Supporting Users of the Wearable Learning Cloud Platform through Improved Usability

A Major Qualifying Project (MQP) submitted to the Faculty of
WORCESTER POLYTECHNIC INSTITUTE
in partial fulfillment of the requirements for the degrees of Bachelor of Science in
Computer Science and Bachelor of Science in Psychological Science

By
Grace Seiche

Date:
9 December 2019

Submitted to:

Professor Ivon Arroyo

Abstract

The Wearable Learning Cloud Platform (WLCP) is a web based software for creating and playing multiplayer games designed as finite state machines. The goal of this project was to assess how to better support the users of the WLCP by conducting two studies about using the program and by making programmatic changes to the software to address feedback received in the studies. While making games in the WLCP, participants were asked to complete surveys about their experiences. The analysis from this data showed that the program ranks below average in terms of usability and seemed confusing and complex to many participants. To address these results, WLCP developers should implement changes to address the concerns of the study participants, add a more robust tutorial or help system to address more nuanced functionality, and continue to reassess usability to monitor progress.

Table of Contents

1. Introduction

Creating well-designed, easy-to-use software that appeals to consumers is one of the biggest challenges facing any software developer. There is an overabundance of software and websites that fail to gain popularity among users because they are not easy to use or do not meet the needs of the consumers. It is crucial for a software developer to collect and analyze information about their users' experiences in order to improve their program and create a more successful product.

The Wearable Learning Cloud Platform (WLCP) is a program created at Worcester Polytechnic Institute under the direction of Professor Ivon Arroyo. The platform enables users to create and play mobile multiplayer games. Rather than having to program games with traditional programming languages such as Javascript or C++, users create finite state machine representations of their games in the WLCP's game editor. These diagrams are then automatically transpiled and can be played on any device that can connect to the internet.

The original goal for the software was to be "a novel infrastructure, which allows for the creation of a myriad of interactive embodied learning experiences for students of all ages" (Cerruti et al, 2015). As such, it has been used in many studies to evaluate learning gains, embodied learning, and game creation with a variety of different participants. Recently, the WLCP has also been the focus of workshops with mathematics and computer science teachers in order to make the tool available to a larger population of users.

Since the WLCP is being used in research studies, the software must work well and be reliable for the study participants. Negative experiences using the platform or bugs in the software can skew a participant's opinion of their experience in the study and can also confound

results of the studies causing problems for researchers trying to draw conclusions. It is critical to evaluate and fix any weaknesses in the system so that research results are not negatively impacted by the WLCP.

Additionally, as the Wearable Learning Cloud Platform grows beyond its usage in research studies led by professional users, it is important to evaluate and improve the user experience with the program, so that those without previous experience can successfully create games and activities with the WLCP. A software that must be taught in person is highly limited in its scalability. Improving usability and adding resources so that users can self-sufficiently learn how to use the platform will increase its potential for growth.

To address these concerns, I attempted to answer the question "How can we better support the users of the Wearable Learning Cloud Platform?" through research, a series of studies, and programmatic improvements to the platform. I broke this research question down into more specific questions that fall into the subcategories of usability, training, and features. The research questions are as follows:

Regarding Usability:

1. How usable is the WLCP currently?
2. How can we make the WLCP more usable?

Regarding Training:

3. How helpful is/would a tutorial (be)?
4. How could we lessen the learning curve of the WLCP?

Regarding Features in the WLCP:

5. What features are or are not easy and intuitive?

6. What features can we implement or improve to ease the user experience?

For this project, I focused exclusively on the game editor as this is the part of the software where users spend majority of their time.

In order to answer these research questions, I performed an initial study based in the US focused on supporting users of the WLCP. The participants were undergraduate and graduate students enrolled in an embodied cognition psychology class to get a more mature perspective of the platform's challenges. Then, based on the results and feedback from the study, my colleagues and I implements changes to the WLCP infrastructure to mitigate some of the issues discussed by the initial set of participants. After these changes were integrated into the system, I performed another study about supporting software users. This time, however, the study was based in Argentina and involved six classes of sixth and seventh graders to gain user experience information from a different perspective. The findings from this entire process were evaluated, summarized, and provided to the team of researchers and programmers working on the Wearable Learning Cloud Platform in order to better inform their future work.

2. Background

2.1 History of the Wearable Learning Cloud Platform (WLCP)

The Wearable Learning Cloud Platform was created out of an interest in combining wearable technology with mathematics games in order to create an embodied learning experience as a fun, effective, alternative way to learn and internalize mathematics topics. The first iteration of wearable learning technology was a CyberHoodie (pictured below in Figure 1) which consisted of a zip of sweatshirt with electronic devices sewn into the clothing. These electronics included "the Arduino, sewn onto a patch on the back of the CyberHoodie along with the battery, relay, wiring, LED, clue button and hint button" (Rountree, 2015). The students used the electronic devices to aid them during the Math Scavenger Hunt game they were playing.

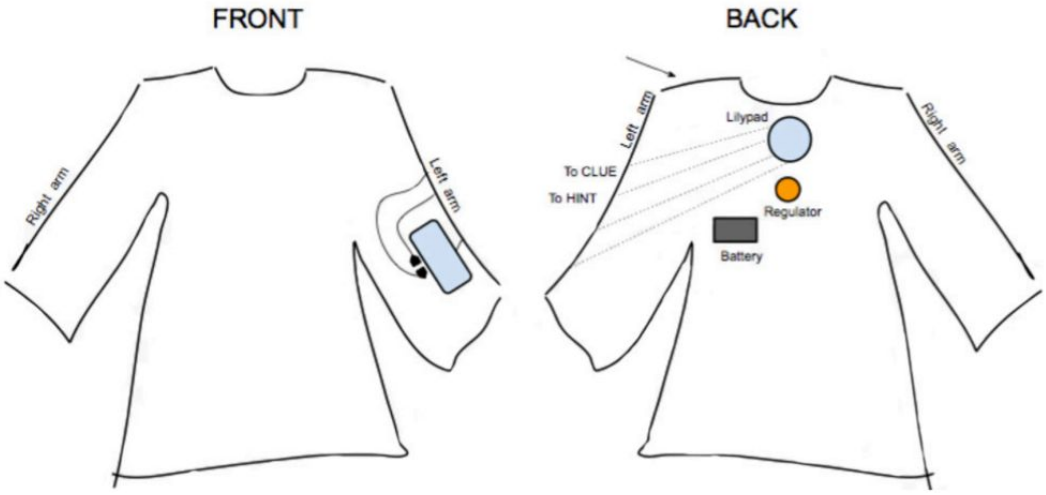


Figure 1. Design of the first CyberHoodies prototype

The next iteration of the wearable technology took the form of a CyberWatch similar to a more modern smart watch because it was "more appealing, safer, and less prone to damage" than the CyberHoodies (Rountree, 2015). This iteration was completed by a team of Worcester

Polytechnic Institute undergraduate students completing their Interactive Qualifying Project (IQP). The first prototype of the CyberWatch simply converted the technology on the CyberHoodies to a single smaller component that could be attached to a player's wrist. The second prototype (shown in Figure 2 below) was modified to work with a new, more complicated game called Estimate It! This game required a server-client interaction, so in addition to redesigning the hardware, the students also created a server and a portal to interface with the server in order to create and play the games (Rountree, 2015).

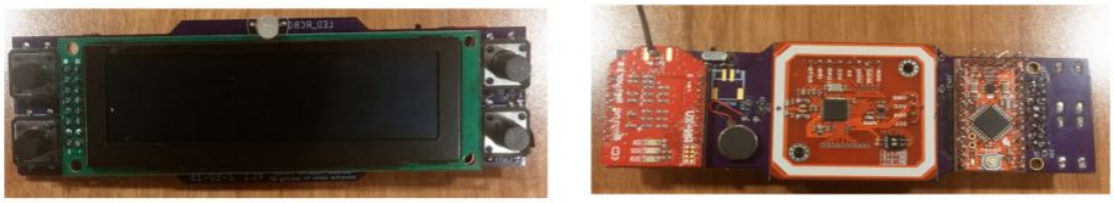


Figure 2. Front (left) and back (right) views of the second CyberWatch prototype

After the IQP project had finished, there were still large limitations to the technology, the largest being that the implementation was not completed and lacked a usable graphical user interface for creating the games to be played with the CyberWatches. Additionally, the software was written in Ruby on Rails, which has limited compatibility on Windows operating systems, where most game facilitators would want to deploy their games. Finally, the system and database were not stable, so only one game could be played at a time and the database had to be wiped between games (Micciolo, 2017). These areas for improvement were the cause of the next iteration of the software, created initially by Matt Micciolo as a Major Qualifying Project.

During MQP rewrite of the software (known as the Wearable Games Engine), the database, frontend, and backend were all overhauled. Even though the database for the IQP iteration was PostgreSQL, the new database was created with MySQL, "due to its large

popularity and large third party support" (Micciolo, 2017). The database was written as relational database with tables for information about teachers, students, games, game states, transitions, game instances, etc. The frontend of the software was completely rewritten to use the Tomcat 7 Web Server with Java Server Faces (JSF) for easy website deployment and interaction with the MySQL database. It was implemented using the model-view-controller design pattern to clearly distinguish the data, the user interface, and their interaction. The frontend was a simple teacher panel for managing students and classes, creating and editing games, and controlling instances of the game being played. Another key component of the frontend rewrite was the creation of a virtual device page in order to be able to mimic the activities and display of the CyberWatches for playing and testing games from a web browser without needing to use the hardware. The final component of the Wearable Games Engine project was the backend rewrite. The backend was converted to a Java server that could be run locally or remotely in order to increase compatibility and allow more users to access the software. It was set up modularly with a ModuleManager singleton to control the logger, server, settings, task manager, and event manager. The backend also implemented different packet types for transmission to and from the server (Micciolo, 2017).

