Altering Heartbeat Using AI Selected Music Software Requirements Specification

Team Members:

May Wandyez, Jeremy John, Ahmad Shah

Tables of Contents:

1. Introduction

1.1 Purpose

This document is intended for all project members involved in the supervision and creation of the Altering Heartbeat Using AI Select Music Project. The purpose of this document is to provide a description of all requirements necessary for the completion of the Desktop Application for altering user heart rate using AI selected music. This document will describe all necessary features required to be implemented as part of the desktop application. This document will also describe all constraints on the desktop application's design and user interfaces. Outlining these requirements will allow the team members of the project to understand what the project must consist of to be considered finished.

1.2 Scope

The desktop application consists of a front end for receiving user interaction, and a backend for giving functionality to user interaction.

The front end of the application will allow the user to add new music to the music folder, select the 'target' heart rate the user wishes to achieve, and automatically select play music from the music folder to achieve the user's desired heart rate when the user clicks on the 'start' button. The front end will also include a brief guide on how to use the front end, as well as an advanced features setting which will allow the user to select 'approach paths' to the user's desired heart rate.

The backend will automatically scan the application's music folder for music, gather characteristics of the music placed inside of the music folder, save these music characteristics to a local comma separated value file, connect to the user's ANT+ enabled heart rate monitoring device, receive the user's heartbeat from a ANT+ enabled heart rate device, calculate the user's resting heart rate based on received heart rate data, generate predictions for how the user's heartbeat will change when listening to a selected piece of music using a machine learning model, and select a piece of music that will most closely get the user to their 'target' heart rate.

1.3 Definitions

Term	Definition
ANT+	ANT+ is a wireless communication protocol

	that is used in heart rate monitors manufactured by Garmin.
AI	AI is a catch-all term for 'artificial intelligence'. In the context of the desktop application it refers to a mathematical model that generates predictions of what a user's heart rate will be should they listen to a particular music file.
Heart Rate	Heart rate is the number of times the human heart performs a pumping cycle per minute.
Resting Heart Rate	Resting Heart Rate is the number of times the human heart beats per minute while at rest. Resting heart rate is unique to each individual. An individual's heart rate tends to converge towards the individual's resting heart rate

1.4 References

LeWine, Howard. "What Your Heart Rate Is Telling You." *Harvard Health*, Harvard Medical School, 13 June 2023, www.health.harvard.edu/heart-health/what-your-heart-rate-is-telling-you.

Barvanovsky, Ben. "CSV Files: Use Cases, Benefits, and Limitations." *OneSchema CSV Importer*, oneschema, 3 May 2023, www.oneschema.co/blog/csv-files

Author Unknown. "Size of the Average Music Library: 7,160 Songs." *Hypebot*, 5 Jan. 2011, www.hypebot.com/hypebot/2011/01/size-of-the-average-music-library-7160-songs.html.

1.5 Overview

This document defines the software requirements for the planned desktop application that will allow users to alter their heart rate towards a target using AI selected music. This document will start by providing the general factors that describe the planned desktop application; this will include the product perspective, the product functions, user characteristics, and general constraints. This document will then describe the requirements for the user interface, the requirements for achieving functionality, and the requirements that are not necessary for functionality. This will be followed by a section describing the design constraints that impose limitations on the project. This document will then give a description of the logical database requirements for this project, followed by a data flow diagram that provides analysis of how the desktop application will function as a whole.

2. General Factors

2.1 Product Perspective

The desktop application for altering heartbeat using AI selected music is a cutting-edge product intended for use on Windows 11 desktop computers. This product is the first known product of its kind to attempt to predict a user's heart rate before the user listens to a piece of music. This product achieves this through a machine learning algorithm trained on data consisting of music file characteristics and heart rate data. The scope of the project encompasses a desktop application capable of receiving heart rate monitor information from ANT+ enabled heart monitoring devices.

