

Software Project

**Ben Sharkey**

**N00212320**

Software Project

e.g. Develop an event ticketing application

Year 2 2022-23

DL836 BSc (Hons) in Creative Computing

Link to resources created as part of the project.

|  |  |
| --- | --- |
| GitHub | <https://github.com/y2-SW-project/swproject23-Poggs5401> |
| Video | Link to your video file (MS Stream, YouTube) |

Table of Contents

[1 Introduction 1](#_Toc96009471)

[2 Business Concept 2](#_Toc96009472)

[2.1 Business Idea 2](#_Toc96009473)

[2.2 Business model 2](#_Toc96009474)

[2.3 Market Research 2](#_Toc96009475)

[2.4 Marketing/Advertising 2](#_Toc96009476)

[2.5 Suppliers 2](#_Toc96009477)

[2.6 Competitors 2](#_Toc96009478)

[2.7 Employees 2](#_Toc96009479)

[2.8 Environmental Impact 2](#_Toc96009480)

[3 Requirements 3](#_Toc96009481)

[3.1 Introduction 3](#_Toc96009482)

[3.2 Requirements gathering 3](#_Toc96009483)

[3.2.1 Similar applications 3](#_Toc96009484)

[3.2.2 Interviews 3](#_Toc96009485)

[3.3 Requirements modelling 3](#_Toc96009486)

[3.3.1 Functional requirements 3](#_Toc96009487)

[3.3.2 Non-functional requirements 3](#_Toc96009488)

[3.3.3 Use Case Diagrams 4](#_Toc96009489)

[3.4 Feasibility 4](#_Toc96009490)

[4 Web application Design 6](#_Toc96009491)

[4.1 Layout 6](#_Toc96009492)

[4.2 Interaction 6](#_Toc96009493)

[4.3 Colour schemes 6](#_Toc96009494)

[4.4 Font choices 6](#_Toc96009495)

[4.5 Wireframes 6](#_Toc96009496)

[5 Database Design 7](#_Toc96009497)

[5.1 Description 7](#_Toc96009498)

[5.2 Business Reporting Requirements 7](#_Toc96009499)

[5.3 Textual Representation of Data-Set 7](#_Toc96009500)

[5.4 Business Rules 8](#_Toc96009501)

[5.5 Entity Relationship Diagram 8](#_Toc96009502)

[5.6 Tables 9](#_Toc96009503)

[5.7 Database Dictionary 10](#_Toc96009504)

[6 System Design/ Architecture Overview 11](#_Toc96009505)

[6.1 Introduction 11](#_Toc96009506)

[6.2 Model View Controller 11](#_Toc96009507)

[6.3 User Authenticaion 11](#_Toc96009508)

[6.4 Routing 11](#_Toc96009509)

[6.5 Templating 11](#_Toc96009510)

[7 Testing 12](#_Toc96009511)

[7.1 Introduction 12](#_Toc96009512)

[7.2 Functional Testing 12](#_Toc96009513)

[7.2.1 Login/Registration 13](#_Toc96009514)

[7.2.2 Navigation 13](#_Toc96009515)

[7.2.3 Calculation 13](#_Toc96009516)

[7.2.4 CRUD 13](#_Toc96009517)

[7.2.5 Discussion of Functional Testing Results 14](#_Toc96009518)

[7.3 User Testing 14](#_Toc96009519)

[7.4 Conclusion 14](#_Toc96009520)

[8 Project Management 15](#_Toc96009521)

[8.1 Introduction 15](#_Toc96009522)

[8.2 Project Phases 15](#_Toc96009523)

[8.2.1 Requirements 15](#_Toc96009524)

[8.2.2 Design 15](#_Toc96009525)

[8.2.3 Implementation 15](#_Toc96009526)

[8.2.4 Testing 15](#_Toc96009527)

[8.3 SCRUM Methodology 15](#_Toc96009528)

[8.4 Project Management Tools 16](#_Toc96009529)

[8.4.1 Github Project 16](#_Toc96009530)

[8.4.2 GitHub 16](#_Toc96009531)

[9 Reflection 17](#_Toc96009532)

[9.1 Your views on the project 17](#_Toc96009533)

[9.2 How could the project could be developed further? 17](#_Toc96009534)

[9.3 Assessment of your learning. 17](#_Toc96009535)

[9.4 Completing a large software development project 17](#_Toc96009536)

[9.5 Technical skills 17](#_Toc96009537)