The MQP rewrite was meant to mimic the structure of traditional learning software like MathSpring that are based on teachers, classes, and measuring learning objectives. However, after completing this iteration, the developers started rethinking the program's use cases and future directions. Through this process, they decided that this concept was something completely unique and it was not practical or feasible to try to fit it into a traditional learning software

structure. Thus began the most recent redesign and creation of the Wearable Learning Cloud Platform (WLCP).

The WLCP was incorporated elements from the Wearable Games Engine but was largely created from scratch. Unlike the Games Engine, the WLCP is completely web and cloud based so it can be accessed from any device that can connect to the internet and it does not require installation of any additional software (Micciolo, 2018). Additionally, this rewrite does not support interfacing with the CyberWatches and intends for games to be played on cell phones or other mobile devices. These initial design changes were implemented to increase usability and scalability. The program architecture was also completely redesigned. Figure 3 illustrates the main aspects of the new architecture as well as their internal and external interactions.

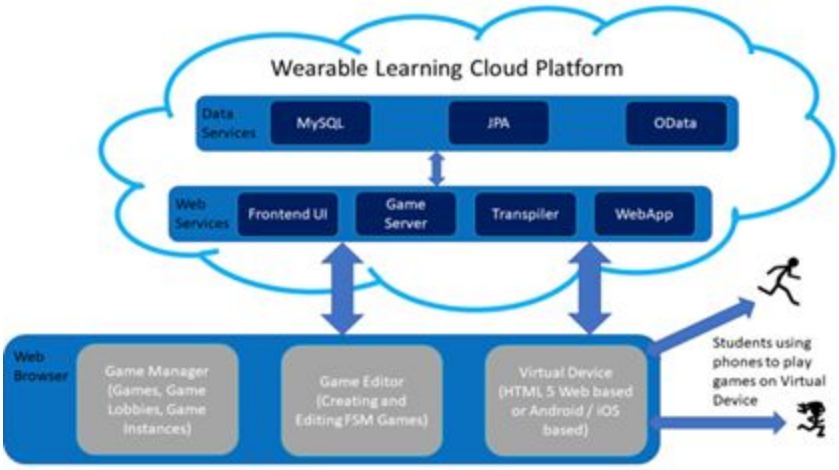


Figure 3. Architecture and interactions of the WLCP

In the new architecture, the MySQL database is managed and generated by Java Persistence API annotations on the classes and members of the OData data models. The front end is a HTML5 and JavaScript based web user interface that uses the SAP OPENUI5 framework to leverage the modern look, rich data binding, and model-view-controller design pattern. This

rewrite also includes a SPRING based web app, a JavaScript transpiler to convert game designs into executable forms, and a multithreaded asynchronous TCP socket game server to handle playing multiple games, instances, and players simultaneously (Micciolo 2018).

The largest change in this most recent iteration is the creation of three game modes: game editor, game manager, and player. Rather than having a teacher portal to manage students/players and create games, when logging into the system, all users have the option to create/edit games, start/manage a game instance (to be played), or play a game someone else is running. In game player mode, the users are taken through the steps of the game they are playing. Each page has some text to read and then a way to change to the next screen either by pressing a button, entering a color code, or filling a text box. This is very similar to the virtual device page created in the MQP rewrite. The main function of the game manager is to be able to create and delete game instances that players can join in order to play the games. Users can also create new users through the game manager. Currently there is a placeholder for a dashboard with information about the games currently running, the server status, and resource usage, but the backend for this has not been implemented. In the games section of the game manager, users can select games to open and modify in the game editor.

The final and most robust mode of the WLCP is the game editor. In the game editor, users create new games and edit existing games. All games are created in the form of finite state machines. Each state is a snippet of text that will appear on a screen in the game. The states are connected with arrows. Attaching a transition to a connection creates a way for the player to move to the next screen (state). While states can only be text, there are three different options for transitions: click a single button, enter a color code, or enter text in a text box. In each state and

transition, the editor can also modify the scope to determine which player(s) or team(s) can see which information. In one state, each play could see a different screen and then have to enter a different transition in order to continue. After creating a finite state machine of their game, editors can save their games and debug them. The debugger will open an instance of the player mode, so the editor can impersonate a player and verify that the game progresses as expected. A portion of a finite state machine created in the WLCP for the Tangrams Race game is shown in Figure 4.

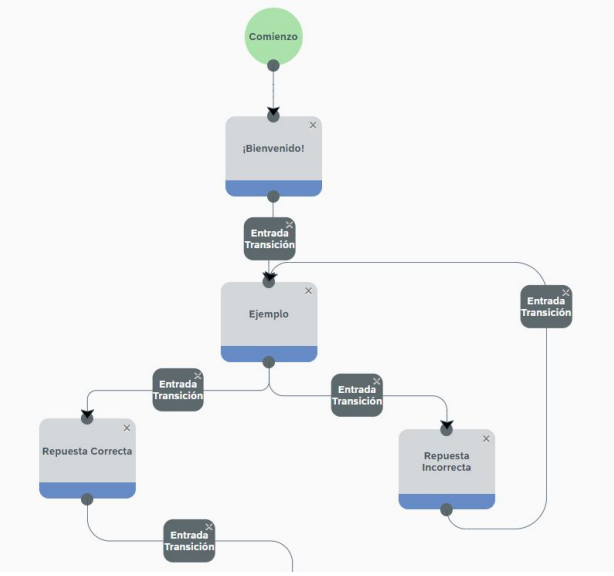


Figure 4. Finite state machine snippet for Tangrams Race

The Wearable Learning Cloud Platform as described above is the most recent version on which the researchers were developing and using in their studies. The creation of an intuitive and visual game editor accessible by all users opened up the field to run studies about computational thinking, finite state machine programs, and the types of games that students create. A selection of those studies are described in the following section.

2.2 Spring 2018 WLCP High School Study

The most robust study of the WLCP was performed with eighteen 11th and 12th graders from The Massachusetts Academy of Math and Science. Some goals of this study were to determine whether creating games with Finite State Machines (FSMs) in the WLCP improves knowledge of FSMs, whether the system would be usable for k-12 students, and whether the WLCP is easy to use and user friendly (Micciolo, 2018). The participants met with the researchers once a week after school for six weeks to complete the study.

During the first class period, students were given a pretest on computational thinking and FSMs. They also played EstimateIt!, a game created in the WLCP, so they could see the capabilities of the system. When playing the game, there was a group of students playing the game and a group of students observing the game play. After playing the game, the students and researchers discussed what happened in the game and what they observed.

During the second class period, participants were put into groups of 3-4 and asked to design their own math games. They were given paper pads to write on and an instructions sheet. The instructions stated that the game needed to incorporate learning math, physical movements (ideally related to the mathematics concepts they chose), and use of cell phones. To keep the game simple, the participants were told to have a maximum of four players and two teams. Participants designed their games on the paper pads they were given and at the end of class, if there was enough time, they quickly presented their games to the other groups.

The third day involved continuing the design of their games, but this time in the form of finite state machines. The class started with a presentation about what a FSM is and how to draw one. Participants were shown examples of FSMs from other study participants from previous

years. The presentation also listed some of the restrictions on the games, such as what kinds of states and transitions the WLCP enables. Then, the students spent the class period converting the descriptions of their games from last class to the finite state machine form.

Days four and five were dedicated to creating their games in the WLCP. Since WLCP games are programmed with FSMs, the participants based their programming off the diagrams they made in the previous class. The researchers gave each group a username and password to log into the system with. They also gave a brief demonstration of how to use the system and perform tasks such as creating a new game, adding and editing states and transitions, and debugging. The participants worked in their groups for the remainder of class four and five to program their games.

Class six was the final class. Participants began with more debugging of their games. Then they were given the chance to play their own games and other teams' games on cell phones, so they were able to see their games in action and better debug their games. The final activity was to complete the computational thinking post test. In addition to the procedure outlined above, the students were also asked to complete short homework assignments which mostly consisted of updating the folder where they were storing copies of their designs and FSMs as well as writing reflections about how the class went, in their opinion.

Two main measures of interest are the gains in computational thinking measured in the pretest and posttest and the opinions survey from the participants about the usability of the WLCP. The results of the pretest and posttest compared improvement in computational thinking from the beginning to the end of the study. This is based on the responses of the nine participants that completed both the pretest and posttest. The posttest was a homework assignment after the

last class, so many students did not complete it. However, those that did complete both tests showed an overall average improvement of 17% across the first three questions, which was shown to be a statistically significant difference (Micciolo, 2018).

Another measure from this experiment was the survey that all participants completed during the last class about their opinions on the WLCP. Most of the participants said that they liked that the WLCP was simple and easy to use. They also liked the aesthetic and that they did not need prior programming experience. Most participants did not like the limited functionality of the system, such as the constraints to using button presses and the inability to add variables. There were also some frustrating bugs in the system. The feature that participants said they wanted included variables, expressions, sensors, and copy/paste (Micciolo, 2018).

Additionally, the participants were asked whether they believed middle school students would be able to use the WLCP's Game Editor successfully. Overall, the results were positive; 61.5% of respondents said yes while the rest said maybe or gave conditions for their success. For example, one participant said that younger students would need more error logging while another claimed it would depend on the school or program (Micciolo, 2018). All of these results informed future ventures in the WLCP to improve the functionality and run more studies.

2.3 Supporting Users through Usability

Usability is one of the most critical aspects for all products. The goal of most products is to be used. If a product is not usable, consumers will not want to use it. There are an infinite number of products to be used by consumers. When a product seems too confusing or a user encounters some difficulty, the user will most likely move on to the next product rather than fighting with the one they are currently using. This is especially true with web applications, where there are so many sites to perform the same activities. Better usability can lead to better success. This seems like a simple concept, but it can be difficult to create a usable product. It is critical for designers and developers to research and test the usability of their product in order to make improvements and a better overall product.

