# 3 S THE MUSIC PLAYER

# REPORT

### A MINI PROJECT REPORT

#### Submitted by

#### Group/Team No: 29/12

**SHIVEK S. MITTAL, 2310992527**

**SHREYA SHARMA, 2310992528**

**SHUBHAM, 2310992529**

#### in partial fulfillment for the award of the degree of

## BACHELEOR OF ENGINEERING

***in***

COMPUTER SCIENCE & ENGINEERING

****

**CHITKARA UNIVERSITY**

**CHANDIGARH-PATIALA NATIONAL HIGHWAY**

**RAJPURA (PATIALA) PUNJAB-140401 (INDIA)**

##### JUL-DEC, 2023

**ABSTRACT**

A music player implemented using HTML, CSS, and JavaScript provides an interactive web-based platform for users to play and control audio tracks seamlessly. This project incorporates the following key components:

**HTML Structure:**

Utilizes the <audio> tag to embed audio content within the web page.

Includes elements for displaying track information, playback controls (play, pause, volume), progress bar, and track duration.

**CSS Styling:**

Applies styles to enhance the visual appeal of the music player interface.

Utilizes CSS to create a responsive and user-friendly design.

**JavaScript Functionality:**

Implements event listeners to capture user interactions with the playback controls.

Dynamically updates the UI elements, such as the progress bar and track information, based on the current playback state.

Allows users to play, pause, adjust volume, and seek through the track using interactive controls.

**Additional Features:**

1. Offers a visually appealing display of album artwork and track information.

2. Implements keyboard shortcuts for enhanced accessibility.

3. Enables users to customize the player's appearance through CSS themes.

4. This music player project showcases the synergy of HTML for structuring content, CSS for styling and layout, and JavaScript for dynamic functionality. It provides a rich and engaging user experience, allowing music enthusiasts to enjoy their favorite tracks with ease and flexibility.

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| Sr. No. | Section | Page No. |
| 1. | Introduction | 1 |
| 2. | Problem Statement | 2 |
| 3. | Technical Details | 3 |
| 4. | Key Features | 5 |
| 5. | Project Advantages | 6 |
| 6. | Results | 7 |
| 7. | Conclusion | 11 |
| 8. | Future Scope | 12 |
| 9. | References | 13 |

**Introduction**

The Music Player project involves the creation of a digital countdown timer using HTML, CSS, and JavaScript. The purpose of the project is to create a user-friendly interface that allows users to play, pause and select songs with the help of buttons according to your requirement as well as set up songs

Music player offers a range of features, such as the ability to create and manage playlists, equalizer controls, and the option to download and stream music,allowing users to access a vast library of songs without having to purchase individual tracks.

Music players can be found on many different platforms, including desktop operating systems like Windows and MacOS, as well as mobile platforms like iOS and Android. They are widely used for personal entertainment, as well as for professional purposes, such as in radio broadcasting or music production.

Overall, music players have become an essential tool for many people who love music, as they provide a convenient and customizable way to listen to their favourite tracks and discover new music.

**Problem Statement**

The objective of this project is to design and implement a sophisticated music player that offers an enhanced user experience, versatile features, and seamless integration with modern technologies. The music player aims to cater to the diverse needs of users, providing an intuitive interface for managing, organizing, and playing music across different platforms.

**Technical Details**

**HTML (Hypertext Markup Language)** is a markup language used for creating web pages. It provides the structure and content of a web page, including headings, paragraphs, images, and links. In the project, HTML was used to create the layout of the music player including the placement and styling of the various elements such as the input fields and the song display.

**CSS (Cascading Style Sheets)** is a style sheet language used for describing the presentation of a web page, including the layout, color, and font of various elements. In the project, CSS was used to style the music player, including the font, color, and size of the music player, as well as the positioning and layout of the input fields and songs.

**AUDIO OUTPUTS**: Music players use audio outputs such as speakers, headphones, and digital audio outputs to play back audio.

**STORAGE DEVICES**: Music players can store music files on various types of storage devices such as hard drives, solid-state drives, and memory cards.

**PYTHON INTEGRATION:** Used for login authentication. For the backend, we opted to use Flask, a lightweight and versatile web framework for Python. Flask's simplicity and extensibility align well with the project's requirements, providing the necessary tools for building a robust authentication system.