2.2 Product Functions

Function:	Description:
ANT+ Enabled Heart Rate Monitor Connectivity	The desktop application will automatically establish a connection to receive heart rate data from ANT+ enabled heart monitor devices.
Live Heart Rate Feed	The desktop application will establish a live feed of the user's heart rate.
Resting Heart Rate Calculation	The desktop application will calculate and store the user's resting heart rate.
Target Heart Rate Selection	The user will be able to select their desired heart rate they wish to achieve.
Music File Transfer	The desktop application will move files to its music folder if the user wishes for the application to consider them for song selection.
Music File Characteristic Calculation	The desktop application will calculate music characteristics for any given music file, including tempo, pitch, and length.
AI Song Selection Based On Target Heart Rate	A machine learning model will generate predictions for every music file the user stores in the application based on the characteristics of each music file (such as the tempo, pitch, and length of the song), the user's current heart rate, and the user's resting heart rate. An

	appropriate music file will be selected based on the user's target heart rate.
Approach Path Selection	The user will be able to alter the song selection process by choosing an 'approach path' which determines how the next song will be chosen. For instance, the user can attempt to change their heart rate by a fixed amount per minute, or by the greatest amount per minute possible.

2.3 User Characteristics

The main types of users who are expected to use the desktop application can be broken down into the following groups: those with heart conditions attempting to better regulate their health, music enthusiasts who desire a better listening experience, and fitness enthusiasts who would be using the application to motivate themselves during exercise. All types of users will only be using the desktop application, however they wish to use it for distinct purposes.

User Type:	User Goals:	User Concerns:
Cardiac-Conscious	Regulate erratic heart rhythms by using the application as an advanced music therapy tool.	Worried about their heart beating too fast or too slowly. Does not want heart rate to fluctuate rapidly.
Music-Enthusiast	Desires a better audio listening experience.	Does not want music that does not fit their current mood, or mood they are trying to achieve.
Fitness-Enthusiast	Wants to raise heart rate during exercise.	Heart rate should not be too low or unsteady.

2.4 General Constraints

The desktop application is only functional should the ANT+ enabled radio communications receiver be in range of the ANT+ enabled heart rate monitoring device. This range is limited to around 16ft. Similarly, the live heart rate feed is limited to the rate at which the ANT+ enabled heart monitoring device broadcasts the heart rate, which is usually around once per second.

The desktop application is also constrained to user provided music - for legal reasons it is not advisable to distribute music that remains copyrighted.

2.5 Assumptions and Dependencies

It is assumed that the desktop application will be functional on any Windows 11. This assumption is reasonable as the processing power necessary for the minimum requirements of Windows 11 should be enough to run a linear regression model.

---

3. Requirements

3.1 External Interface Requirements

3.1.1 User Interface

The user interface will consist of a home page, a user guide page, and an advanced features page. A navigation bar will be present on all pages to allow the user to navigate between them.

3.1.1.1 Home Page

A user of the desktop application should be greeted with a homepage. At the top of the homepage will be a navigation bar. The first option on the homepage will be to allow the user to select their target heart rate. The user will only be able to select heart rates within the safe range of 40-150[1] beats per minute.

In the middle of the page will be a button labeled 'Measure Resting Heart Rate'. Upon pressing the measure resting heart rate button, the user will be greeted with a popup and prompted to take a deep breath and try to relax. In the upper right corner of the popup will be a button for the user to manually input their resting heart rate instead of taking a direct measurement. If the user has already had their resting heart rate measured, the button will be filled automatically with the user's resting heart rate.

At the bottom of the page will be a button labeled 'Start', which when pressed will automatically select and play a song within an audio widget for the user that will get the user to their target

---

[1] 40 is the resting heart rate of athletes in top condition, 150 is the maximum safe heart rate for those over the age of 70. LeWine, Howard. "What Your Heart Rate Is Telling You." *Harvard Health*, Harvard Medical School, 13 June 2023, www.health.harvard.edu/heart-health/what-your-heart-rate-is-telling-you.

heart rate. The user will be able to interact with the audio widget to pause and resume the song. The user will also be able to skip the song, and a new recommended song will be selected.