[9.6 Further competencies and skills 17](#_Toc96009538)

[10 References 18](#_Toc96009539)

# Introduction

Application area

Technologies

PHP, MySQL, Bootstrap, CSS, Vanilla

Tools

IDE, phpMyAdmin, Miro

Project management

GitHub

Business Concept

Requirements

Design

Implementation

Testing

Reflection

# Business Concept

## Business Idea

For my Software Project I have decided to create a website based off music. On this website you will be able to read about artists and listen to music that the artist has created. My website would offer similar features to Spotify and Itunes such as, allowing you to listen to music by the artist and donate to have access to listen to a FLAC version of that song. This is a higher quality of music than would be available on Spotify.

## Business model

I plan to earn revenue from this site based off donations to support artists as well as ads. You are given the option to pay a small monthly subscription to not have to listen to an ad. However, if you have purchased a song you will not have to listen to ads for the duration of that song. I think I would make a profit off of this site because you do not need to pay to listen to music unless you are a fan of a particular song and want access to the FLAC version, which would attract the attention of consumers who are really into music.

## Market Research

People are constantly listening to music and the music industry is one of the largest industries in the world. By allowing users to freely use my website unlike Spotify or Itunes, I would be able to attract attention to my website. If a user is not forced to spend money, then they are more likely to spend it as they feel like it is their choice, and they don’t feel forced into anything.

I am hoping to attract specific types of users with my site. An example would be a user who does not have enough money to afford a monthly subscription or does not listen to music enough to be paying for a monthly subscription. This type of audience would gain the most out of using my website as they can use it freely.

## Marketing/Advertising

My main type of advertisement in the long run would be word for word as customers would tell their friends about my website and how it is free to use. However to get to this point I would first have to attract a large enough audience to the point where I could make a profit off doing this. I think setting up an ad on YouTube would be the most profitable advertisement I could do when starting off. YouTube is a free to use app which means all type of social classes use this app. This could attract the type of audience I am looking to start off with.

## Suppliers

There are plenty of Suppliers that I could contact when starting off such as, BMG Rights Management, Kobalt Music Group, Sony/ATV Music Publishing, Universal Music Group and Warner Music Group. All these distributors sell rights to songs. By purchasing songs in the most popular music category such as RAP/Hip Hop and Pop. I could gain a particular audience for my site and over time as my site starts to make profit, I could then spread out into different genres of music to attract different customers.

## Competitors

My main competitor would be Spotify. Spotify is a digital music service that gives you access to millions of songs all over the world. Basic functions such as playing music is free but you are restricted on the site unless you upgrade to a premium subscription which can charge up to $12 a month.

## Employees

If I decide to sell vinyl’s and merch from artists on my website I would also need a warehouse to store the goods. This would mean that I would have to hire warehouse staff to send out the purchased goods to the consumers. I would also need to hire a distributor so that I would gain the rights to sell the FLAC versions of songs.

## Environmental Impact

It is estimated by the Climate Group that digital applications could replace up to 25% of paper consumption. I will not be selling CD’s on my site and instead I’m allowing customers to listen to music digitally. By doing this it has a positive effect on the environment.

