

Music Learning Application

Software Requirements Specification

CSCE 247

Ian Attmore, Matthew Radin, Ryan Leadbitter, Josiah White, Owen Coulam

Table of Contents

1. Introduction
2. Stakeholders
3. Constraints
4. Overall Description
5. Business Use Cases
6. Functional Requirements
7. Non-Functional Requirements
8. Definitions and Acronyms
9. Competitive Analysis
10. References
11. Appendices

1. Introduction

Purpose

Learning music can often be intimidating to newcomers. This application is designed to assist new musicians in learning to read sheet music and the fundamentals of playing their instrument of choice. Various instruments including piano, guitar, snare drum, and more will be available for users to broaden their musical horizons.

Scope

This document includes:

- The personas of potential users and stakeholders invested in this project.
- Any constraints that this project may have.
- A description of the website/app and its use cases.
- Shows both the app's functional and non-functional requirements.
- A competitive analysis to outline the purpose and advantages of the project.

2. Stakeholders

- Aspiring Musicians
 - Middle School/High School Band/Orchestra Students
 - Professional Musicians (looking to improve skillsets)
 - Freelance Musicians

- Hobbyist Musicians
- Music teachers
 - Public school teachers
 - Secondary and Post-Secondary Teachers
 - Private/Online Tutors