3.1.1.2 Navigation Bar

Contained within the navigation bar will be a button labeled 'home', which when clicked will take the user back to the home page. The navigation bar will appear on all pages of the front-end. Contained within the navigation bar will be a question mark, which when clicked will take the user to the User Guide page. Contained within the navigation bar will be a small gear icon, clicking on the small gear icon will lead the user to the 'advanced features' page.

3.1.1.3 User Guide Page

The user guide will consist of a page with a text description of how to navigate and use the user interface, as well as a video demonstration of the application in use to guide the user.

3.1.1.4 Advanced Features Page

The advanced feature page will contain advanced features for the user once they are more familiar with the application. One such advanced feature will be the 'approach path' selection. The approach path selection will allow the user to determine how the music selection algorithm will choose the next song. How this choice is made is based on how quickly the user wants their heart rate to change. The available choices to the user will be:"Shallow", "Linear", "Steep", "Fastest" , "Parabola", and "Rollercoaster". Shallow, linear, and steep refer to the rate to which the heart rate will change, while parabola refers to overshooting the target heart rate then bringing the user back in the direction of their target. Rollercoaster refers to the song selection alternating towards increasing and decreasing heart rate in the direction of the target heart rate.

3.1.2 Hardware Interfaces

The desktop application will have a designated hardware interface with an ANT+ USB Stick. The ANT+ USB stick provides the means for receiving ANT+ wireless communications signals from ANT+ enabled heart rate monitor devices. The hardware interface will be managed by the openANT+ 1.3.1 python library.

3.1.3 Software Interfaces

The desktop application is a self-contained entity that does not communicate with any external software.

3.1.4 Communications Interfaces

The only communication that occurs that is required for the operation of the desktop application is between the ANT+ USB receiver hardware and the ANT+ enabled heart rate monitoring device. Communication between the receiver and the heart rate monitoring device is managed through ANT+ wireless communication protocol.

3.2 Functional Requirements

ID	TITLE:	PRIORITY:	DESCRIPTION, INPUTS, PROCESSING, OUTPUTS, ERROR HANDLING,
FR 3.2.1	Retrieve Heart Rate from ANT+ wireless connection	HIGH	**Description:** In order for the desktop application to be able to predict change in heart rate, the desktop application must know the user's current heart rate. The current heart rate must be retrieved from a ANT+ compatible heart rate monitor and received through a ANT+ USB stick. Current heart rate does not need to be displayed in the front end.
			Inputs: Wireless ANT+ data stream.
			Processing: ANT+ data stream is converted into heart rate.
			Outputs: Continuous data stream of current heart rate.
			Error Handling: In the event of no connection being established, an attempt to connect should be repeated for a one minute period. If no connection is established within one minute, an error handling message should appear alerting the user that there has been a failure to connect to a ANT+ wireless heart rate monitor.
			Forward Dependencies: FR 3.2.6, FR 3.2.2.
			Backward Dependencies: None.

FR 3.2.2	Establish Resting Heart Rate	MEDIUM	**Description:** In order for the desktop application to be able to predict change in heart rate, the desktop application must know the user's resting heart rate. The user should be instructed to stay still and attempt to relax. During this period, the user's resting heart rate will be determined when the user's heart rate does not vary by more than 3 beats per minute during a 20 second time period. This heart rate will then be saved locally for reuse. The user should also have an option to manually input their resting heart rate.
			Inputs: Current heart rate.
			Processing: Current heart rate will be repeatedly sampled into a fixed length queue representing a span of 20 seconds. When the range in values does not exceed 3, then the average value of the queue will be returned
			Outputs: Current resting heart rate.
			Error Handling: In the event of no resting heart rate being established within a one minute time period, an error message will appear saying that there is too much variation within the user's heart rate, that the user should try to move as little as possible, and they should try again. Alternatively the user will be presented with the option to input their resting heart rate manually.
			Forward Dependencies: FR 3.2.11, FR 3.2.6
			Backward Dependencies: FR 3.2.1
FR 3.2.3	User Selects Target Heart Rate	HIGH	**Description:** In order for the desktop application to be able to predict change in heart rate, the desktop application must know the user's desired heart rate. The user should be prompted to click a button to select their target heart rate from a range of 40-150 beats per minute.
			Inputs: Desired heart rate.

