and Aim :

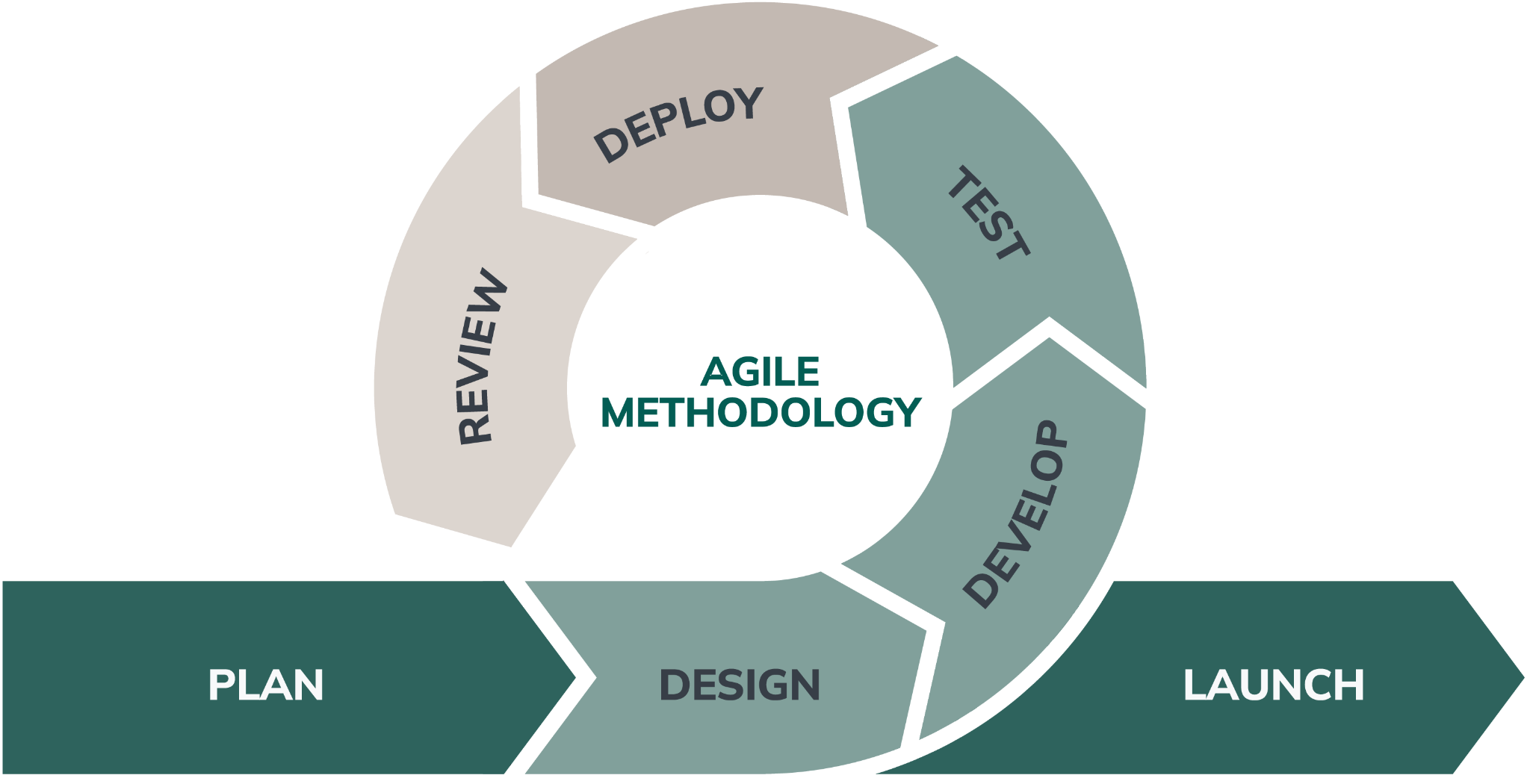
* Provide a user-friendly platform for uploading, streaming, and managing music libraries seamlessly.
* Ensure fair revenue sharing, with artists earning 70% of the income generated from their streams.
* Introduce personalized music recommendations to increase user engagement.
* Build a community-driven feature for collaborative playlists to encourage user interaction and creativity.
* Introduce mood-based playlists that adapt to user preferences.

# 

# 

# Chapter 4: Methodology

The proposed system has well defined and distant requirements and needs to increment or add new features so best agile software methodology for this system is agile software system development model.



## 

## 

## 4.1 Requirement Identification

### 4.1.1 Study of existing system

Existing music streaming platforms, such as Spotify, Apple Music, YouTube Music, and SoundCloud, have revolutionized the way people consume and share music. Here’s a brief analysis of their strengths and weaknesses:

### 

### 4.1.2 Literature Review

Music streaming platforms have transformed the way users consume music, offering on-demand access to vast libraries of songs. Key studies and trends include: Spotify and Apple Music: These platforms dominate the market, offering personalized playlists, high-quality streaming, and cross-device synchronization. Research highlights their use of recommendation algorithms to enhance user engagement (Schedl et al., 2018). SoundCloud: Known for its focus on independent artists, SoundCloud allows users to upload and share music. Studies emphasize its role in promoting emerging talent but criticize its low royalty payouts (Wikström, 2020). YouTube Music: Combines music streaming with video content, offering a unique multimedia experience. Research notes its reliance on user-generated content and ad-supported models (Burgess & Green, 2018). Gap: While existing platforms excel in personalization and content delivery, they often lack fair revenue models for artists and robust community features.