# Requirements

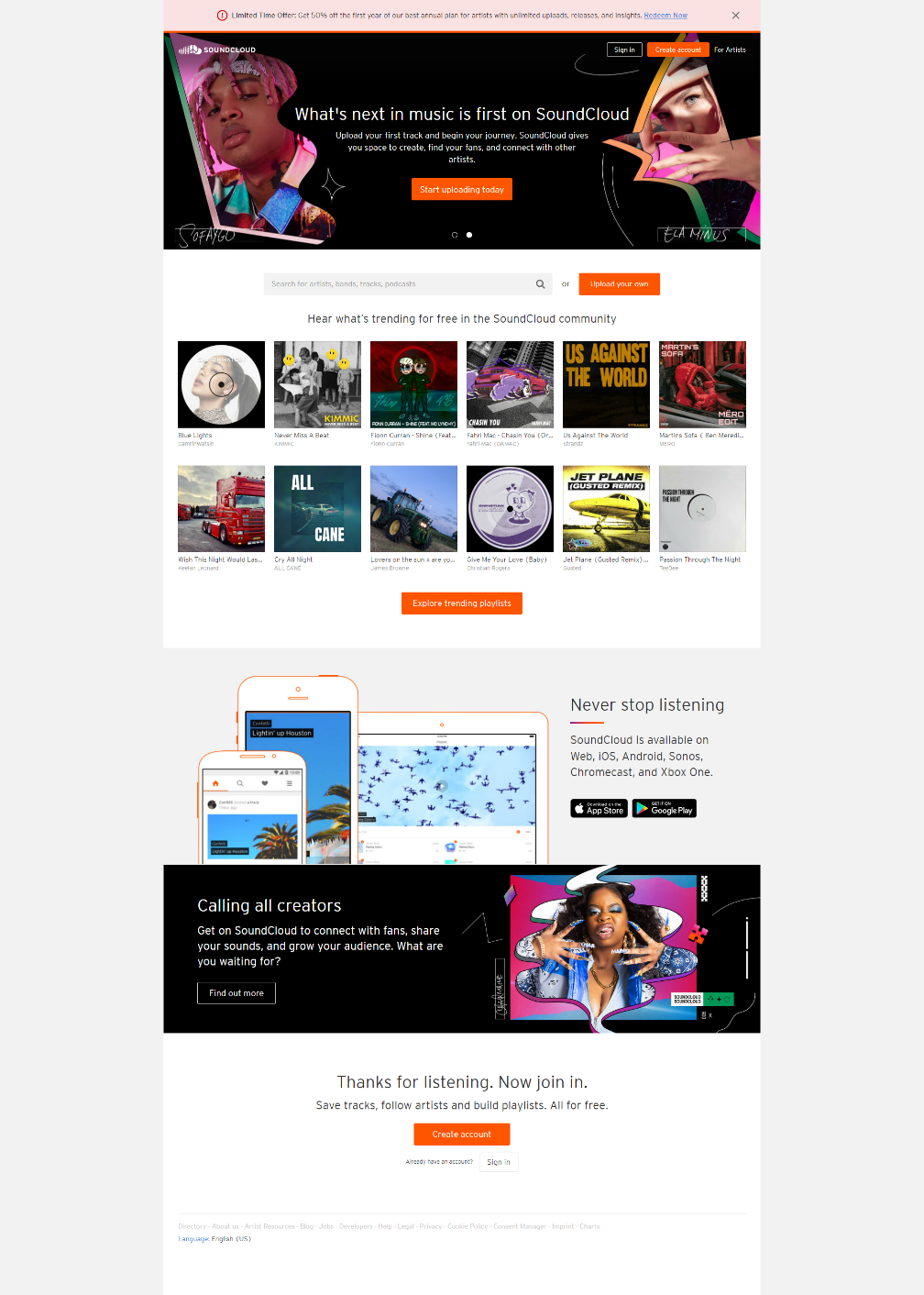
## Introduction

So far, I have only looked at this website with a developer’s perspective. If I were to look at the website I want to create through the eyes of a user what would I want to be on this site that would make me feel inclined to go back to it? As a user I would want a way to save my favorite music and for it to be easily accessible, I would also like a way to upload my own music onto the website if I ever made any as it could be a way for more people to discover my music. I would also want any subscriptions to the website to be cheaper then Spotify as that is my main competitor and Spotify have an app for your phone which allows you to listen to music anywhere.