The authentication logic involves checking entered credentials against the stored user data. If the username and password match, a session is initiated.

Enhance authentication logic by using Flask-Login, which simplifies the management of user sessions.

**TAGS USED:**

**<a> (Anchor Tag):**

Purpose: Used to create hyperlinks, allowing users to navigate to another web page or resource.

Attributes: Common attributes include href, target (specifying where to open the linked page, such as in a new tab or window), and title (providing additional information about the link).

**<form>:**

Purpose: Defines an HTML form that collects user input. Forms are used for various purposes like user authentication, data submission, and more.

Attributes: Common attributes include action (specifying the URL where the form data will be sent), method (specifying the HTTP method for form submission, usually "GET" or "POST").

**<audio>:**

Purpose: Embeds audio content in a web page, allowing users to play and control audio files.

Attributes: Common attributes include src (specifying the URL of the audio file), controls (adding playback controls like play, pause, and volume), and autoplay.

**<input>:**

Purpose: Defines an input field that allows users to enter data. The type attribute determines the type of input, such as text, password, checkbox, radio, etc.

Attributes: Common attributes include type (specifying the type of input), name (used to identify the input when the form is submitted), placeholder (providing a hint to the user), and value (setting a default value).

**<label>:**

Purpose: Associates a label with a form control, enhancing accessibility and user experience. Clicking on a label can also focus or activate the associated form element.

Buttons:

Use the button element and the button selector for styling play, pause, stop, and other control buttons.

Progress Bar:

Use the progress element and the progress selector for styling the progress bar.

Volume Control:

Use the input element with type="range" for volume control, and the

properties for the volume control \*/ }

Album Cover:

Use an img element for the album cover, and the img selector for styling.

Text and Labels:

Use appropriate HTML elements for text and labels (e.g., p, span, label), and their respective selectors for styling.

**Key Features**

* Play, pause, and stop buttons: These buttons allow the user to control the playback of the music.
* Progress bar: A progress bar indicates the current position of the music playback and how much time has elapsed.
* Volume control: A volume slider allows the user to adjust the volume of the music.
* Search bar: Users can search a song of their choice.
* Playlist: A playlist allows the user to select which songs they want to play.
* Album artwork display: The music player can display the album artwork of the currently playing song.

**Project Advantages**

1. Users can effortlessly navigate their music collections and enjoy a seamless playback experience.

2. As part of an academic project, this endeavour provided valuable learning opportunities in web development, user interface design, and project management. It allowed us to apply classroom knowledge to a practical project, improving our skills in the process.

