Applying User-Centered Techniques in the Design of a Usable Mobile Musical Composition Tool

Conference Paper · April 2019

CITATIONS
0

READS
13

5 authors, including:

Jordan Aiko Deja
De La Salle University
22 PUBLICATIONS **13** CITATIONS
SEE PROFILE

Kevin Gray Chan
De La Salle University
6 PUBLICATIONS **4** CITATIONS
SEE PROFILE

Migo Andres Dancel
De La Salle University
2 PUBLICATIONS **1** CITATION
SEE PROFILE

Applying User-Centered Techniques in the Design of a Usable Mobile Musical Composition Tool

Kevin Gray Chan
De La Salle University
Manila, Philippines
kevin_gray_chan@dlsu.edu.ph

Jordan Aiko Deja
De La Salle University
Manila, Philippines
jordan.deja@dlsu.edu.ph

John Patrick Tobias
De La Salle University
Manila, Philippines
johm_patrick_tobias@dlsu.edu.ph

Allen Vincent Gonzales
De La Salle University
Manila, Philippines
allen_gonzales@dlsu.edu.ph

Migo Andres Dancel
De La Salle University
Manila, Philippines
migo_dancel@dlsu.edu.ph

ABSTRACT
In this study, we present the music composition tool Flow and how an interaction was designed that led towards introducing balance in the work of musicians across all stages in musical composition. Observation and user research led to having a deeper understanding of the various needs, gains and pain points musicians encounter when composing. Musicians and composers who participated in the study, came from varying levels of expertise from beginner (those with less than 7 years) and veteran (those with beyond 10 years experience). An iterative process of design and development was continuously employed which led to improving the interaction design within the prototype. The processes described in this study show how insights were uncovered from a comprehensive set of usability tests and inspections done. These insights led to the development of a more usable and acceptable musical composition tool as seen from the results in the user tests. It can be observed that varying levels of expertise in music composition leads to different expectations and needs with regards to a music composition prototype. Results of the user tests show that Flow achieved a level of satisfaction and usability at par with the industry-standard tools.

ACM Classification Keywords
H.5.2. Information Interfaces and Presentation (e.g. HCI): User–Centered Design

Author Keywords
Human Centric Computing; Human Computer Interaction; Interaction Design; User Interface Design; Interaction Techniques, Gestural Input

CHIUXID'19, March 31–April 09, 2019, Bali, Indonesia

ISBN 123-4567-24-567/08/06…$15.00

DOI: http://dx.doi.org/10.475/123_4

Figure 1. The Cyber Glove, as seen in the work of [13].

INTRODUCTION
Musical composition can generally be summed up into three (3) main activities: ideation, sketching, and revision [1, 10]. But given that it is a creative endeavor, composers have the freedom to create and follow their own process when composing music [18, 16, 9, 4, 14]. The fact that the process has become personal makes it possible that one composer's approach to composing music may not be effective, or even possible for another [8, 11, 3, 5]. This makes it especially hard for some composers to use modern musical composition tools, because they are forced to adapt to an unnatural process. In this paper, we present how a usable mobile musical composition tool was designed to assist composers in their natural composition process, rather than force them to follow a different approach. Following the works of [7] and [6], an iterative and user-centric approach was followed in designing the application. Interviews, user tests, and other relevant methodologies are discussed in the succeeding sections.

RELATED WORK
Existing studies have also recognized that existing systems can hinder composers from being expressive in their composition

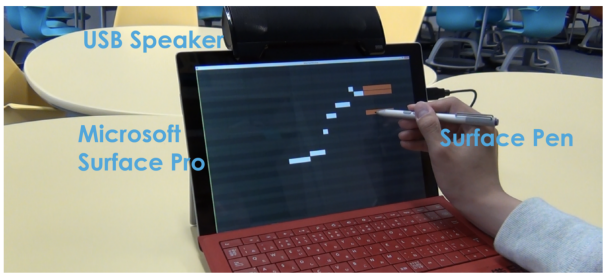

Figure 2. The interface of the composition application developed in the work of [14].

