Impromptune

Sprint 2 Planning Document

Team 5

Ben Ahlbrand

Chris Doak

Sean Phillips

Jacob Richwine

1. Sprint Overview

This sprint will primarily concern itself with adding more functionality to manual input, including redo/undo, save as MusicXML, modify opened scores, and fixing existing issues from previous sprint. This Sprint also starts work on implementing a way to analyze the music for generation. Even though we completed many of the same user tasks in Sprint1, this sprint expands upon that feature set.

Completed user functionality by end of sprint include:

- Fix GitHub issues/sharing issues
- Undo/Redo Functionality
- Modify Opened Files
- Display multiple score pages
- Save MusicXML files
- Fix manual composition issues
- Customize new composition(choose time, tempo,clef) (GUI)
- Customize new generation(choose options) (GUI)
- Send manual input to analyzer
- Implement generation control
- Add tab to display temporary generation (GUI)
- Full GUI Implemented

Scrum Master: Jacob Richwine

Meeting Schedule: We will meet at least once a week on Tuesday/Thursday from 3 to 5.

Risks/Challenges:

- User tasks code completed in Sprint 1 must be redone or completely changed due to requirements of this sprint.
- Incompatibilities discovered between 3rd party code
- Modification of Zong! renderer code is infeasible within our time constraints
- Music analysis via code
    - It will be very challenging because no one in the team has done anything like it before and it is a seldom implemented concept
- Generation of accompaniment complexity
    - Will it be pleasing to the human ear?
    - Not a 'solved' problem, results will be interesting to see/hear
    - Deciding which parameters to allow the user to define, and which choices to assume

2. Current Sprint Detail

As a user, I would like to be able to compose via a GUI.	Estimated Time	Responsible
Continually Polish GUI	10 Hours	Chris
Implement keyboard shortcuts	5 Hours	Chris
Embed Generation Controls to Main View	3 Hours	Chris
Create Generation Dialog View	5 Hours	Chris
Apply CSS to GUI	5 Hours	Chris
Display Multiple Pages of the Composition	7 Hours	Chris
Display a Preview of Generated Music	6 Hours	Chris

As a user, I would like to have control over the generative patterns' parameters.	Estimated Time	Responsible
Decide on appropriate parameters and create corresponding graphical controls.	4 Hours	Sean
Create data store and retrieval mechanisms for parameters	3 Hours	Sean

As a developer, I would like to analyze music based on parameters.	Estimated Time	Responsible
Research Music Analytics/Theory, Markov Models, and Java based State Machines	10 Hours	Sean
Define all necessary classes/methods/structures for musical analytics.	6 Hours	Sean
Parse XML / Zong data into custom analytic data	8 Hours	Sean
Create probability engine that will determine possible progressions / chords based on given notes	10 Hours	Sean
Form JFugue object from Piano Input and send to Analyzer	2 Hours	Jacob

As a developer, I would like to generate music based on parameters.	Estimated Time	Responsible
Research layered state generation w/anchored Markov chains	10 hours	Ben
Define/extend associated classes/structures	5 hours	Ben
implementation utilizing analytics classes, generate basic accompaniment (investigate stylistic impact of different constraints) and prepare results for display	10 hours	Ben
Finish manual inputs w/Zong to Composition	5 hours	Ben

As a user, I would like to see the composed music as sheet music.	Estimated Time	Responsible
Allow multiple renderings to be shown and tie playback/piano to each one	4 Hours	Jacob
Modify current Zong! renderer to display multiple pages of the loaded song	8 Hours	Jacob

As a user, I would like to be able to edit / interact with the composition.	Estimated Time	Responsible
Extend Zong! code to allow cursor position for input	8 Hours	Ben & Jacob
Investigate Zong! possibility for selecting notes from GUI	10 Hours	Jacob
Allow modification of loaded files	2 Hours	Jacob

Continuing From Sprint1	Estimated Time	Responsible
Fix known bugs introduced in Sprint 1	4 Hours	Ben & Jacob
Implement Undo/Redo	6 Hours	Ben & Jacob
Add File-Save Logic	2 Hours	Ben

Miscellaneous Tasks	Estimated Time	Responsible
Fix and optimize intellij and git issues	3 Hours	Jacob
Test Sprint 1 Demo	1 Hour	Jacob
Fix bugs found by testing	1~3 Hours	Jacob
Code Review of Sprint 1	1 Hour	Jacob
Update Backlog to current Project Status and adapt to project changes	2 Hours	Jacob

3. Backlog

Functional:

1. As a user, I would like to choose from different parameters for algorithmic composition.
2. As a user, I would like to be able to re-generate pieces of previously generated patterns.
3. As a user, I would like to add dynamics to my music.
4. As a user, I would like to be able to use a MIDI controller for manual input (if time allows).
5. As a user, I would like to manipulate the musical renderer to allow for easier modification
6. 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).
7. As a developer, I would like to easily add new musical generation templates (if time allows).
8. As a developer, I would like to easily create new styles of music (if time allows).
9. As a developer, I would like to create a new file format to be used by the program.
10. 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 software on all operating systems.
2. The interface needs to be intuitive and allow experienced musicians sufficient control (simple suggestions that can be easily edited, to the note)
3. The GUI must be responsive.
4. The application has a place to display current processing tasks (if time allows).