## Requirements gathering

### Similar applications

**Image:**



**Description:**

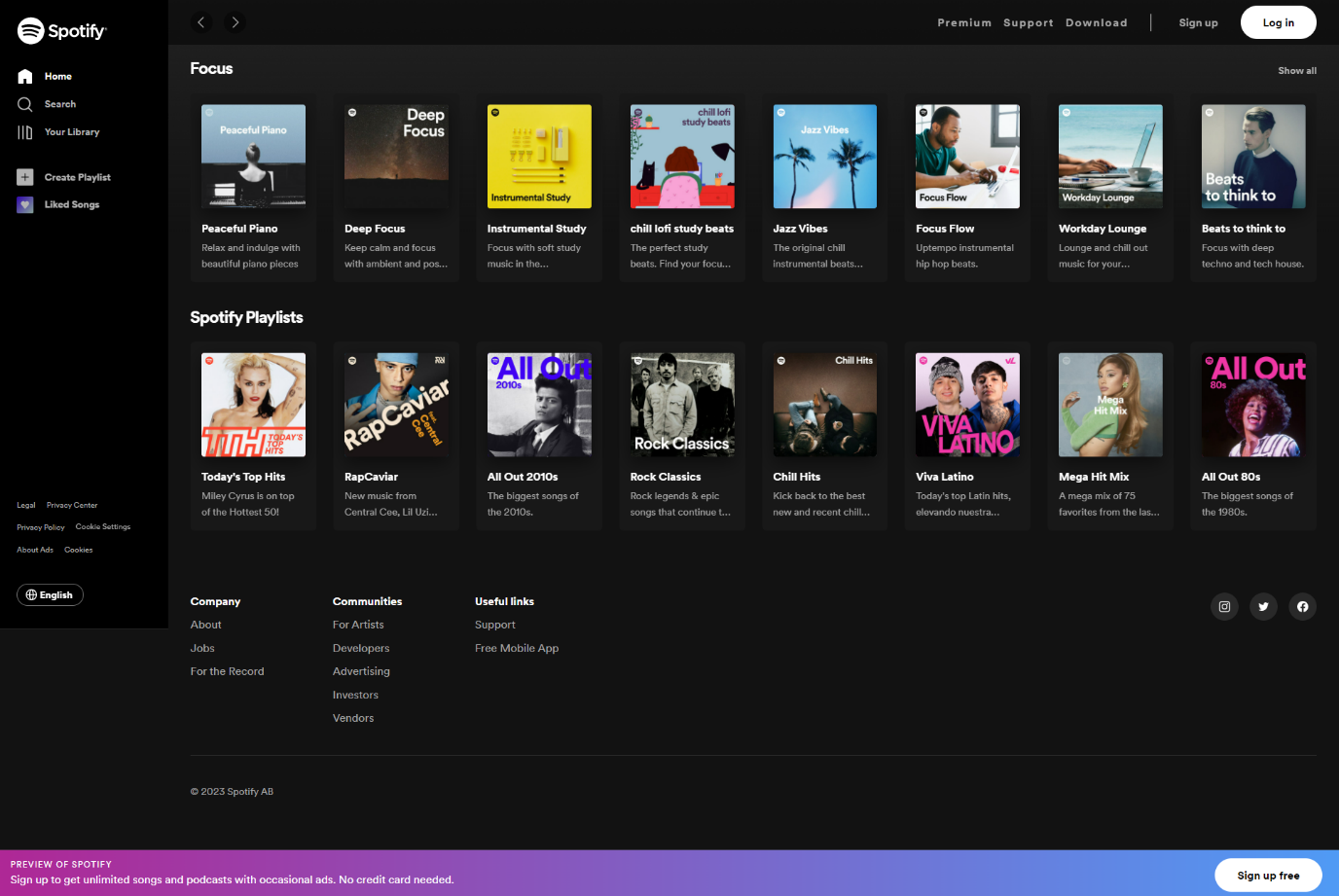
Soundcloud is a popular online platform that allows users to upload, share, and discover music and other audio content. With millions of tracks available, it’s a hub for aspiring musicians to showcase their work and connect with a global audience. Soundcloud’s unique social features and easy to use interface have made it a favorite among the music community, as it offers a low barrier entry for independent artists to distribute their music and get noticed. With its vast library of user generated content, Soundcloud has become a go-to destination for discovering new and diverse music that might not be available on traditional streaming platforms.

**Advantages:**

* Soundcloud provides a platform for independent artists to showcase and distribute their music to a global audience, offering a low-barrier entry for those who may not have the resources or connections to promote their work through traditional channels.
* With its vast library of user generated content, Soundcloud offers a unique opportunity for listeners to discover new and diverse music that may not be available on other streaming platforms.
* Soundcloud prioritizes community engagement, offering users the ability to interact with creators and comment on their work in real-time, providing a more social and interactive music listening experience.

**Disadvantages:**

* Due to the platform’s open and democratic nature, the sheer volume of content on Soundcloud can make it difficult for new or lesser-known artists to get noticed or receive meaningful feedback on their work.
* Soundcloud has faced criticism over its royalty payment structure, which some artists argue doesn’t adequately compensate them for their work. Additionally, Soundcloud has experienced financial difficulties in the past, leading to uncertainty about the platform’s long-term viability.
* As a user generated content platform, Soundcloud may be more susceptible to copyright infringement or other legal issues related to the unauthorized use of copyright material. This can pose a risk for both users and the platform itself.

