**ITCS 4155**

**SD2: Software Design Document**

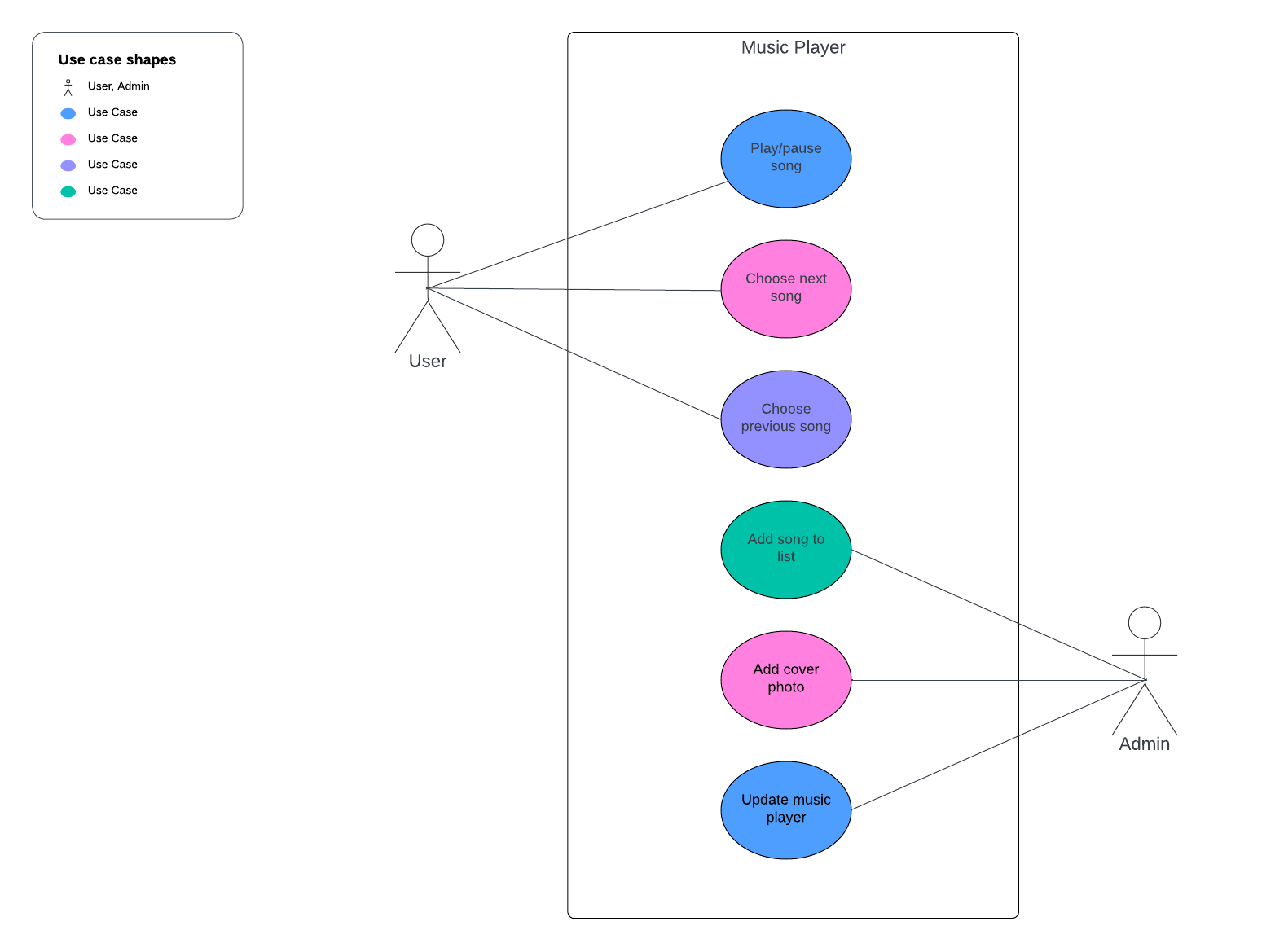
1. Project Overview

The purpose of the music player is to listen and have access to music when there is no internet connection. Most music apps such as Spotify and YouTube music require access to the internet, but this music player can play songs while offline. It is useful in situations where you don’t have access to the internet, such as traveling by plane. Users can play and pause songs, skip songs, and go back to the previous song with the click of a button. Admins can change the layout of the music player, add or remove songs, and change album covers. The users can also become admins themselves by changing the codebase.

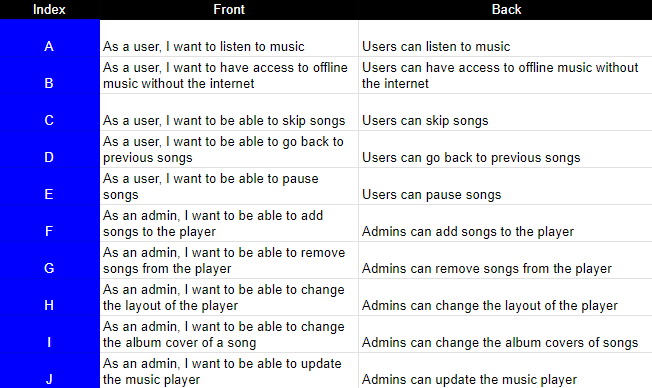
2. Architectural Overview

I decided to build the music players using only HTML, CSS, and JavaScript. I thought about using Python but later decided that no backend is needed, so I built the music player using only frontend programming languages. To use the music player, the user will need to simply clone the repository on their device and open the index.html file on their browser. It is a good alternative to Spotify and YouTube music because it enables offline listening, although Spotify has a feature that allows premium users to save songs to listen offline. However, unlike the other music services, this music player is completely free and allows for offline listening.

**Use Case**

****

**User Stories**

****

3. Detailed System Design

The music player was built using the frontend programming languages, HTML, CSS, and JavaScript. The files included are index.html, style.css, data.js, app.js, and the images and music folders. The index.html and style.css files control the layout of the music player, such as the background colors, disk size, borders, box shadows, and text. The data.js file contains a list of all the songs in the music player, with the song name, file path, artist name, and cover image. The app.js file contains all the variables and methods for the music player to work. It controls the play/pause and forward and backward buttons and matches the slider with the current time and the song duration.

Repository: <https://github.com/ekkanikanti/music-player.git>