# Muse

Liam Easton

Contents

[Muse 1](#_Toc152792033)

[Statement Of Work 3](#_Toc152792034)

[Data Dictionary 4](#_Toc152792035)

[ER Diagram 5](#_Toc152792036)

[Feature List and User Stories 6](#_Toc152792037)

[User Interfaces 7](#_Toc152792038)

[Review 24](#_Toc152792039)

# Statement Of Work

Muse

Technologies used

TSQL, Microsoft SQL Server, Visual Studio, Visio, Adobe XD, GitHub

Project Description

Muse is a music-sharing multi-user data-driven application, built off of the C# .NET framework with n-tier architecture. Users will be able to play other user's uploaded songs, create custom public profiles to share with friends and curate their own playlists. This project will be created throughout 8 weeks and 4 sprints.

Scope of Work

Week 1 & 2: Documentation and planning

1. Creating a Feature list with both minimal viable product and desirable features.

2. Making extensive UIs within Adobe XD

3. Creating the Data Dictionary

4. Creating the ER Diagram

5. Creating the User Stories

Week 3 & 4: Project Setup and Coding

1. Building the SQL Script

2. Creating the .BAT file to run the database

3. Initial Setup with N-Tier Architecture

4. Coding, testing using the AAA methodology, and debugging feature by feature

Week 5 & 6: Continuing Code

1. Continue building out each feature

2. Making sure that all tests needed are here

Week 7 & 8: Final Touchups

1. Look over all code to ensure neatness and continuity with styling

2. Creating comments where needed

3. Making sure all names are understandable to read

4. Fixes where needed in the UI

Using these sprints, and the AGILE style of development, I plan to create the best project I possibly can for my users.