[12, 17]. Creating good compositions is time-consuming and requires the composition tool to work with the composer and not against them [15, 14, 19, 2]. This concept was applied in the work of [13] through the use of gestures. The goal of the study was to create a method of composition that felt natural. Hence, they attempted to create a tool, called the Cyber Glove, that detected the hand gestures of a person and translated them into corresponding sheet music (see Figure 1). This had a downside because although it felt natural, the tool was found to be obtrusive and inaccessible. The work of [14] also attempted to develop a usable mobile musical composition tool. In their study, they made use of a Microsoft Surface Pro 3 and a Surface Pen for as the platform and interaction method (see Figure 2). Their study differed in that they used a piano-roll type interface for their application. This was simpler and easier to understand for people new to composition, but also made it unnatural and far from the music sheet and notation that composers often used. This study is a continuation of the work of [7]. Given that it was an initial study, only first two (2) iterations were presented. There was also an emphasis on the quantitative data from the tests and how they drove the changes in between the two (2) iterations. This study continues that work with further development and iterations and gives more focus on the qualitative analysis of the data from the tests.

METHODS

This study was done with three (3) main phases: (1) Understanding Composer Needs, (2) Interaction Design and Implementation, and (3) Verification and Evaluation. This framework has been patterned after the work of [7]. The purpose of the first phase was to understand the pains, gains, and needs of the composers while during the musical composition process. The data and information we gathered during the first phase was then used for the second phase to aid in the design of the solution. The resulting solution was used in the third phase to evaluate it with composers.

Participants

Testers were recruited through snowball sampling. As a prerequisite, they needed to have at least three (3) years of experience with musical composition and notation by the time they participated. They were classified into two (2) groups based on their years of experience: experts, and amateurs. A composer must have at least seven (7) years of experience to classify as an expert. The demographics of the characteristics of these testers are shown in Table 1.

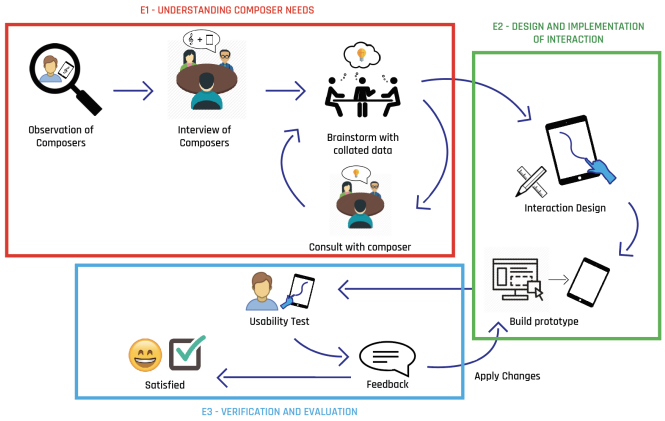

Figure 3. Phases of the research as based from the work of [7]. These has been divided into three main phases namely Understanding Composer Needs, Interaction Design and Implementation and Verification. Notice how each stage involves the users and are put at the center of the process. It is ensured that for every comments or changes made in the application, their insights and feedback are gathered and used to create the improved newer version of the prototype.

Study Design

We applied an iterative process in our study to allow the application to improve from the user feedback over time. We were able to perform a total of four (4) iterations with each iteration including building the prototype (Phase 2: Interaction Design and Implementation), evaluating it and getting feedback from composers (Phase 3: Verification and Evaluation), and using that feedback to identify necessary adjustments and improvements to the prototype for the next iteration. The goal of the tests was mainly to improve the usability and user experience of the application. As such, qualitative data was gathered through the Think Aloud Protocol, video recordings of the testers' expressions and hand movements while using the prototype (see Figure 4, and post-user testing interviews. Quantitative data was also gathered using a questionnaire to provide metrics to show whether the application was improving with each iteration. During tests, the composers had to go through different test setups, making use of three (3) different applications, with one of them being the application we designed (called Flow) and the other two being commercially available applications (komp and Notion). In each test setup, the composers were asked to perform specific use cases in an attempt to simulate different musical composition scenarios. At the same time, they were meant for composers to make use of the different features of the application and how they would they would approach them. Given however, that musical composition is a creative process and that some composers would have a hard time making up a completely new composition in a short span of time, they were advised that the goal of the tests was to see how they would interact with the application. Hence, the quality of their output did not matter.

