Music rhythm game development and usability testing

Melodie Y. Wang
Mentored by C. Blake Ledford

Introduction

Music is a key part of our world. It lends emotion to shows and movies, entertains us on car rides, and streams on our devices. Music is all around us, even when it goes unnoticed, making music education critical. Rhythm is the baseline structure for making music that all musicians need, but for being foundational, there is surprisingly little time in music education formally put into learning it. When students begins learning an instrument, all their time is put towards learning to read notes and learn techniques unique to their instrument. Later in a student's music education, teachers expect students to have the ability to sight read music, but students struggle so much with reading rhythms properly that progress is halted.

Music rhythm games and sight-reading sites exist, but they do not require the reading of music notes or have no fun aspect. Music-based games have become one of the most popular digital games genres and have been increasing interest in real-life music making (Cassidy & Paisley, 2013). Studies conducted involving such games indicate a positive increase in learning outcomes. Playing games has also shown to increase time-on-task, the time spent on school-related tasks (Ballwebber et al., 2014). This project was intended to fill the intersection of fun and productivity by creating a more enjoyable way to encourage children to practice and further their talent while still teaching or reinforcing reading skills.

The purpose of the program was to create a game for use in educational settings to teach and help young children practice reading music rhythms; and to evaluate the success or failure of the application.

Methods and Materials

A game was created in which the player progressed through levels by reading and playing music rhythms. Each level consists of a bar line where a combination of notes fill four measures of 4/4 time signature as shown in Figure 1.

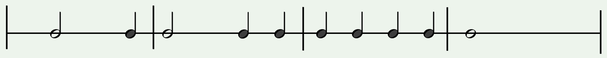


Figure 1: Four measures of notes in 4/4 time signature

The user controls a sprite, a fox, through tap actions at the correct time according to the notes on the bar line. Points are earned based on accuracy as a means of positively reinforcing the user's practice.

The game's design was first sketched out on paper and redesigned multiple times before being redrawn as digital files. All game objects and sprites were created in Unity Game Engine and all scripts were written in C#, using VS Code. The main framework of the game was built as multiple scenes where the user would begin at a start screen, enter to a level selection screen, and finally an actual level. A tutorial was created with the same workings as a standard level of the game for new players.

Methods and Materials (cont.)

A game play screen is shown in Figure 2. Each scene has main layers where background object were placed and the user interface (UI) layer which contained objects the user could interact with. Music notes and other interaction sounds were recorded and added into game scenes. The character sprite was also animated to walk during play.

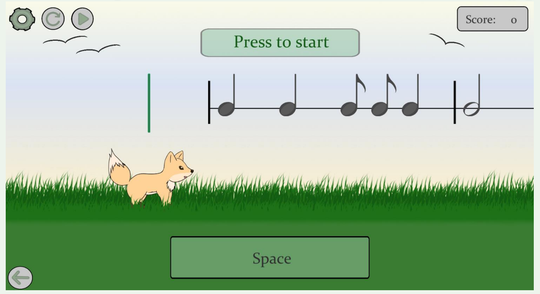


Figure 2: Game's generic level interface

The singleton and notification systems are object-oriented design techniques were used to provide a static instance. The singleton provided a static instance that would not be destroyed when new scenes were loaded; which allowed data to persist between scenes. Notifications allow C# classes to behave independently of each other, but still interact without the use of inheritance or polymorphism.

Upon starting a level, the code loaded in a different set of notes depending on the level. Within the code each note was created and assigned a unique ID which was referenced to change color depending on whether the user had correctly tapped and earned points. The player's score was visible in the upper right corner of the game and was shown again upon completing the level, as part of increasing player engagement.

A tutorial was created as a variation of the standard level framework to walk the user through the UI objects visible on the screen, explain the goal of the game, and show how to earn points. This was directed by a dialog box which was programmed to follow through a queue of instructions to guide the player through a level. Once complete the user could return to the main menu and begin playing other levels.

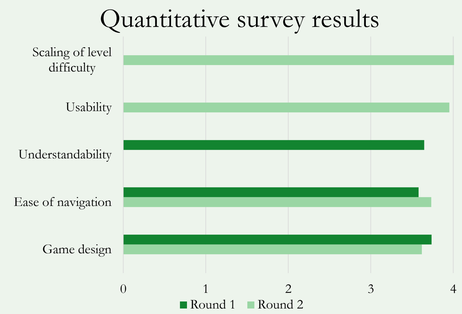
Each of the following levels were created by calling methods to procedurally generate notes. Each type of note was created as a prefab so that calling a method would cause the existing prefab object to be instantiated. These methods allowed different combinations of notes to be easily added or changed to fit the appropriate rhythm difficulty.

A sample of 19 sixth grade students participated in testing. In the first round of testing the students were asked to play through the music game's tutorial and then complete a survey with questions related to their experience. The feedback was used to develop the actual levels of the game and make fixes and changes. During the second round of testing the same students were asked to play through multiple levels and complete a second survey.

Results

All qualitative data compiled from the first and second rounds of testing was used to gather input on additions or changes to make to the game and to understand if certain factors, such as previous exposure and experience in music, would affect game experience. Some survey questions were only asked during one round of testing because the questions were not relevant to the other round. Scaling of level difficulty and usability could not be assessed untill the second round due to levels not existing in the first round. Understandability was not assessed a second time due the participants already having background knowledge on the goal and how to play from round 1 participation.

Qualitative feedback from round one, such as an interest in increased level difficulty, additional visual detail, and final score display were added to the final version.



Graph 1: Average student survey results on a scale of 0 (low) to 4 (high). Ease of navigation had an increase when comparing the first and second round of testing. The average rating for game design had a slight decrease.

Conclusion

The purpose of the project was to create a game for use in educational settings to teach and help young children practice reading music rhythms.

While quantitative ratings of the program did not all increase, the scores remained high. The majority of qualitative responses were positive as well. Program bugs mentioned in the first round's surveys were fixed and not mentioned again in the second round's feedback.

The game could be developed further by creating more rhythm levels, recording additional or better audio, and drawing a more consistent art style for the game. Players could also be given a choice of sprites. The game could have a wider range of difficulty with adjustability to allow students to filter based on their own ability.

References

Cassidy, G. G. & Paisley, A. M. J. M. (2013). Music-games: a case study of their impact. *Research Studies in Music Education, 35*(1), 119–138. doi.org/10.1177/1321103X13488032

O'Rourke, E., Haimovitz K., Ballweber, C., Dweck, C., & Popovic, Z. (2014). *CHI '14: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM Digital Library. doi/10.1145/2556288.2557157