3. Users can stream music ad free.

**Results**

**1. Login Page**

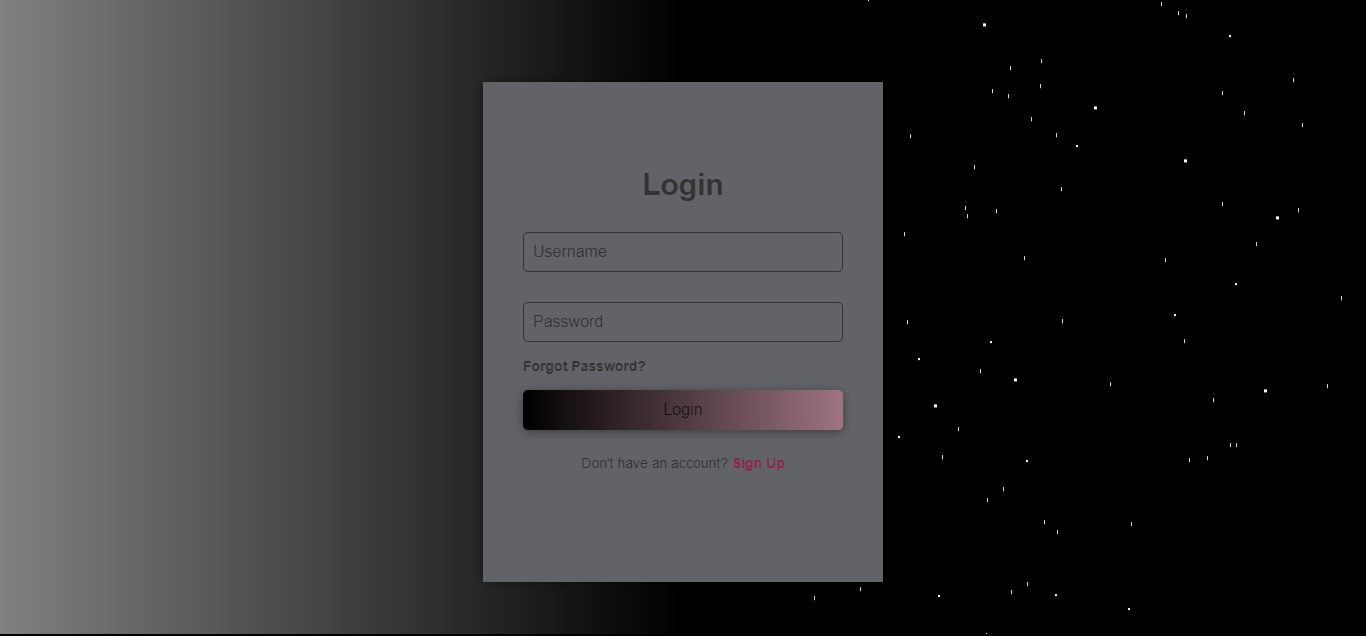


Figure 1:login page

On clicking Sign In :