Cloud computing has enabled scalable, reliable, and cost-effective music streaming services. Key insights include: Scalability: Cloud-based systems like AWS and Google Cloud provide the infrastructure to handle millions of users and streams. Research highlights the importance of load balancing and auto-scaling for seamless performance (Armbrust et al., 2010). Storage and Accessibility: Cloud storage allows users to upload and access their music libraries from any device. Studies emphasize the need for data security and privacy in cloud-based systems (Subashini & Kavitha, 2011). Cost Efficiency: Cloud computing reduces the need for physical infrastructure, making it ideal for startups and small businesses (Marston et al., 2011). Gap: While cloud computing offers scalability, there is a need for platforms to optimize costs and ensure data security for users and artists.

User experience is critical for the success of any web-based application. Relevant research includes: Personalization: Platforms like Spotify use AI-driven recommendations to enhance user satisfaction. Studies show that personalized playlists increase user engagement (Anderson et al., 2020). Accessibility: Ensuring that platforms are usable by individuals with disabilities is essential. Research highlights the importance of WCAG compliance and inclusive design (Henry, 2007). Social Features: Community-driven features like collaborative playlists and live listening rooms foster user interaction. Studies suggest that social integration enhances user retention (Zhang et al., 2017). Gap: Many platforms lack robust social features and accessibility options, limiting their appeal to diverse user groups.

### 4.1.3 Requirement Collection

**Table 1:** **Requirement Collection table**

| **Technological requirements** | **Hardware requirements** | **Software requirements** |
| --- | --- | --- |
| HTML | Laptop | MS WORD |
| CSS | Keyboard | Photoshop |
| Java Script | Mouse | Visual studio code |
| MYSQL |  | Xampp server |
| Python, Django |  |  |

## 4.2 feasibility study

Feasibility study of the proposed system is given below:

### 4.2.1 Technical feasibility

To develop the proposed system we have HyperText Markup Language(HTML), Cascading Style Sheet(CSS) and Javascript for frontend and Python Django for the server side scripting which are Opensource and free to use.

Therefore, the proposed system is technically feasible.

### 4.2.2 Operational feasibility

To make the system work, we have various web hosting service providers for domain registration and web servers, as well as  Django server to manage and serve the services within the system.

Therefore, the proposed system is operational.

### 4.2.3 Economical feasibility

Study of the Estimated cost to develop and deploy is affordable. We have various web servers to host the system at a low cost.

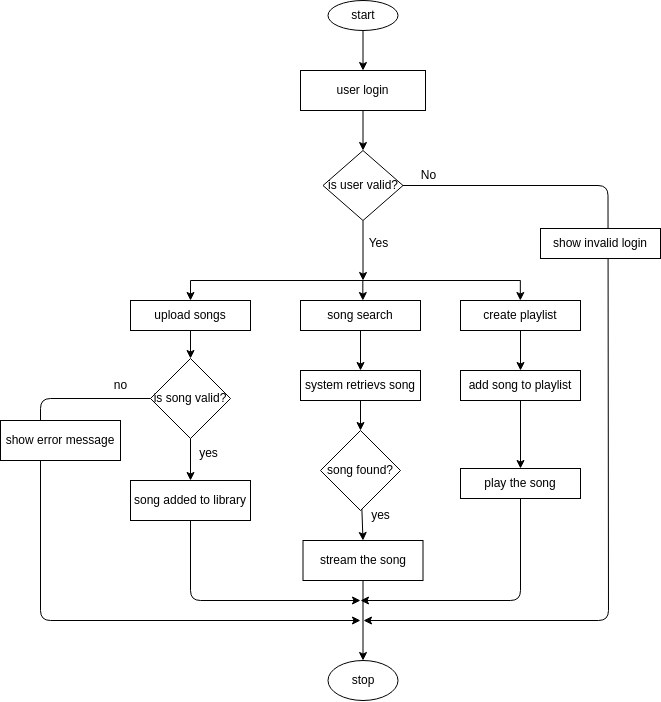
Upon the requirement of the System we can easily update the resources such as database size increment, servers increment  at an affordable cost.

Therefore, the proposed system is Economically feasible.