There are many qualities of usability. According to Neilson, the five main components of usability are learnability, efficiency, memorability, errors, and satisfaction (2012). Learnability is how easy it is for users to figure out how to use the system for the first time. Efficiency is important because users want to be able to do things fast; too many clicks or roundabout steps can cause a user to leave. Memorability is whether a user is going to remember the product and use it again. Errors means how easy is it to make a mistake and how does the product support users to recover from their mistakes. Finally, satisfaction is about whether using the product was an enjoyable experience that the user would want to do again.

Quesenbury has also developed a list of the five main components of usability. She lists efficiency, error tolerance, easiness to learn, and engagingness, which correspond directly with Neilson's components of efficiency, errors, learnability, and satisfaction, respectively. They differ on their final component. Rather than memorability, Quesenbury cites effectiveness, how

successfully and accurately users can reach their goals (2001). As a whole, these components are meant to be used by researchers to break down the concept of usability into more than just "ease of use" to form a more detailed perspective on a product's usability.

When researching a product's usability, it is important to break down the concept into smaller categories as Quesenbury and Neilsen did. From there, each of these qualities can be researched, measured, and improved upon to increase the usability of a product. There are many ways to collect data about usability, such as user testing, focus groups, and user surveys. Giving research participants a goal, letting them use the software to try to achieve their goal, observing their actions, and then allowing them to honestly reflect on their experience is one of the most simple and effective ways to learn about the user's perspective and gain insight into areas of improvement.

Since society, technology, and user perspective are always adapting and changing, evaluating and improving usability is a cyclical experience. It starts with a question, then some user research, analysis, and finally some improvements to the product. Then the cycle starts all over again. It is critical for a product's success to be constantly improving and becoming more usable to attract new users.

2.3.1 System Usability Scale: A measure of usability

The System Usability Scale (SUS) is a tool for measuring the overall perceived usability of a system, specifically a piece of software or a website. It was created by John Brooke in 1986 as a "quick and dirty" way to assess usability and has since become an industry standard. It has been proven to be valid and reliable for both large and small sample populations. The SUS is a

10 question survey where respondents respond either strongly disagree (1), disagree (2), neutral (3), agree (4), and strongly agree (5). The questions are listed below (Sauro, 2011).

1. I think that I would like to use this system frequently.
2. I found the system unnecessarily complex.
3. I thought the system was easy to use.
4. I think that I would need the support of a technical person to be able to use this system.
5. I found the various functions in this system were well integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.
9. I felt very confident using the system.
10. I needed to learn a lot of things before I could get going with this system.

To score the SUS results, change all of the verbal responses to numerical responses. Then subtract one from all the odd question responses. Subtract all the even numbered question responses from five. Sum up all the converted scores and multiply the total by 2.5. This will produce a single numerical value out of a possible 100 that represents the system's perceived usability.

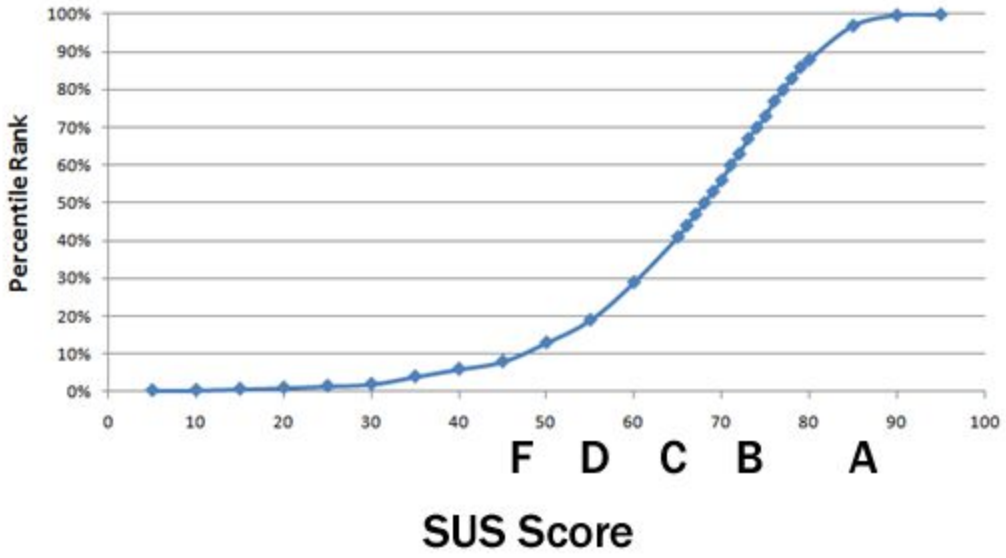


Figure 5. Percentile ranks of SUS scores (Sauro, 2011)

The average usability score is 68 based on 500 usability studies. This score however should not be interpreted as the percent usable. Instead, use Figure 5 above to determine the percentile of the score as a method of comparison (Sauro, 2011). An SUS score is not meant to be a diagnosis of what parts of a system needs improving as it does not ask or report on system specific details. It can, however, be used as a benchmark to measure progress as the system improves and to measure against other systems.

3. Study 1

3.1 Methodology

3.1.1 Participants

Nine participants from the Embodied Cognition class at Worcester Polytechnic Institute participated in an abbreviated WLCP workshop and were surveyed. They varied from college undergraduate freshmen to graduate students continuing their education after working for 20+ years. The pool included 2 males and 7 females. Most participants had no previous experience with the Wearable Learning Cloud Platform (WLCP), but some had more experience and had previously administered studies using the WLCP. No one was an expert user of the system.

3.1.2 Procedures

On Tuesday 2/5/18, ten participants spent about 20 minutes creating an embodied game that used cell phones. Students were split into groups of 2 or 3 and given the instructions found in Appendix A.

The students drew the games that they created on whiteboards in the classroom. Those with more knowledge of the system were instructed to act as if they did not know about the study or the end goal. After the designing stage, students presented their games to the class.

On Friday 2/8/18, nine participants were given a fifteen minute presentation about finite state machines and the Wearable Learning Cloud Platform. The presentation can be found in Appendix B. The presentation focused heavily on slides 4 through 9, with emphasis on 4, 6, and 8. It also included a brief introduction to how to use the Game Editor of the WLCP, spending 2 minutes showing participants how to make a new game, add states, and add transitions.

After the presentation, participants got into the same groups that they were working with on Tuesday to try to adapt their game to a finite state machine and then the WLCP. The participants spent 15 minutes trying to adapt their games to a finite state machine either on paper or on a whiteboard.

The participants were then told to transition into creating their game through the WLCP. They were instructed to transfer their games from the written finite state machines form to the finite state machine builder in the game editor. Some teams switched the programming role among team members half way through, while others kept the same team member as a programmer for the entire experience.

Following 15 minutes of game building, the participants were given a 10 minute survey about their experience, which served as the data collection method. The survey is located in Appendix C. After finishing the survey, the participants discussed as a group the struggles that they encountered with the process, especially regarding the game editor. The researcher took notes during this feedback session.

3.1.3 Materials

Each participant was provided with whiteboard space and markers. They were each asked to bring their own laptops on which they could access and use the WLCP. All the instructional materials were created by the faculty member running the workshop.

3.1.4 Measures

The data for this survey was collected through the survey that participants completed at the end of the experience. This survey is located in Appendix C.The first section consisted of the System Usability Scale (SUS), which has a strong backing and is a common way to calculate the

usability of software. This was followed by an evaluation of the difficulty of various actions in the game editor, where participants were asked to rate the difficulty from 1 (very easy) to 5 (very hard). The second half of the survey was more qualitative and will be coded by hand by the researcher. It asked open ended questions in order to understand the more specific individual-level strengths and weaknesses of the program. Finally, there were three demographic questions to control for factors like previous experience with the program.

3.1.5 Methods of Analysis

The SUS section of the survey was analyzed at an individual question level and a whole score level. First, all question responses had to be normalized to a 0-4 scale where 0 is most negative and 4 is most positive. Each score of the odd numbered questions was reduced by 1, but since the odd numbered statements reflect positive usability of the system and the even statements reflect negative usability of the system, each score for the even numbered statements was normalized by subtracting the score from 5. By normalizing the results, all scores now reflected 4 as the best score possible and 0 as the worst score possible and allowed for analyzing which questions received the most negative responses.

The normalized scores were graphed in a stacked bar chart to better visualize how responses leaned for each question. It also helped to identify the questions with the greatest amount of positive responses and negative responses. To calculate the total SUS score, the normalized responses for each participant were summed and multiplied by 2.5 to create a score out of 100. Averaging SUS scores given by each participant gave the system an overall score that was then compared to national standards.

For the questions about the easy of performing various actions in the WLCP, the responses were graphed in a stacked bar chart similar to the one created for the SUS scale, again to better visualize how the participants generally felt about the usability of certain features. The open ended questions that followed were hand coded by the researcher. First categories and themes in the responses were listed out and then the categories mentioned in each response were noted. Then number of mentions for each category in each question were reported. The final section about demographics were calculated at percentages to describe the study pool.

3.2 Results

3.2.1 System Usability Scale

Chart 2 below shows the normalized responses to the SUS questions. The darkest red sections represent the most negative responses while the lightest green sections represent the most positive responses.

