Software Requirements Specification for Software Engineering: subtitle describing software

Team 8 – Rhythm Rangers

Ansel Chen
Muhammad Jawad
Mohamad-Hassan Bahsoun
Matthew Baleanu
Ahmed Al-Hayali

October 11, 2024

Contents

1 Purpose of the Project **vi**
- 1.1 User Business . . . vi
- 1.2 Goals of the Project . . . vi

2 Stakeholders **vi**
- 2.1 Client . . . vi
- 2.2 Customer . . . vi
- 2.3 Other Stakeholders . . . vi
- 2.4 Hands-On Users of the Project . . . vi
- 2.5 Personas . . . vi
- 2.6 Priorities Assigned to Users . . . vi
- 2.7 User Participation . . . vii
- 2.8 Maintenance Users and Service Technicians . . . vii

3 Mandated Constraints **vii**
- 3.1 Solution Constraints . . . vii
- 3.2 Implementation Environment of the Current System . . . vii
- 3.3 Partner or Collaborative Applications . . . vii
- 3.4 Off-the-Shelf Software . . . vii
- 3.5 Anticipated Workplace Environment . . . viii
- 3.6 Schedule Constraints . . . viii
- 3.7 Budget Constraints . . . viii
- 3.8 Enterprise Constraints . . . viii

4 Naming Conventions and Terminology **viii**
- 4.1 Glossary of All Terms, Including Acronyms, Used by Stakeholders involved in the Project . . . viii

5 Relevant Facts And Assumptions **ix**
- 5.1 Relevant Facts . . . ix
- 5.2 Business Rules . . . ix
- 5.3 Assumptions . . . x

6 The Scope of the Work **x**
- 6.1 The Current Situation . . . x
- 6.2 The Context of the Work . . . x
- 6.3 Work Partitioning . . . x

6.4 Specifying a Business Use Case (BUC) xi

7 Business Data Model and Data Dictionary xi
7.1 Business Data Model xi
7.2 Data Dictionary xi

8 The Scope of the Product xi
8.1 Product Boundary xi
8.2 Product Use Case Table xi
8.3 Individual Product Use Cases (PUC's) xi

9 Functional Requirements xi
9.1 Functional Requirements xi

10 Look and Feel Requirements xii
10.1 Appearance Requirements xii
10.2 Style Requirements xii

11 Usability and Humanity Requirements xii
11.1 Ease of Use Requirements xii
11.2 Personalization and Internationalization Requirements xii
11.3 Learning Requirements xii
11.4 Understandability and Politeness Requirements xii
11.5 Accessibility Requirements xii

12 Performance Requirements xii
12.1 Speed and Latency Requirements xii
12.2 Safety-Critical Requirements xiii
12.3 Precision or Accuracy Requirements xiii
12.4 Robustness or Fault-Tolerance Requirements xiii
12.5 Capacity Requirements xiii
12.6 Scalability or Extensibility Requirements xiii
12.7 Longevity Requirements xiii

13 Operational and Environmental Requirements xiii
13.1 Expected Physical Environment xiii
13.2 Wider Environment Requirements xiii
13.3 Requirements for Interfacing with Adjacent Systems xiv
13.4 Productization Requirements xiv

13.5 Release Requirements xiv

14 Maintainability and Support Requirements **xiv**
14.1 Maintenance Requirements xiv
14.2 Supportability Requirements xiv
14.3 Adaptability Requirements xiv

15 Security Requirements **xiv**
15.1 Access Requirements xiv
15.2 Integrity Requirements xiv
15.3 Privacy Requirements xv
15.4 Audit Requirements xv
15.5 Immunity Requirements xv

16 Cultural Requirements **xv**
16.1 Cultural Requirements xv

17 Compliance Requirements **xv**
17.1 Legal Requirements xv
17.2 Standards Compliance Requirements xv

18 Open Issues **xv**

19 Off-the-Shelf Solutions **xv**
19.1 Ready-Made Products xv
19.2 Reusable Components xvi
19.3 Products That Can Be Copied xvi

20 New Problems **xvi**
20.1 Effects on the Current Environment xvi
20.2 Effects on the Installed Systems xvi
20.3 Potential User Problems xvi
20.4 Limitations in the Anticipated Implementation Environment That May Inhibit the New Product xvi
20.5 Follow-Up Problems xvi

21 Tasks **xvi**
21.1 Project Planning xvi
21.2 Planning of the Development Phases xvii

22 Migration to the New Product **xvii**
22.1 Requirements for Migration to the New Product xvii
22.2 Data That Has to be Modified or Translated for the New System xvii

23 Costs **xvii**

24 User Documentation and Training **xvii**
24.1 User Documentation Requirements xvii
24.2 Training Requirements xvii

25 Waiting Room **xvii**

26 Ideas for Solution **xviii**

Revision History

Date	Version	Notes
Date 1	1.0	Notes
Date 2	1.1	Notes

1 Purpose of the Project

1.1 User Business

Insert your content here.