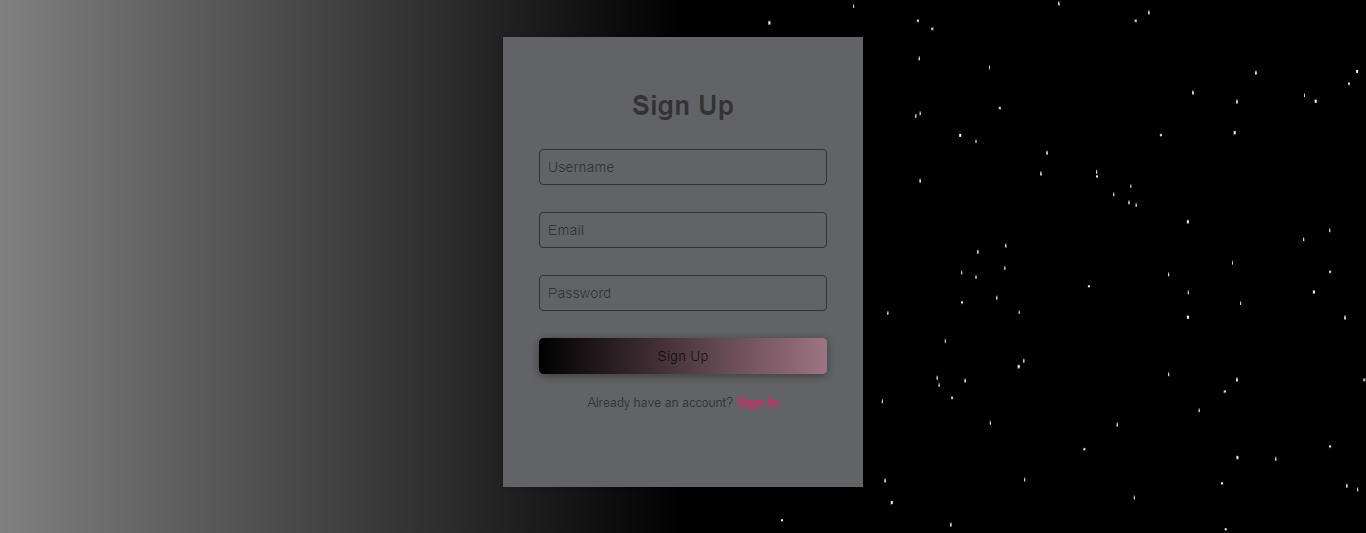
****

Figure 2:sign up page

**2. Music Player**

a) Menu b) Songs

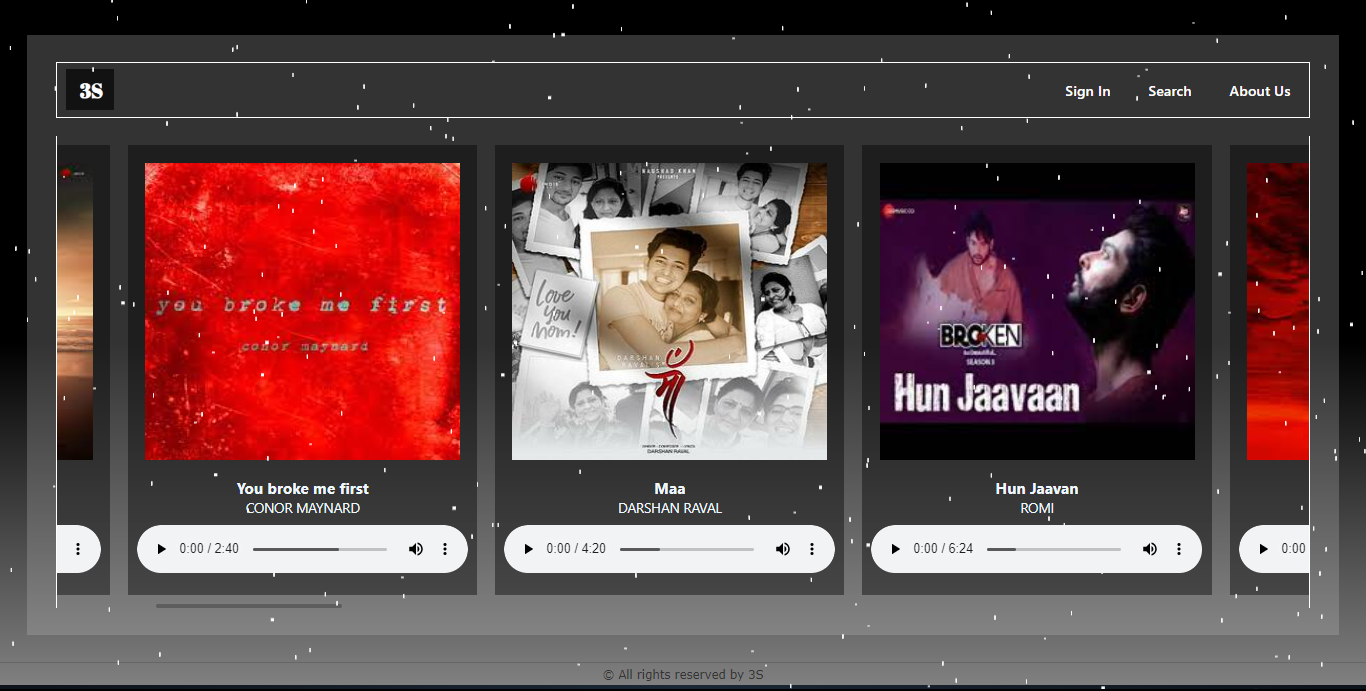


Figure 3: music player (main page)