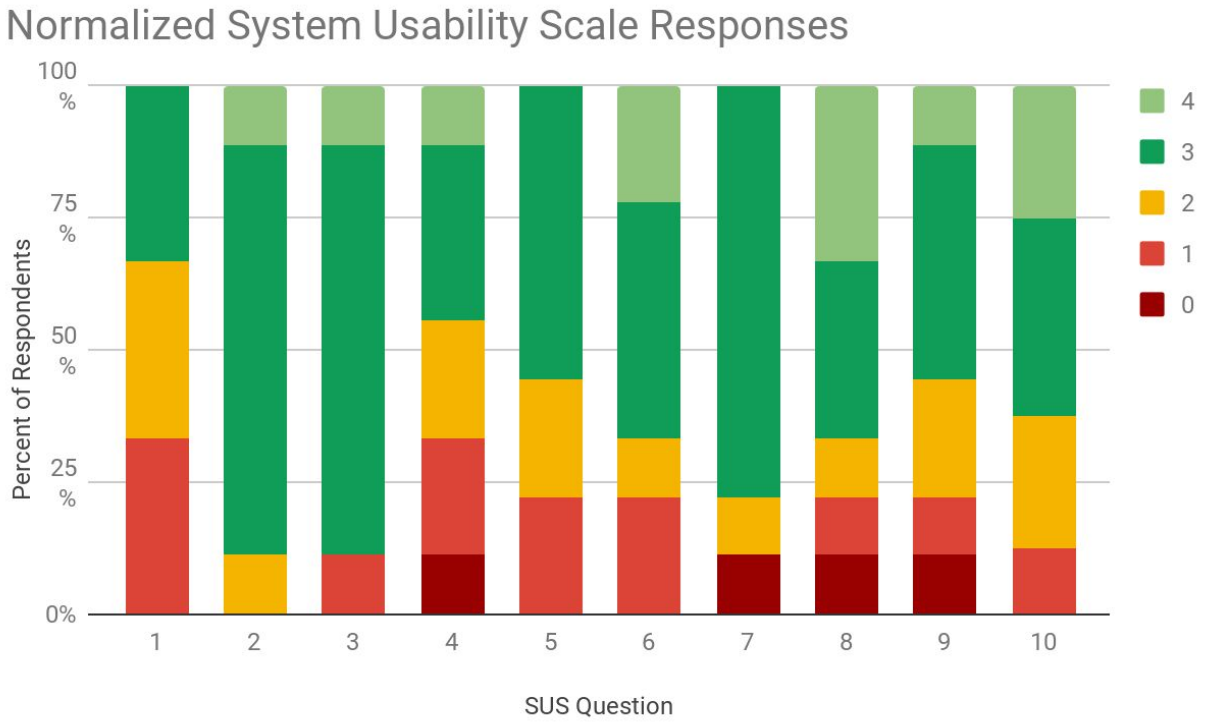


Figure 6. The normalized usability for the Study 1 SUS results.

From Figure 6 we can see that most respondents indicated that the system had high usability with 8 of the 10 statements showing over 50% agreement (responses of a normalized 3 or 4). The statement least agreed with is the first statement which says "I think that I would like to use this system frequently." The statements with the most normalized agreement were statements two and three. Statement two says "I found the system unnecessarily complex" to which 88.9% of respondents either disagreed or strongly disagreed. Statement three says "I thought the system was easy to use" to which 88.9% of respondents either agreed or strongly agreed. Additionally, the SUS allowed the researcher to calculate the overall usability of the system as a whole. The WLCP earned an overall mean score of 63.3.

3.2.2 Ease of Completing Tasks

The second section of the survey asked about how easy performing certain tasks were. The results from the survey are shown in Chart 3 below.

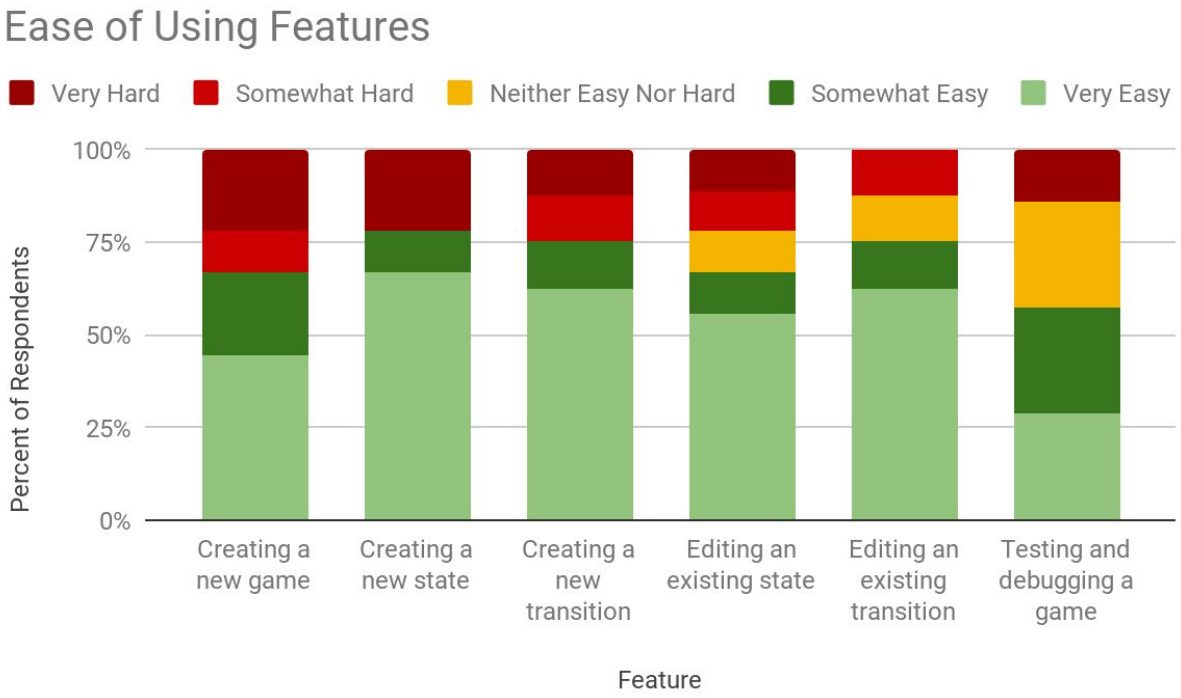


Chart 3. Ease of completing tasks during the workshop

At least 50% of respondents rated each task either somewhat easy or very easy. The easiest tasks were creating a new state and creating a new transition, while over 25% of respondents claimed that creating a new game was either somewhat or very hard.

3.2.3 Open Ended Responses

The third section of the survey asked 4 open ended questions about how the participants felt about the system. These questions are available in Appendix C. In response to the first

question about what participants like about the WLCP, seven of the nine respondents mentioned how easy or simple the interface was to use and three people talked about how visually well designed the system appeared. When talking about what they did not like, two of the six people who answered this question referred to technical difficulties they experienced due to bugs in the system and three people discussed how the system's options and features were too limited. On the topic of desired features, three people wanted to be able to use pictures while creating games, three requested a tutorial to better understand how to use the software, and two requested the ability to add hyperlinks that students could click on. Additionally, one person mentioned having the ability to separate the public games from the user's games. Finally, respondents were asked about the difficulties they faced. Eight of the nine respondents stated that their difficulties stemmed from technical issues they experienced and the last respondent stated that lack of options was the reason for his/her difficulties.

3.2.4 Group Discussion

From the group discussion we learned that in addition to the results above, the participants would have liked to be able to add images to states, copy and paste sections of their finite state machines in the game editor, have the option to include hyperlinks in text, make math text in text inputs, include variables in the games, change the order in which the questions were presented to the players, have groups of different sizes, and add interaction with the phones' GPS systems. The participants experienced bugs related to not being able to find the input text box option in states, using large group sizes, editing input states, and adding and editing transitions.

3.3 Discussion

3.3.1 System Usability Scale

The WLCP scored a 63.3 out of 100 on the SUS. Compared to an industry average of 68, this is in approximately the 33rd percentile and considered to be a C- on the typical A through F grading scale. This suggests that it is important to keep improving the usability of this product and there is significant room for growth. Looking at the results, the developers should work towards removing some of the technical layers that make the program difficult to use without prior instruction. Making the product more accessible to everyone and all levels of experience is one way to work towards improving the usability of the system to meet the industry average.

3.3.2 Ease of Completing Tasks

The data suggests mostly easy ability to perform the basic tasks required for the Game Editor of the WLCP. However, this data may not be completely accurate. Some markings on the paper response sheets indicate that a few respondents may have misinterpreted the scale as 5 being the easiest instead of 1. This could be due to the fact that in the previous section, 5 meant strongly agree which is the most positive reaction while in the ease of task section, 5 meant very hard which was the most negative reaction. Some of the data points could be flipped. Due to the small sample size and the questions on the accurate interpretation of the survey, this data may not be a reliable depiction of people's opinions on the ease of use of the system's features.

Since there is doubt about the accuracy of the survey results, conclusions and recommendations cannot be formed from these results. It is important to repeat this study and

survey with clarification on how to answer the questions, in order to gather accurate data on the ease of performing the basic functions of the game editor. This is one limitation of the study.

3.3.3 Open Ended Questions and Group Discussion

The open ended questions and group discussion provided much more direction for the developer on how to improve the system. The next iteration in development should do the following

1. Fix bugs that presented themselves during the study
2. Create pop up messages for the user in situations where the system is behaving as desired but it appears to the user to be a bug
3. Add more options and features such as
    a. Adding pictures to states
    b. A click through tutorial to orient new users
    c. Adding hyperlinks to states
4. Separating private from public games.
5. Comments/Feedback that they made during discussion
    a. Variables
    b. Randomization
    c. Copy/paste sections of charts

These were all specifically requested areas of improvement for the system that would improve the user experience and overall usability of the WLCP.

4. Programmatic Changes

After the first study was completed, I presented the list of recommendations discussed in the previous section to the team of programmers working on developing the WLCP. We worked diligently to fix bugs and implement features to improve user experience before the next study began. The following sections give an overview of the changes that were implemented between the first study (conducted in February 2019) and the second study (conducted in August through October 2019).