1.2 Goals of the Project

Insert your content here.

2 Stakeholders

2.1 Client

Insert your content here.

2.2 Customer

Insert your content here.

2.3 Other Stakeholders

Insert your content here.

2.4 Hands-On Users of the Project

Insert your content here.

2.5 Personas

Insert your content here.

2.6 Priorities Assigned to Users

Insert your content here.

2.7 User Participation

Insert your content here.

2.8 Maintenance Users and Service Technicians

Insert your content here.

3 Mandated Constraints

3.1 Solution Constraints

Insert your content here.

3.2 Implementation Environment of the Current System

Insert your content here.

3.3 Partner or Collaborative Applications

Insert your content here.

3.4 Off-the-Shelf Software

There are several existing solutions that could serve as part of the music generation and recommendation system. These include:

- **Spotify API**: Provides access to a vast library of music, including song previews and metadata, which can be leveraged for generating recommendations.
- **Librosa Library**: An open-source Python package for analyzing and processing music files, suitable for extracting features from songs and facilitating generative components.
- **TensorFlow and PyTorch Pre-trained Models**: Both frameworks offer pre-trained models that could be adapted for music generation

tasks. These solutions provide a basis for deep learning models without having to build and train from scratch.

- **OpenAI Jukebox**: A generative model that is capable of producing music, which could potentially be adapted and integrated into our system.

These off-the-shelf software solutions provide a foundation upon which we can build our custom features, significantly reducing the development time and leveraging existing technologies to enhance the functionality of our platform.

Insert your content here.

3.5 Anticipated Workplace Environment

Insert your content here.

3.6 Schedule Constraints

Insert your content here.

3.7 Budget Constraints

Insert your content here.

3.8 Enterprise Constraints

Insert your content here.

4 Naming Conventions and Terminology

4.1 Glossary of All Terms, Including Acronyms, Used by Stakeholders involved in the Project

Insert your content here.

5 Relevant Facts And Assumptions

5.1 Relevant Facts

- Music contains the following core features
  - Tempo
  - Key signature
  - Time signature
  - Pitch
  - Timbre
- Song files have metadata that contains information such as:
  - Song title
  - Artist
  - Release date
- Most songs can be classified into a particular genre

5.2 Business Rules

- The user should be able to generate their own music
- The user should be able to figure out what musical features a song contains
- The user should be able to ask for similar songs
- the user shouldn't have to download an application to interact with the system

5.3 Assumptions

- Users will have at least some familiarity of music theory
- The analysis and recommendation systems will use as many well-established musical features as possible

- All API inputs will be easily accessible and reliable enough to support the recommendation and analysis systems
- The system will be written in a language that all developers are familiar with
- The system will use a local server to handle the processing of the machine learning model and large datasets
- Handling of niche features and cover art are designed to enhance the user experience, but these will not be a part of the core functionality of the system
- The generative system will be completed by the POC demo date
- The recommendation and analysis systems will be completed by the Revision 0 date

6 The Scope of the Work

6.1 The Current Situation

Insert your content here.

6.2 The Context of the Work

Insert your content here.

6.3 Work Partitioning

Insert your content here.

6.4 Specifying a Business Use Case (BUC)

Insert your content here.

7 Business Data Model and Data Dictionary

7.1 Business Data Model

Insert your content here.

7.2 Data Dictionary

Insert your content here.

8 The Scope of the Product

8.1 Product Boundary

Insert your content here.

8.2 Product Use Case Table

Insert your content here.

8.3 Individual Product Use Cases (PUC's)

Insert your content here.

9 Functional Requirements

9.1 Functional Requirements

Insert your content here.

10 Look and Feel Requirements

10.1 Appearance Requirements

Insert your content here.

10.2 Style Requirements

Insert your content here.

11 Usability and Humanity Requirements

11.1 Ease of Use Requirements

Insert your content here.

11.2 Personalization and Internationalization Requirements

Insert your content here.

11.3 Learning Requirements

Insert your content here.

11.4 Understandability and Politeness Requirements

Insert your content here.

11.5 Accessibility Requirements

Insert your content here.

12 Performance Requirements

12.1 Speed and Latency Requirements

Insert your content here.

12.2 Safety-Critical Requirements

Insert your content here.

12.3 Precision or Accuracy Requirements

Insert your content here.