**3. Searching songs**

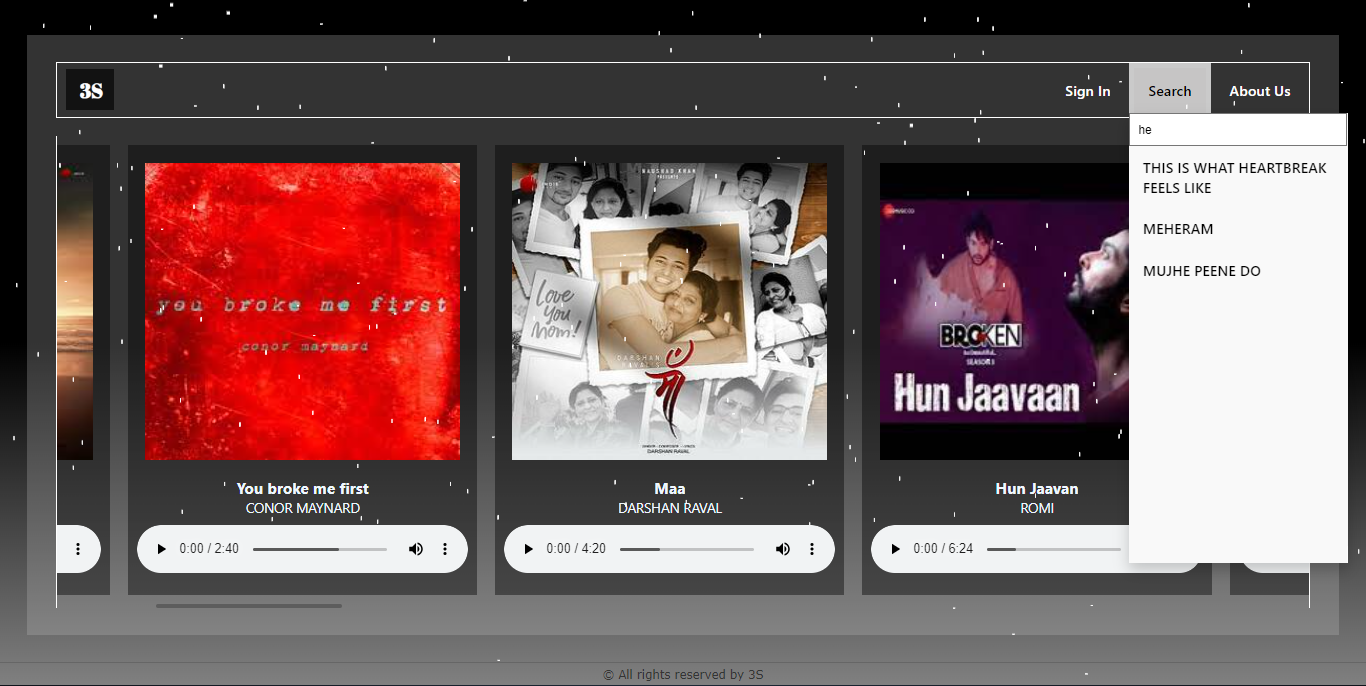


Figure 4:search bar in music player

**4. About Us Page**

a) Menu b) Tagline

c) Our Work d) Team Members