# Data Dictionary



# ER Diagram

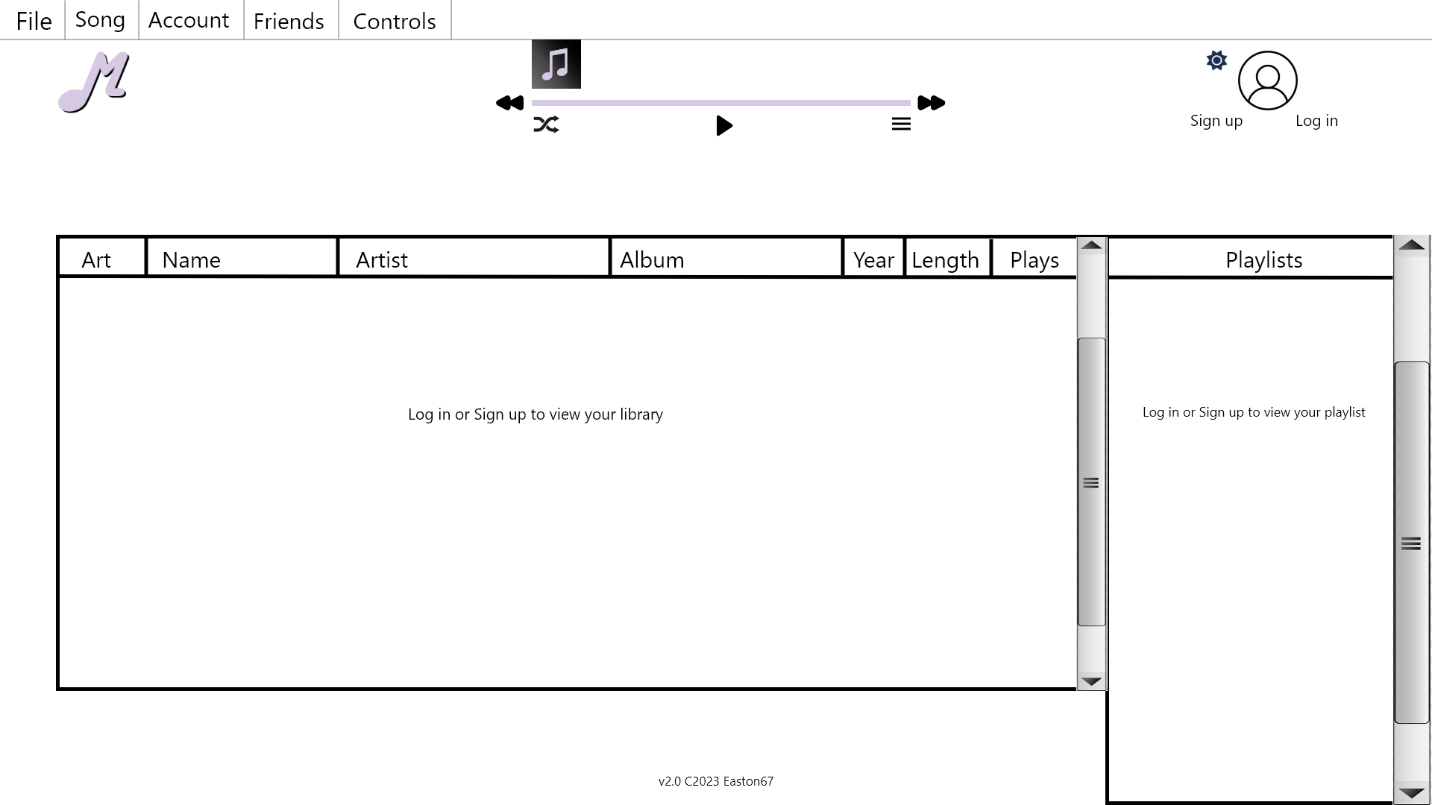
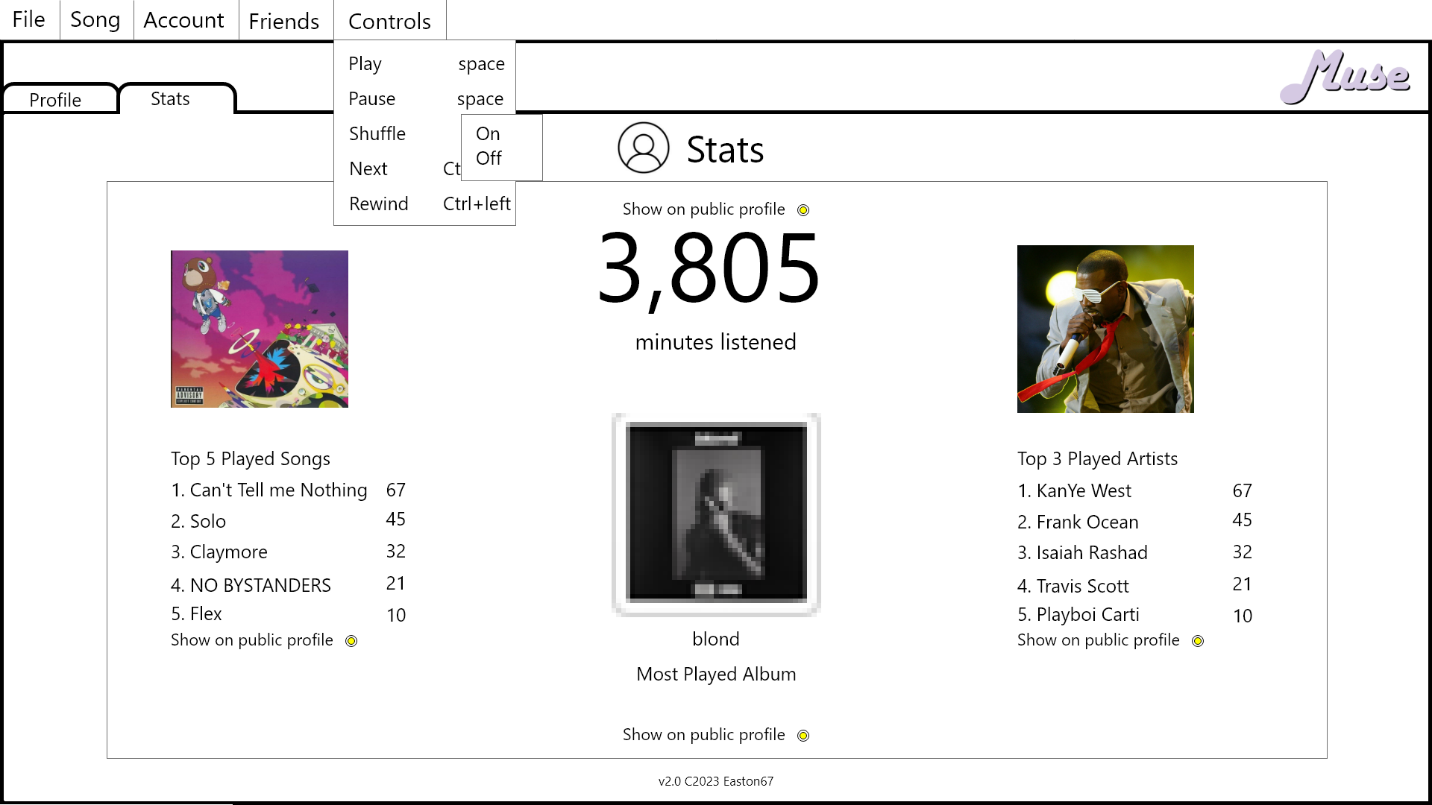
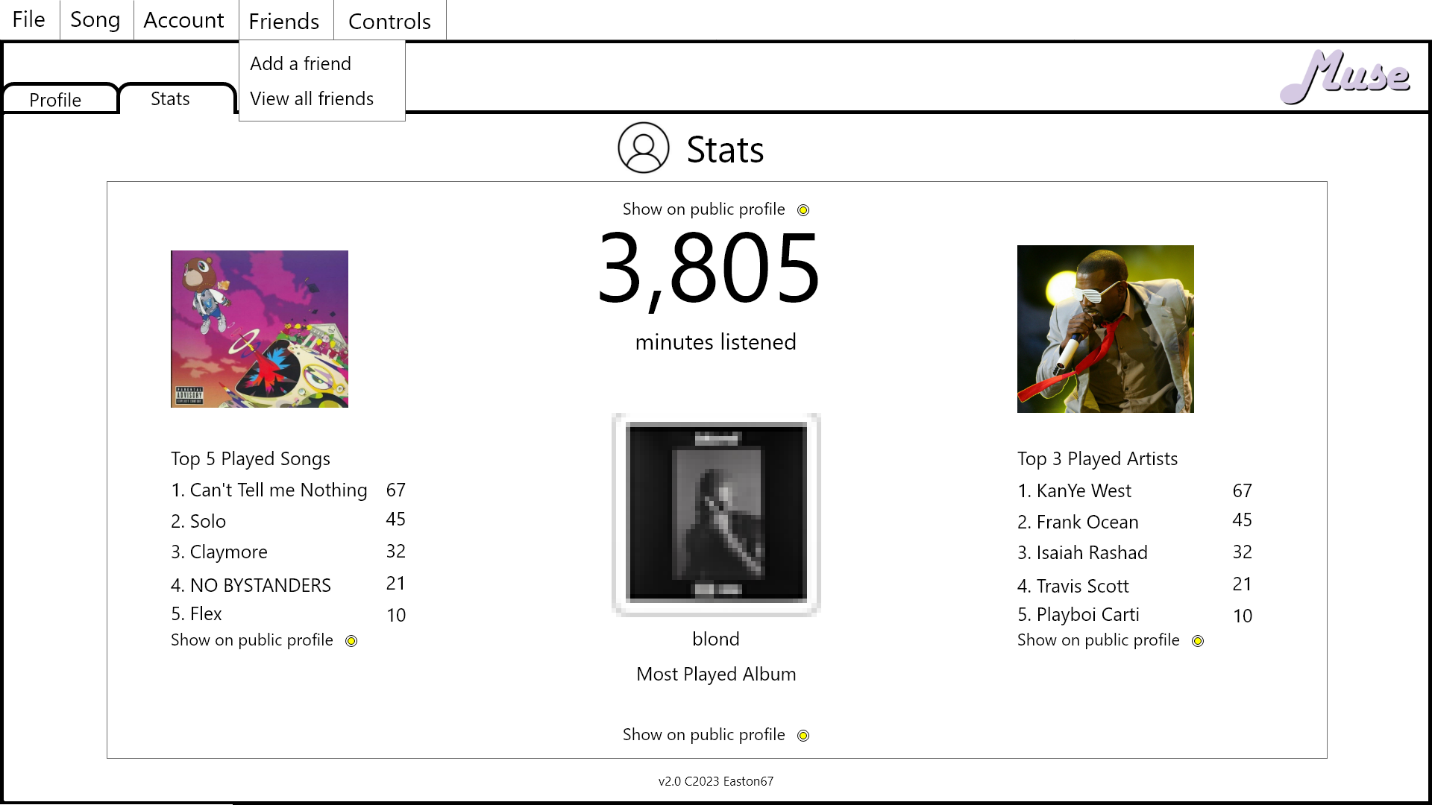
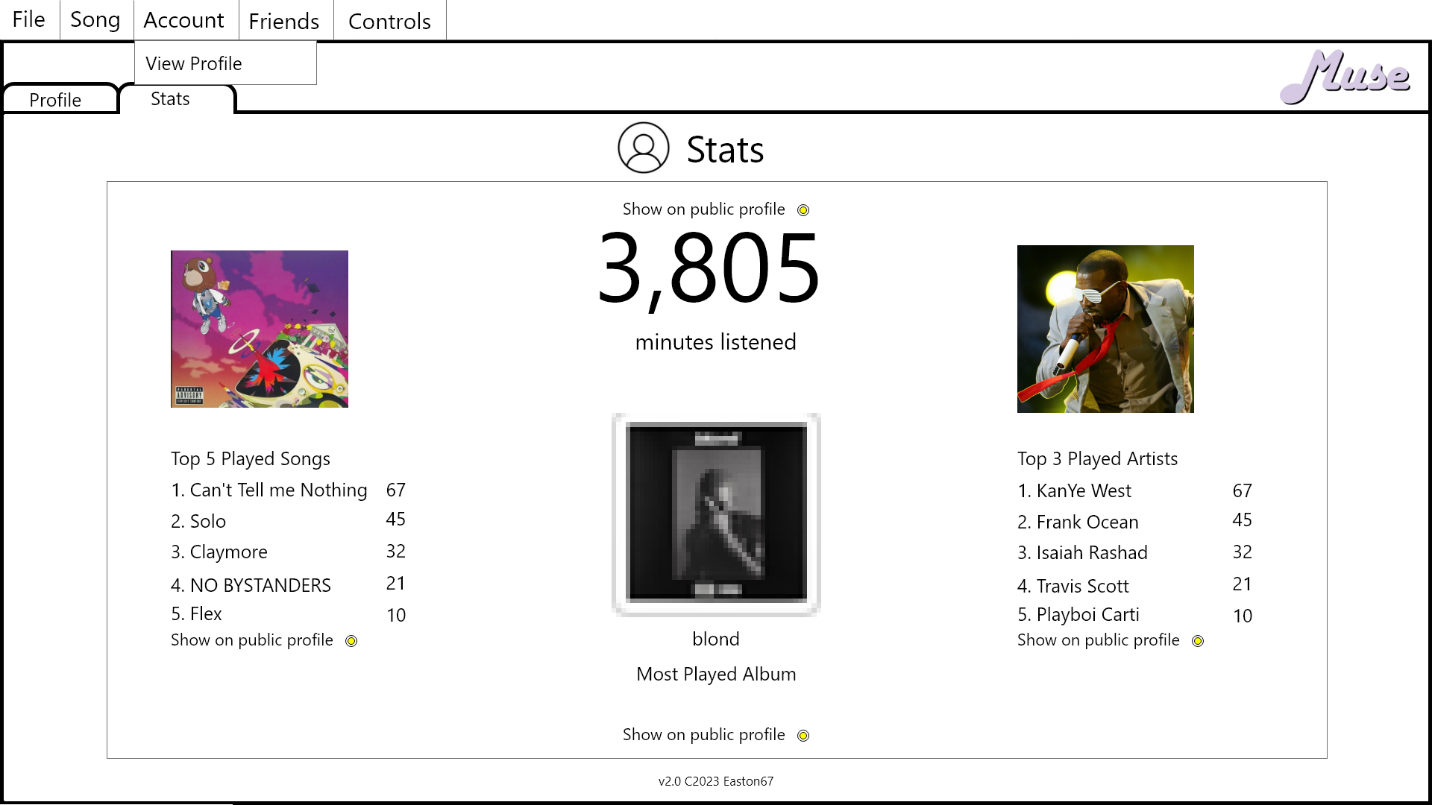
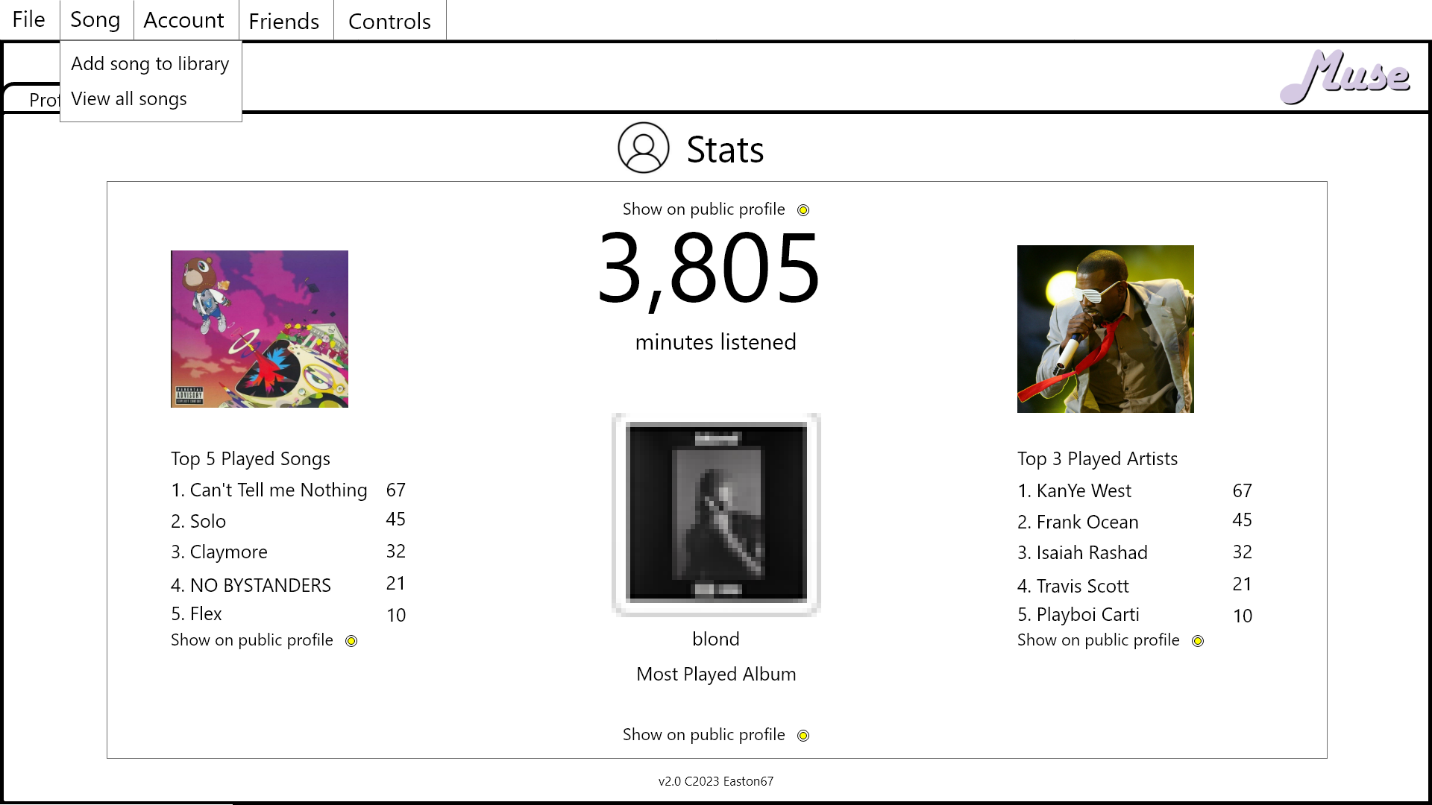
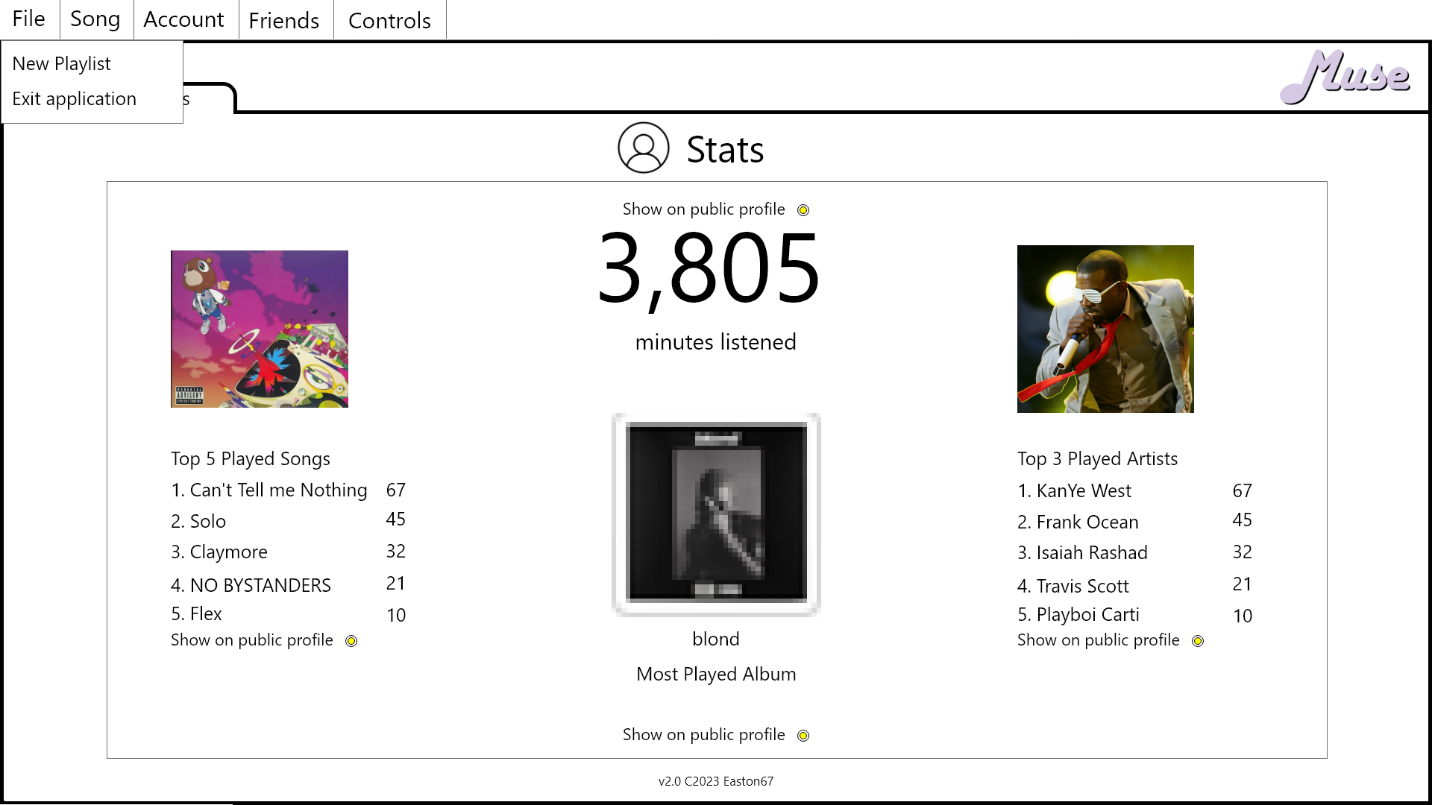
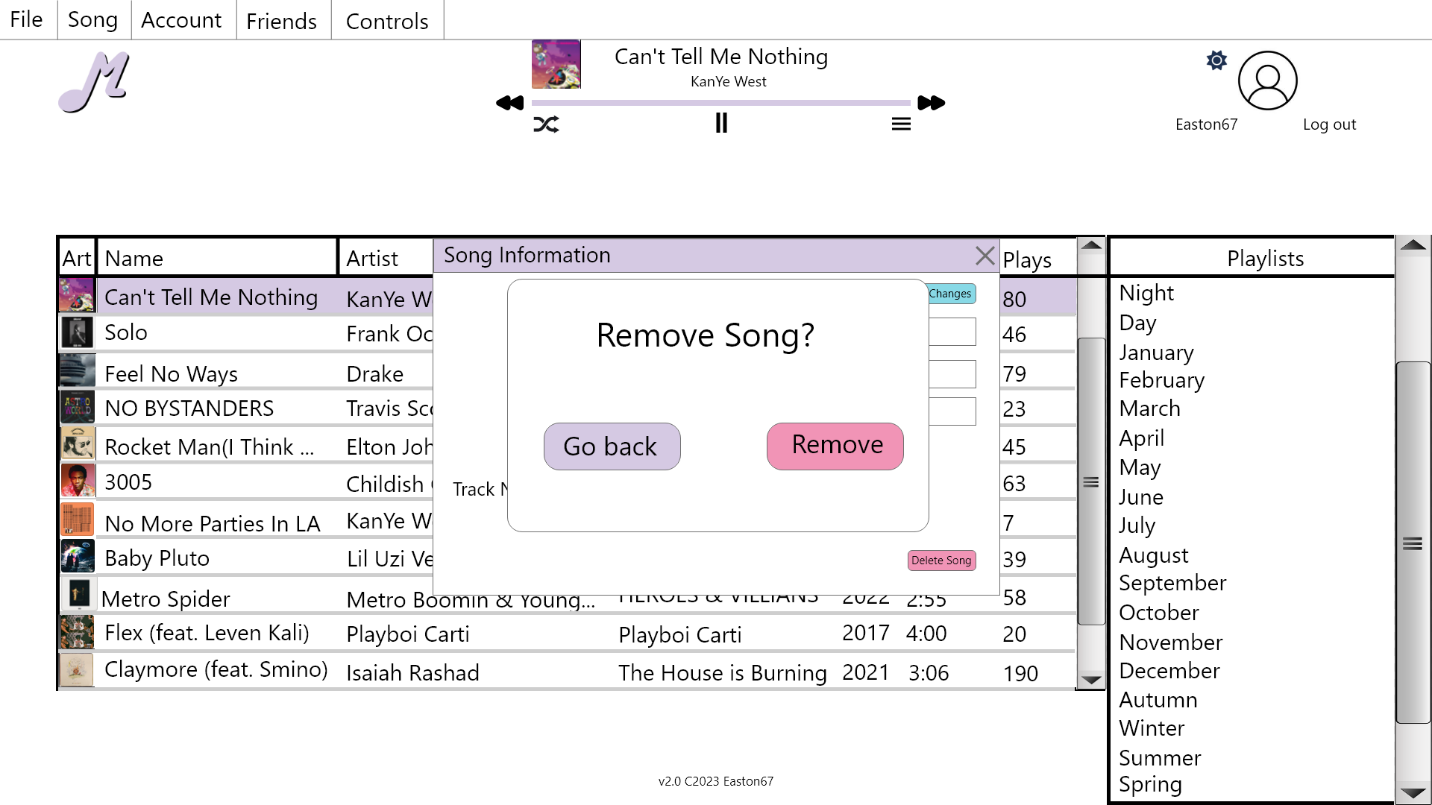
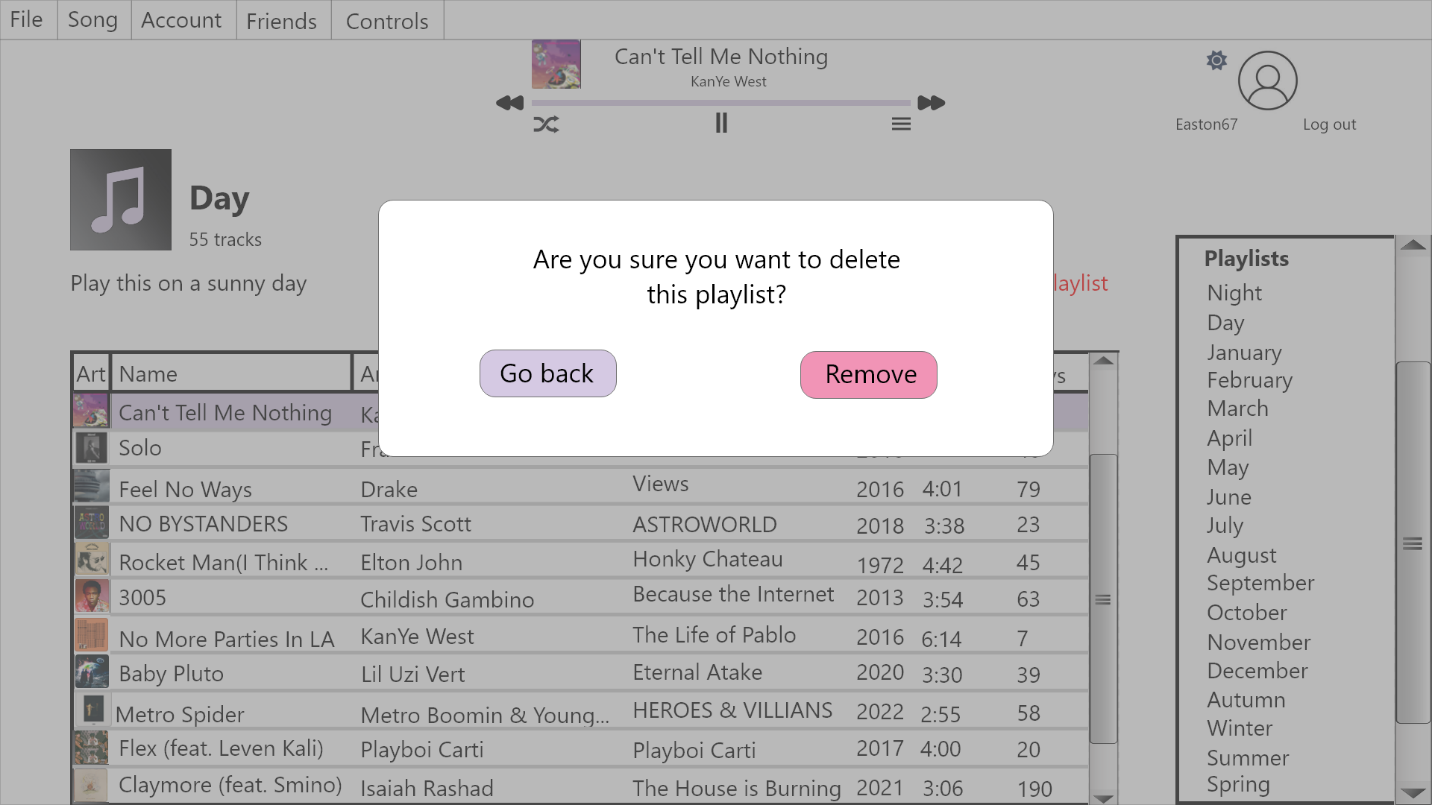
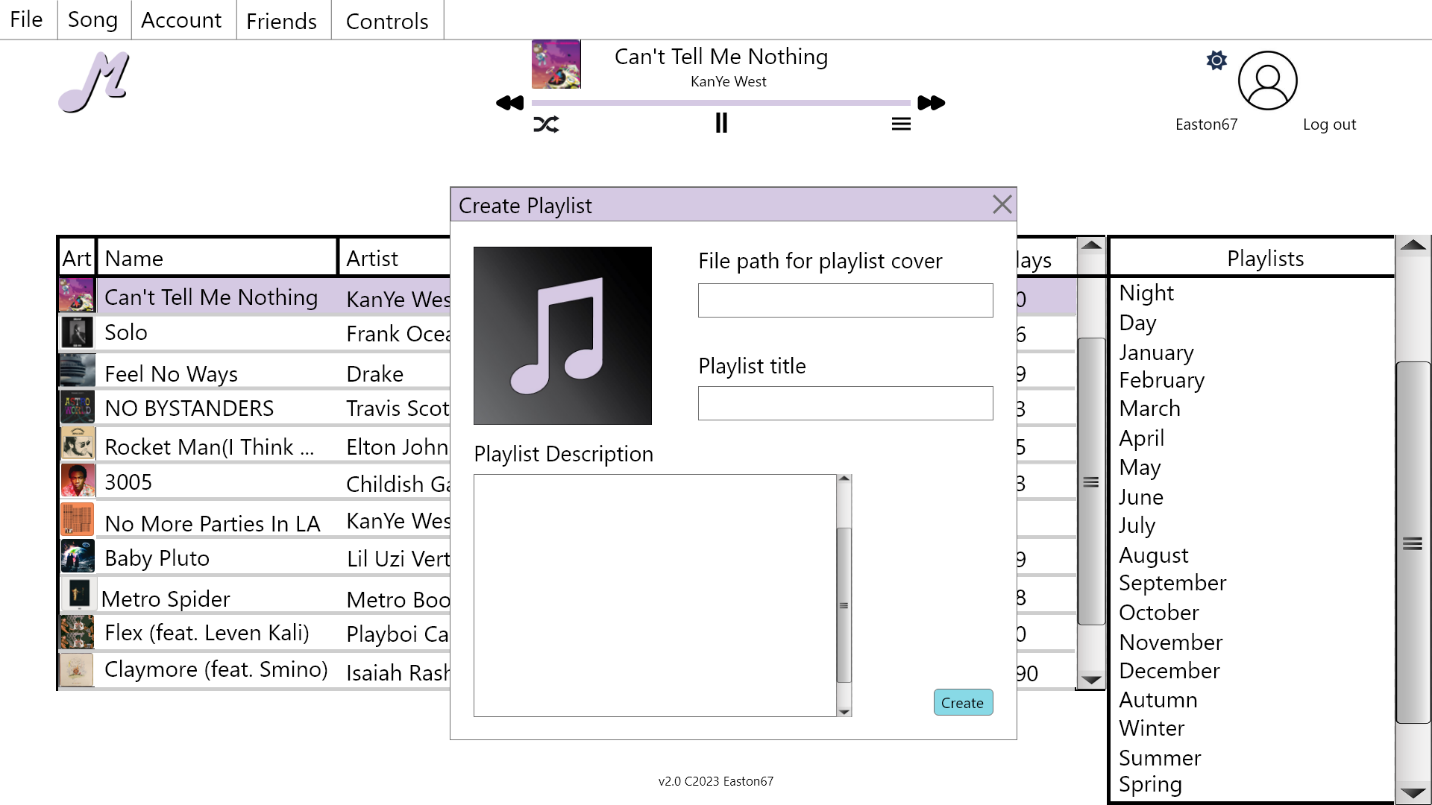