The use cases that the testers performed are shown below:

- Compose a familiar song
- Compose from scratch

Table 1. Demographic characteristics.

Characteristic	Experts (n=17)	Amateurs (n=8)	Total (n=25)
Age (mean ± SD [range])	25.8 ± 8.8 [20-47]	22.1 ± 1.4 [21-25]	24.6 ± 7.5 [20-47]
Sex (n [%])			
Male	12 [71]	6 [75]	18 [72]
Female	5 [29]	2 [25]	7 [28]
Marital Status (n [%])			
Never Married	15 [88]	8 [100]	23 [92]
Married	2 [12]	-	2 [8]
Educational Attainment (n [%])			
Secondary	9 [53]	4 [50]	13 [52]
College	5 [29]	4 [50]	9 [36]
Graduate	3 [18]	-	3 [12]
Occupation (n [%])			
Employed	5 [29]	3 [37]	8 [32]
Self-employed	12 [71]	5 [63]	17 [68]
Years of Experience	12.7 ± 5.6 [7-29]	4.9 ± 1.2 [3-6]	10 ± 6.0 [3-29]

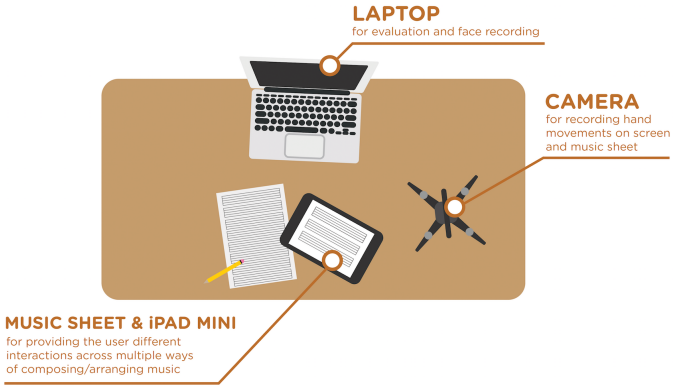

Figure 4. The physical map describing the setup for user tests. It important that in all rounds of user testing, the scenarios are the same and recreate the same atmosphere. This way, it is expected that external features that could potentially affect evaluation such as stress, backpain have been minimized.

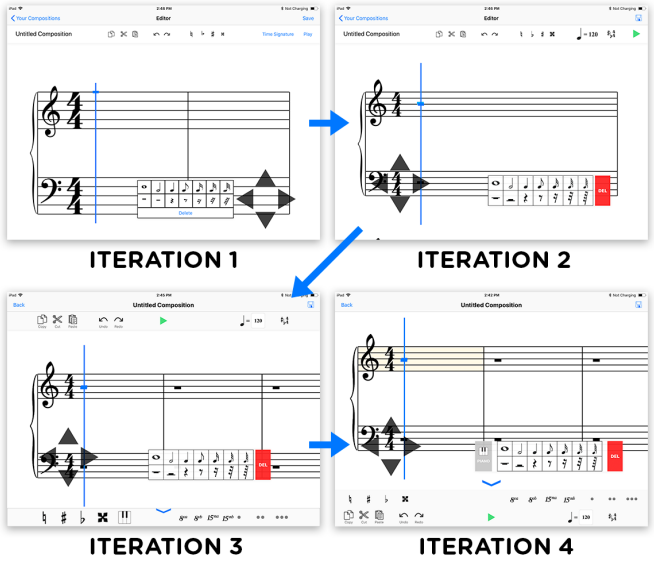

Figure 5. Screenshots of the application from each iteration. We were able to produce 4 milestone versions of the prorotype. Changes and improvements were made and derived from the feedback of the usability tests and comprehensive user research studies done after each version.