				Processing: None
				Outputs: None.
				Error Handling: If the user attempts to select a heart rate outside of the established bounds the target heart rate will be set to the edge of the bounds and an error message will appear telling the user that the selected heart rate is unsafe
				Forward Dependencies: FR 3.2.11, FR 3.2.6
				Backward Dependencies: None.
FR 3.2.4	Get Character istics Of Music In Desktop Applicati on Music Folder	HIGH		Description: In order for the desktop application to be able to predict change in heart rate, the desktop application must know the tempo, pitch, and length of every song within the desktop application's music directory. This information for each song will be saved within each row in a local CSV file titled "music_characteristics.csv"
				Inputs: File path of desktop application's music directory.
				Processing: Each music file should be broken down into an audio time series and analyzed to determine its length, average pitch, average pitch of the first 30 seconds of the music file, average pitch of the last 30 seconds of the music file, tempo of the first 30 seconds of the music file, tempo of the last 30 seconds of the music file, and tempo of the entire data file.
				Outputs: CSV file with each row consisting of the following for each song present within the music directory: song length, average pitch, pitch of the first 30 seconds of the song, pitch of the last 30 seconds of the song, tempo of the song, tempo of the first 30 seconds of the song, tempo of the last 30 seconds of the song.
				Error Handling: If a file is unable to be processed, its information will not be

			saved to the CSV file.
			Forward Dependencies: FR 3.2.6
			Backwards Dependencies: None.
FR 3.2.5	User Selects Approach Path	LOW	**Description:** In order for the desktop application to be able to predict change in heart rate, the desktop application must know how the user wishes to get to their current heart rate. Under the advanced settings page in the desktop application the user will be able to select from a variety of approach paths. The available choices to the user will be:"Shallow", "Linear", "Steep", "Fastest" , "Parabola", and "Rollercoaster". Shallow, linear, and steep refer to the rate to which the heart rate will change, while parabola refers to overshooting the target heart rate then bringing the user back in the direction of their target. Rollercoaster refers to the song selection alternating towards increasing and decreasing heart rate in the direction of the target heart rate.
			Inputs: User selects approach path
			Processing: None.
			Outputs: String of approach path type selected.
			Error Handling: If no approach path is selected, the approach path will default to the default approach path.
			Forward Dependencies: FR 3.2.10, FR 3.2.6.
			Backward Dependencies: None.
FR 3.2.6	Select Music Feature	HIGH	**Description:** When the user wishes for the machine learning model to select music for them, they should be able to initiate it via a button at the bottom of the homepage. The select music

			feature will then use a machine learning model to generate predictions for what the change in heart rate will be for every song present in the music_characteristics.csv. A song will be chosen based on the user's specified 'approach path'. This feature will then begin automatically playing the song. This feature will then be automatically repeated when a song ends or is skipped until the user selects "stop".
			Inputs: User selects start, music characteristics.csv file path, current heart rate, resting heart rate, approach path.
			Processing: A machine learning model will comb through every row of the csv generating predictions for each song. The way a song is chosen as a prediction will be based on the approach path specified.
			Outputs: File Path of selected song.
			Error Handling: If no song is capable of achieving the user's target heart rate, an error message will prompt the user that no song within the song directory is suitable to achieve the desired change in heart rate. If the user has not connected their ANT+ enabled heart rate monitoring device or entered their current heart rate the user will not be allowed to start the select music feature.
			Forward Dependencies: FR 3.2.11
			Backward Dependencies: FR 3.2.6, FR 3.2.5, FR 3.2.4, FR 3.2.3, FR 3.2.2, FR 3.2.1
FR 3.2.8	Navigation bar	LOW	**Description:** The user will have the ability to traverse from the home page to the user guide page to the advanced features page via a navigation bar.
			Inputs: Button click.
			Processing: None.