Personas

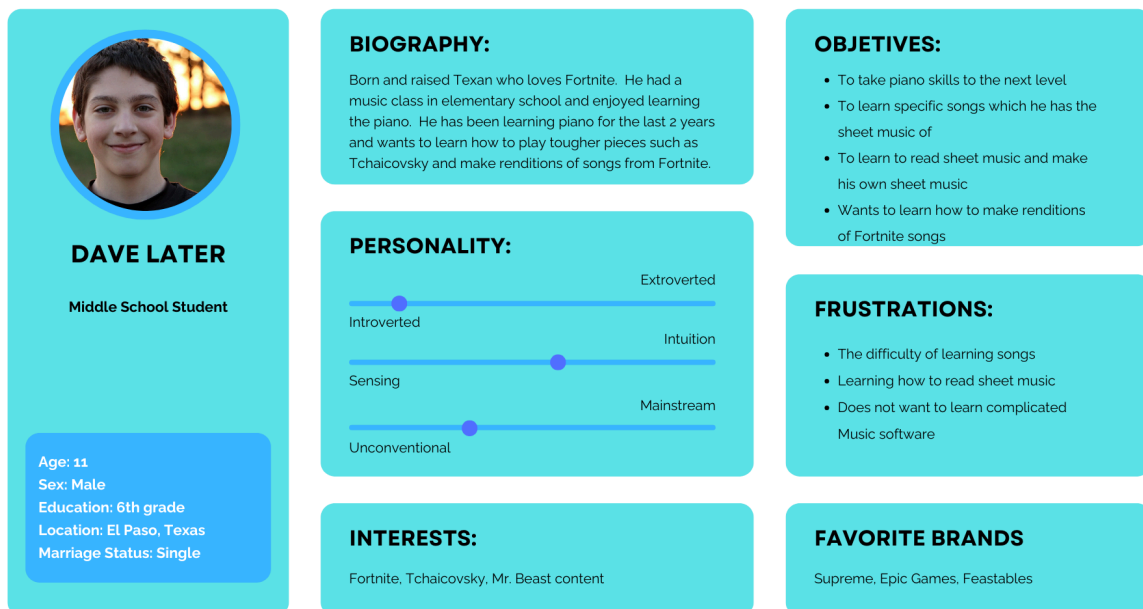


Figure A

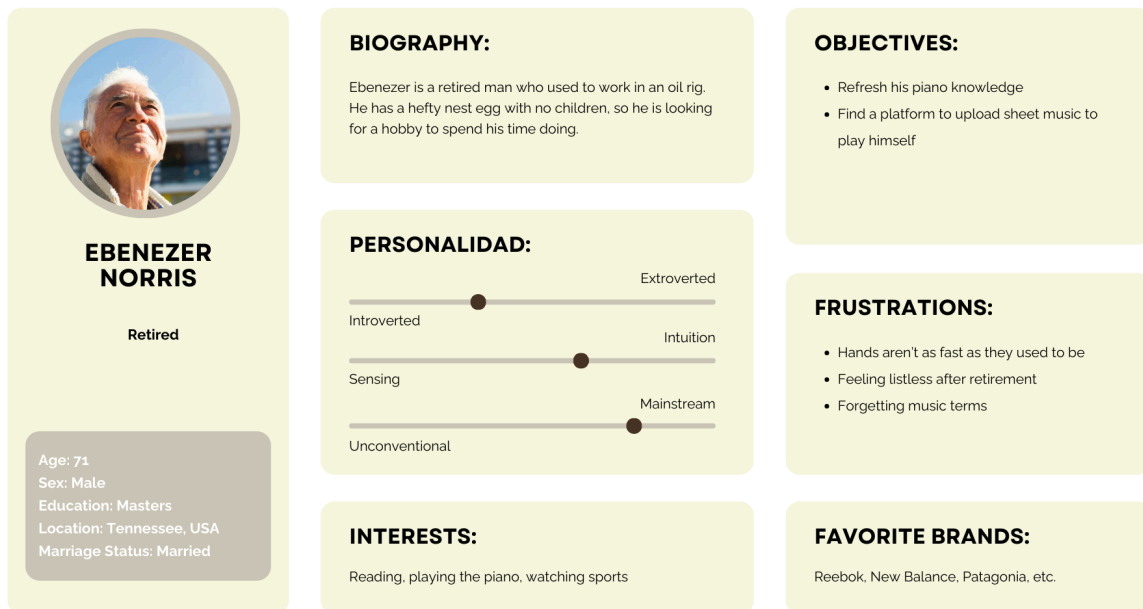


Figure B

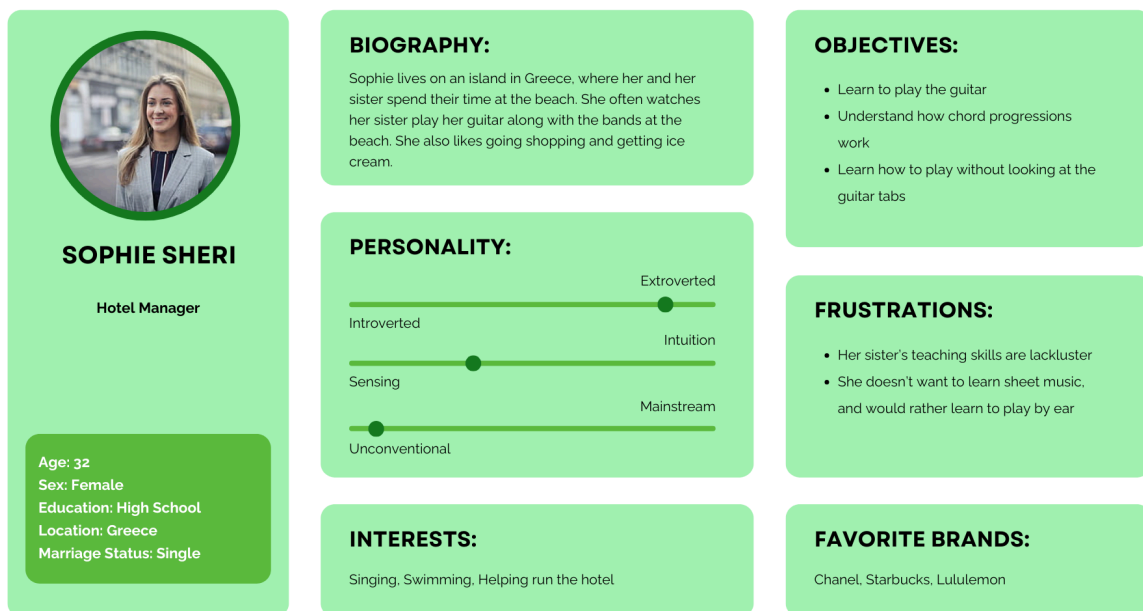


Figure C

3. Constraints

Time Constraints

- This project is to be completed within a timespan of one month.

Monetary Constraints

- This project's development has been allocated a budget of \$0.