- Modify a composition

Immediately after the tests, we conducted interviews with the composers. The goal of these interviews was to confirm observations during the testing and also identify how they felt and their experience while interacting with the applications. Quantitative data was collected through questionnaires. They were used to gather usability data on the different features of the applications, and easily identify problematic features. The question, "How easy was this activity to perform?" was asked for each feature with the answer being on a scale of 1 - 4, 1 (Very Difficult) being the lowest, and 4 (Very Easy) being the highest.

RESULTS

This section contains the quantitative and qualitative results from the tests and interviews that we have conducted in each iteration and the changes made according to these results.

Table 2. Summarized Main Feature Scores from Iteration 1 - 4

Feature	I1	I2	I3	I4	Average
Select/highlight notations	3.1	3.6	3.7	3.4	3.4
Add notations	2.8	3.5	3.7	3.2	3.3
Edit notations	2.2	3.2	3.3	3.0	2.9
Delete notations	2.2	3.6	3.7	3.4	3.2

Feature Usability Scores

The usability scores of the main features obtained from the quantitative data gathering are shown in Table 2.It can be observed that each feature increases in its average rating from iteration 1 to iteration 3. This can imply an improvement in the user experience. Scores went down a bit in iteration 4, but this could be attributed to the demographics of the testers since all but one (1) have never used the application before and most were experts.

Iteration 1

The first version of the high-fidelity prototype (see Figure 5) took inspiration from existing musical composition applications available on computers. This can be observed in the presence of the notation menu and the arrow keys. Since adding and editing was one of the tasks that composers would be doing the most, they needed to be the easiest to perform. We added a notation menu to allow composers to quickly add notations when they had an idea in mind. A full set of arrow keys was provided to control the cursor, taking inspiration from the "speedy entry" from Finale (a musical composition application for computers) where users would press the arrow keys on the keyboard to move the cursor that tells where the note or rest would appear. However, users were also given the ability to tap on any valid location they wanted on the measure to move the cursor.

Iteration 2

The biggest change for iteration 2 came with the revised highlight/select interaction. In the iteration 1, we used a two-finger drag interaction for highlighting/selecting since a one-finger drag was commonly used for scrolling in mobile applications and we wanted to keep that retention for users. Although from the tests, it was found that the two-finger drag for highlighting/selecting notations in the previous iteration was hard to figure out. This directly affected the Edit notations and Delete notations features, giving them the low scores they have in iteration 1 (see Table 2). It was observed that they would try to use a one-finger drag to highlight (see Figure 6). Hence, the gestures for the highlight and scroll interactions were switched, resulting in the highlight using only a one-finger drag, while the scroll used a two-finger drag (see Figure 7).

Iteration 3

Iteration 3 added a new bottom menu containing modifiers like accidentals, dots, and ottavas. This was done so that the top menu would not be crowded. A keyboard, which can be shown using the show/hide keyboard button, was also added to allow users to test out melodies on the keyboard while composing. For this version, we also improved on the time and key signature menu (see Figure 9). In the previous version,

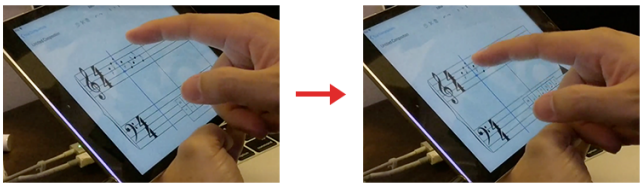

Figure 6. Segments from the user testing video of a tester trying to perform the highlight interaction.

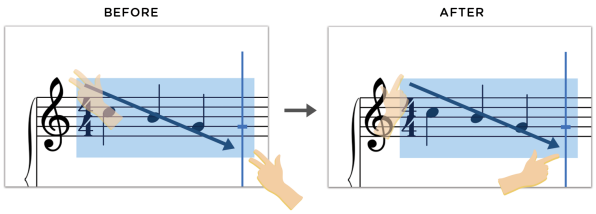

Figure 7. The highlight interaction before and after changes were made due to the user testing. The old interaction used two (2) fingers which the users did not find intuitive. The new interaction uses only one (1) finger.