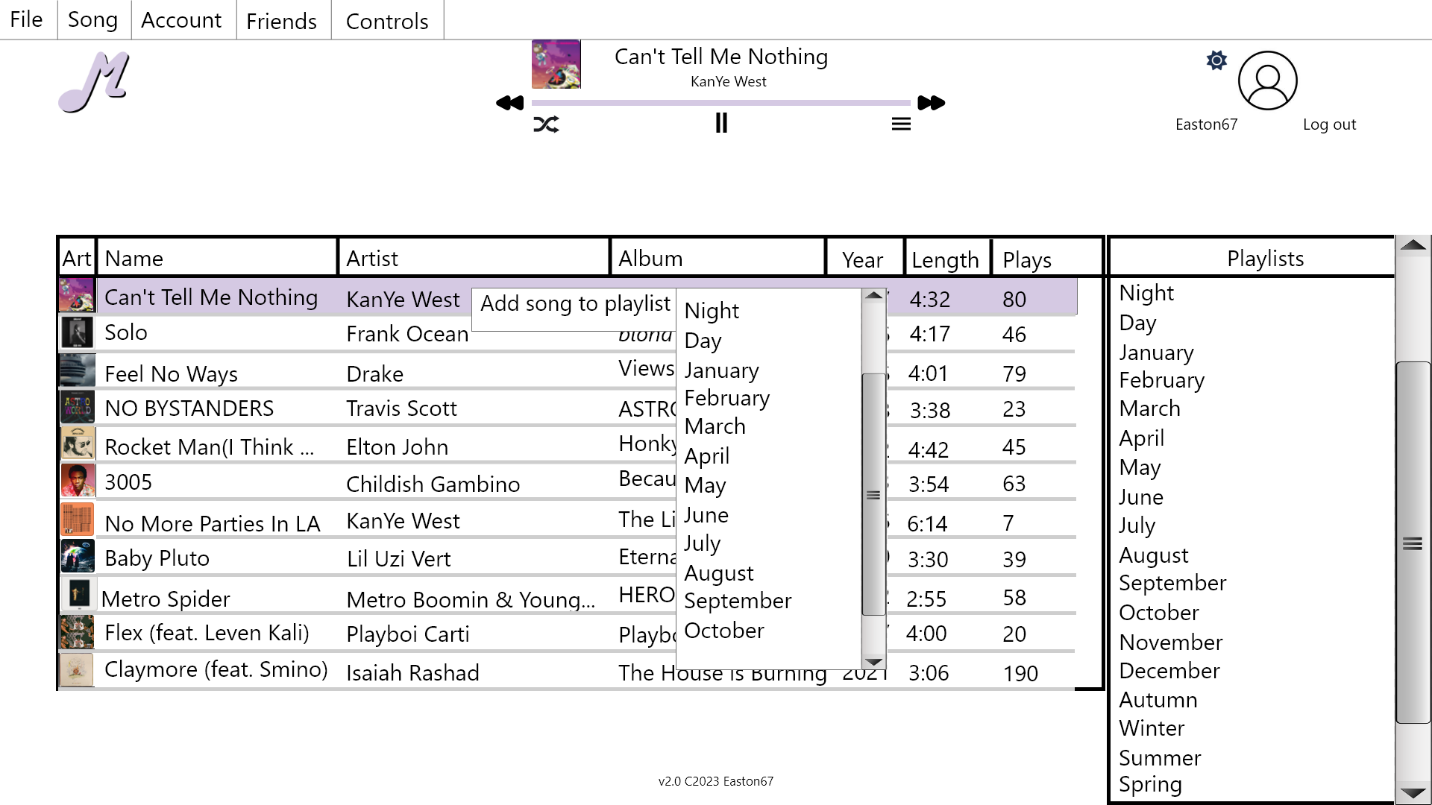
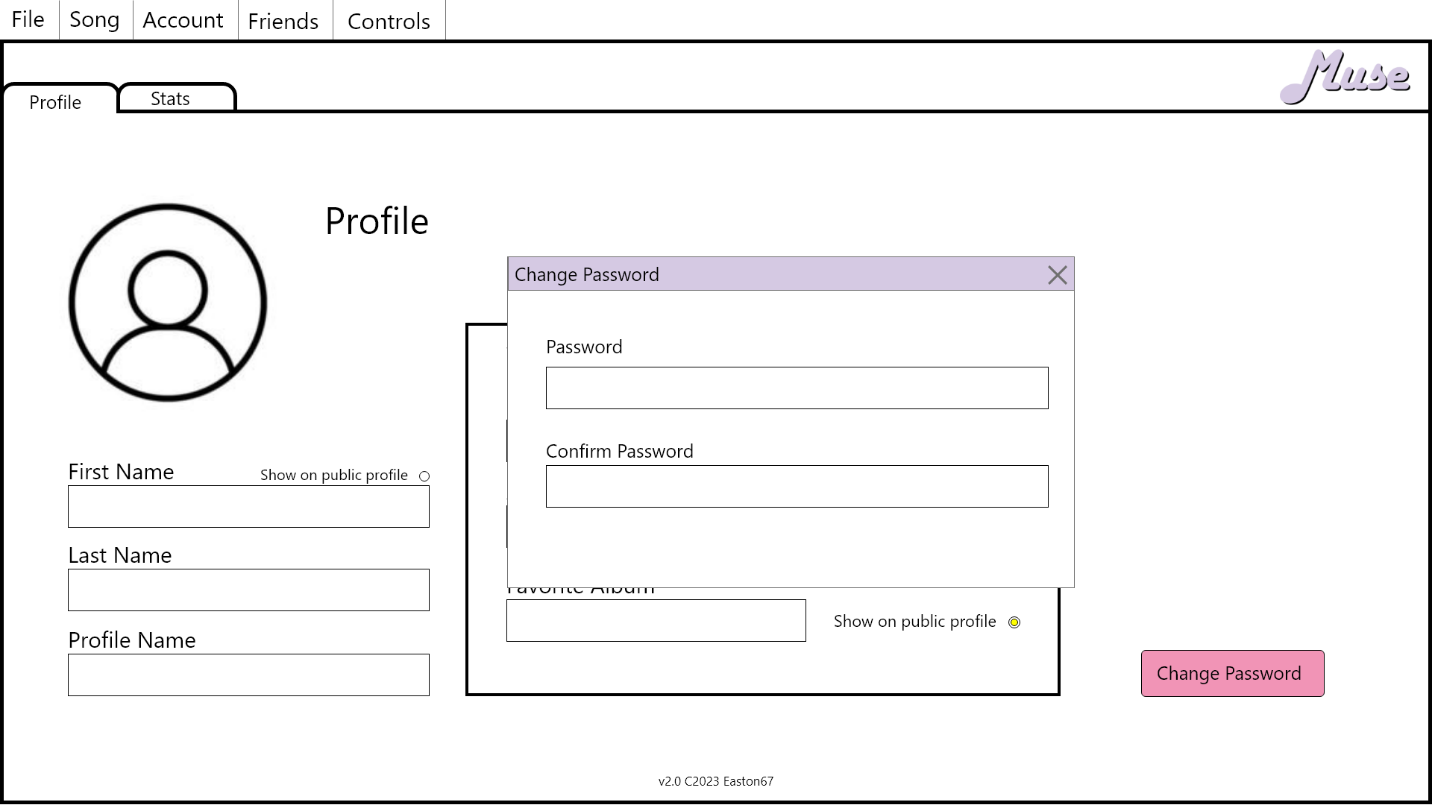
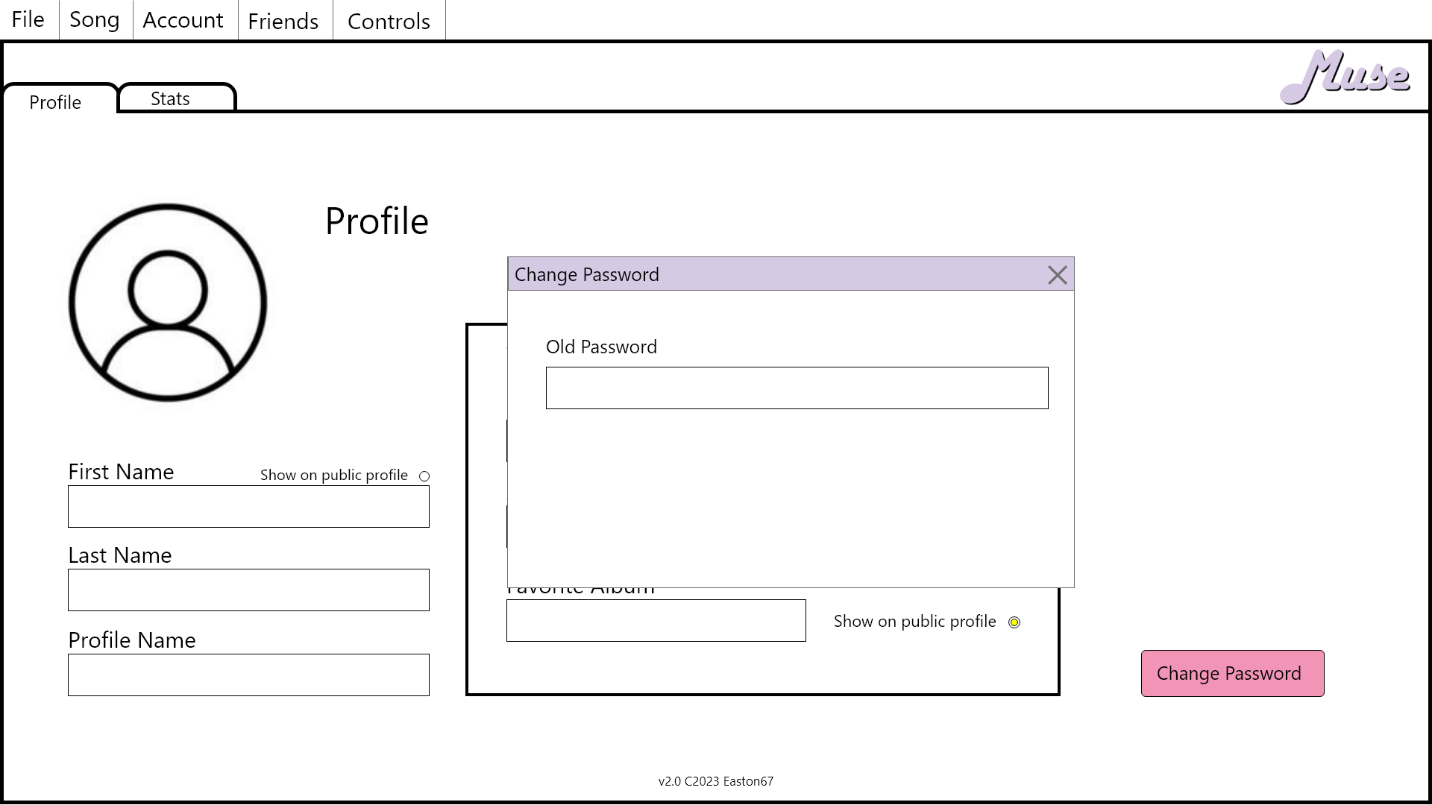
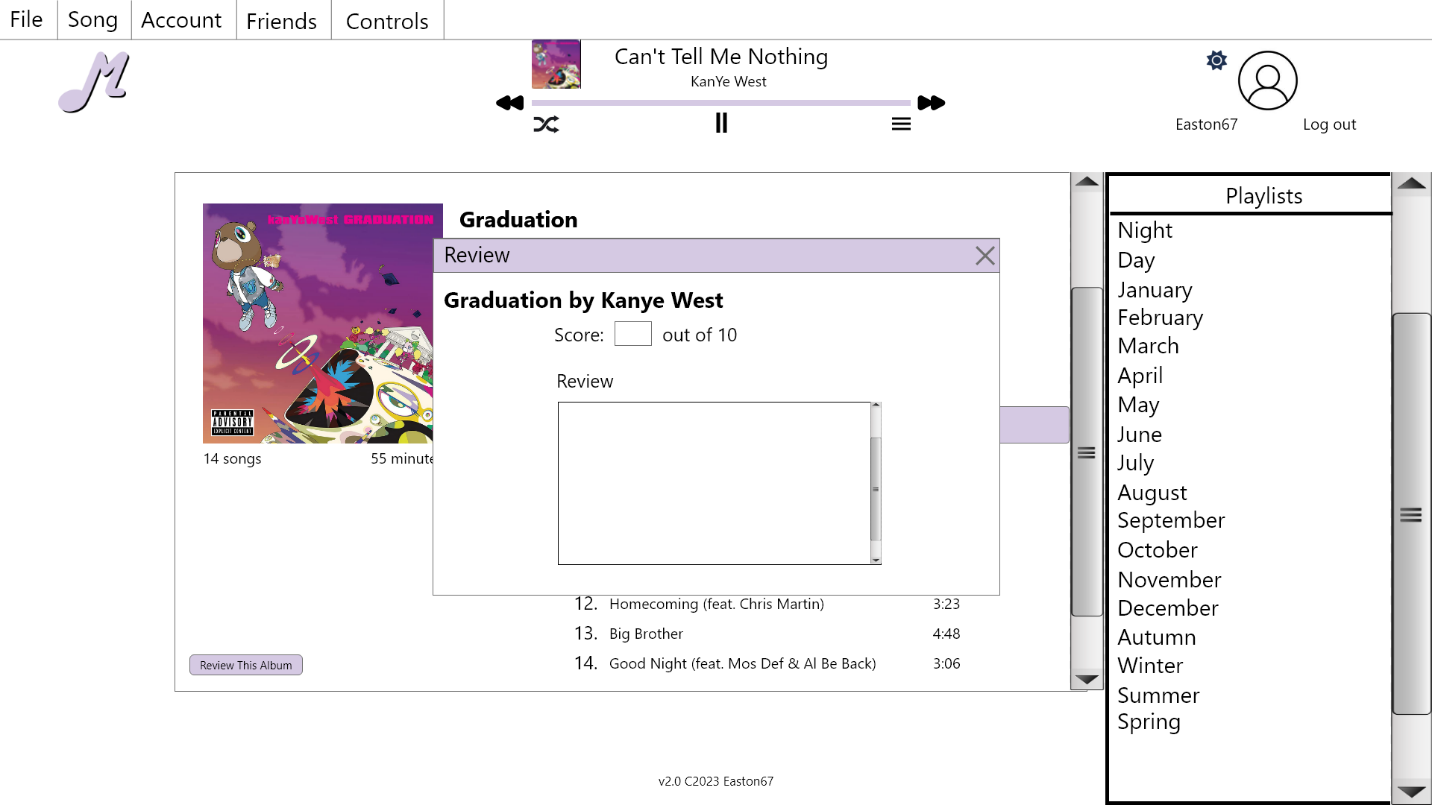
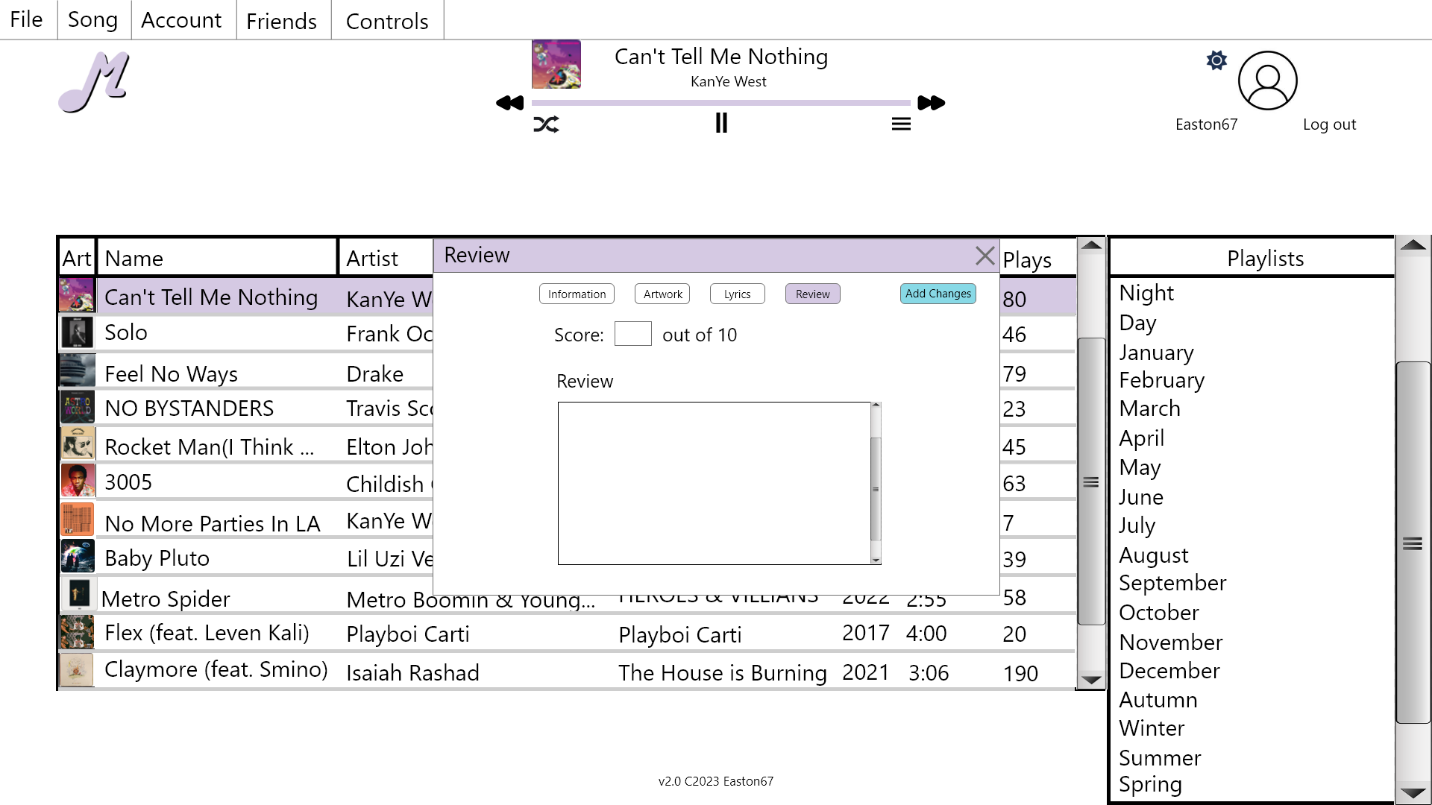
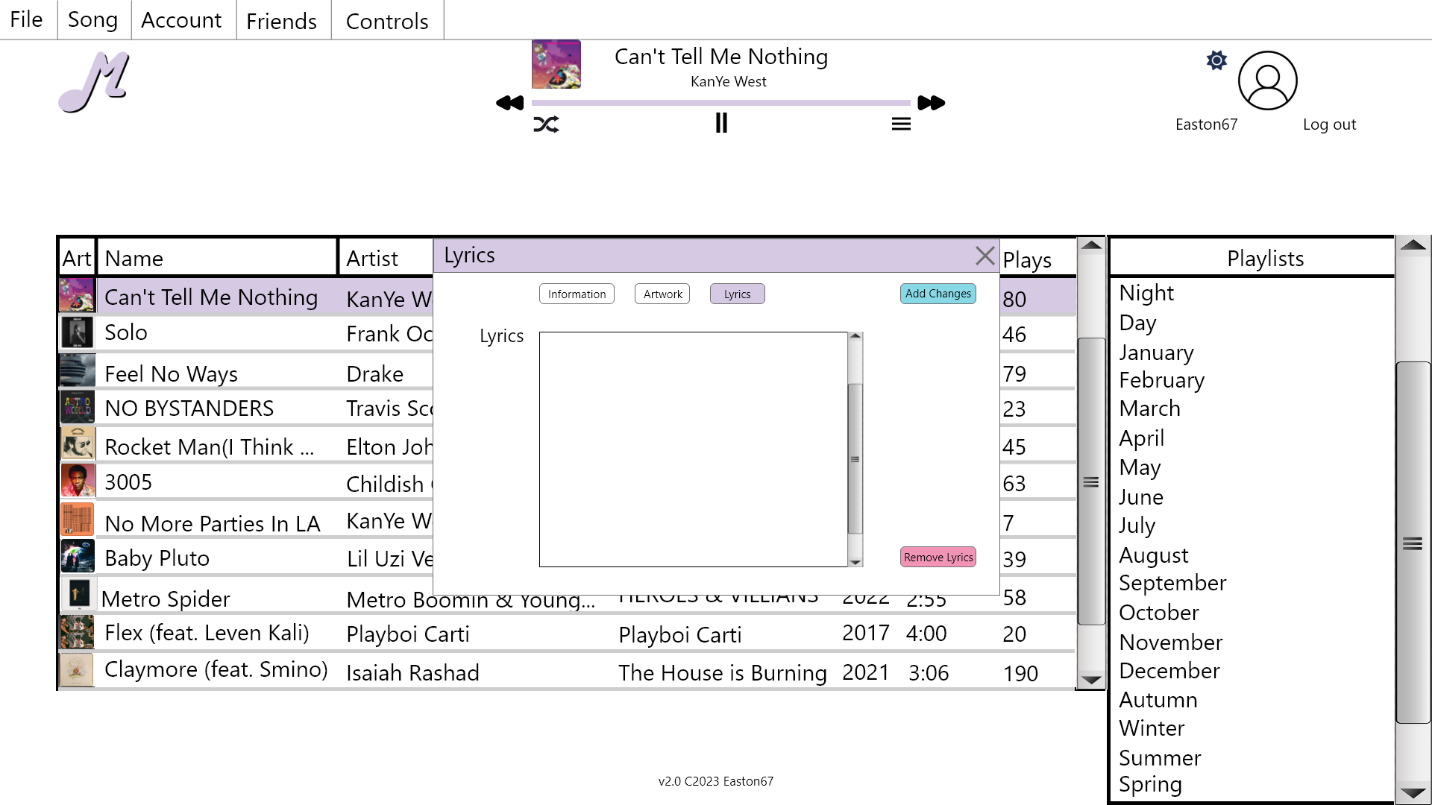
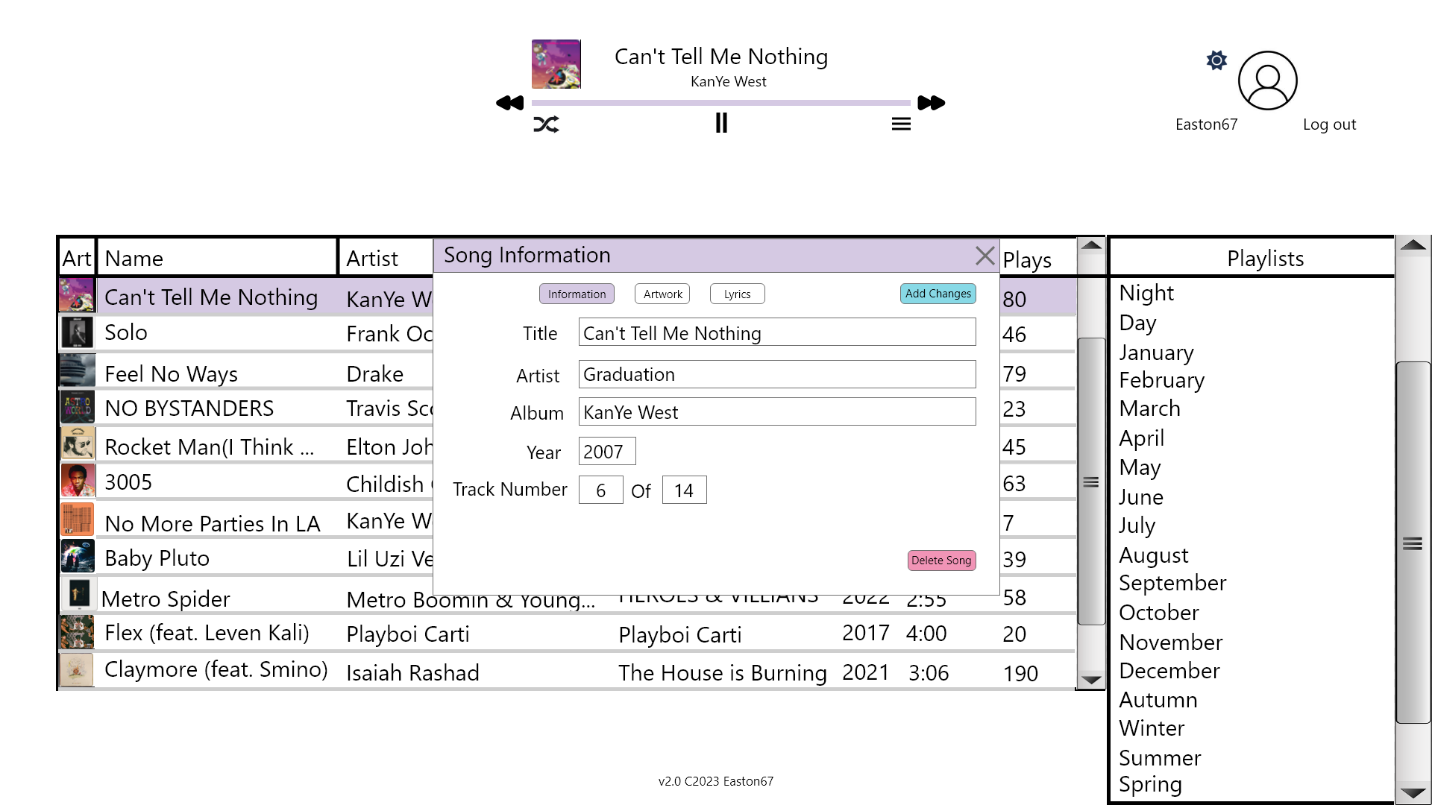
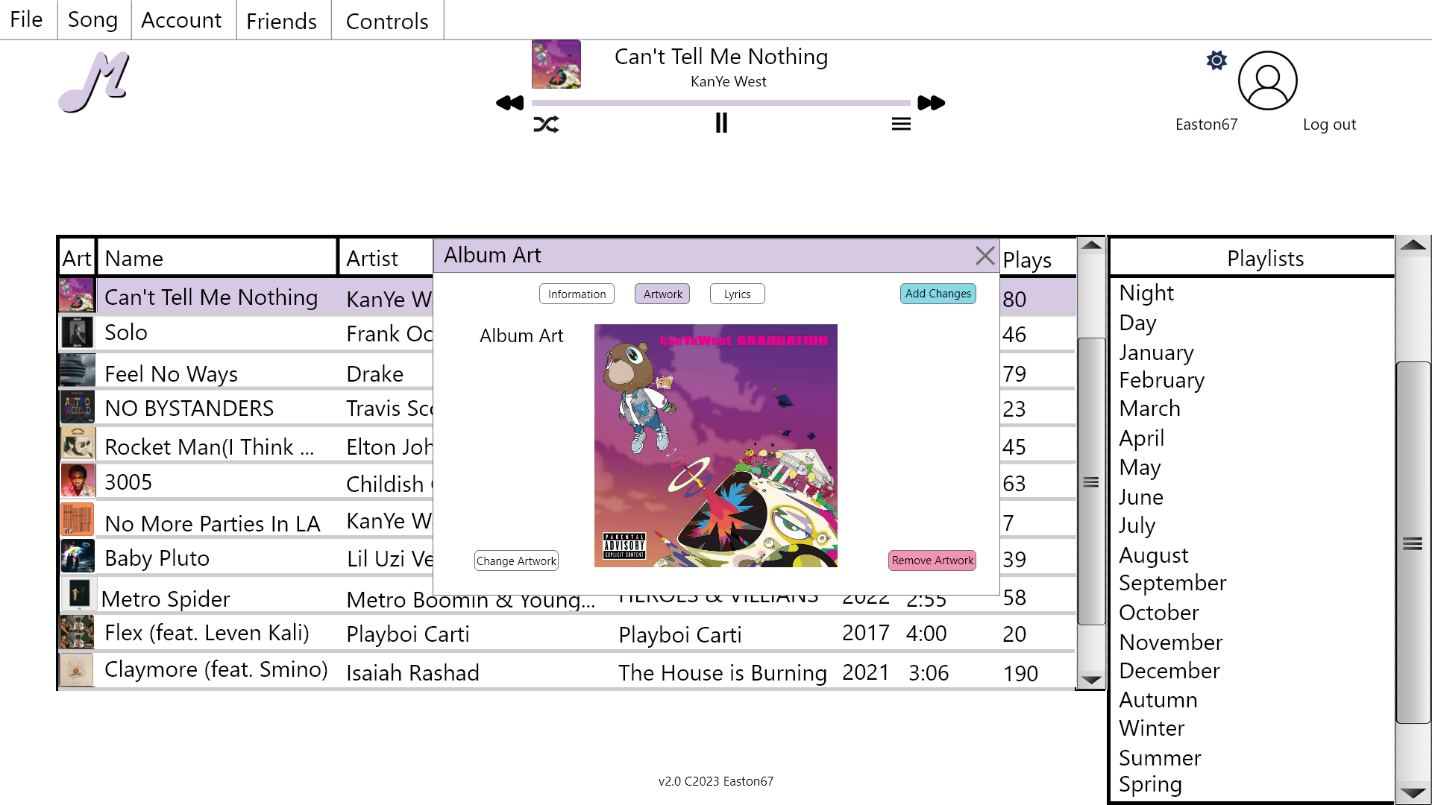
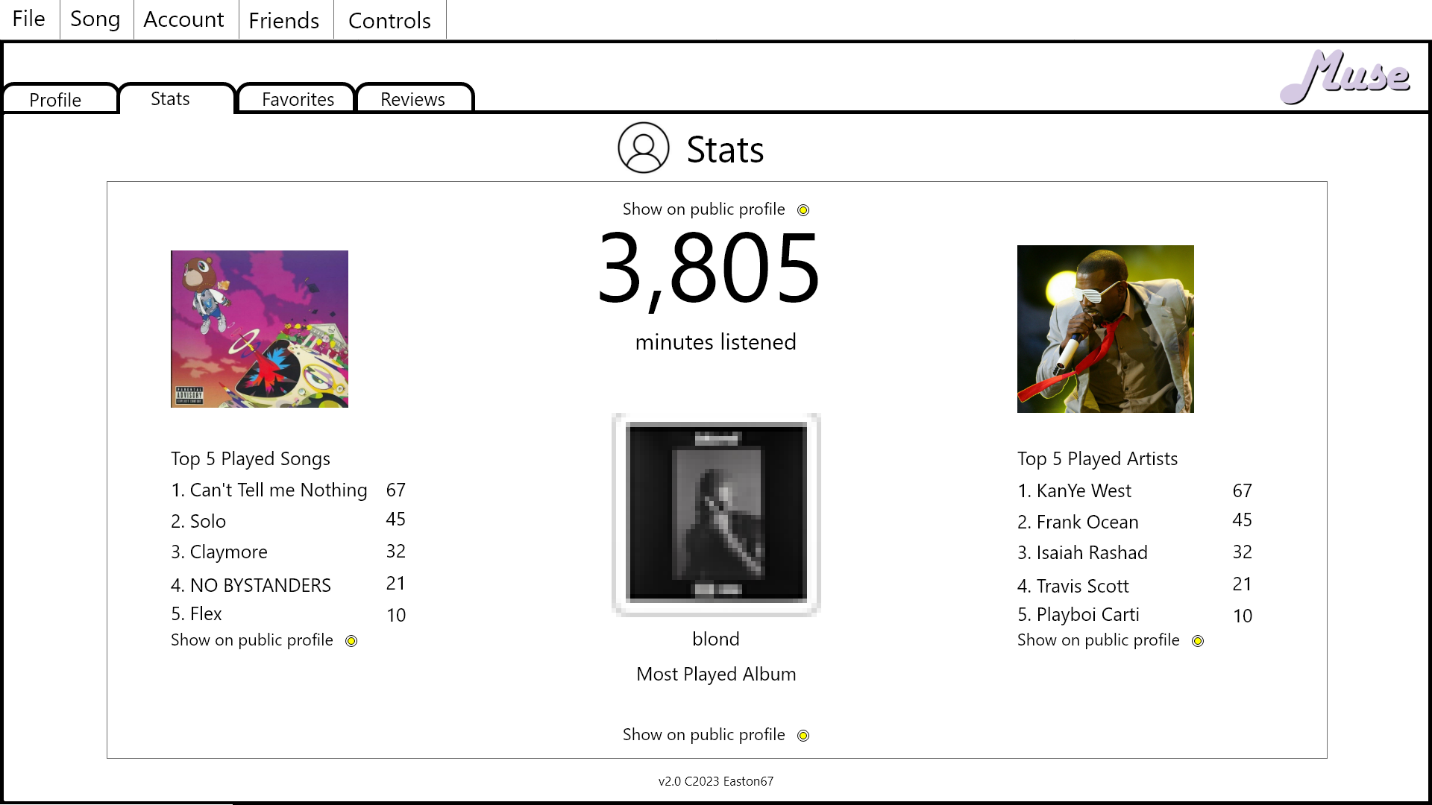
