**Daniel Morrissey**

**X21118701**

**HDWD\_SEPOL**

**Project Report**

**Contents**

1. Introduction…………………………………………………………………………………………..2
2. Purpose………………………………………………………………………………………………….3
3. Design and Planning………………………………………………………………………………4
4. Homepage
5. Band page
6. New bands page
7. Test
8. External gems
9. Constructed gem
10. Conclusion
11. Bibliography/references

**Introduction**

For this Cloud Application Development project, the task was to create a web application in which was to be deployed on a cloud-based platform. We are to pick a topic and then to choose a programming language and framework in order to create this web application that is to be deployed to a cloud-based platform.

Once the topic, the programming language and framework has been chosen, preparation is to be made in order to complete this project. The first step is to make at least one wireframe in order to design what the web application will look like. This can be done using online tools or applications like Figma as an example.

The next step is to design a database so as to have an optimal database with no redundant data. After the design of the database, the next step is to choose the language used to query the database via the use of object-oriented programming. Once that has been set up, choosing external libraries is to be next. To choose a library, a certain level of research is needed in order to use the external libraries that can be chosen. (Add something about creating a gem)

The type of tests that are to be used is also needed to be decided upon in order to ensure that the web application works as intended.

Once all these steps are completed the next step is to create the web application in question and to choose a suitable cloud platform such as Heroku or Amazon Web Services.

Graphical user interface, text, application

Description automatically generated

Screenshot of Heroku Website

**Purpose**

The purpose of this web application is to create a website that a person can go to and simply add a band or an artist to a database so that other people can look at bands and artists that other people have recommended to listen to. One advantage to this music recommendation site is that there is no need to register and only the band’s name or the artists name and the genre of music is needed.

This type of web application was chosen with the goal of creating a non-committal and simple website that someone can just enter a bands or artists details or check out the other bands and artists that people have recommended.

While it is understood that there are other websites that also recommend music such as Gnoosic (screenshot shown below), it felt like that there that some effort is involved in getting those music recommendations such as Gnoosic would ask a series of questions before you get a music recommendation of a band or an artist and it seems to be one at a time.

Graphical user interface, text, application

Description automatically generated

(Screenshot of Gnoosic homepage)

In order to create a music recommendation website, the Ruby programming language and the framework Ruby on Rails 7 were chosen to be the tools to create this web application.

**Design and Planning**

Before attempting to create this web application of a music recommendation site, a certain level of planning is needed. To plan for this web application, a use case diagram and entity relation diagram was created.

Diagram

Description automatically generated

(Use Case Diagram for web application, created using Visual Paradigm)

For the use case diagram above, a user (represented by Actor) would visit the site and then visit the Band list webpage and/or add a new band to the listing. Checking the list of bands and adding a new band are not always going to happen as the user may just visit the homepage.

In regard to the web application receiving and showing band and artists, the web application would need to interact with a database. Therefore, a database table needs to be designed to give the web application the ability to receive and show bands and artists.

To design a database table, the types of data required needs to be decided. In the end, it was decided to have five attributes and one of the attributes will be auto incremented. The attributes are: ID, Name, Country, Members and Genre. The ID and Members attributes require integer data while the rest are string data. The Name attribute accepts the name of the band or artist. The Country attribute accepts the country of origin of the band or artist. The Members attribute accepts the number of members in the band and the Genre attribute accepts the music genre.

When a user enters data, only the Name and Genre attributes are required, and the minimum number of members will be 1. The entity relationship diagram is on the next page.

Diagram

Description automatically generated

(Entity Relationship diagram of Band table)

After the database has been designed, the next step is to design a general layout of what the web application will look like. To design the general layout of the web application an application called Figma was used.

It was decided to create three wireframes to outline the general layout of each web page of the music recommendation web application to make a total of 3 web pages. The three web pages will be the homepage, the band and artist listings page and a web page that deals with the addition of new bands and artists.

The homepage is an introduction to the music recommendation web application that describes what the web application is and what it does. The band and artist listings page lists all the artists and bands that other users have recommended starting with the most recently added band or artist. The list of bands and artists show the name of the band or artist, the country of origin, the number of members and the music genre that the band or artist plays.