users had a hard time with the menu because of the sliders. The qualitative data shows that the sliders made selecting inaccurate and tedious (see Figure 8). It was common for users to slide their finger to get the time signature they wanted, then after lifting their finger, the slider moved to the next value forcing them to repeat the process again. For the next iteration, buttons for common time signatures were added, but users could also type the time signatures they wanted using the keyboard. The key signature menu was redesigned to be a circular menu that follows the circle of fifths. We also observed that the transposition interaction needed improvement. In the previous prototype, users can tap on the up or down arrow keys to instantly transpose the selected notes to a higher or lower pitch respectively. This was sometimes confusing and not easy to find (see Figure 10) because the users' assumption was that the arrow key was only used to move the cursor and not the notes. They would eventually be able to figure out after some messing around, but this still needed to be improved. Hence, in this iteration a menu was added containing the transpose arrow keys and other modifiers that would only appear when a user highlights a set of notes (see Figure 11. This not only made it more obvious, but also saved space by only showing the necessary buttons when needed.

Iteration 4

With the addition of the bottom menu came a new problem: it split the focus of the users. Since there was now a menu at the top and a menu at the bottom, some users had a harder time finding specific functions. Their initial instinct was to first look at the bottom menu so it added extra load just to find the top menu functions like cut, copy, or paste (see Figure 12). To fix this, we moved the top menu to the bottom so users would only have to look in one segment (see Figure 13). Users also mentioned some problems with the music playback. These

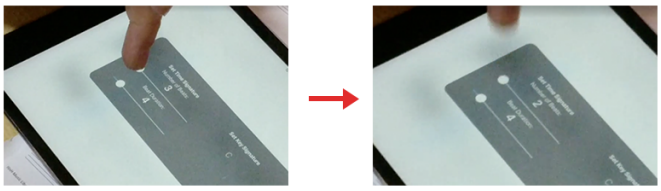

Figure 8. Segments from the user testing video of a tester trying to set a time signature using the menu. It can be noticed that after pointing to the time signature they want, lifting their finger moved the slider to the previous value.

revolved around two main issue: (1) the playback always started from the first measure, and (2) the playback did not show the current note/rest that was playing. The users tests showed that the composers would usually place the cursor at the measure where they wanted to start playing from. They would be surprised to find out that the playback always started from the first measure, regardless of the cursor placement (see Figure 14). We incorporated this feature in this iteration to reduce the tediousness when writing a long composition. We also fixed the second issue to reduce confusion and to make it easier to find the notes/rests they need to change in case they want to modify the melody (see Figure 15).

Comparison Between Applications

During the tests in the final iteration, we also asked to rate the usability of the three (3) applications they used through a scale of 1 - 4 with 4 being the highest. Table 3 lists the results from the survey. Notion ends up as the highest with an average score of 3.6, followed by Flow at 3.0, with Komp being the last at 1.9. The scores were to be expected as a lot of people commented that Notion was the application they liked using the most because of its completeness. They would often mention that they felt like Notion would be able to create a full piece hence they would almost always rate it the highest. However, when it came to the input method, a lot said that they found Flow to be the easiest and most natural. They would often say that the only problem they had with Flow was that it lacked features to become a full musical composition application, but in terms of interaction, it was very good. Komp was the least liked mainly because of its input. Although a lot of them liked its concept of writing as you would on a physical music sheet, it did not work too well and would often cause frustration for its users. In terms of the average of the feature scores, Flow rated a bit higher than Notion in iteration 3 (see Table 4), and tied with it in iteration 4 (see Table 5). According to the testers, they gave Notion low scores mainly because of its overwhelming user interface. Looking at Figures 16 and 17, it can be observed that Notion has a lot of buttons and features that do not make sense at first. It was often observed that testers had to go through all of the menu items just to find specific functions they wanted to use. Unlike Flow which testers found to be cleaner and more organized, Notion had too many features crammed in its menus. The testers usually mentioned that Notion's most problematic features was its selection interaction. There are