4.1 Added Features

The two most prominent changes to the WLCP between Study 1 and Study 2 were the additions of localization and showing pictures in states. Localization was a tactical change before starting the second study. Since the next study was performed in Argentina and was conducted completely in Spanish, developers had to implement i18n localization and translate the program so the WLCP would support both English and Spanish based on the users' browser settings.

Adding pictures to states was part of another IQP that involved measuring the impact of visual hints. Users can now add pictures in addition to text to the states in a game by opening a state, adding a true link to an image, resizing the image, and saving the state. When players reach those states in the game, the image will show up on the screen along with the text for the state. Not only did this improvement aid in the visual hints study, it also added a functionality that users requested in the initial study.

Additionally, many minor changes were made to the game editor. They are listed below.

1. Added a warning when refreshing or leaving the page in game editor or player mode
2. Changed the appearance of the start state to be distinct from other states since it does not function the same as the other states
3. Added ability to copy, rename, and delete games
4. Separated personal and public games when loading an existing game
5. Prevented users from editing games they do not own
6. Enabled switching modes from game editor
7. Added click and drag scrolling on the canvas
8. Add a warning message for removing a connection

4.2 Bug Fixes

There were also a series of bugs that were fixed between the two studies. During the first study, there were many bugs or perceived bugs that negatively impacted the participants' experiences and contributed to poor rating of the software. The bugs fixed are listed below.

1. Double clicking on button in the player mode no longer unintentionally moves the player through multiple states
2. Increased state character limit from 255 to 2048 so users do not run out of characters when programming their states
3. Fixed connection issues when reloading the page while in game player mode
4. Added more restrictions and validation on the number of players and teams so users cannot create non transpilable games

5. Fixed bug where new, load, save, run & debug, and options disappears when the window is too small
6. Fixed security vulnerabilities

4.3 My Changes

The changes outlined in the previous sections were completed by other members of the team, but in this section I will describe the changes that I personally implemented in the game editor. The first change that I implemented was adding a couple pop up messages to the game to add clarity and reduce confusion. The purpose of the start state has been on source of confusion for WLCP users because it is not editable. I added the following message in a pop up window when a user double clicks the start state to try to edit it: “This is where your players will enter the game. Draw an arrow to the first state you want them to see!” The goal of this message is to give the users direction when they try to edit the start state and realize that they cannot.

The next set of pop up messages that I implemented were for editing states and transitions. Running in the background of the game editor is the validation engine, which is constantly checking that the game can be transpiled and preventing users from creating games that cannot be transpiled. This prevention is usually in the form of not allowing a user to edit a state or transition because there are other empty states/transitions that need to be filled first or that have already filled the scopes. Previously, when this occurred, the editor pop up would still show up, but there would be a big blank box instead of the editing tools. This behavior is shown in Figure 7.

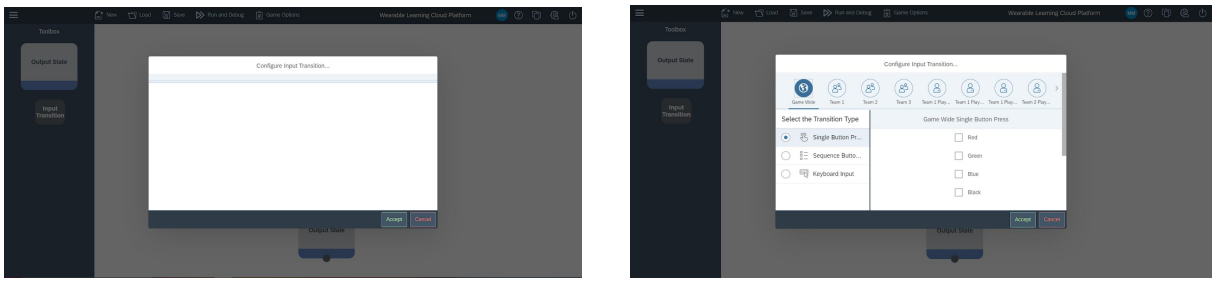


Figure 7. Transition editor when the game fails (left) and passes (right) validation

Users participating in studies often announced to the researchers that they had found a bug and the software was broken when in reality this was the desired behavior. To fix this, I added pop up error messages that the users would see in place of the blank editor. These messages remove the misconception that this was a bug in the software, and they provide the users with possible action steps they can take to get past the errors. When trying to edit a state while all prior states are empty, the users see a message that says "All of the input states are empty. Fill in at least one input state to edit this one!" When trying to edit a transition while the previous state is empty, users see this message: "Fill in the state above to edit this transition." Finally, when a user tries to edit a state that does not have available scopes, they see the following message: "All players and teams have been assigned in neighboring states. Do you really need this state? Who do you want to see this state? Check the neighbor states to see what those players are seeing." The goal for adding these messages is to support the users and give them direction when they try to program something invalid, rather than making it appear that there is a bug in the software.

The next feature I implemented was a QuickStart tutorial. Previously, there was no tutorial and all help for the user was provided through an in-person demonstration. However, as the software grows, it is important to create a way to learn the software without a demonstration

from a previous user. This is the reason for creating the QuickStart. It is not meant to be a complete tutorial, but rather just help users get started when they first open the software. The QuickStart is a pop up window that appears every time at login. The first page of the QuickStart is shown in Figure 8 below.

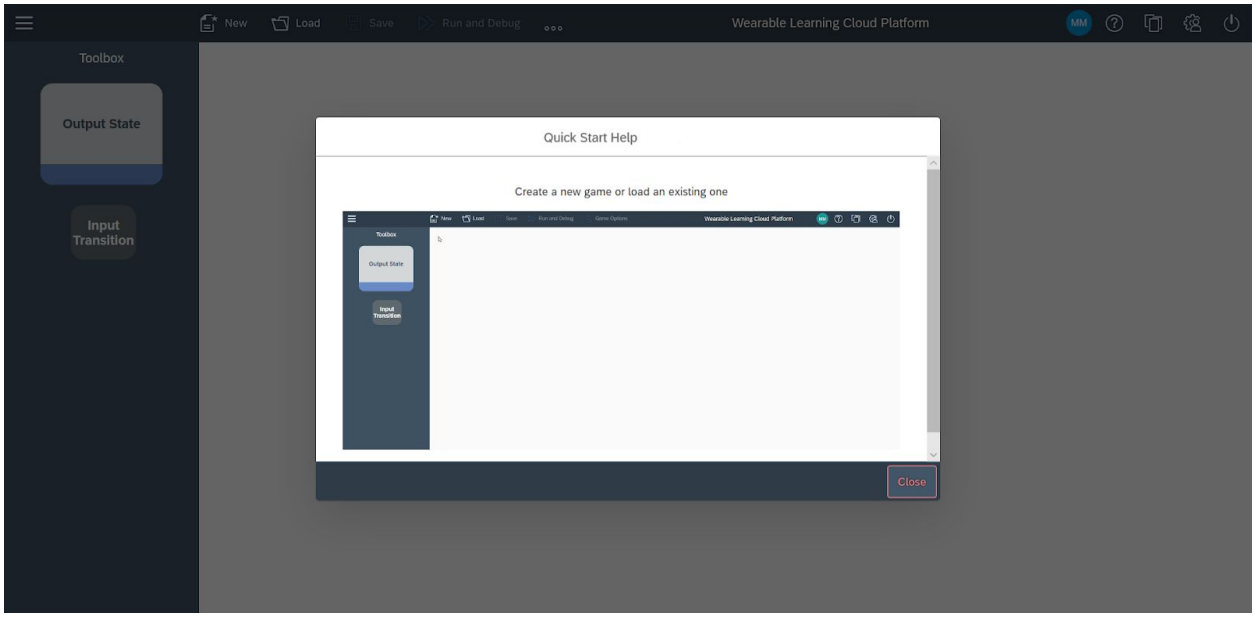


Figure 8. First page of the QuickStart tutorial

It does have the capacity to be cookie enabled to reduce the frequency of appearance, but that has not been enabled. In the window, the users see a description of an action they can take in the game editor along with a video of that action. There are seven screens that users can click through and the text descriptions are listed below.

1. Create a new game or load an existing one
2. Drag an output state onto the canvas
3. Drag an arrow from state to state and add a transition
4. Double click states and transitions to edit them
5. Don't forget to save your game

6. Run and debug to test your game before playing
7. Use these buttons to relaunch the tutorial, change game modes, or log out

Additionally, experienced users can click out of the tutorial at any time by selecting the close button. To reference the QuickStart again, users can click the question mark button in the upper right corner of the game editor. The goal of this feature was to provide a prototype for a tutorial or help section and then get feedback before fleshing it out and making a more comprehensive tutorial.

The final improvement that I made to the WLCP game editor between the two studies was expanding translation of the game editor. Internationalization (i18n) and localization is the process of adapting a piece of software to be functional and accessible cross culturally. A large part of this process is enabling translations of not just text but also images, graphics, and cultural references. My colleagues implemented initial infrastructure for the i18n localization in heavily used features. I expanded the localization and translation to cover the entire game editor. This was especially critical for all of the error messages, so participants in the Argentina study could problem solve autonomously. Completing the localization also required translating the QuickStart tutorial and its videos so that the videos in the tutorial show the game editor's user interface in a language that corresponds to the language they have loaded the website with. The goal of completing the game editor translation was to remove language barrier as a confounding variable in the second study and to make the software more accessible to non native English speakers.

5. Study 2

5.1 Methodology

5.1.1 Participants

A total of 143 participants participated in a seven week game development workshop in Argentina and completed both study surveys. The participants were 6 different classes: 2 from School C, 2 from School M, and 2 from School N. They varied in age from 10 years old to 14 years old. The pool was 42% male and 52% female with 6% of students not responding. None of the students had previous experience with the Wearable Learning Cloud Platform, but 29% of students said they had previous programming experience.