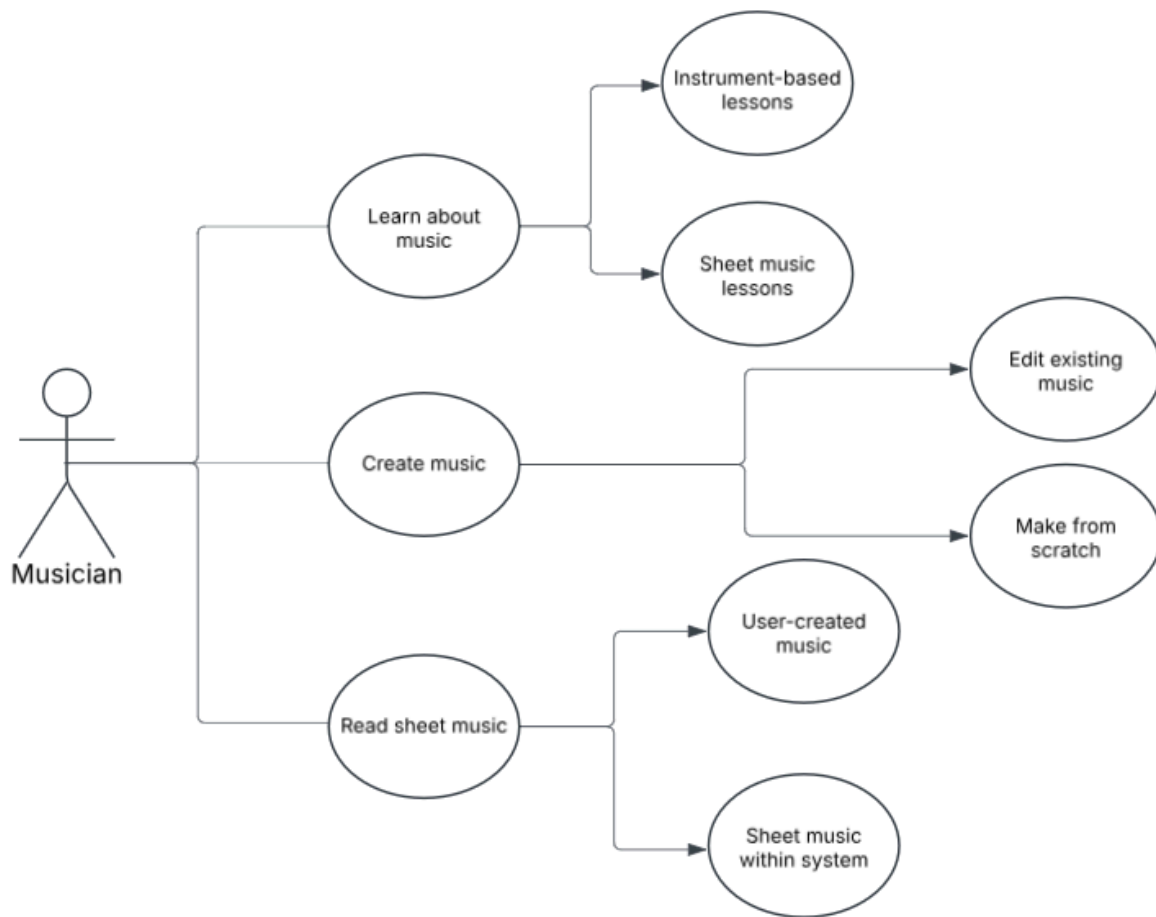
Technical Constraints

- The code must be done in Java.
- The website must be accessible on a variety of different devices, including PC, laptops, and mobile devices.

4. Overall Description

The app must be accessible by users from their mobile devices, PCs, and laptops. The user will be given options of what instrument they want to play. The user will have a database of sheet music to choose from. The app will have a simple, intuitive interface for the user to interact with.

5. Business Use Cases



6. Functional Requirements

[Requirements Spreadsheet](#)

7. Non-Functional Requirements

- User Experience Requirements
 - The system's user interface shall be implemented in an intuitive way that is easy for all users to understand.
- Usability Requirements
 - The system shall be used by users with a variety of musical experience

- Performance Requirements
 - The system shall be able to run on multiple platforms with or without an internet connection
- Maintainability and Support Requirements
 - The system shall be able to run on Window, Linux, and Mac systems
- Security Requirements
 - The system shall store user's information securely
- Legal Requirements
 - The system shall comply with all state and local law.

8. Definitions and Acronyms

- Tablature: A musical notation designating the location of the notes on the instrument rather than the musical pitches themselves.

9. Competitive Analysis

	Flat.io
Strengths	<p>This website is designed to be able to compose sheet music for a wide variety of instruments. This app is very easy to learn and understand for someone with a background in music.</p> <p>Users are able to both create sheet music from scratch, as well as upload existing sheet music from their computer onto the website. They are able to select multiple instruments and have the sheet music display each instrument independent from one another.</p> <p>The website features a brief tutorial that goes over the basics of using the website, and is fairly simple to use</p>