12.4 Robustness or Fault-Tolerance Requirements

Insert your content here.

12.5 Capacity Requirements

Insert your content here.

12.6 Scalability or Extensibility Requirements

Insert your content here.

12.7 Longevity Requirements

Insert your content here.

13 Operational and Environmental Requirements

13.1 Expected Physical Environment

Insert your content here.

13.2 Wider Environment Requirements

Insert your content here.

13.3 Requirements for Interfacing with Adjacent Systems

Insert your content here.

13.4 Productization Requirements

Insert your content here.

13.5 Release Requirements

Insert your content here.

14 Maintainability and Support Requirements

14.1 Maintenance Requirements

Insert your content here.

14.2 Supportability Requirements

Insert your content here.

14.3 Adaptability Requirements

Insert your content here.

15 Security Requirements

15.1 Access Requirements

Insert your content here.

15.2 Integrity Requirements

Insert your content here.

15.3 Privacy Requirements

Insert your content here.

15.4 Audit Requirements

Insert your content here.

15.5 Immunity Requirements

Insert your content here.

16 Cultural Requirements

16.1 Cultural Requirements

Insert your content here.

17 Compliance Requirements

17.1 Legal Requirements

Insert your content here.

17.2 Standards Compliance Requirements

Insert your content here.

18 Open Issues

Insert your content here.

19 Off-the-Shelf Solutions

19.1 Ready-Made Products

Insert your content here.

19.2 Reusable Components

Insert your content here.

19.3 Products That Can Be Copied

Insert your content here.

20 New Problems

20.1 Effects on the Current Environment

Insert your content here.

20.2 Effects on the Installed Systems

Insert your content here.

20.3 Potential User Problems

Insert your content here.

20.4 Limitations in the Anticipated Implementation Environment That May Inhibit the New Product

Insert your content here.

20.5 Follow-Up Problems

Insert your content here.

21 Tasks

21.1 Project Planning

Insert your content here.

21.2 Planning of the Development Phases

Insert your content here.

22 Migration to the New Product

22.1 Requirements for Migration to the New Product

There are no migration requirements as this project is not a replacement or upgrade of a previous project

22.2 Data That Has to be Modified or Translated for the New System

Similarly, there currently is no data that needs to be modified

23 Costs

Insert your content here.

24 User Documentation and Training

24.1 User Documentation Requirements

Insert your content here.

24.2 Training Requirements

Insert your content here.

25 Waiting Room

Insert your content here.

26 Ideas for Solution

- **Hybrid Recommendation System**: A hybrid recommendation system combines content-based filtering and collaborative filtering techniques to provide a more personalized experience for users. Content-based filtering analyzes song features, such as genre, key, and rhythm,

to suggest similar tracks. Collaborative filtering uses user preferences and historical listening patterns to suggest music. By combining these approaches, the system can offer users personalized suggestions while also helping them discover new genres and music styles.

- **Generative Music Model**: To enable the creation of new music, a generative model will be used. This model could be based on techniques such as a Generative Adversarial Network (GAN) or Recurrent Neural Network (RNN). A GAN would allow for the generation of realistic music by having the generator and discriminator work together to produce convincing compositions. An RNN, on the other hand, would be well-suited for learning the sequential nature of music, generating new melodies based on learned patterns. This solution provides users with an innovative way to create new music based on their inputs and preferences.

- **Feature Manipulation Interface**: This interface will allow users to interact directly with song features, such as tempo, key, and rhythm, enabling them to create customized versions of existing tracks or generate entirely new compositions. By adjusting different musical parameters, users can personalize their musical experience and experiment with creative variations, providing a high level of control over the output.

- **Integration with Existing Platforms**: Integrating the system with existing music platforms, such as Spotify, will allow users to easily access and analyze a large library of songs. Users will be able to input their favorite tracks from these platforms and generate variations or receive recommendations. This integration ensures a smooth user experience, allowing seamless interaction between existing music libraries and the platform's generative capabilities.

Appendix — Reflection

The information in this section will be used to evaluate the team members on the graduate attribute of Lifelong Learning. Please answer the following questions:

1. What knowledge and skills will the team collectively need to acquire to successfully complete this capstone project? Examples of possible knowledge to acquire include domain specific knowledge from the domain of your application, or software engineering knowledge, mechatronics knowledge or computer science knowledge. Skills may be related to technology, or writing, or presentation, or team management, etc. You should look to identify at least one item for each team member.

2. For each of the knowledge areas and skills identified in the previous question, what are at least two approaches to acquiring the knowledge or mastering the skill? Of the identified approaches, which will each team member pursue, and why did they make this choice?