5.1.2 Procedures

Before the researchers arrived, all families were notified of the study being done and were informed that their children were going to be photographed with an option to opt out. The letter sent home is included in Appendix D. Additionally, each student completed a pretest on computational thinking and finite state machines that is shown in Appendix E. This pretest was administered by the participants' teachers unlike previous studies where the pretest was administered by the researchers.

Over the seven weeks of the study, the researchers saw each class for one 80 minute block each week. Though the study was seven weeks long, the researchers only planned to meet with the students for six weeks. This was due to the numerous holidays and special events that interrupted the study. Each class missed one week due to scheduling conflicts with the exception of the participants from School C who missed two classes. To make up for the extra missed

period, those classes received an accelerated Week 2 and Week 3 curriculum using the same materials and instruction that the other students received but was sped up to fit into one class.

The first week, the students received an introduction to the study which was referred to a workshop. They then played the Spanish version of Tangrams Race. Two teams of 4 students played the game while the rest of the students watched and observed. After playing the game, the students discussed their observations, first in small groups and then in a whole class discussion facilitated by the researchers. The discussion focused mostly on the questions on the second page of their activity notebooks which can be found in Appendix F. Finally, the teachers separated their students into groups of three (with the occasional group of four) and the students began to make their own games. The researchers gave verbal instructions to create a game that involved a math concept and some movements that could eventually be made in the software that was used to make Tangrams Race. Students were also given a set of written instructions for designing their games available in Appendix G. At the end of this class period, most students had made significant progress if not finished designing their games.

The second week introduced students to finite state machines (FSM), which are used to program in the WLCP. Both School N classes spent time between week one and week two working on their games outside of the study, so they began with brief presentations to their classmates about their games. Both School M classes needed more time for designing their games, so they all got 20 min at the beginning of class to finish working on their games. From this point, five of the classes received a powerpoint presentation on finite state machines and how to draw them. A copy of this presentation is located in Appendix G. Instead of this presentation, the first School M class received more discussion and interaction based instruction

to test out a new curriculum that the researchers want to implement in future studies. After the finite state machine instruction, the students were given time to work in their groups to draw a finite machine of what the phone would be doing in their game. The participants were told that the states of the FSMs should represent the screens that the players see and the transitions should represent the ways to change the screens. Additionally, the participants were told that the screens could have text and/or pictures, while the transitions could be pressing a single button, entering a color coder, or entering text. They were given the rest of the class period to finish drawing their FSMs in their groups. The first School N class finished over half an hour early and started the third week curriculum. They were working in the WLCP for about 15 minutes.

During the third week, the classes began programming in the WLCP. The first School M class needed more time to finish their FSM drawing, so they spent the first 15 minutes working on their diagrams. All other classes started with an introduction to the WLCP. The introduction told the students the web address of the page, their login credentials, and how to use the QuickStart tutorial. This was purposefully brief and not comprehensive to be able to measure the effectiveness of the QuickStart later on. The groups were given about 20 to 30 minutes to work in the WLCP before taking the first of two surveys as part of this study. The week three survey is located in Appendix H and will be discussed further in the Measures section. Most classes ended after the students took the survey, but the second School N class had about 10 minutes after taking their surveys to continue programming in the WLCP.

In the fourth week, the students continued programming in the WLCP. At the beginning of the class, they were given more information about how to use the debugger and why debugging is important. They were also given a hand out with some design patterns for how to

structure their questions. This handout is located in Appendix I. The majority of the class period was spent working on creating the games. Unlike the previous week when researchers were only providing minimum support necessary for the students in order to prevent skewing the usability survey result, during this week researchers were actively helping participants as much as possible and jumping in unpromptedly to provide help. At the end of this period, the students were given a five question exit ticket about how they used or did not use the design patterns. Additionally, the researchers presented demonstrations of two possible features to implement in the WLCP and the students voted for their favorite on their exit tickets. See Appendix J for the exit ticket. All classes completed the exit ticket at the end of week four with the exception of the second School C class, who completed it at the end of their fifth week since half the class was absent due to other school events during class four.

The fifth week was the last week of programming in the WLCP. The students were strongly encouraged to ask for help and debug their games. Additionally, the researchers taught the students how to start and stop a game instance to they could try playing their games. The researchers brought in cell phones so students could debug on mobile devices as well as their computers. Most of this class was spent working on the games and finalizing everything. At the end of the class, students were also asked to create a list of any physical materials that they needed to play their games so that the researchers could pick up additional supplies for the last class.

The sixth and final week was spent playing all of the games. The researchers brought cell phones and the physical materials requested at the end of the previous class. The participants took some time to get their physical materials together and then each group took turns presenting

their games to the class and playing their games with other classmates. The final step of the study was for the teachers to administer the post test to their students after the researchers left and then send the results to the researchers. The post test is located in Appendix K.

5.1.3 Materials

Each group of students received one notebook which contained pages for the activities in the study. There was one page with questions for observing Tangrams Race, three pages to brainstorm ideas for their own games, three pages to draw their own games as finite state machines, and then a number of blank pages that the students could use for whatever they needed. There were a number of handouts for the students as well, including the game creation instructions, design patterns cheat sheet, and the two surveys.

Additionally, the researchers provided physical materials for playing Tangrams Race and the students designed games. This included bringing the tangram pieces to play Tangrams Race and buying materials like baskets, rocks, and spinners for student games. Nine cell phones were also provided by the researchers to use while playing Tangrams Race, debugging participant games, and playing participant games.

5.1.4 Measures

Results of this study were measured through the survey and the exit ticket which are located in Appendices H and J respectively. The survey pertained to the usability of the system and the effectiveness of the QuickStart tutorial. The survey began with the ten multiple choice questions from the System Usability Scale to calculate the overall usability of the game editor. This was followed by six questions about certain features that asked participants to rate their ease of use from one to five. The next section was three yes-or-no question and one rate-on-a-scale

question about the effectiveness of the QuickStart. Then there were two open ended questions for general positive and negative feedback and then four demographic multiple choice questions.

The exit ticket was a much shorter measure that focused on the value of providing students with the design patterns. There were three yes-or-no questions about the design patterns cheat sheet. The next question was also yes-or-no but focused on the possibility of adding design patterns to the game editor. The final question asked participants to choose which of two new functionalities they would prefer to have. Majority of the data collected from the survey and exit ticket is quantitative data. The qualitative data was collected from the open ended feedback questions on the survey and from researcher observations.

5.1.5 Methods of Analysis

The first two sections of the survey were identical (with the exception of the translation) to the survey given in the first study and thus followed the same analysis process but with some addition analysis. See Section 3.1.5 for a description of how the SUS questions and ease of use of features sections were analyzed. Since the study population for the second study was a significant size, mean response and standard deviation were also calculated for the first two sections. For the SUS questions, the mean, standard deviation, and percent no response were calculated for the normalized score of each question and for the SUS scores from each participant. It is important to note that the percent no response is higher for the overall score because the average overall score was calculated from only complete responses. If any of the SUS questions were left blank, that participant's responses were included in the individual question analysis but no overall SUS score was calculated. For the features' ease of use

questions, the average response, standard deviation, and percent of population who did not respond were calculated and reported in a table.

The open ended questions were analyzed with the same methods described in Section 3.1.5, except instead of reporting counts for each category, the results were described as percentages of all responses that fit the category. As in the previous study, the demographic questions were calculated as percentages and used to describe the study pool.

There were two sections of data that were new in the second study: the QuickStart tutorial feedback and the design patterns feedback. All of the questions in both of those sections were choose one of two responses, so the results were analyzed as the percent of respondents who chose each option and who did not respond to the question.

5.2 Results

5.2.1 System Usability Scale

Table 1 below shows the mean normalized response, standard deviation for all participants who responded, and also the percent of participants who did not respond for each of the ten SUS questions. Additionally, the last row shows the overall mean SUS score, with the standard deviation and the percent of participants that did not answer all ten SUS questions.

Table 1. SUS results by question and total score

	SUS		
	Mean	SD	% No Response
Q1 - Normalized	2.40	0.99	2.80%
Q2 - Normalized	2.32	1.09	2.80%
Q3 - Normalized	2.50	1.22	4.20%
Q4 - Normalized	2.29	1.37	6.99%

Q5 - Normalized	2.79	0.95	2.80%
Q6 - Normalized	2.62	1.11	3.50%
Q7 - Normalized	2.60	1.17	2.80%
Q8 - Normalized	2.45	1.14	2.80%
Q9 - Normalized	2.76	1.05	2.80%
Q10 - Normalized	2.00	1.34	3.50%
Total Score	62.20	16.91	11.89%

The mean SUS score for the WLCP is 62.20 with a standard deviation of 16.91. There is a 90% confidence interval between 59.87 and 64.53. Of the 143 participants, 11.9% skipped at least one of the SUS questions or did not answer the survey from which this data was taken. Questions five (M=2.79, SD=0.95, NR=2.80%) and nine (M=2.76, SD=1.05, NR=2.80%) about a well integrated system and user confidence respectively had the most positive responses, while question ten (M=2.00, SD=1.34, NR=3.50%) about needing a lot of prior knowledge had the most negative response.

