

Impromptune

Team 5

Project Backlog

Ben Ahlbrand

Chris Doak

Sean Phillips

Jacob Richwine

Problem Statement:

In the current music creation software landscape there is a lack of software that dynamically generates music based on the user's input and musical ideas. There is an overabundance of composition software, but lack of user-friendly computer assisted generative tools, particularly for those with a formal musical background. This software is different in the fact that it will take pre-existing motifs and produce entirely unique patterns, ideas, and riffs that compliment the user's musical input, and allow for manual manipulation of generated output with a graphical user interface.

Background Information:

There is currently a market gap in the available music generation software arena, and our software will target composers, students, or any individual that wishes to create their own music with the help of an algorithm that analyzes musical input then dynamically generates output. There are three current programs that have similar features:

1. Microsoft Songsmith: Generates background music from vocal input. Lacks capability for analyzing or generating from instrumental input.
2. Rocksmith: Session mode is part of this videogame that focuses on real-time processing of a guitar signal with backing tracks as opposed to assistive composition with editing. Lacks ability to save or influence output.
3. Band-in-a-Box: A complex software suite that provides dynamically generated songs from inputted chords. Lacks the ability to generate songs from notes, and other non-chord musical input.

All of the above are lacking simple features that target musical composers willing to experiment with generative algorithms for instrumental input. Our software will provide features to accomplish that and fill in the areas where other software is lacking.

Requirements:

Functional:

1. As a user, I would like to manually compose music.
2. As a user, I would like to choose from different parameters for algorithmic composition.
3. As a user, I would like to be able to re-generate pieces of previously generated patterns.
4. As a user, I would like to be able to edit a generated pattern.
5. As a user, I would like to save and load my work.
6. As a user, I would like to see the composed music as sheet music.
7. As a user, I would like to add dynamics to my music.
8. As a user, I would like to be able to use a MIDI controller for manual input (if time allows).
9. As a user, I would like to import a music file to load into the program (if time allows).
10. As a user, I would like to choose from various instruments.
11. As a user, I would like to have control over the generative patterns' parameters.
12. As a user, I would like to play each channel individually or concurrently.
13. As a user, I would like to have multiple projects open at same time.
14. As a user, I would like to print my music.
15. As a developer, I would like the music generation process to play live feedback when a user provides input via a MIDI device (If time allows).
16. As a developer, I would like to easily add new musical generation templates (if time allows).
17. As a developer, I would like to easily create new styles of music (if time allows).
18. As a developer, I would like to create a new file format to be used by the program.
19. As a developer, I would like to have the music generation process take a maximum of a few seconds.

Non-Functional:

1. We must be able to run this device on Microsoft Windows
2. We must be able to run this device on Unix based operating systems (if time allows)
3. The interface needs to be intuitive and allow experienced musicians sufficient control (simple suggestions that can be easily edited, to the note)
4. The GUI must be responsive.
5. The program is written in Java, using JavaFX as the GUI library.
6. Use the MusicXML format and JFugue library.
7. The application has a place to display current processing tasks (if time allows).