e) Contact Us

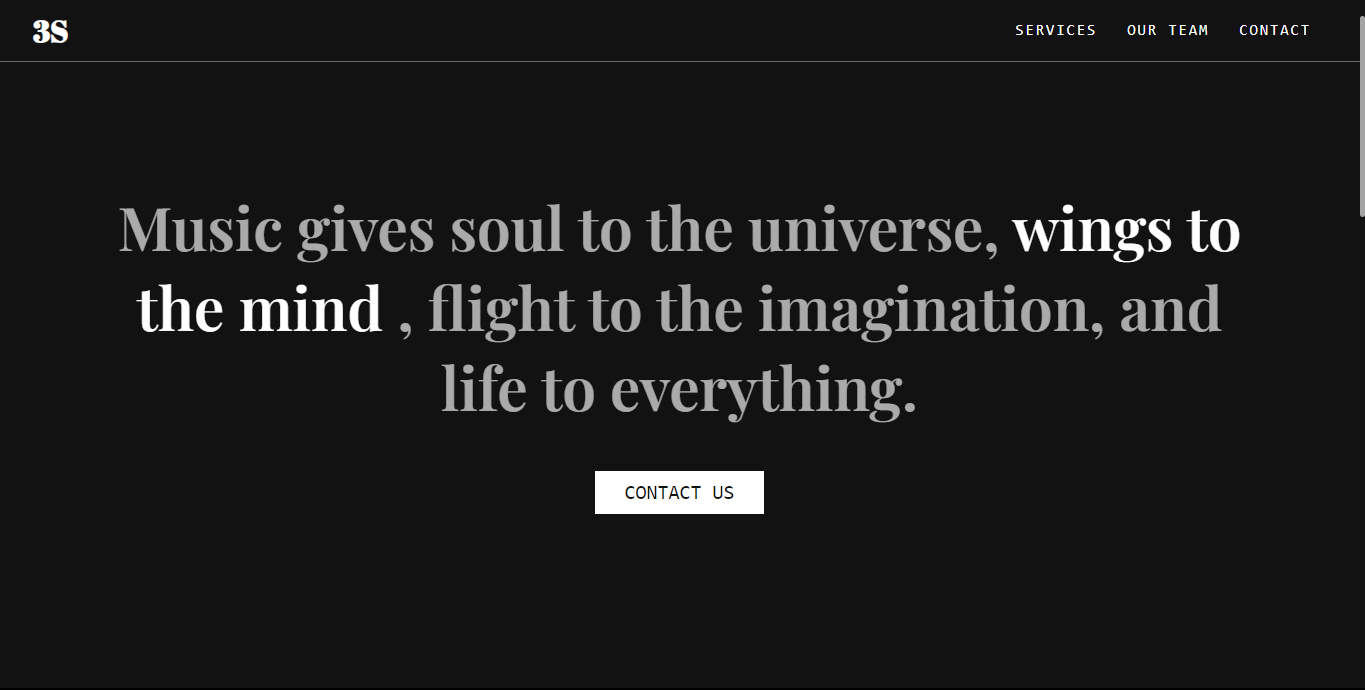


Figure 5: About us

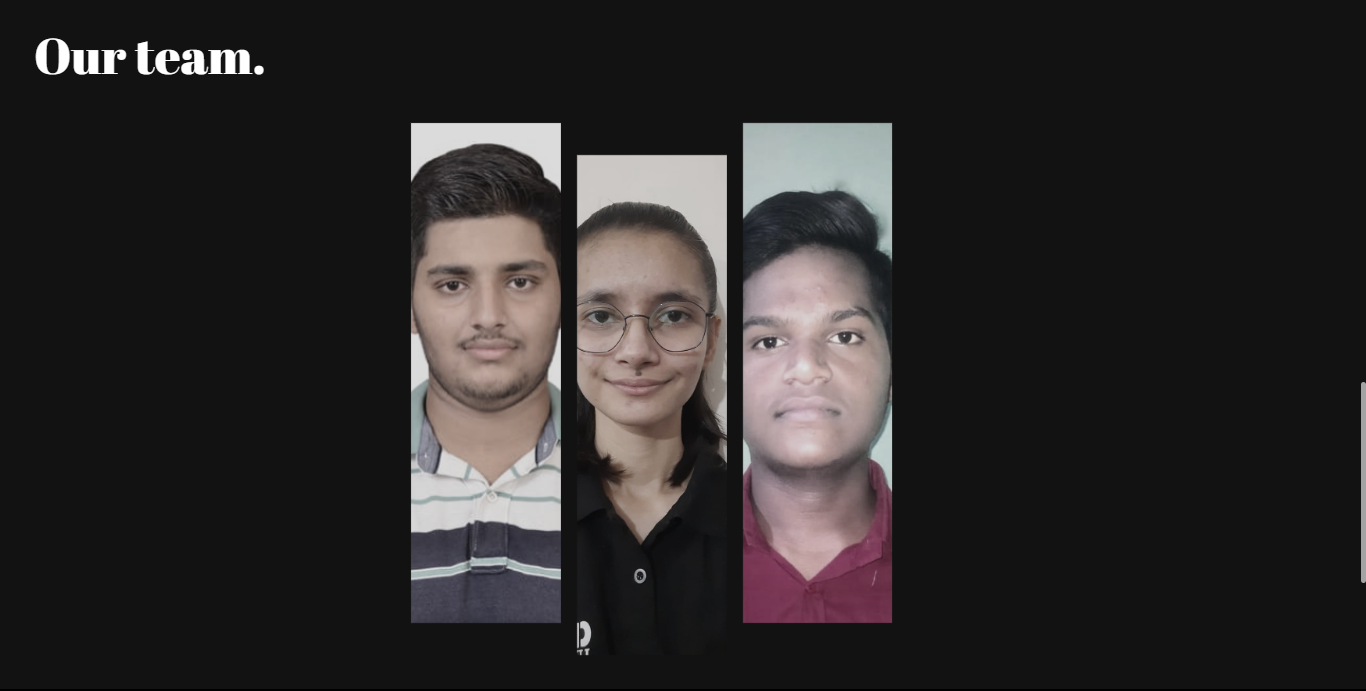


Figure 6: about our team

On hovering on any image :