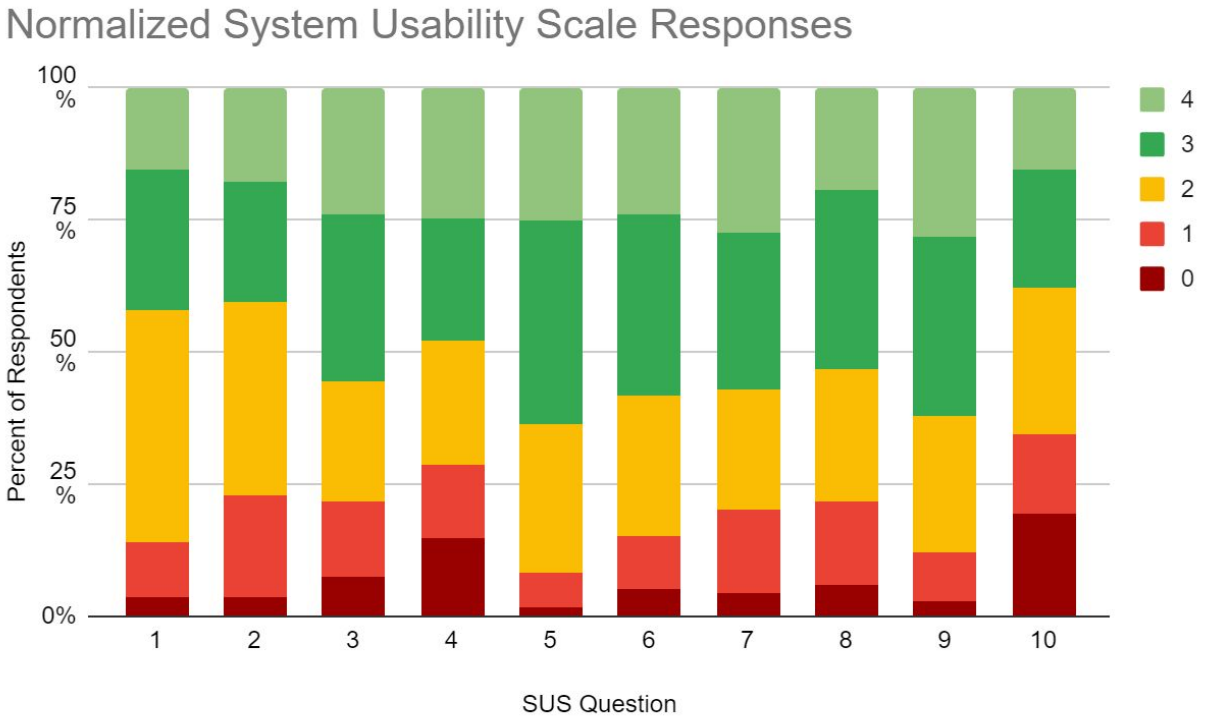


Figure 9. Normalized scores of SUS questions

Figure 9 above visualizes all of the normalized responses and supports the claims above that questions 5 and 9 were the most positively answered while question 10 was the most negatively answered.

5.2.2 Ease of Completing Tasks

Table 2 below shows the average score and standard deviations for the responses regarding the ease of performing various tasks in the WLCP. It also includes the percent of participants who did not respond to each question.

Table 2. Ease of using features means and standard deviations

Ease of Use			
	Mean	SD	%No Response
Creating a new game	3.08	1.08	3.50%

Creating a new state	3.77	1.09	3.50%
Creating a new transition	3.76	1.24	4.90%
Editing an existing state	3.52	1.10	4.90%
Editing an existing transition	3.56	1.08	5.59%
Debugging a game	3.46	1.16	3.50%

Creating a new state had the highest mean (M=3.77, SD=1.09, NR=3.50%). Creating a new transition followed closely behind with a mean of 3.76, but it had a relatively large standard deviation of 1.24 (NR=4.90%). Editing an existing transition (M=3.56, SD=1.08, NR=5.59%), editing an existing state (M=3.52, SD=1.10, NR=4.90%), and debugging a game (M=3.46, SD=1.16, NR=3.50%) were situated in the middle of the range of means. Participants ranked creating a new game the lowest with a mean of 3.08 (SD=1.08, NR=3.50%).

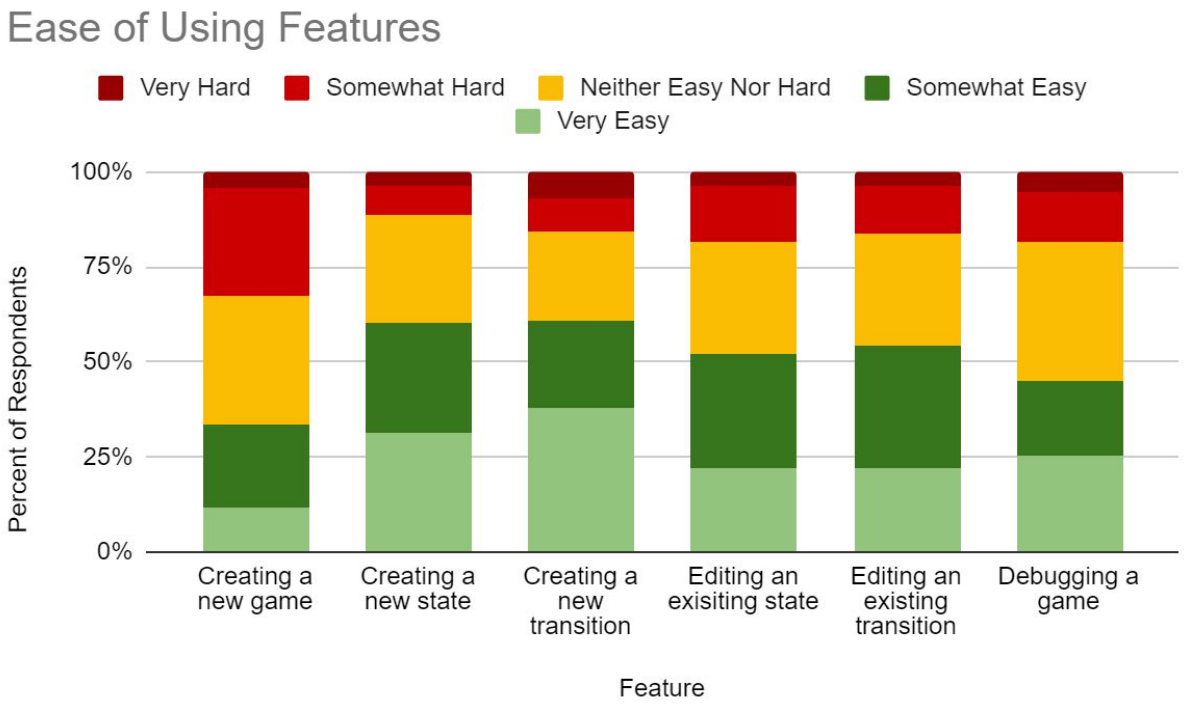


Figure 10. Stacked bar chart of ease of using features

Figure 10 above shows all of the responses about the difficulty of tasks. The lightest green represents the highest scores while the darkest red represents the lowest scores. The chart supports the claims made earlier that the participants ranked the creating a new state and creating a new transition to be the highest and creating a new game to be the lowest.

5.2.3 QuickStart Tutorial

Table 3 below shows the participants responses to the yes or no questions asked about the QuickStart tutorial as well as the mean ranking of the QuickStart with the standard deviation. It also shows the percent of participants who did not respond to each question.

Table 3. Opinions on the QuickStart tutorial

QuickStart			
	% Yes	% No	%No Response
Did you use the QuickStart when you logged in?	83.22%	11.19%	2.80%
Did you use the QuickStart when you were programming?	81.82%	11.19%	4.20%
Did you like the QuickStart?	81.82%	12.59%	2.80%
	Mean	SD	%No Response
How helpful was the QuickStart?	3.69	0.93	1.40%

With regard to the QuickStart tutorial, 83.22% of participants said that they used the tutorial when they logged in while 11.19% did not use the tutorial when they first logged in (NR=2.80%). 81.82% of participants used the QuickStart while they were programming, 11.19% did not use the tutorial while programming, and 4.20% did not respond to the questions. When

asked if they liked the QuickStart, 81.82% said yes, 12.59% said no, and 2.80% did not respond. Participants gave an average rank of 3.69 out of 5 when asked how helpful with the QuickStart was (SD=0.93, NR=1.40%).

5.2.4 Open Ended Questions

Table 4 below shows the percent occurrence of various themes when asked what they liked about using the WLCP.

Table 4. Percent occurrence of user likes

What do you like about the program?	
	% Occurrence
Creating Games	20.28%
Easy to use	18.88%
Learning	16.08%
Programming	10.49%
Fun	6.99%
Playing games	4.20%
Good/Yes	4.20%
No response	2.80%

The most popular response was that the participants liked the game creation aspect of the software (20.28%). 18.88% of participants said the program was easy to use, 16.08% said they liked learning new things, 10.49% stated that they enjoyed the programming aspect of the software, 6.99% called it fun, and 4.20% said they liked it because they enjoyed playing games. Another 4.20% of the participants simply responded with something akin to “good” or “yes”. Finally, 2.80% of participants did not answer the question.

Table 5. Percent occurrence of user dislikes

What do you NOT like about the program?	
	% Occurrence
Difficult/Complex	27.27%
Nothing	23.78%
Don't understand	11.89%
Not enough options/functionality	8.39%
Not enough explanation	5.59%
States and transitions	4.20%
Not enough space	3.50%
No response	6.29%

When asked what they did not like about the WLCP, the most prevalent responses were that the system was difficult or complex (27.27%), there was nothing they did not like (23.78%), or that they did not understand aspects of the software (11.89%). Some participants (5.59%) commented about not having enough explanation about using the system and 4.20% complained about creating and/or manipulating states and transitions. Another 3.50% of participants disliked the limited amount of space on the software. Finally, 6.29% did not respond to the questions.

5.2.5 Design Patterns

Table 6 below shows the responses to the two option questions given with regard to the design pattern materials provided during the study and the integration of design patterns into the WLCP.