**Image:**

**Description:**

Spotify is one of the most popular and widely-used music streaming services in the world. It offers a vast library of millions of songs and other audio content, accessible to user across multiple devices and platforms. One of Spotify’s key features is its ability to provide personalized recommendations and playlists based on a user’s listening habits and history, making it easy to discover new music that is likely to appeal to the user. Additionally, Spotify’s social feature allow users to follow friends and influencers, share music and playlists, and collaborate with others to create unique and collaborative listening experiences, With its extensive library of music, seamless user experience, and innovative features, Spotify has become a go to platform for music fans and casual listeners alike.

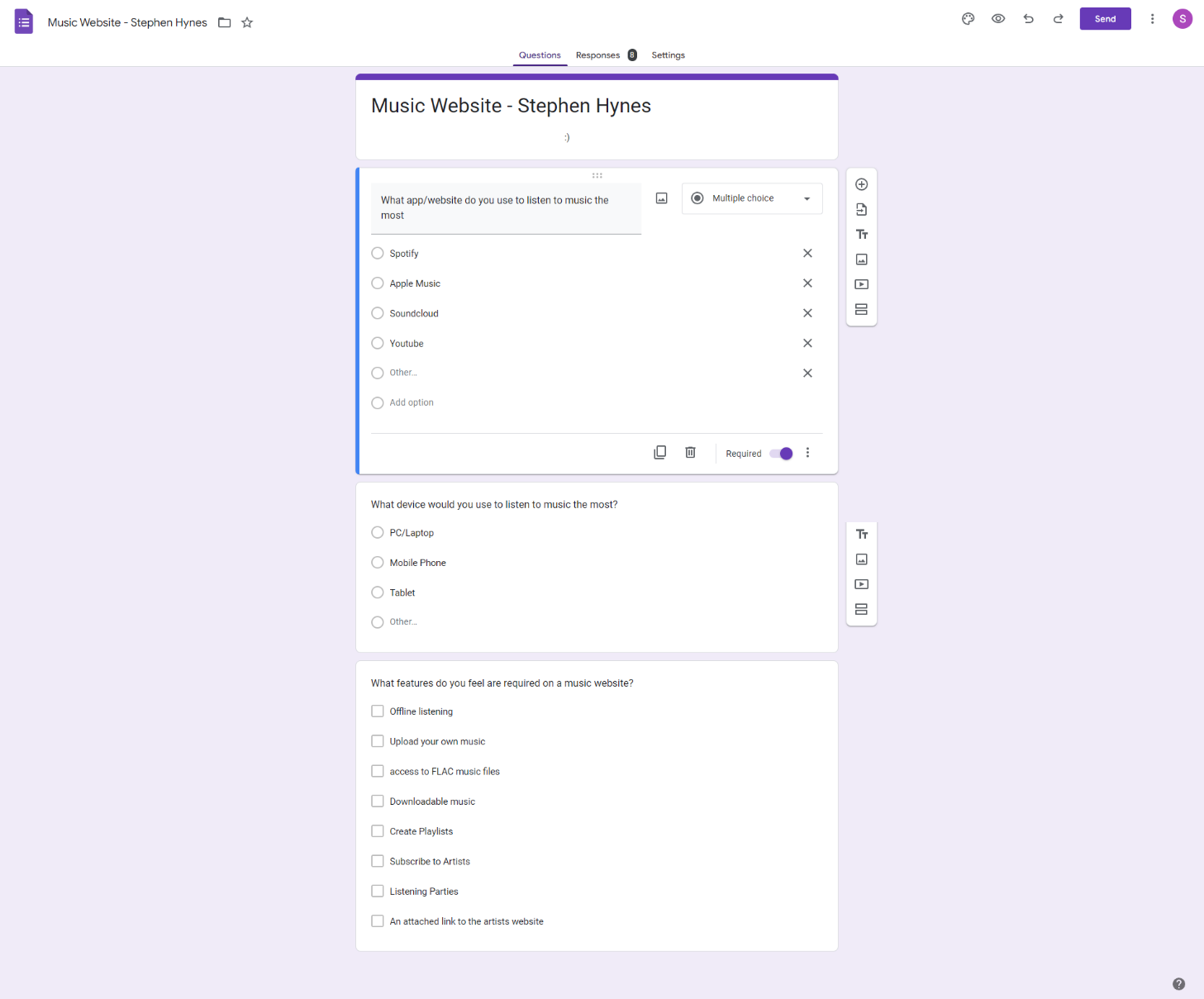
**Advantages:**

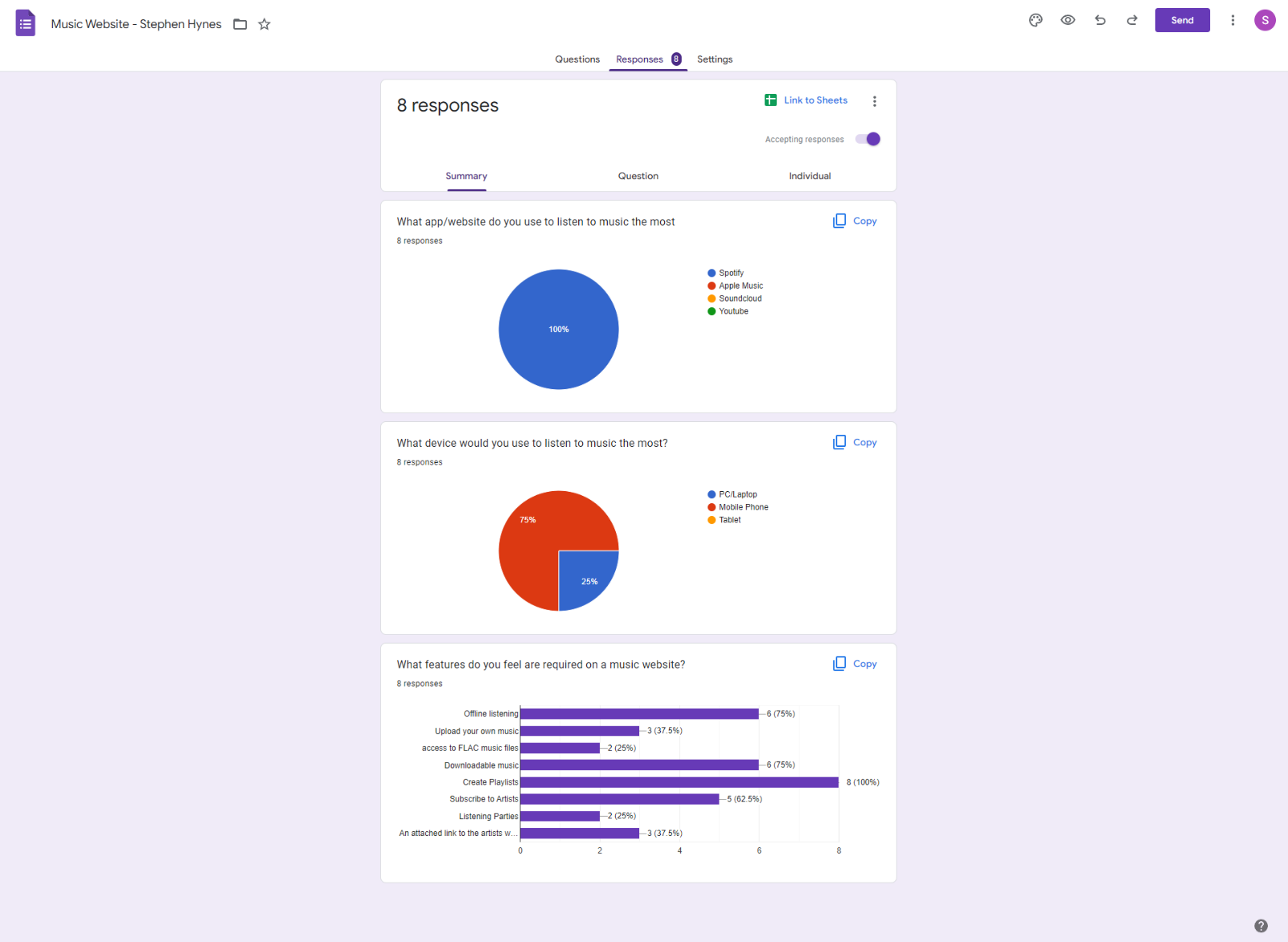
* Spotify’s extensive library of millions of songs and other audio content provides users with a seemingly endless array of music to discover and enjoy. This vast selection includes everything from classic albums and hits to emerging artists and new releases.
* Spotify’s personalized recommendations and playlists, based on a user’s listening history and preferences, help users discover new music that is tailored to their individual tastes. This makes it easier for users to discover new artists, genres, and styles of music that they might not have otherwise explored.
* With its social features, Spotify allows users to connect with friends and influences, share music and playlists, and collaborate with others to create unique and collaborative listening experiences. This social aspect of the platform makes it a great way to connect with like-minded music fans and discover new music through shared interests.