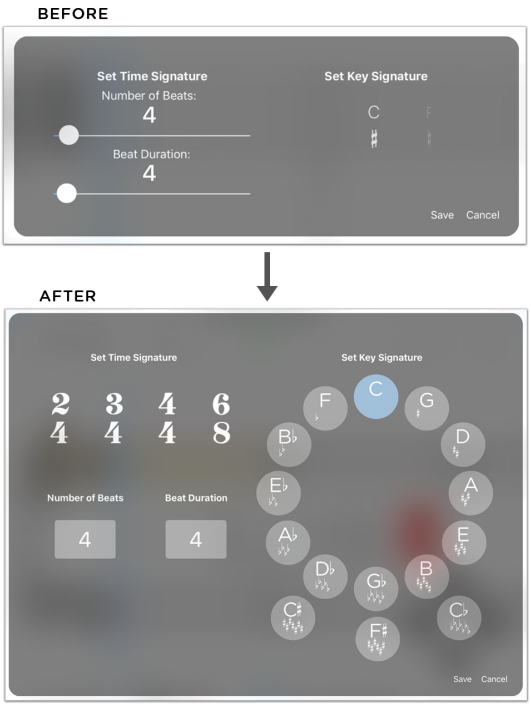

Figure 9. The time and key signature menu before and after changes were made due to the user testing. Sliders in the previous iteration made selecting inaccurate and tedious (see Figure 8). The revised time signature menu adds buttons for the common time signatures and also allows users to input any valid time signature they want. The revised key signature menu is now circular and follows the circle of fifths.

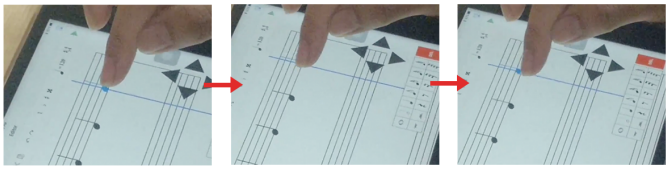

Figure 10. Segments from the user testing video of a tester trying to transpose a note. The tester was trying to drag to transpose and was not able to figure out that they had to use the arrow keys to transpose.

two (2) ways to select in Notion: (1) the user can either hold on the measure they want to select, or (2) press the select button in the menu. A lot of the testers were not able to figure this out initially since the hold gesture did not feel natural and the select button did not seem obvious to them. Even when they were able to figure out how to perform it, they still felt that it was tedious to perform and required too much time. This became a problem because actions like delete, cut, copy, or paste were only made available when notes or rests were selected (see Figure 17). Deleting notes or rests became a hassle for the testers since they had to select first so a commonly observed workaround was that they would just use the undo feature. This did not always work well because they would sometimes want to change a note even after adding other notes. They would use undo repeatedly just to edit that note and add all the previous notes again.

Table 3. Application Usability Scores

Application	Average	St. Dev	Min	Max
Flow	3.0	0.7	2.0	4.0
Notion	3.6	0.7	2.0	4.0
Komp	1.9	0.8	1.0	3.0

Table 4. Comparison of the Feature Scores in Iteration 3

Feature	Flow	Notion	Komp
Select/highlight notations	3.7	3.3	2.4
Add notations	3.7	3.2	2.2
Edit notations	3.3	3.0	2.5
Delete notations	3.7	2.6	3.2
Cut/copy/paste notations	3.3	2.8	2.0
Undo/redo an action	3.8	3.9	3.6
Music playback	3.1	3.2	2.8
Average	3.5	3.1	2.7
Standard deviation	0.3	0.4	0.6

Table 5. Comparison of the Feature Scores in Iteration 4

Feature	Flow	Notion	Komp
Select/highlight notations	3.4	3.1	3.1
Add notations	3.2	3.2	2.4
Edit notations	3.0	2.8	2.2
Delete notations	3.4	3.1	3.2
Cut/copy/paste notations	2.9	3.2	1.4
Undo/redo an action	3.6	3.6	3.3
Music playback	2.9	3.4	2.9
Average	3.2	3.2	2.7
Standard deviation	0.3	0.3	0.7

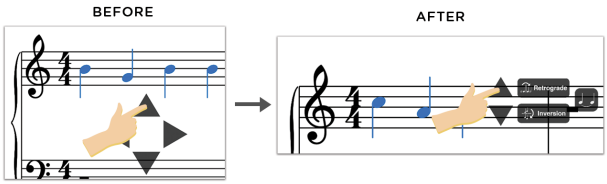

Figure 11. The transpose interaction before and after changes were made due to the user testing. The old interaction made use of the cursor arrow keys which made it confusing for the users. The new transpose interaction adds a menu containing separate transpose arrow keys as well as additional modifiers.