			Outputs: None.
			Error Handling: None.
			Forward Dependencies: None.
			Backward Dependencies: FR 3.2.11 FR ,3.2.10, FR 3.2.9
FR 3.2.9	User Guide Page	LOW	**Description:** The user guide page will consist of text and image descriptions of how to operate the desktop application. A video instruction will also be present.
			Inputs: None.
			Processing: None.
			Outputs: None.
			Error Handling: None.
			Forward Dependencies: None.
			Backward Dependencies: None.
FR 3.2.10	Advanced Features Page	LOW	**Description:** The user guide page will consist of a page where the user can select the desired approach path.
			Inputs: None.
			Processing: None.
			Outputs: None.

			Error Handling: None.
			Forward Dependencies: None.
			Backward Dependencies: FR 3.2.5
FR 3.2.1 1	Home Page	HIGH	**Description:** The home page will consist of establishing a connection to the ANT+ compatible heart rate monitor, the button to select the resting heart rate, the button to select target heart rate, and the button to begin the select music feature.
			Inputs: None.
			Processing: None.
			Outputs: None.
			Error Handling: None.
			Forward Dependencies: None.
			Backward Dependencies: FR 3.2.6, FR 3.2.4, FR 3.2.3, FR 3.2.2, FR 3.2.1
FR 3.2.1 2	Upload File	LOW	**Description:** The user should be able to drag and drop any mp3 file into the desktop application's front end to make uploading new files easy. When a new file is added, its music characteristics should be gathered.
			Inputs: Drag and drop mp3 audio file.
			Processing: Change file path of mp3 file to music directory.
			Outputs: None.

			Error Handling: None.
			Forward Dependencies: None.
			Backward Dependencies: FR 3.2.4

3.3 Non-Functional Requirements

ID	TITLE	DESCRIPTION
NFR 3.3.1	Performance	The user will not likely tolerate the desktop application taking more than 10 seconds to choose a new song. Therefore the desktop application must choose a new song within 10 seconds of the start button being pushed. The desktop application should also take less than 5 seconds to load..
NFR 3.3.2	Safety	It may be possible for the user to select dangerously high heart rate targets. To prevent injury to the user the heart rate targets should be limited to between 40 and 150 beats per minute.
NFR 3.3.3	Availability	The desktop application should not require internet availability in order to function.
NFR 3.3.4	Security	The desktop application should not store any heart rate data that might legally be considered private healthcare information.
NFR 3.3.5	Maintainability	To ensure maintainability the desktop application should be constructed with a HTML front end. HTML has been the web standard for the future, and nearly all browser applications are still backwards compatible with the first HTML3 pages created in 1995. Similarly CSV files have been in use since

		1972 and should ensure continued functionality. To ensure that the code is easy to maintain, the code should be broken down into separate files, with each file only performing one purpose - thus ensuring that understanding its purpose is clear.. The front end application should then access each of these singular files and use them for their intended purpose.
NFR 3.3.6	Portability	The desktop application should be able to run on any computer running Windows 11.

4. Design Constraints

ID	Title	Constraint
DC 4.1	Song Limit	The music folder can store no more than 10,000 songs. This limit is imposed due to the size of the CSV file limits.
DC 4.2	File Compatibility	The desktop application will only be compatible with .mp3 file types.
DC 4.3	ANT+ Compatibility	The desktop application will only be compatible with ANT+ compatible heart rate monitoring devices.