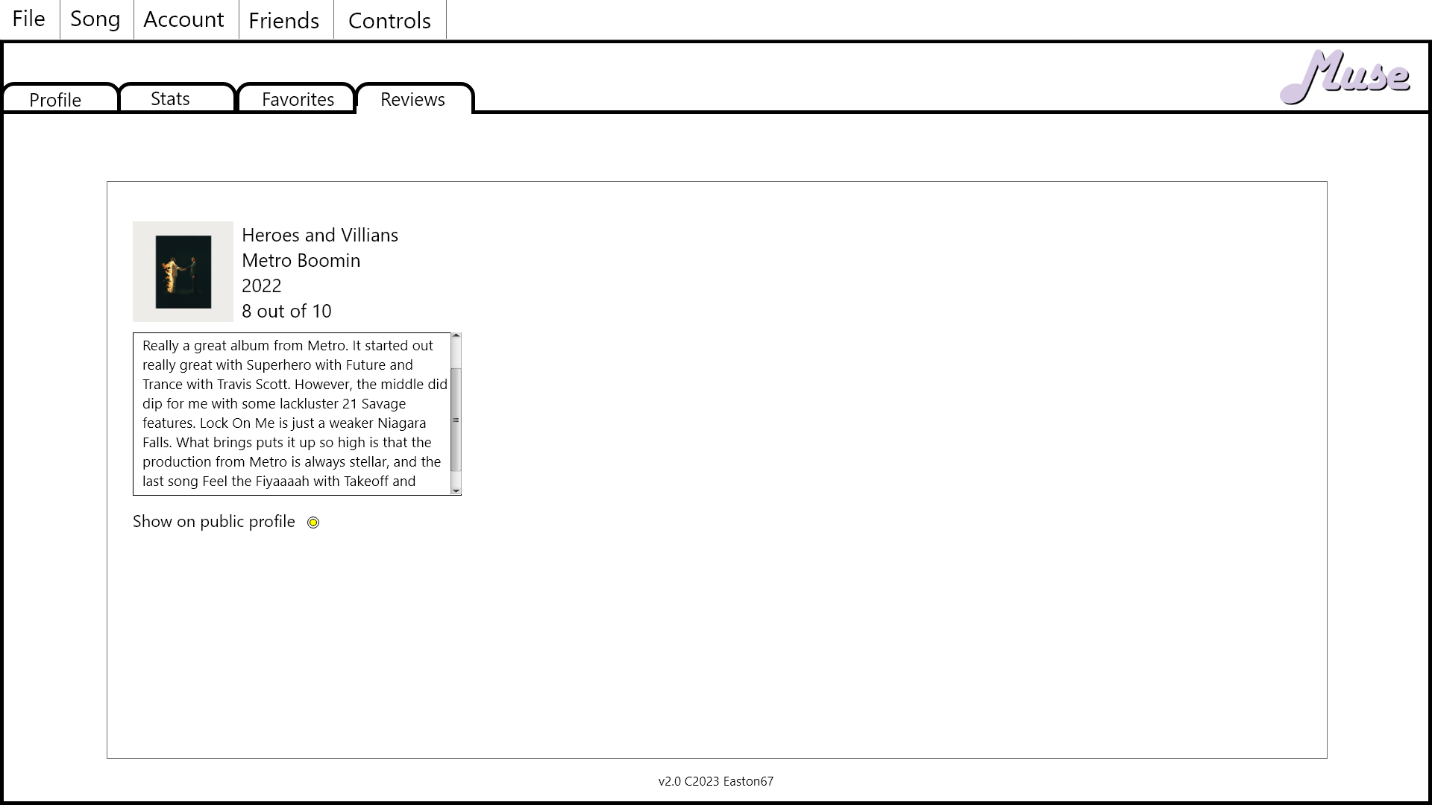
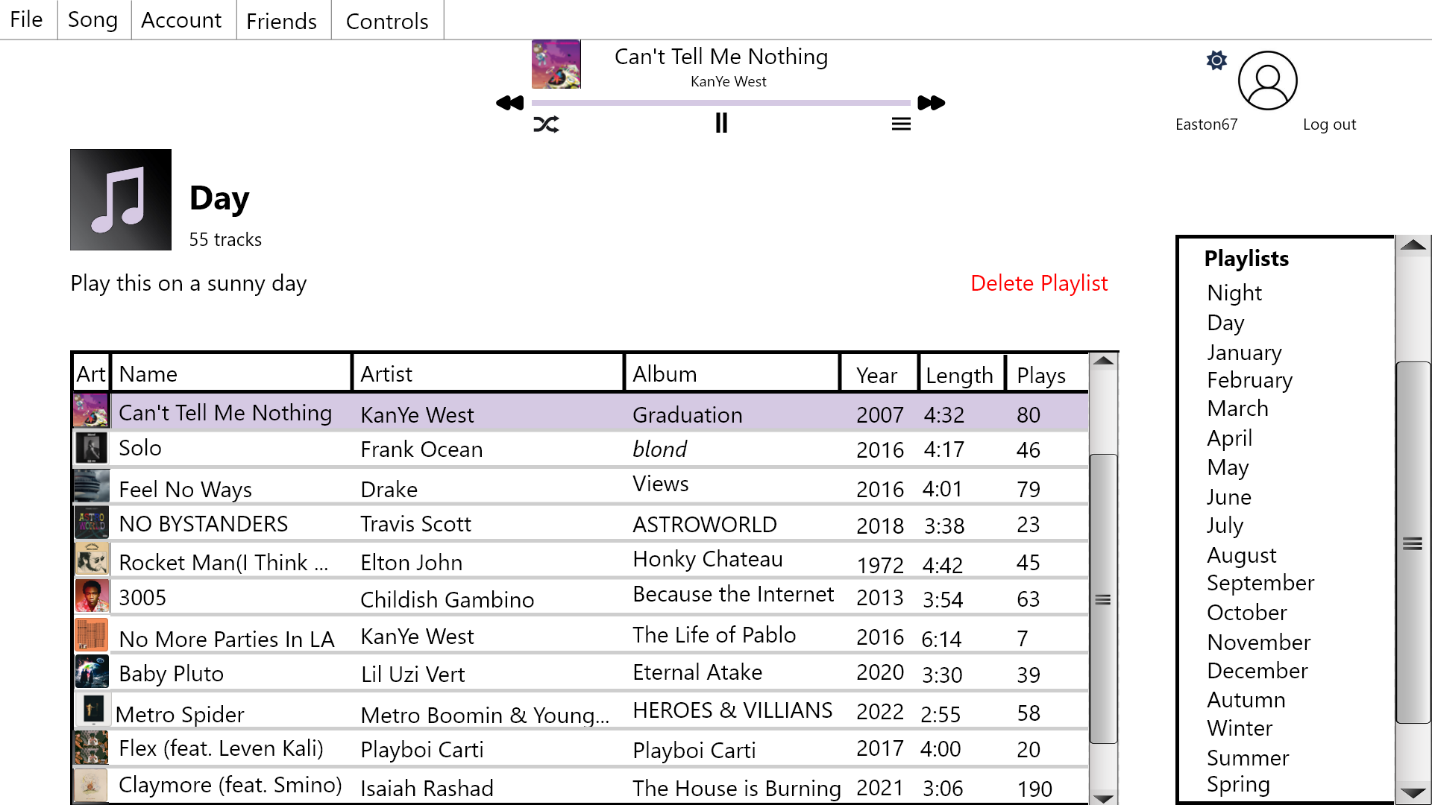
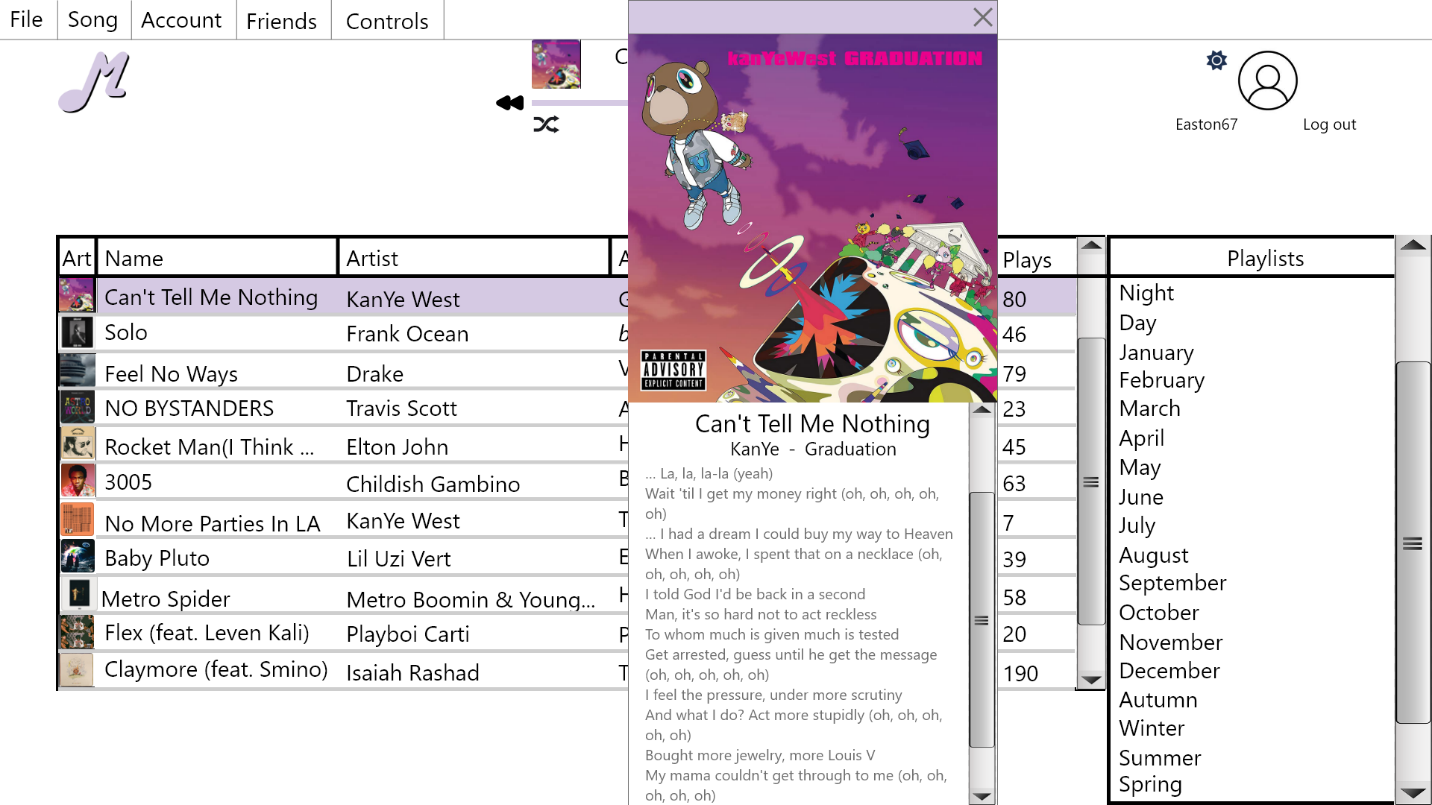
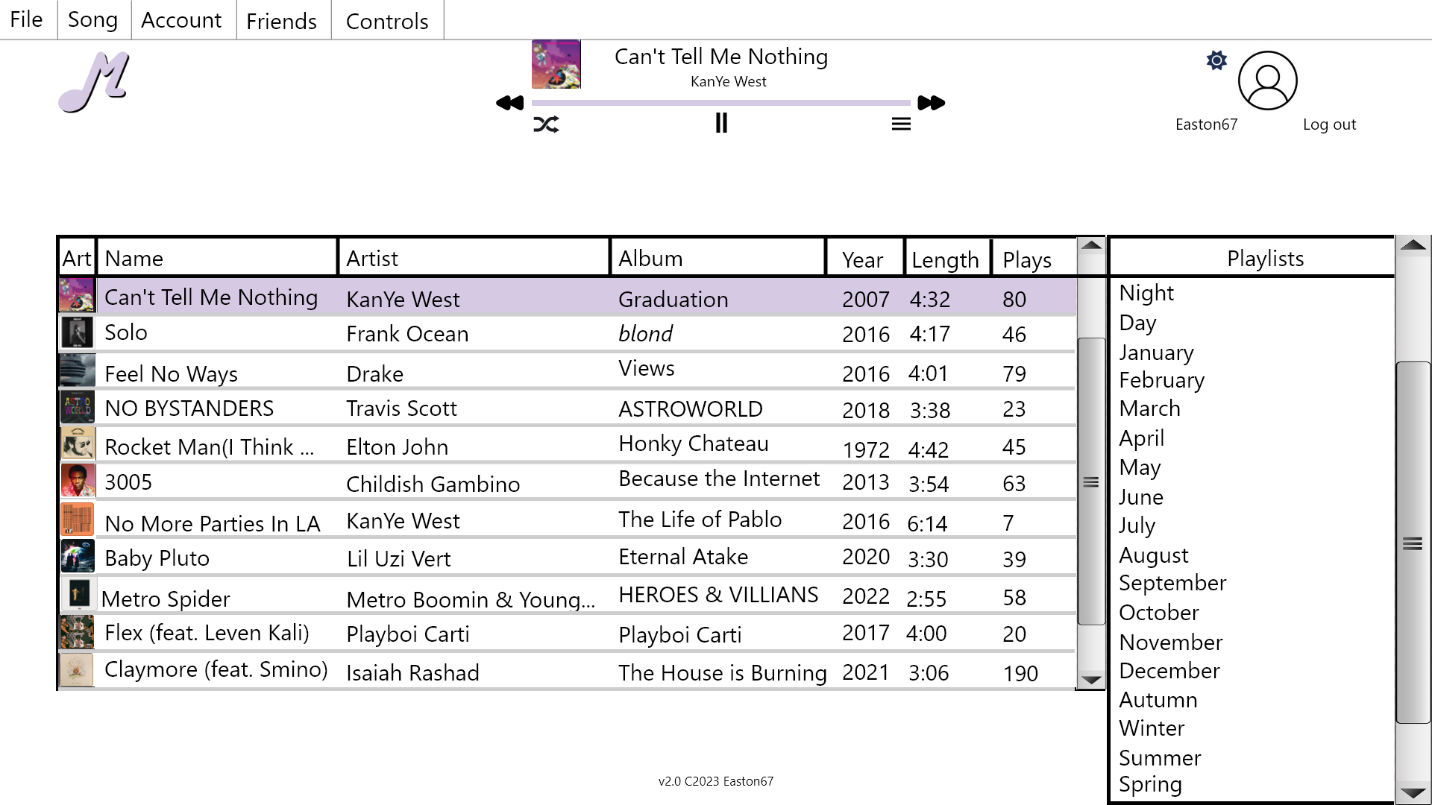
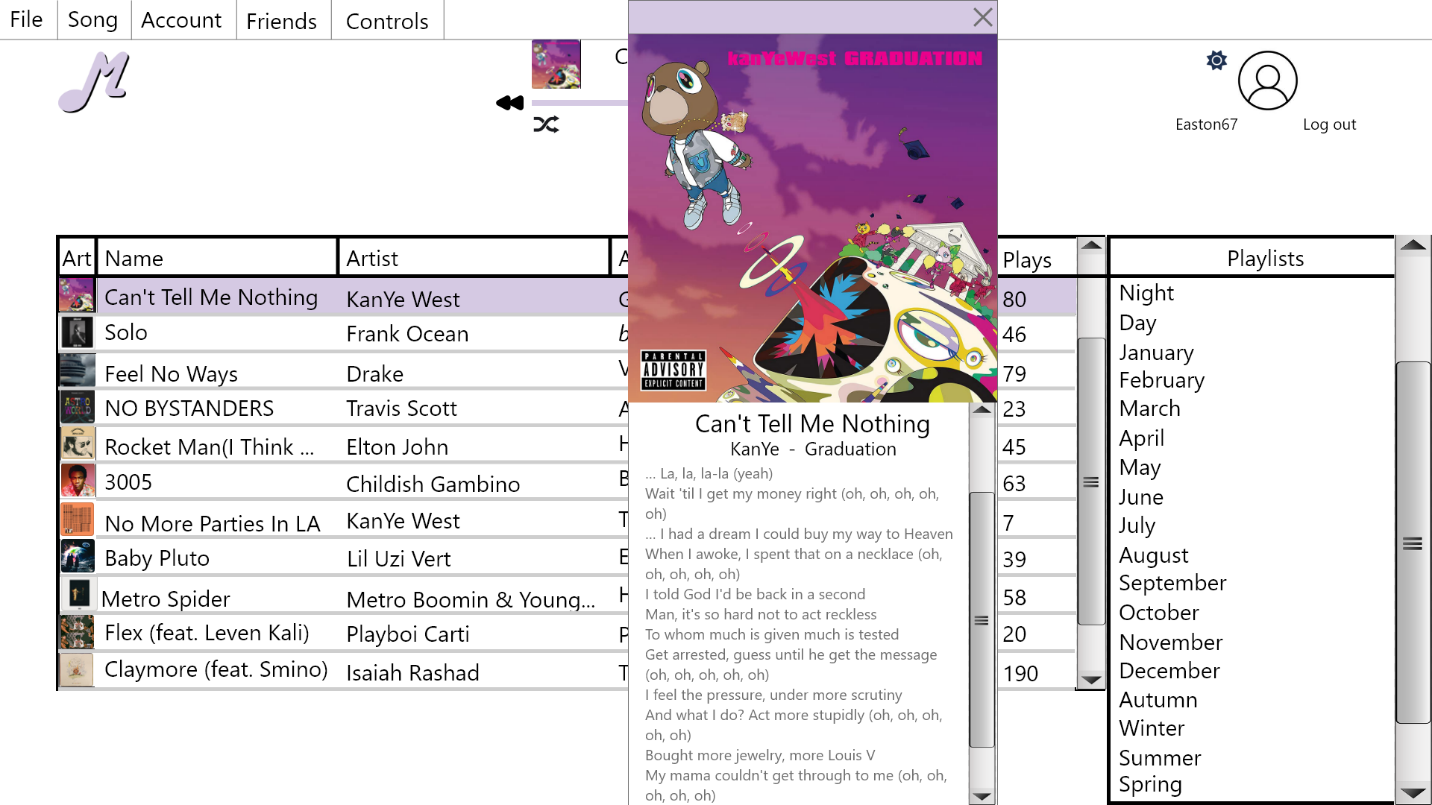
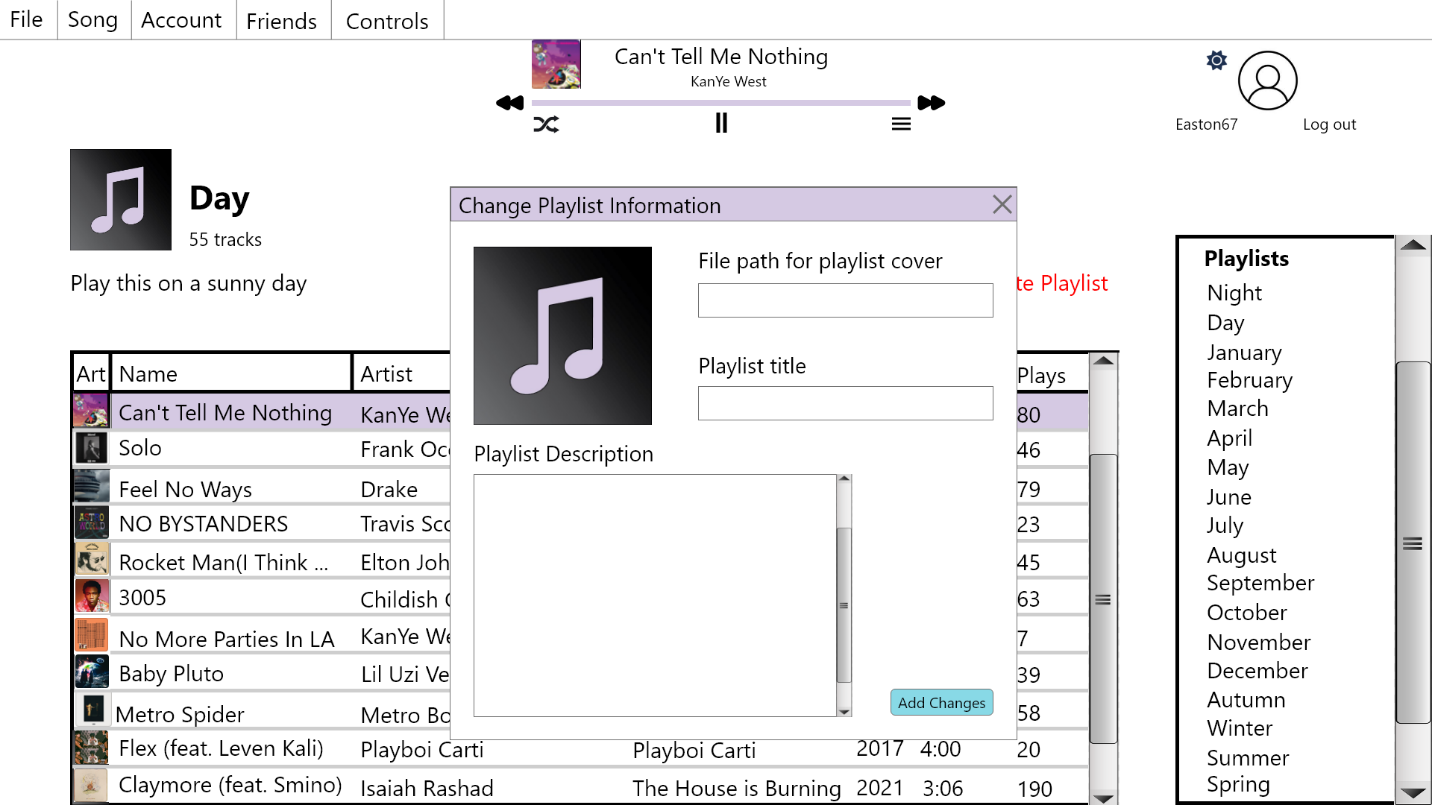
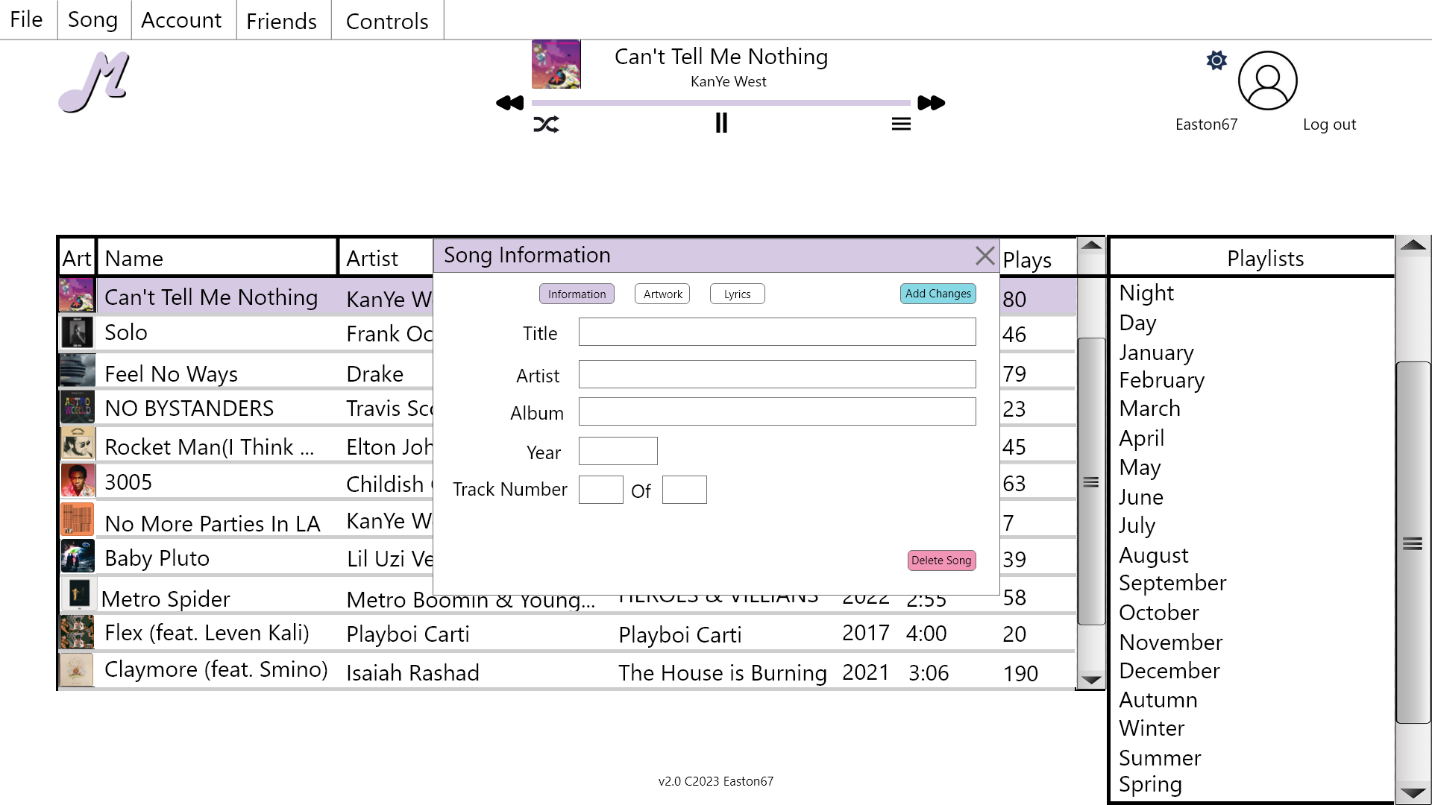
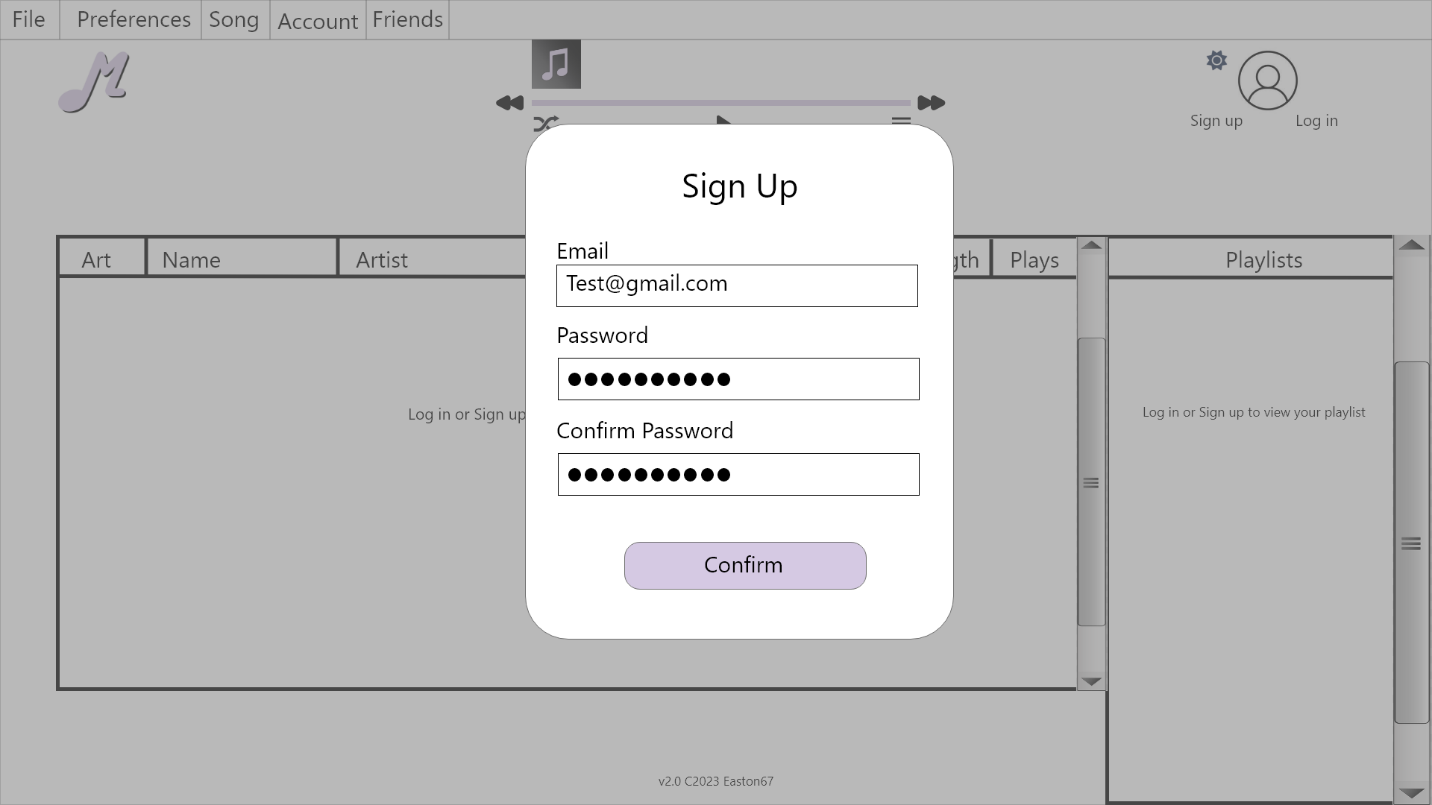
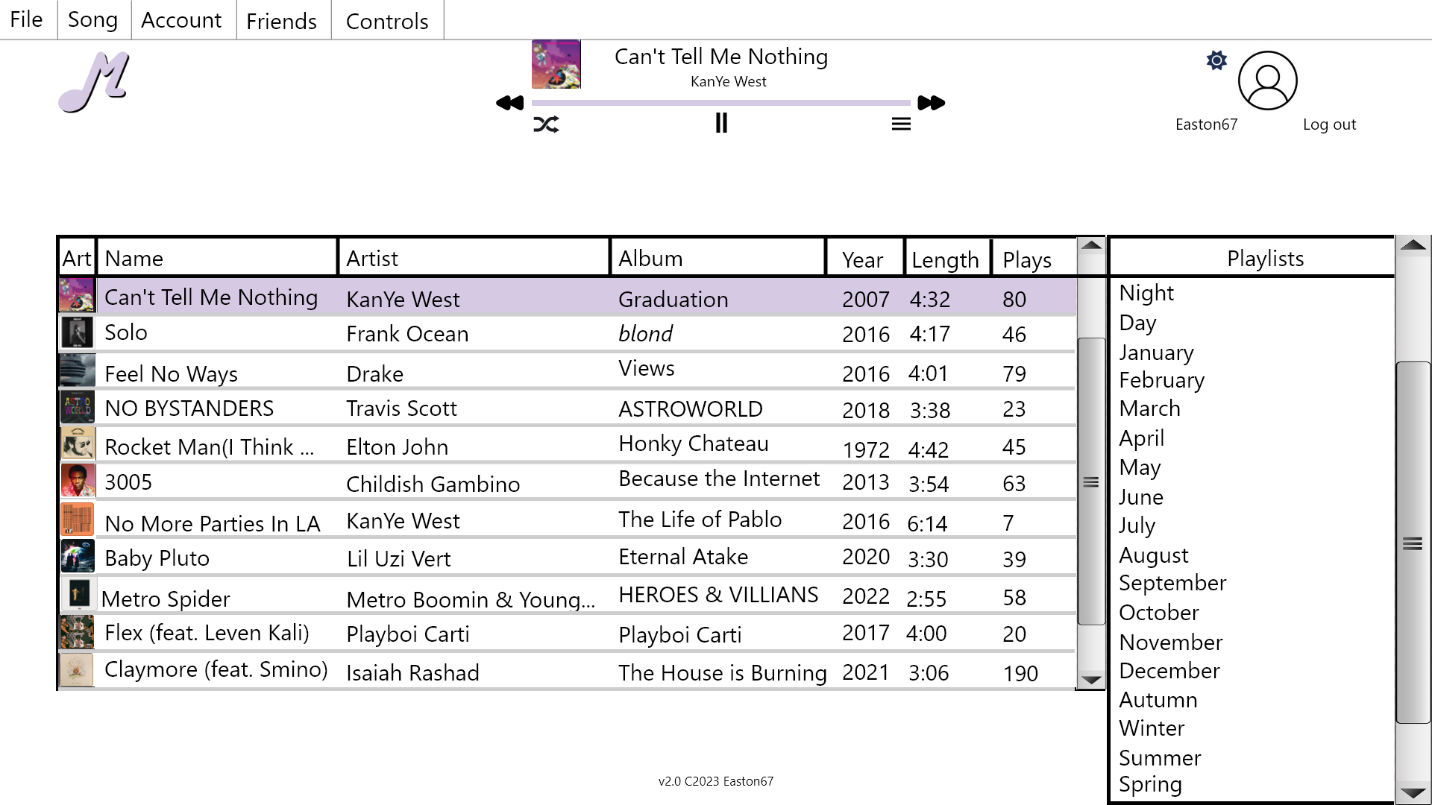