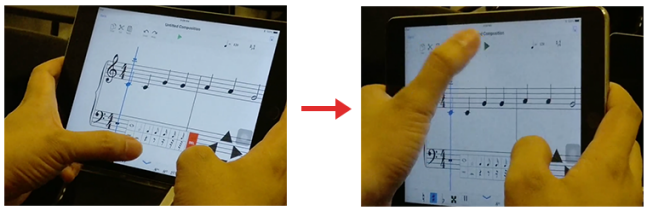

Figure 12. Segments from the user testing video of a tester looking for the play button in the bottom menu.

CONCLUSION AND FUTURE WORK

Our study provides a framework for designing mobile musical composition applications. We did this through extensive user research which included performing interviews with composers and observing their creative processes. The results of the user research led to the design and development of a usable mobile musical composition tool. This was improved and redesigned repetitively over the course of four (4) iterations. In each iteration, we tested with composers, gathered quantitative and qualitative feedback, analyzed the results, and improved the prototype for the next iteration. The user tests also led to a deeper understanding of the composers' musical composition process and their way of thinking when using musical notation applications. Initial assumptions were also proven wrong like in the case of the two-finger drag interaction for highlight which, although followed common mobile application conventions, did not work well for the composers. It was also found that including digital instruments within the application were helpful when writing and thinking of melodies. As with any application, improvements can always be made. In this study, only accidentals, ties, slurs, and dots were implemented. Some advanced modifiers like accents, or fermatas were not included given the time. It was mentioned by most of the composers that the application would be better with more features and musical notation modifiers. Future work could incorporate these in the application. Tests would be needed in this case to ensure that the additional features do not make the experience harder for composers. Further testing with more composers is also needed to gather more feedback and continue improving the application.

ACKNOWLEDGMENTS

We would like to thank De La Salle University College of Computer Studies and the Center for Complexity and Emerg-

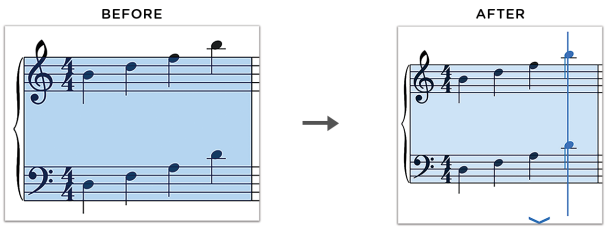

Figure 15. The music playback before and after changes were made due to the user testing. Previous versions only showed the current measure, making it hard to find specific notes or rests that were off in a measure or melody. The playback was improved to also show the current note/rest that is playing.

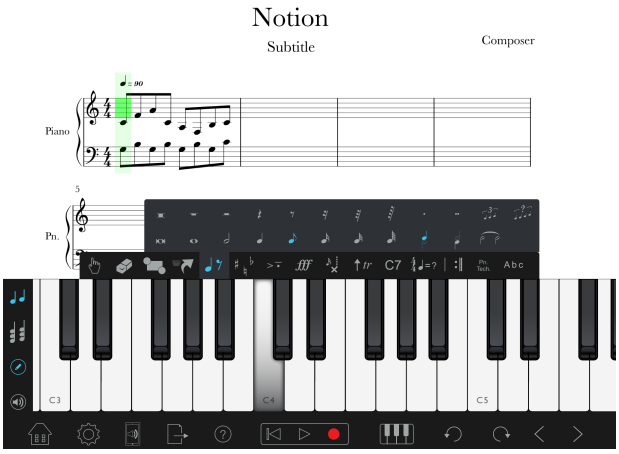

Figure 16. The composition screen of Notion.

