Software Requirements Specification

for

EarWorm

Version 1.0
21 October 2022

Team: Keyboard Buddies
Alvaro Lopez-Romero, Mitchell Mercer,
Nathalie Claire Tivar, Rodolfo Rivera, Charles Quebral

Supervisor: Edmund Dantes
Computer Science Department
California State University, Northridge

Revision History

Version	By	Change Description	Date
1.0	All team members	Original Document	10/21/2022
1.1			
1.2			
1.3			

Table of Contents

1. **INTRODUCTION**

1.1 Purpose

This senior capstone seeks to connect individuals and artists with like minded individuals based on criteria such as music taste,sex, age, and location.

1.2 Scope

The EarWorm Software Requirements Specification defines requirements for the EarWorm web application produced at CSUN by the group Keyboard Buddies as their senior capstone project.

EarWorm will be available to any individual, over the age of eighteen, to find and collaborate with individuals of similar music taste and garner semi-professional and personal relationships.

1.3 Product Value

EarWorm matches individuals based on their music tastes and thus will match like minded individuals. Users can use these matches to form romantic bonds and/or catapult these connections into personal and business opportunities.

1.4 Intended Audience

The intended audience of this product is any individual 18-65 years of age. These individuals should be looking to cultivate interpersonal relationships with the purpose of dating or furthering musical pursuits.

1.5 Intended Use

The goal is that these individuals can then proceed to make romantic connections or business connections to produce further pieces of art and/or music.

1.6 Definitions and Acronyms

- User - Someone who uses the web application.
- ML - Machine learning
- UI - User Interface

2. OVERALL DESCRIPTION

2.1 Product Perspective

Earworm can only be run in a web browser for now.

2.2 Product Functions

Take users' music taste and age input and match with potential partners. Search for potential partners in a given area. This web application will be easy to navigate and be in a nice simple layout for a user friendly experience.

2.3 User Characteristics

Users of this application will need to have access and know how to use their web browsers.

2.4 Constraints

- Will be written with Javascript, Java, Python, CSS, and HTML
- User must be using an up to date modern web browser
- Connection to internet is required

2.5 Assumptions and Dependencies

Assumes that all users have an up to date version of their web browser and a strong internet connection.

3. SYSTEM ARCHITECTURE

3.1 External Interface Requirements

Earworm relies on the use of a web browser. The app also interfaces with Spotify Analytics as well as other sources like current top charts, apple music, etc. Tools like web scraping and relevant API's are implemented.

3.2 Functional Requirements

Machine Learning Module:

Calculate Music Taste:

> The Machine Learning module will use a model trained by supervised learning to assign a user with a "music taste"
>
> Ex. User Will be assigned a backend value of "A1"

User Compatibility:

Logic must be implemented to filter users displayed by preference

> Ex. Only users 18-22 within a 10 miles radius will be displayed in the end product.

3.3 Performance Requirements

New User creation should take no longer than 10 minutes to input user preferences and link spotify accounts etc.

After setup, displaying other individuals of interest should take no longer than 60 seconds.

3.4 Logical Database Requirements

Maintain Unique Users:

Database must keep track of unique users:

A SQL database will be maintained and regularly updated that tracks unique users, and all relevant data pertaining to said users

Database must be able to be contacted by the Machine Learning backend.

3.5 Design Constraints

Definitively assigning music choice is impossible. All assignments are completely arbitrary and will have to be reassigned often, especially during early implementation stages.

Software

3.6 Software System Attributes

3.6.1 Security

Users should have unique Identification and passwords prerequisite to application use.

3.62 Capacity

This app requires no local storage capacity, but will require significant database storage.

3.6.3 Compatibility

This application requires the ability to run current web browsers.

3.6.4 Reliability

The app will not crash or hang except as the result of operating system error, the app will also be available 95% of the time between 5am - 12am.

3.6.5 Scalability

In early implementation the system will rely on a single SQL database with a single server backend, however in a full production environment container systems like kubernetes would enable high volume usage.

3.6.6 Usability

With zero technical experience users should be able to setup an account and browse the online application with ease.

3.6.7 Other

Not applicable on this version of the document.