VIBE WAVE

Submitted by:
A. Yugesh
242237
I B.Sc Computer Science (DAS)

ABSTRACT

The "Vibe Wave" project is a music platform where users can upload, download, and record music samples. It features both a website and mobile app, built using HTML, CSS, JavaScript, Python, and MongoDB. Users can access their music on any device, securely store it, and record new samples which is really helpful for emerging artist. The backend, powered by Python, handles file management and user sign-ins, while MongoDB stores user data and music. The front-end provides an intuitive interface, with the mobile app offering an optimized experience on phones and tablets.

SYSTEM ANALYSIS

Existing System:

Music platforms like Spotify, Apple Music, and Wynk Music

let users upload and stream music, but they lack features like

syncing across devices, recording custom samples, and

managing music in a personalized way. Users often need

separate apps for uploading, recording, and syncing, leading to

a difficult experience. Privacy and security issues also arise

with personal music files

Proposed System:

Vibe Wave solves these issues by providing a single platform

for uploading songs, recording samples, and syncing across

devices. Users can access their music from any device they log

into. The system is secure, with encrypted data storage and user

authentication, ensuring privacy. It offers a seamless

experience across both desktop and mobile devices.

SYSTEM REQUIREMENTS

Hardware Requirements:

- Processor: Intel Pentium 4 / AMD Athlon 64
- RAM: 4 GB or more
- Storage: At least 2 GB free space
- Display: Any standard resolution screen

Software Requirements:

- Operating System: Windows 10 or later
- Programming Language: Python (3.x version)
- Database: MongoDB
- Frontend: HTML, CSS, JavaScript
- IDE/Editor: Visual Studio Code

T ASK/MODULE DESCRIPTIONS

Module 1: User Authentication & Profile Management

Secure user sign-up, login, password recovery, and profile

editing.

Module 2: Song Upload & Download

Upload songs to cloud storage, download across devices,

and sync libraries.

Module 3: Recording Music Samples

Record, save, and edit music samples for future use.

Module 4: Music Library Management
Organize songs and recordings with sorting, filtering, and playlist creation.

Module 5: Cross-Device Synchronization
Sync music, recordings, and user data across web and
mobile devices.

Module 6: User Interface & Experience:
Responsive, user-friendly design for seamless navigation
on web and mobile platforms.