**Disadvantages:**

* One potential disadvantage of Spotify is that it operates on a subscription-based model, which means that users must pay a monthly fee in order to access its full range of features. This can be a barrier for some users, particularly those who are on a tight budget.
* While Spotify’s vast library of music is impressive, there are some notable gaps in its collection. This can be frustrating for user who are looking for specific songs or albums that are not available on the platform.
* Spotify has faced criticism from some artists and industry professionals who argue that its royalty payment structure does not adequately compensate artists for their work. This has led to tension and dispute between Spotify and some of the musicians and labels whose music is featured on the platform.

### Interviews

Instead of doing an interview I created an online survey using Google Forms as I found I could get a more accurate answer by getting 6-10 people to answer the same question rather than basing my conclusions on 2 people. I was able to get 8 people to answer my questions and the answers I received surprised me. Below you will find 2 images, one of the images shows the questions that I asked in my survey and the other shows the responses.



## Requirements modelling

### Functional requirements

By doing my survey I was able to get an understanding of what features are most essential when creating a music website and this is the conclusion, I have come up with starting with the most obvious.

1. Being able to listen to music.
2. Creating an account
3. Two Factor Authentication

### Non-functional requirements

These are requirements which if not met do not stop the application from working, but which mean that the application is not working as well as it should. They are usually based on issues such as:

1. Creating a playlist
2. Offline Listening
3. Downloadable music
4. Subscribing to an artist
5. Uploading music
6. A link to the artists website
7. Access to FLAC files
8. Listening parties

### Use Case Diagrams

Consists of actors and use cases. You should document each individual use case.

Delete the following diagram and insert your diagram. Use draw.io



## Feasibility

This section describes which technologies are planned to be used in the development of the application. It then explains if there are any issues in terms of the technical feasibility of the project, for example, if there are two different types of software which may have compatibility issues.

# Web application Design

## Layout

Describe the layout of your web application. Does this depend on a framework like bootstrap? Is it responsive?

## Interaction

What are the navigation elements, form elements. How does the user interact with the application?

## Colour schemes

Describe the colour palette that you will use consistently across the web application

## Font choices

Specify the fonts that you will use for different types of text. Include samples for paragraph text, headings and bold and italicised text.

## Wireframes

Describe how to navigate from one page to the next by adding a diagram of the different screens and what the main functionality is.



# Database Design

## Description

A company has a website that sells video games for different consoles. They would need a database for all their games and order places. For each order place, they would need games bought, total price, date of the order, and how long it will take to deliver. The database needs to keep track of all games that are being sold. Customers will have to input their information when registering an account. Customers will also have to input their card details when making a payment for their order.

## Business Reporting Requirements

Substitute in here the information the users of your application will want to be able to view.

1. Organisers need to be able to create, read, update, and delete: festivals, stages, shows, performers, and genres.
2. Users will need to be able to find all festivals ordered by their start date.
3. Users may want to find a festival by a specific start date.
4. Users need to find all festivals using a list of genres.
5. Users need to find the stage for a specific show.
6. Users need to find the shows using the performer's name.
7. Performers may need to find the list of festival contacts.
8. Users need to find festivals by location and the location needs to be displayed on a Google Map
9. Users may need to find festivals by city
10. Users need to find stages within a festival by the stage’s location
11. Organisers need to display a list of employees that are assigned to a specific festival

## Textual Representation of Dataset

Substitute in here the tables for your database

**FESTIVAL** (title, description, latitude, longitude, city, start\_date, end\_date, image\_id)

**PERFORMER** (title, description, contact\_email, contact\_phone, image\_id)

**GENRE** (title, description)

**IMAGE** (id, filename)

**SHOW** (date, start\_time, end\_time, performer\_id, stage\_id)

**STAGE** (title, description, location, festival\_id, image\_id)

**GENRE**\_**PERFORMER** (id, genre\_id, performer\_id)

**EMPLOYEE** (name, phone, email)

**FESTIVAL\_EMPLOYEE** (employee\_id, festival\_id, role)

## Business Rules

Substitute in here the business rules for your database

 A **Festival** has many **Stages**.

 A **Stage** belongs to one **Festival**.

 A **Stage** hosts many **Shows**.

 A **Show** is performed on one **Stage**.

 A **Performer** can perform in many **Shows**.

 A **Show** is performed by one **Performer**.