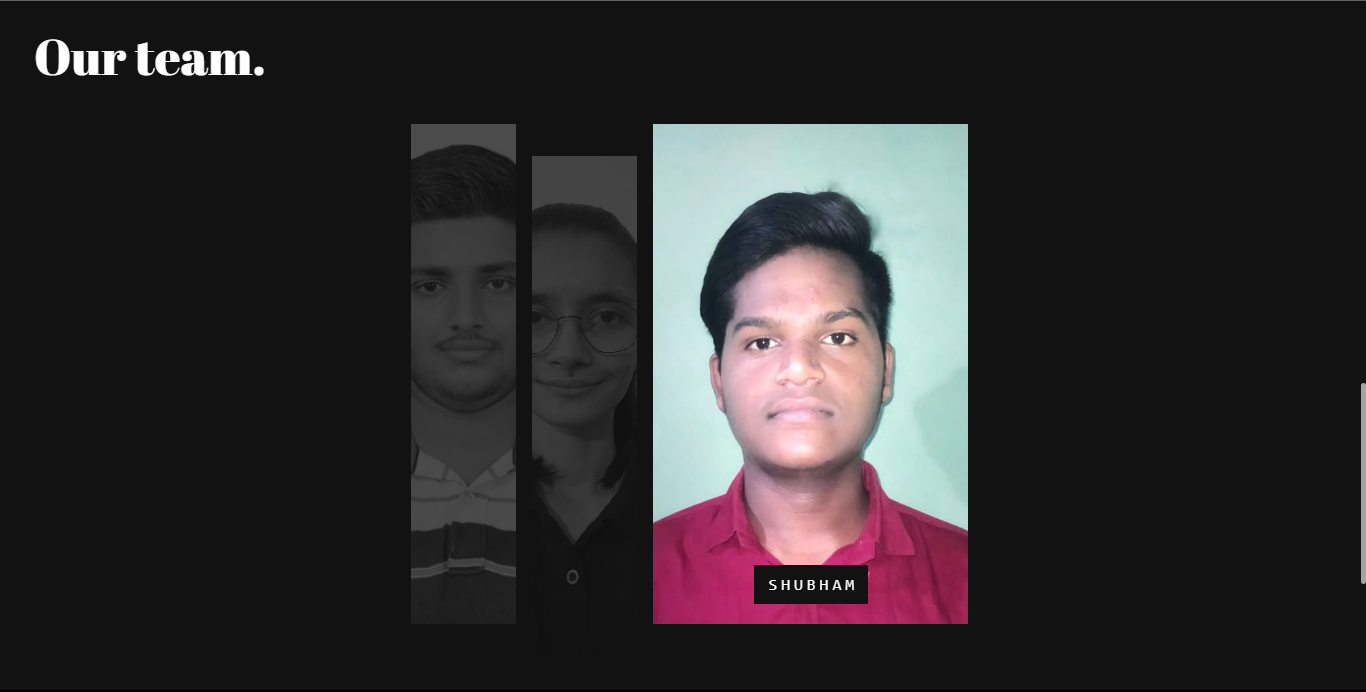


Figure 7 :showing hovering of image

**Conclusion**

In conclusion, developing a music player can provide significant advantages, such as a large user base, revenue opportunities, personalization, integration with other services, and innovation. As the music industry continues to evolve and grow, music players will continue to play an essential role in how people consume and interact with music.

**1. User-Centric Design:**

The emphasis on creating a user-friendly interface has driven the decisions in layout, styling, and interactive elements. The user experience is at the forefront, ensuring that music enthusiasts can effortlessly engage with their favorite tracks.

**2. Integration of Technologies**:

The seamless integration of HTML, CSS, and JavaScript has been the backbone of this project. Each technology plays a crucial role in contributing to the structure, style, and functionality of the music player, demonstrating the power of a well-rounded front-end development approach.

**3. Scalability and Customization:**

The modular design of the music player sets the stage for future expansion and customization. Whether it's adding new features, integrating with back-end systems, or incorporating additional visual elements, the project serves as a robust foundation for growth.

**4. User Interaction and Accessibility:**

Leveraging JavaScript for user interaction and accessibility considerations has resulted in a dynamic and inclusive music player.

**Future Scope**

In terms of future scope, there are several exciting opportunities for music players, including:

1. Advanced AI and machine learning algorithms for better recommendations and personalized playlists

2. Integration with virtual and augmented reality technologies for immersive music experiences

3. Increased focus on social and community features, allowing users to connect and collaborate with other music lovers

4. Expansion into emerging markets, such as Asia and Africa, where music consumption is rapidly growing

5. Integration with emerging technologies,.

Overall, the future of music players looks promising, with many exciting opportunities for innovation and growth. Music players will continue to evolve and adapt to meet the changing needs and preferences of music lovers around the world

**List of References**

References of this project have been taken from the below mentioned sites-

1. <https://www.w3schools.com/tags/tag_audio.asp>
2. HTML & CSS: Design and Build Web Sites by Jon Duckket