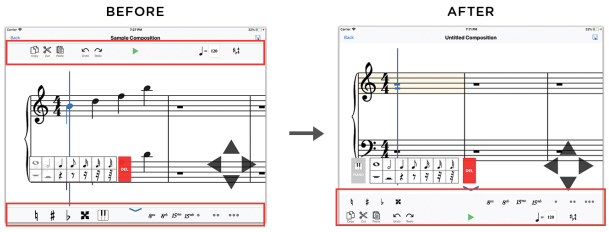

Figure 13. The menus before and after changes were made due to the user testing. In the previous iteration, having the menus separated split the focus of the users (see Figure 12). The top menu was moved to the bottom as well so users would only have to look in one area of the screen.

Figure 14. Segments from the user testing video of a tester trying to start the playback from a specific measure.

ing Technologies for all the support that they have given us.

We would also like to thank the Musicians, Experts and Composers who sacrificed their time for our interaction study.

REFERENCES

1. Stan Bennett. 1976. The process of musical creation: Interviews with eight composers. *Journal of research in music education* 24, 1 (1976), 3–13.
2. Dom Brown, Chris Nash, and Tom Mitchell. 2017. A user experience review of music interaction evaluations. In *Proceedings of the International Conference on New Interfaces for Musical Expression 2017*.
3. Christine Cheng. 2016. On approaching a performance of Paul Hindemith's Der Schwanendreher. (2016).
4. David Collins. 2005. A synthesis process model of creative thinking in music composition. *Psychology of music* 33, 2 (2005), 193–216.
5. Dave Collins. 2016. *The Act of Musical Composition: Studies in the creative process*. Routledge.
6. Jordan Aiko Deja, Patrick Arceo, Darren Goldwin David, Patrick Lawrence Gan, and Ryan Christopher Roque. 2018a. MyoSL: A Framework for Measuring Usability of Two-Arm Gestural Electromyography for Sign Language. In *International Conference on Universal Access in Human-Computer Interaction*. Springer, 146–159.
7. Jordan Aiko Deja, Kevin Gray Chan, Migo Andres Dancel, Allen Vincent Gonzales, and John Patrick Tobias. 2018b. Flow: A Musical Composition Tool Using Gesture Interactions and Musical Metacreation. In *International Conference on Human-Computer Interaction*. Springer, 169–176.
8. Kevin Donnelly. 2005. *The spectre of sound: Music in film and television*. British Film Institute; University of California Press.
9. Andrew Gartland-Jones and Peter Copley. 2003. The suitability of genetic algorithms for musical composition. *Contemporary Music Review* 22, 3 (2003), 43–55.
10. Max Graf. 2013. *From Beethoven To Shostakovich-The Psychology Of The Composing Process*. Read Books Ltd.
11. Stan Hawkins and Lori Burns. 2013. *From Pac-Man to pop music: interactive audio in games and new media*. Ashgate Publishing, Ltd.
12. D. Herremans and E. Chew. 2016. MorpheuS: Automatic music generation with recurrent pattern constraints and tension profiles. In *IEEE TENCON*. IEEE, IEEE, Singapore.
13. H. H. S. Ip, K. C. K. Law, and B. Kwong. 2005. Cyber Composer: Hand Gesture-Driven Intelligent Music Composition and Generation. In *11th International Multimedia Modelling Conference*. 46–52.
14. Junki Kikuchi, Hidekatsu Yanagi, and Yoshiaki Mima. 2016. Music Composition with Recommendation. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. ACM, 137–138.

Figure 17. The composition screen of Notion with its selection menu shown.

15. David A Levitt. 1992. A representation for musical dialects. In *Machine models of music*. MIT Press, 455–469.
16. Carolyn R Miller. 1984. Genre as social action. *Quarterly Journal of Speech* 70, 2 (1984), 151–167.
17. Philippe Pasquier, Arne Eigenfeldt, Oliver Bown, and Shlomo Dubnov. 2017. An Introduction to Musical Metacreation. *Comput. Entertain.* 14, 2, Article 2 (jan 2017), 14 pages.
18. John Rothgeb. 1975. Strict counterpoint and tonal theory. *Journal of Music Theory* 19, 2 (1975), 260–284.
19. Valerio Velardo and Steven Jan. 2016. Study Day on Computer Simulation of Musical Creativity. (2016).