**Computer Science and Engineering, University of Nevada, Reno**

**MelodyBot**

**Team #19**

**Nicholas Harris**

**David Neilsen**

**Joseph San Nicolas**

**Stefan Stukelja**

**Instructors: Sergiu Dascalu & Devrin Lee**

**External advisor(s): Dave Feil-Seifer**

**Benjamin Brown**

**March 6th, 2018**

# **Table of Contents**

Abstract……………………………………...2

Project Updates and Changes…………….2

User Stories and Acceptance Criteria…...2

Testing Workflow…………………………...5

Testing Strategy…………………………….5

Contribution of Team Members…………...7

# Abstract

MelodyBot is is an artificial intelligence application whose purpose is to create unique music samples based on the genre and other parameters that are specified by the user. MelodyBot is able to create the desired music sample thanks to its use of a Recurrent Neural Network (RNN). The neural network has been able to generate music samples ranging in length from 10 seconds to 3 minutes from multiple genres, even making use of musical chords instead of only individual notes. The developers of MelodyBot have created a functioning user interface using Visual Basic. The current user interface is able to load sound files in a viewable list, play and stop/pause said sound files, and remove the sound files from the list. It is also able to invoke the artificial intelligence and provide information about the product.

# Project Updates and Changes

The development of MelodyBot has made significant progress. The application has essentially completed its proof-of-concept phase, and is now capable of producing a sample of three different genres: classical violin, jazz piano, and blues guitar. The AI is also replicating musical chords to create fuller musical samples. The ability to invoke the AI’s sampling is being incorporated into the user interface, at which point the core functionality of the application will be implemented. Future steps will involve training the neural network on more and more genres and selections to increase options for the user. In addition, improved versions of the user interface and the AI can be continuously made and incorporated into the application. Finally, additional small features can start to make their way into the application, such as the ability to select a tempo or key for the musical sample.

The only significant change to the project’s original design, as described in the previous report, is the switch to using a recurrent neural network (RNN) for the initial implementation of MelodyBot, instead of the initially proposed generative adversarial network (GAN). The reason for this switch was to make the initial implementation of MelodyBot simpler. It is still quite possible that the developers will wish to implement a GAN in future iterations of MelodyBot.

# User Stories and Acceptance Criteria

**Neural Network**

* As a recurring customer, I want the music that I receive to match the genre that I asked for.
  + AC1. AI is able to generate music samples of different genres.
  + AC2. AI is able to differentiate between the genres it can produce.
  + AC3. AI is able to identify the genre that the user wants.
  + AC4. AI is able to generate a music sample that matches the genre specified by the user.
* As a music lover, I want the music that I receive to sound good so that I can be satisfied with transaction and possibly ask for more music.
  + AC1. The AI is able to produce music samples that sounds appealing to the ear.
  + AC2. The AI is able to consistently produce terrific sounding music samples to increase customer satisfaction.

**User Interface**

* As a person who is not tech-savvy, I want a product that easily to understand because I may have limited knowledge with computers.
  + AC1. The UI will give easy to understand and non-technical instructions on how to use the product.
* As a customer, I want user interface that looks good and works well so that it is easier for me to use.
  + AC1. The UI is able to run the AI correctly with no errors or crashes.
  + AC2. The UI is design in a manner that is both aesthetically pleasing and professional looking.

**Website**

* As a curious customer, I would like to know more about the project so I can understand how the product works.
  + AC1. The website will include a “About” section that will educate the user on how the product is able to produce these music samples, both in a non-technical manner as well as a more in-depth technical manner.
* As a cautious customer, I would like to know more about the team that created this product so I know they are trustworthy and know what they are doing.
  + AC1. The website will include a “Team” section that will give a brief description of each of the team members and their contribution towards the product development.

**Project Management**

* As a frequent customer, I want to know when the product comes out with new, interesting things so I can use them.
  + AC1. The website as well as the UI will include an “Updates” section which talks about the updates that the team have done to the product to better improve the customer’s experience.
* As a vocal customer, I want to give the team my opinion about their product so they can make changes to their product that would give the users a better experience.
  + AC1. Upon achieving a large user base, the team will implement a discussion board so users can post their issues/ideas to better improve the product.

# Testing Workflow

Happy Paths

1. User successfully enters a seed that is accept by the neural network. Is validated by the neural network successfully generating the music from the seed.
2. When the user updates the settings they will save and alter the characteristics of the generated music. This will be validated after the music is produced and seen to have adhered to the settings changes.
3. User interface for the audio controls works as intended. Validation will be observed when the pause button stops the music, play button resumes the music, volume slider correctly adjust the loudness and so on for every widget in the audio player.
4. Launching the application successfully loads the UI and loads the settings. Validated when the UI window appears with the settings that were saved from the previous session.

