# Full-Stack Project Documentation

---

Project 1:  Music App

## 1. Objective

V Music is a fully functional music application designed to allow users to browse, search, and play their favourite tracks. The app provides a seamless and engaging music experience with a modern and responsive user interface.

---

## 2. Technologies Used

### Frontend:

- **ReactJS**: For building dynamic and interactive user interfaces.
- **Material-UI**: For implementing a clean, professional, and responsive design system.
- **HTML/CSS**: For structuring and styling the application.
- **JavaScript**: For application logic and interactivity.

---

## 3. User Interface (UI) and User Experience (UX)

- **Search Bar**: Enables users to quickly search for tracks or artists.
- **Music Player Controls**: Provides play, pause, and skip functionalities with an intuitive design.
- **Pin/Like Button**: Enables user to Pin or like your tracks and can access to those tracks directly from the home screen

### UI Design Approach:

- Minimalistic and user-centric design principles.
- Responsive layout for a consistent experience across devices.

---

## 4. Hosting and Deployment

- **Hosting Platform**: GitHub Pages.
- **Deployment Process**:
    1. Application code is pushed to a GitHub repository.
    2. The app is built and deployed using the `gh-pages` package in React.

**Project Link:**

doodleboii.github.io/music/

---

# Project 2: Weather Forecasting App

## 1. Objective

The Weather Forecasting App provides real-time weather updates and a 7-day forecast for any user-entered location. The app focuses on delivering accurate weather information with an elegant and easy-to-navigate interface.

---

## 2. Technologies Used

### Frontend:

- **ReactJS**: To handle UI components and state management.
- **Tailwind CSS**: For modern and responsive styling.
- **Material-UI**: For pre-built UI components like cards and input fields.

---

## 3. User Interface (UI) and User Experience (UX)

- **Location Input Field**: Allows users to enter a city name to get weather data.
- **Weather Display Section**:
    - Shows current temperature, humidity, and weather conditions with corresponding icons.

### UI Design Approach:

- Clean, modern, and minimalistic interface for a better user experience.
- Optimized for both desktop and mobile devices.

---

## 4. Hosting and Deployment

- **Hosting Platform**: Netlify.
- **Deployment Process**:
    1. Application code is pushed to a GitHub repository.
    2. Netlify automatically builds and deploys the application.

**Project Link:**

**sturdy17.netlify.app/**

**GitHub Link:**

https://github.com/doodleboii/weatherapp

---

## Future Scope

To transform these into full-stack applications:

- **Backend Development**:
  - Implement a backend using **Node.js** and **Express** for server-side logic.
  - Use **MongoDB** or **Firebase** as a database for user data, playlists, and weather history.