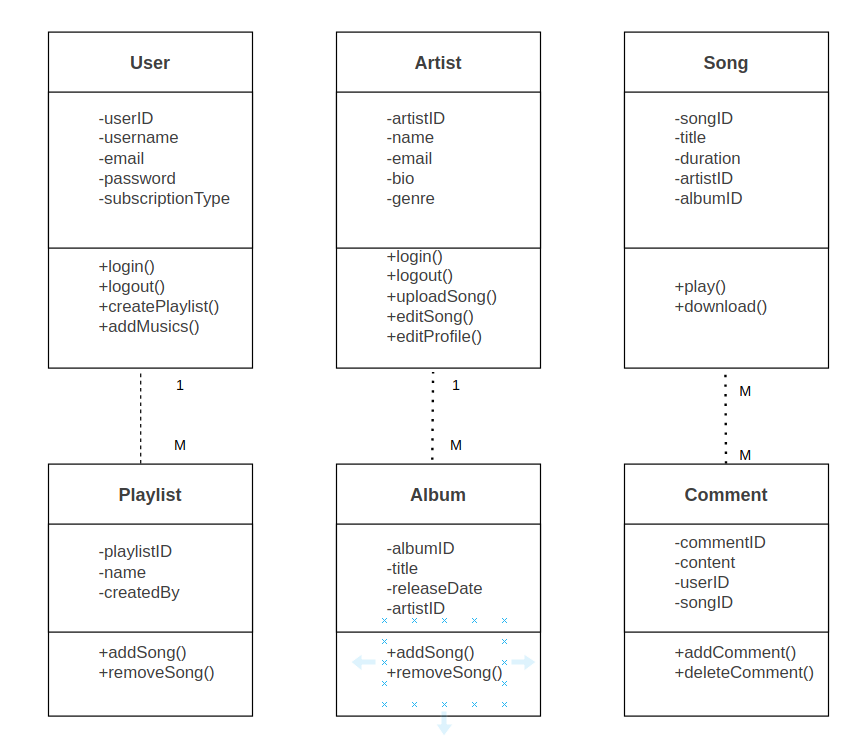
# 4.3 High Level Design of System

The proposed system has well defined and distant requirements and needs to increment or add new features so best software methodology for this system is agile software system development model.

### 4.3.1 System Flowchart



### 4.3.2 Object Diagram

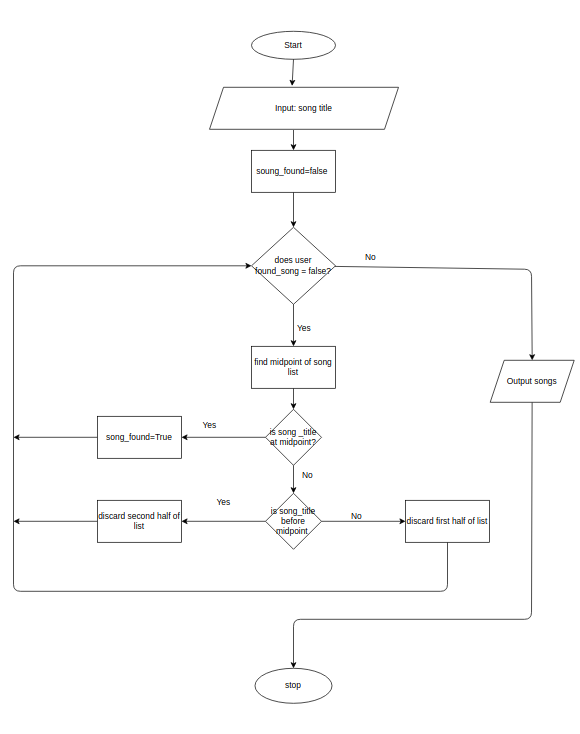


### 4.3.3 Description of Algorithm used

Binary Search

Binary Search is an efficient searching algorithm used to locate a specific element in a sorted collection (e.g., an array or list). It works by repeatedly dividing the search interval in half, reducing the problem size exponentially with each step. This makes it significantly faster than linear search, especially for large datasets.

Here is the detailed Flowchart for Binary Search in the System



# Project timeline

|  |  |  |
| --- | --- | --- |
| **Task** | **Start Date** | **End Date** |
| Planning | 01/March/2025 | 07/March/2025 |
| System Design | 08/March/2025 | 20/March/2025 |
| Development | 21/Mar/2025 | 30/April/2025 |
| Testing | 01/May/2025 | 05/May/2025 |
| Deployment | 06/May/2025 | 15/May/2025 |
| Review | 15/May/2025 | 18/May/2025 |
| Launch | 19/May/2025 | 25/May/2025 |
| Maintenance | 26/May/2025 | Till the system works |

# 

# Conclusion

Clearly , the MusicCloud project will provide numerous advantages to both user and the music artists. To add their musics in the platform and can monetize for fair royalty of their creation.

The music artist can add their musics, collaborate with other artist, can add their event schedule so that users can book the ticket.

***References***

* ***Wikström, P. (2020). The Music Industry: Music in the Cloud. Polity Press.***
* ***Armbrust, M., et al. (2010). A View of Cloud Computing. Communications of the ACM.***
* ***Zhang, Y., et al. (2017). Social Features in Online Streaming Platforms: A Survey. Computers in Human Behavior.***
* ***Subashini, S., & Kavitha, V. (2011). A Survey on Security Issues in Service Delivery Models of Cloud Computing. Journal of Network and Computer Applications.***
* ***Schedl, M., Zamani, H., Chen, C.-W., Deldjoo, Y., & Elahi. Current Challenges and Visions in Music Recommender Systems Research. International Journal of Multimedia Information Retrieval.***