 A **Performer** can have many **Genres**.

 A **Genre** can belong to many **Performers**.

 A **Performer** can have a single **Image**.

 A **Festival** can have a single **Image**.

 A **Stage** can have a single **Image**.

 An **Image** can be associated with a **Performer**, **Festival**, or **Stage**

 A **Festival** can have many **Employees** associated with it

 An **Employee** can be assigned to one **Festival** at a time

## Entity Relationship Diagram

Substitute in here your ERD from draw.io



## Tables

Substitute in here your tables and the relationships between tables from draw.io in the format you used in DBMS with Mohammed.



## Database Dictionary

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Table | Attribute | Datatype | Range | Required | PK/FK | FK Ref Table |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

# System Design/ Architecture Overview

* 1. Introduction

This section will describe the internal functionality of the web framework that you have chosen for the implementation. Add further sections if required by the specification of your web application

* 1. Model View Controller

Explain the follows a model-view-controller design pattern and how it is implemented in your web application.

* 1. User Authentication

Explain how user authentication is implemented in the web application framework.

* 1. Routing

Describe the routes that were defined in the web application

* 1. Templating

Describe the templating engine and how it was used to configure/ style the web application.

Add a sequence diagram in this section and other diagrams that illustrate the architecture clearly.

Diagram

Description automatically generated

# Testing

* 1. Introduction

This chapter describes the testing that has been undertaken for the application. This chapter is presented in two sections:

1. Functional Testing
2. User Testing

Functional testing is a type of software testing whereby the system is tested against the functional requirements. The app is tested by looking to see if the actual output for a given input corresponds with the expected output. The tests should be based on the requirements for the app. The results of functional testing can indicate if a piece of software is functional and working, but not if the software is easy to use.

User testing looks to see if a piece of software is easy and intuitive for the user.

* 1. Functional Testing

This section describes the functional tests which were carried out on the app. These functional tests can be categorised as: (whatever is relevant to your app)

* Login/Registration
* Navigation
* Calculation
* CRUD

Functional testing generally uses a Black Box Testing technique which means that the internal logic of the system being tested is not of interest to the tester. The tester is only interested in whether the actual output agrees with the expected output.

* + 1. Login/Registration

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test No | Description of test case | Input | Expected Output | Actual Output | Comment |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

* + 1. Navigation

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test No | Description of test case | Input | Expected Output | Actual Output | Comment |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

* + 1. Calculation

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test No | Description of test case | Input | Expected Output | Actual Output | Comment |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

* + 1. CRUD

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test No | Description of test case | Input | Expected Output | Actual Output | Comment |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

### Discussion of Functional Testing Results

Describe the results from the tests. Address any functionality where unexpected behavior could not be debugged.

* 1. User Testing
  2. Conclusion

Discussion of test results.

# Project Management

## Introduction

This chapter describes how the project was managed. It shows the phases of the project, going from the project idea through the requirements gathering, the specification for the project, the design, implementation and testing phases for the project. It also discusses GitHub as a tool which assists in project management.

## Project Phases

In this section, describe each of the following project phases. Explain any issues which arose for each of the phases.

### Requirements

### Design

### Implementation

### Testing

Include a Gantt chart



## SCRUM Methodology (optional)

Sprints

## Project Management Tools

### GitHub Project

Description

Include screen shots

How it worked in practice

### GitHub

Description

How it is used

How it worked in practice

# Reflection

## Your views on the project

Describe how you feel the project went from your perspective.

## How could the project be developed further?

## Assessment of your learning.

Critically assess your learning. List what skills and competencies you have learned developed in this Continuous Assessment.

List which part of the project would need further development and itemize where you feel you have not satisfactorily completed the continuous assessment.

## Completing a large software development project

Describe what you have learnt from the project, from the point of view of completing a large software development project.

## Technical skills

Describe what you have learnt from the project, from a technical skills viewpoint.

## Further competencies and skills

Describe any extra competencies and skills that would help you with your development in the workplace.

# References

Add a list of references that you used to complete the project.

The Department of Technology and Psychology in IADT uses APA 7th referencing style.

Use alphabetical order for your references.

This site gives details about how to cite websites using APA:

https://www.wikihow.com/Cite-a-Website-in-APA

The following is a useful site for creating citations for APA for websites.

<http://www.citationmachine.net/apa/cite-a-website>

You can also use the Referencing tab within Microsoft Word to enter reference information manually. Word then creates an APA style reference.