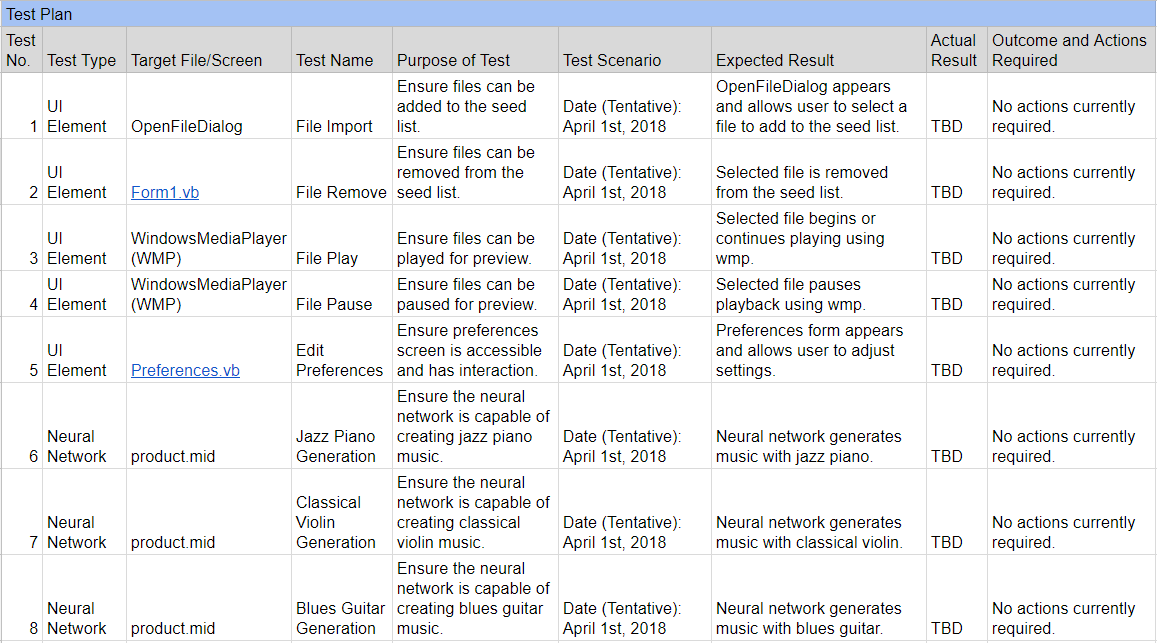
Unhappy Paths

1. The user enters an unreadable seed. Validated when the neural network returns an error.
2. Settings fail to save changes. After the user attempts to make changes to the neural network an error could cause them to either not change or just not reach the neural network before the next song is generated where the error will be validated.
3. Any audio control errors will be validated as soon as any of the audio widget buttons do not have the intended effect on the audio player.

# Testing Strategy

Testing for MelodyBot will be conducted over the course of the next two months with specific details documented in the test plan below. MelodyBot will use acceptance testing and unit testing strategies to prepare for the release version. Acceptance testing will be done to ensure user requirements such as file importing, file removal, file preview, and preference editing are all working correctly. Unit testing will be conducted on the generation feature with varied seeds to confirm all generation options are operational and able to be processed correctly by the artificial intelligence. Test automation will not be incorporated into MelodyBot’s testing due to the complexity that would be required of the automation system to analyze and judge whether the output from the artificial intelligence is adequate. User testing is not necessary for the initial release version due to the simplicity of MelodyBot’s use. Future releases of MelodyBot will necessitate user testing as the user interface gets more cluttered and complex to navigate with added features.

Acceptance testing will be conducted by Joseph San Nicolas and completed by April 1st, 2018. The unit testing will be conducted by David Neilsen and also be completed by April 1st, 2018. Defects will be documented with what the issue is, how to replicate it, and any suggestions for its resolution and then reported to the group discussion board used by the development team. The distribution of work for the rectification of issues will be designated on a case-by-case basis depending of factors such as familiarity with the issue and the systems involved. The project will be deemed complete when it has passed all tests outlined in the testing plan.



# 

# Contribution of Team Members

**Time Log**

This is a record of each team member’s time spent on each section of this report:

* **Nicholas Harris**
  + **Abstract: 30 min**
  + **Project Updates and Changes: 1 hr**
  + **Time Log: 30 min**
  + **Paper Revisions: 30 min**
* **David Neilsen**
  + **Testing Workflow: 2 hr**
  + **Revising paper: 30 min**
* **Joseph San Nicolas**
  + **User Stories/Acceptance Criteria: 2 hr**
  + **Revisions: 30 min**
* **Stefan Stukelja**
  + **Testing Strategy: 2 hr**
  + **Paper Revisions: 30 min**