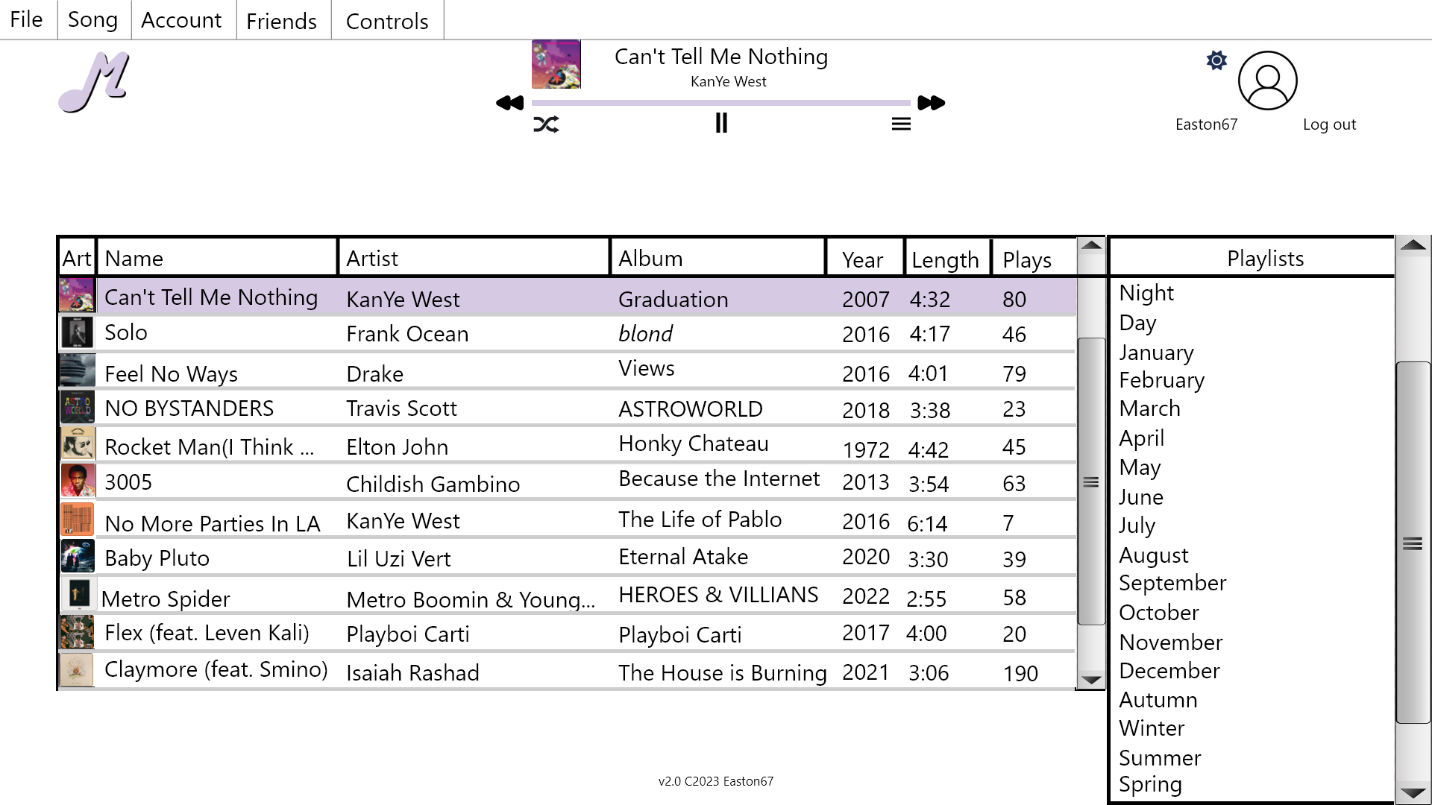
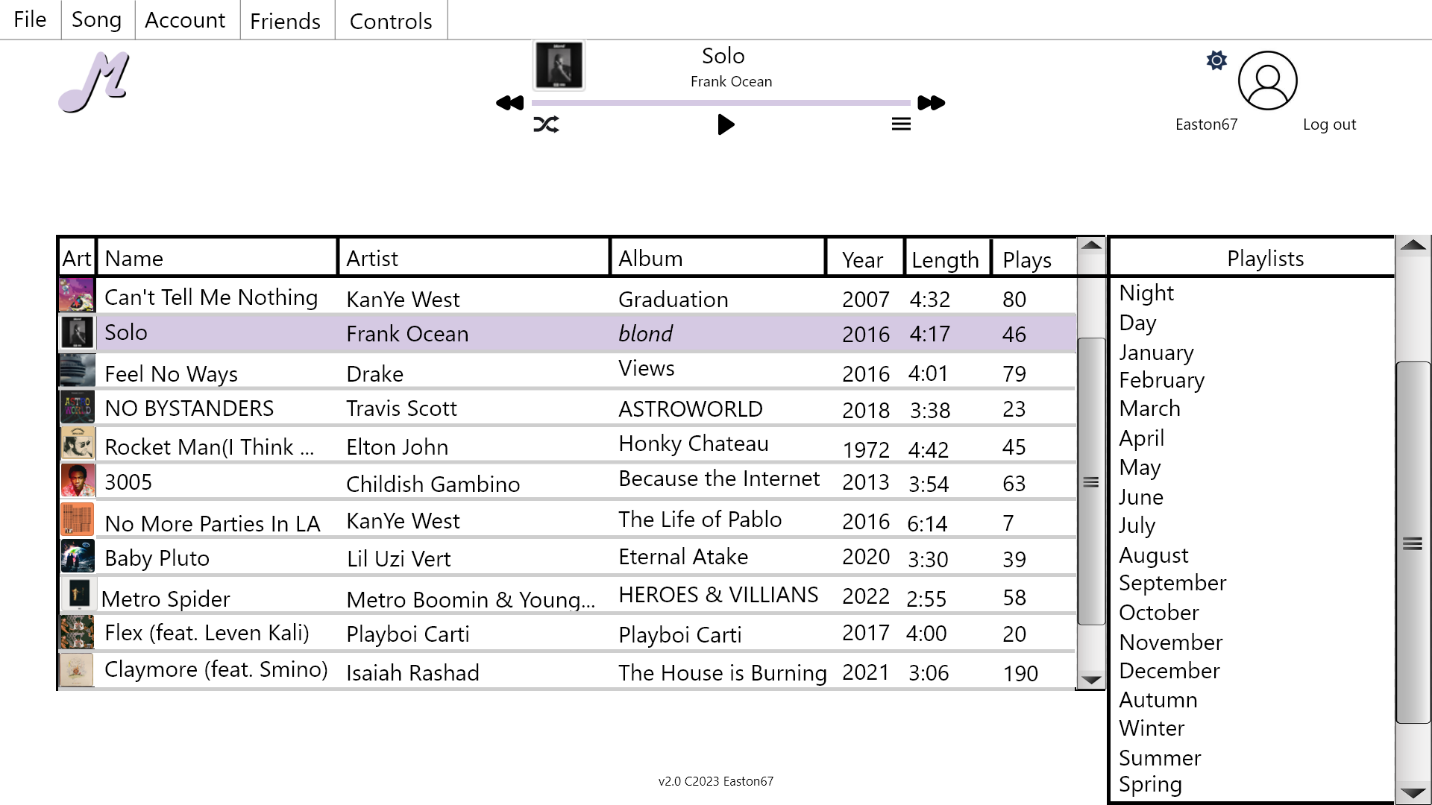
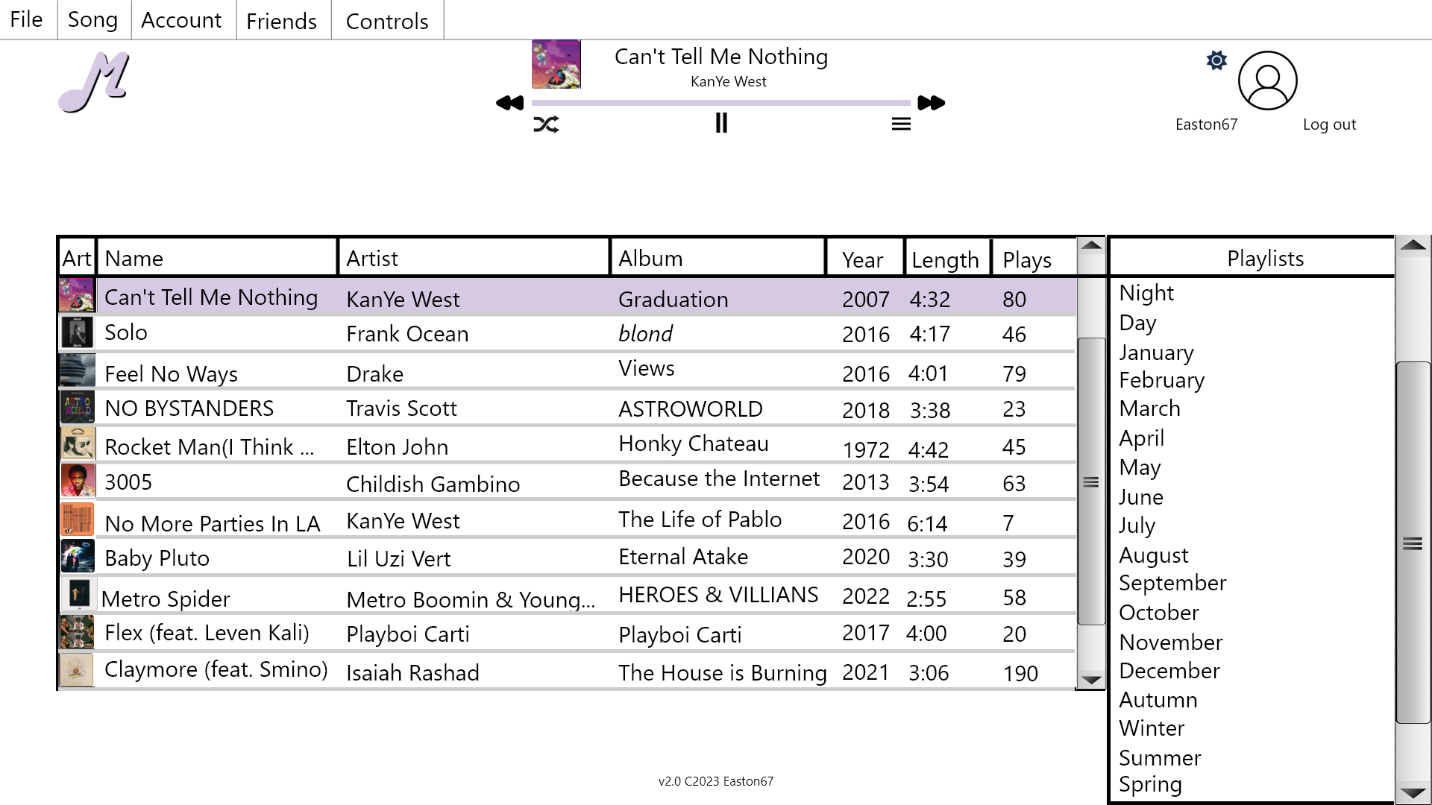
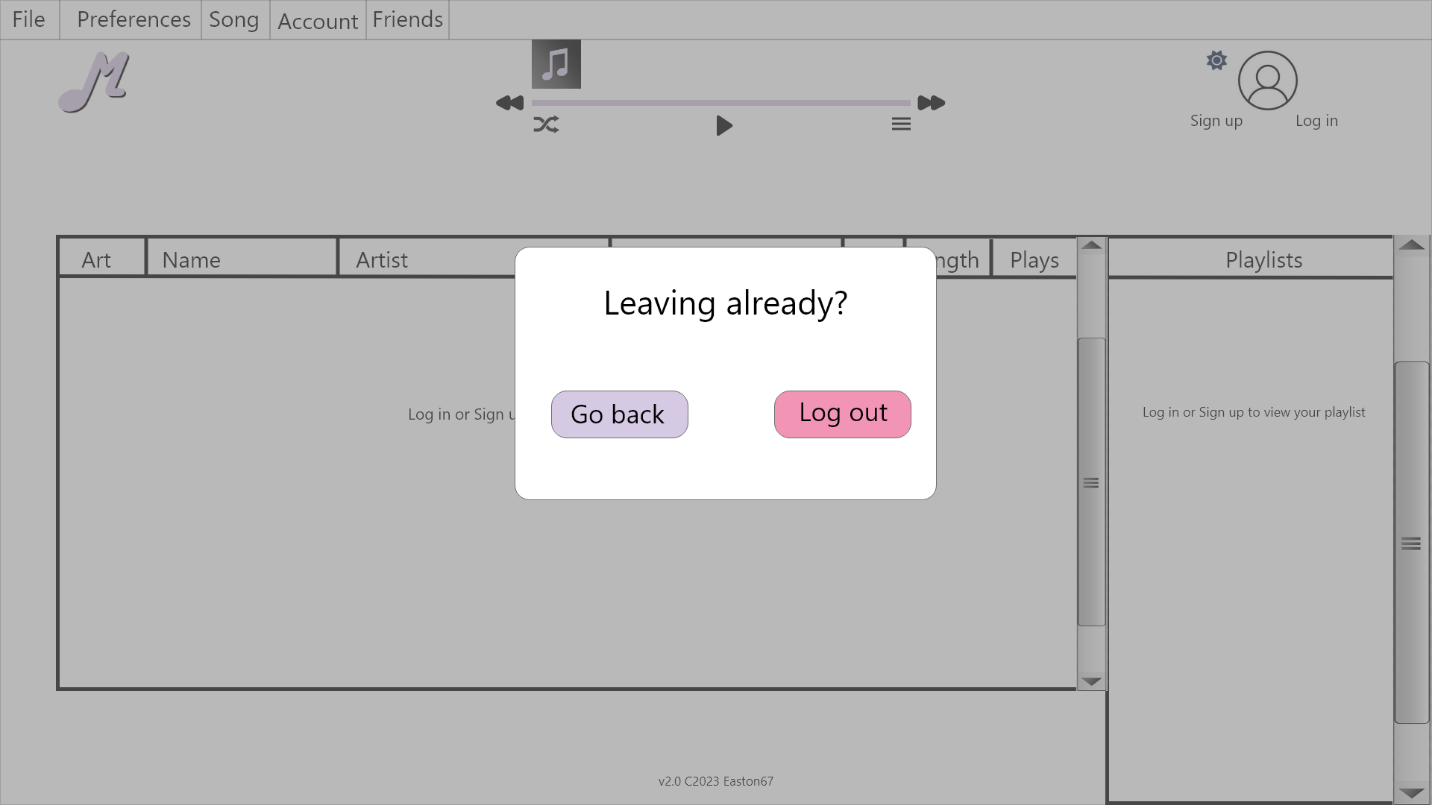
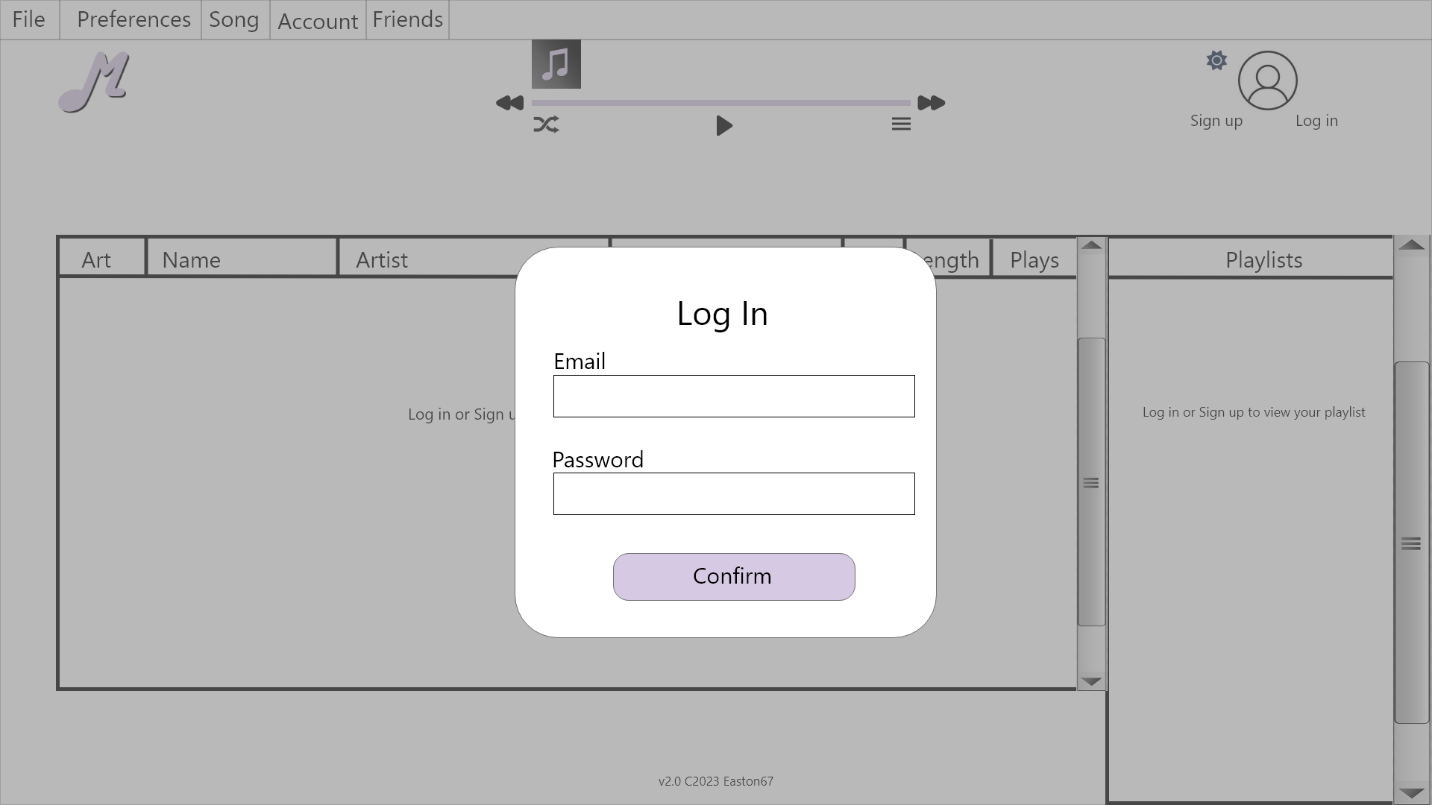
A diagram of a music album

Description automatically generated

# Feature List and User Stories



# User Interfaces



# Review

Over the last almost 2 months now, I have been updating my project from .NET 1. Muse was born out of wanting a well running music application and wanting to push myself to become fluent in the C# language. Going from the WinForms version of my original project, to this version using the local database is a night and day improvement. I finally added some much-needed features like playlists, and even something as crucial as adding the delete functionality to songs.

I’ve learned a lot this semester, and I have spent many late nights trying and experimenting with WPF to get this version that I am proud to turn in as my final project. This might be my final project, but not the final version. I intend to add a lot of what I originally envisioned over winter break to keep my mind sharp, and to build this app out to what I know I am capable of coding. If I could go back, I would tell myself to really think about what each feature is doing and how they interact with each other. If I knew what I knew now at the start of the class, I would have a much more robust application, and since I have about a month off, I intend to use that time wisely to build something I can show off to future employers.