	<p>after. There are many different options and objects that are available (even for free) to use for sheet music, making this a great tool for practiced musicians who are looking to compose free of charge.</p>
Weaknesses	<p>Many features, such as more than half of all instruments, are locked behind a paid monthly subscription. This may limit the user from being able to compose complicated pieces of music.</p> <p>The website can overwhelm users who have little to no musical background. It features a plethora of note types, measures, and articulations beyond the standard for music, which may intimidate users when creating sheet music. Additionally, no tutorial is given for what these features are; you, as the musician, are expected to know them.</p>
Audience/Focus	<p>This website is designed for musicians who have some musical background and are looking to compose for free or very cheap. This app can be intimidating for new musicians, which can create a high barrier for entry.</p>

	Songsterr
Strengths	<p>This website is designed to be a site to collect and showcase user-made guitar tabs. A wide variety of songs, genres, and musical instruments are available to see.</p> <p>Users can use the site to create a tab for any song and upload it to their library, allowing other users to rate and use their tablature.</p> <p>The site also features a help tab that details information such as website navigation, how-to-read tabs, and a section for contacting support.</p> <p>Users can play along to either synthesized audio of the instrument or a synced YouTube video of the original</p>

	audio.
Weaknesses	<p>Many features, such as speed, looping, and pitch shift, are locked behind a paid subscription. (This includes lengthy synchronized audio of the original)</p> <p>It has a difficulty meter to rate tabs, but it's unclear exactly how hard a song is and can be rather inaccurate.</p> <p>The website doesn't have support for sheet music, so any song will be in guitar notation (besides bass and drums).</p> <p>The site depends on user submissions, so non-mainstream songs are less likely to be available.</p>
Audience/Focus	This app is designed for any musician, from beginner to advanced, who is aiming to learn or tabulate songs. The features are designed to make the learning and play-along experience as clear and simple as possible.

	Garageband
Strengths	<p>This app provides a variety of interactive instruments for the user to practice with virtually. This allows the app to be accessible to a wide audience of different musicians and aspiring musicians.</p> <p>It features a robust sound library, allowing the virtual instruments to produce realistic sounds. The app can be used as an adequate substitute for many instruments.</p> <p>Users can record their own music that they create using the app and share it with others.</p>
Weaknesses	One weakness of the app is its complicated user interface, especially for newcomers. This can make the app intimidating to get into and create a high barrier to entry.

	<p>This app has no education features, expecting users to already be familiar and experienced with the instruments.</p> <p>Sheet music is not incorporated into the app's design.</p>
Audience/Focus	<p>This app is designed for intermediate to advanced level musicians, specifically those interested in producing original music. The interactive features are designed to simulate real instruments as much as possible in a digital environment, rather than teach users how to play musical instruments and read music.</p>

Summary

	Strengths	Weaknesses	Focus
Flat.io	<p>+Robust</p> <p>+Wide range of tools</p>	<p>-Limited educational value</p> <p>-Difficult for beginners</p>	Sheet music creation
Songsterr	<p>+Easy-to-use</p> <p>+Wide library of songs</p>	<p>-Limited features</p> <p>-User dependent</p>	Song learning
GarageBand	<p>+Realistic</p> <p>+Robust</p>	<p>-Complicated</p> <p>-Limited educational use</p>	Music creation

Following the competitive analysis above, we have learned much about the successes and failures of similar music programs. Throughout this process, we have learned about some of the differences between music software and some of the things they did well that we could try to implement in our software. When we checked Flat.io, we found that it had great tools for working with sheet music but was relatively hard to understand for beginners. With that in mind, the tutorial was

helpful and could be implemented in our program to help with ease of use but it was frustrating how many features are blocked behind a paywall. In researching Songsterr, we found that it was much more easy to use and has an expansive library of songs due to the ability of the users to upload songs. Our biggest issue with Songsterr, however, like Flat.io, is how many features are locked behind a paywall as well as the lack of support for sheet music. In looking at GarageBand, we learned about the importance of having a wide variety of instrument sounds to play with, which was one of its strengths, and how important it is to make a simple interface that is welcoming to newcomers and will not overwhelm them, which is one of GarageBands biggest weaknesses. Additionally, we noticed the lack of resources for people who want to learn music, rather it is expected that you know how to play before picking up GarageBand. Overall, it could be useful to create a program that combines the sheet music capabilities of Flat.io, the song-learning elements of Songsterr, and the music-creation abilities of GarageBand.

10. References

- The websites of: Garageband, Songsterr, and Flat.io including all of their features and add-ons.

11. Appendices

Figure A - A middle school boy named Dave Later who wants to learn how to read sheet music as well as go to the next level with his piano skills

Figure B - A retired old man named Ebenezer Norris who wants to spend his time learning the piano at his own pace

Figure C - A young adult woman named Sophie Sheri who wants to learn how to play the guitar better but doesn't care too much for sheet music