5. Logical Database Requirements

Entity Relationship Diagram:

music_characteristics

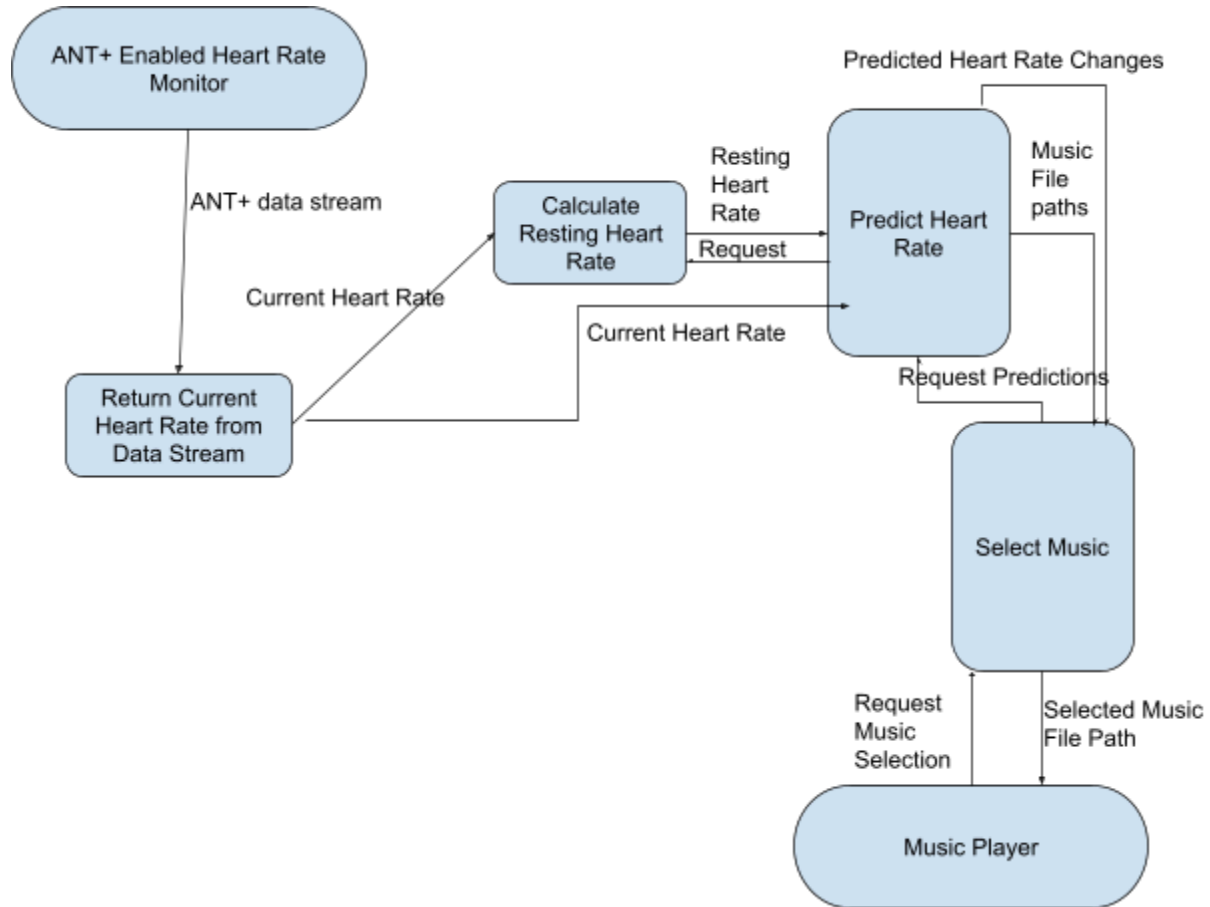| |
|---|
| Song filepath |
| Tempo of Song (bpm) |
| Tempo of First 30 Seconds of Song (bpm) |
| Tempo of Last 30 Seconds of Song (bpm) |
| Length of Song (seconds) |
| Average Pitch of First 30 Seconds of Song (hz) |
| Average Pitch of Last 30 Seconds of Song (hz) |
| Average Pitch of Entire Song (hz) |

6. Other Requirements

ID	TITLE	REQUIREMENT
OR 6.1	Accuracy	The machine learning model should have 80% or more accuracy in predicting the direction of change in heart rate for any given song.

7. Analysis Models

7.1 Data Flow Diagram

8. Appendices