Table 6. Opinions on design patterns WLCP integration

Design Patterns

	% Yes	% No	%No Response
Did you use the design patterns worksheet?	38.46%	58.04%	3.50%
Were the patterns useful/beneficial?	52.45%	42.66%	4.90%
Would you like to see the design patterns in the WLCP?	83.22%	12.59%	4.20%
Would you like to be able to drag design patterns into your game in the WLCP?	77.62%	17.48%	4.90%
	%Drag Patterns	%Copy Paste	%No Response
Would you prefer being able to drag design patterns or copy paste?	37.76%	58.74%	3.50%

Majority of the participants did not use the design patterns handout given to each of the groups (58.04%), but 38.46% said they did use the handout (NR=3.50%). When asked if the design patterns sheet was helpful, 52.45% said yes, 42.66% said no, and 4.90% did not respond to the question. 83.22% of participants said they would like to see the design patterns in the WLCP while 12.59% said they would not (NR=4.20%). Additionally, 77.62% said they would like the ability to drag design patterns into a game in the WLCP (No=17.48%, NR=4.90%). When given the option of being able to drag patterns or copy and paste, 37.76% chose to be able to drag patterns, 58.74% preferred copy and paste, and 3.50% did not respond.

5.2.6 Researcher Observations

In addition to the quantitative, the researchers administering the study also made observations of difficulties that participants faced and bugs they encountered while using the system. They are listed below in no particular order.

1. In the error message about minimum and maximum teams, there is a space character missing.
2. Participants were often confused about the purpose of the start state and they tried to put a transition between the start state and the first state.
3. Participants often tried to put spaces and accent marks into the titles of their games.
4. When participants clicked the login button more than once, the QuickStart would not close.
5. When logging in, if there is no password, the user still needs to touch all of the fields to be able to log in successfully.
6. Many participants were confused as to why they could not drag a connection from the bottom of one state into the bottom of another state.
7. The error message about all users being assigned to other states was not easily understood by the users and could appear in nonapplicable situations.
8. Some participants were confused about how to leave the page because there is always a pop up message confirming that a user wants to leave the game editor or debugger.
9. Many participants requested to be able to change the text on the buttons.
10. For the spanish translation, participants did not understand to click the “grabar” button to save. Many expected a button with “guardar.”

11. The ability to leave a color code transition or text box transition blank in order to accept all unassigned inputs was not clear or visible and had to be explained by the researchers.
12. The error message about not being able to loop back to a neighbor was not translated into Spanish.
13. Some participants did not realize that they could drag a state to the bottom of the screen to add more space for their FSM in the Game Editor.
14. When dragging a state to the bottom of the screen to add more space, once the addition space appeared, sometimes the state being dragged was no longer positioned under the cursor.
15. Sometimes the message for the previous state being already filled gave a false positive.
16. Participants were able to have two different types of transitions coming from the same state which caused two types of transitions to appear when debugging.
17. Some participants requested a timer or a way to determine which player finished first.
18. One participant asked if there was an undo button.
19. Many participants filled out the titles of the states without filling in the body of the states and then were confused why nothing showed up in the debugger.
20. One group was not able to add input transitions to their connections, but after naming some of their unnamed states, they were then able to add input transitions.
21. One group asked if there was a way to select their whole game and move it, rather than moving states one at a time.
22. In the game manager, the selection box for choosing a new game when creating a new instance overflowed and did not display all of the games.

23. When playing a game, if a user is already registered to playing another game, the states showed up blank.
24. Very few groups used scopes other than the default global scopes in the states and transitions.

5.3 Discussion

5.3.1 System Usability Scale

The average SUS score for the WLCP was 62.2 which is significantly lower than the recognized standard of 68. Additionally, a score of 62.2 is in approximately the 32nd percentile which is also quite low. This suggests that there is a significant margin for improvement. The questions most positively responded to were questions five and nine which were about well integrated functions and users' confidence in using the system, suggesting that these are two strengths of the system. Question ten about needing a significant amount of prior knowledge to use the system received the most negative response. This system currently does not have a strong help infrastructure and is often used in the context of a workshop with many prior activities, so this may contribute to participants feeling they need to learn a lot before being able to use the system. Either way, the developers of the WLCP should consider this feedback when moving forward and make more of an intentional choice about how much help to provide programmers. However, it is important to note that the range of the average normalized scores was between 2.00 and 2.79 which is a relatively small range. There were no questions that garnered significantly negative responses or significantly positive responses. Overall though, the SUS results show that the WLCP is lacking in usability.

5.3.2 Ease of Completing Tasks

When ranking the difficulty of tasks, the participants ranked creating a new state to be the easiest, followed closely by creating a new transition. This may allude to the intuitive nature of being able to drag and drop state and transitions onto the board to create them. However, the participants rated creating a new game as the most difficult task. This may be due to the login procedure. After logging into the Game Editor, no game is automatically created or loaded, so the user must select a button at the top to choose to create or load a game. However, the initial page looks nearly identical to a new game page, with the exception of the start state. This could be confusing to the user, so the developers and researchers should explore this further. Overall, all of the mean ease of use scores were between 3.08 and 3.77 which is relatively high. That said, there is always room for more improvement.

5.3.3 QuickStart Tutorial

The implementation of the QuickStart tutorial was very successful. An overwhelming majority of the participants used the tutorial when they logged in and while they were programming. Over 80% of users said they liked the tutorial. They also said it was very helpful and the mode score was four out of a highest possible five. This implies that the QuickStart was a good and valuable addition to the WLCP. The developers should look into refining the QuickStart and creating a more robust tutorial system to supplement the benefits of the QuickStart.

5.3.4 Open Ended Questions

When asked what they liked about the WLCP, the top responses from the participants were that they liked creating games, that the system was easy to use, that they enjoyed learning new things, and that they liked the programming aspect of the software. This question was meant to get feedback about what features and system components were strong, but the answers do not match that intent. In future studies, this question should be reworded to get more specific feedback. The question about what participants disliked about the system gathered more constructive feedback. The top comment was that the system was difficult and/or complex. Some other popular comments were that they did not understand how to use the system and that there was not enough explanation. These three comments could stem from the procedure followed during the study. In order to evaluate the usefulness of the QuickStart, the participants were not given instructions about how to use the system and were only told how to access the program and log in. However, from these responses, it seems that the QuickStart was not sufficient to remove confusion and clearly explain the software. Therefore, the developers should work on developing a more robust tutorial program in addition to the QuickStart to help new users get acquainted with the software. Another theme discussed was the lack of options and functionality in the program. This has also been a theme in other studies, so the developers should continue adding functionality. One final comment of note was that some users said that they did not feel there was enough space on the FSM canvas in the Game Editor. More experienced users know that if a state is dragged to the bottom of the screen, more space will appear, but this was apparently not intuitive for the users. The developers should work to try to make it more intuitive or to advertise this functionality more clearly, so newer users can take advantage of it.

5.3.5 Design Patterns

Over half of the participants did not use the design patterns sheet that they were given on the second day of programming. This may have been because they had already figured out how to set up their questions during the first day, but this was a purposeful decision so that the participants had to think and make an attempt before getting help. That said, over 80% of participants said they would like to see the design patterns as a reference on the WLCP and over half said that the sheet was helpful and beneficial. Over three quarters of the participants said they would like to have the ability to drag design patterns into their games in the WLCP to be able to create their games faster, but when given the choice between dragging design patterns into their games and being able to copy and paste sections of their game, nearly 60% of participants said they would prefer to be able to copy and paste. This suggests that the developers should spend their time implementing copy and paste, and the design patterns could be helpful as a static webpage that users could access as a reference when designing their games. According to the participant feedback, enabling a system to drag and drop design patterns would not be worth the effort and would not be more helpful than other features like copy and paste.

5.3.6 Researcher Observations

The following list contains all of the actions that the developers can take to address the researcher observations noted in Section 5.2.6.

1. Fix the error message about minimum and maximum teams, by adding a space character to make it more readable.

2. Change the start state to be an editable state that cannot have an entrance connection or step up all new games with a connection from the start state to a new state to eliminate confusion about the start state's purpose.
3. Enable the use of spaces and accent marks in the titles of games.
4. Fix the bug where when participants clicked the login button more than once, the QuickStart would not close.
5. Fix the bug where when logging in, if there is no password, the user still needs to touch all of the fields to be able to log in successfully.
6. Create a pop up message when users try to drag a connection from the bottom of one state into the bottom of another state that tells them to drag a connection from the bottom of one state to the top of the next.
7. Clarify the error message about all users being assigned to other states and try to add more messages for more specific situations.
8. Disable the pop up message confirming that a user wants to leave the game editor or debugger if the player has saved their game in the last minute or reached the last programmed state.
9. Enable the ability to change the text on the colored buttons.
10. Change the Spanish translation of "save" from "grabar" to "guardar."
11. Make the ability to leave a color code transition or text box transition blank in order to accept all unassigned inputs more visible, possibly by adding that option directly on the transition editor, instead of just in a pop up message.
12. Translate the error message about not being able to loop back to a neighbor into Spanish.

13. Make the functionality to drag a state to the bottom of the screen to add more space for their FSM in the Game Editor more visible to the user or change the implementation to be more obvious/intuitive.
14. Fix the bug where when dragging a state to the bottom of the screen to add more space, once the addition space appeared, sometimes the state being dragged was no longer positioned under the cursor.
15. Perform more intensive testing for the message for the previous state being already filled, specifically searching for false positive cases that can be eliminated so that the message is more reliable.
16. Once an input transition from a state has been filled, limit all other transitions coming from that same state to be of the same type.
17. Add a timer feature or a way for the system to progress the players to a new state without the players performing a transition.
18. Add infrastructure to support and implement an undo button and a redo button.
19. Add a warning when a user attempts to close a state where they have filled out the title of the state without filling in the body of the state.
20. Investigate and fix a bug where users were not able to add input transitions to their connections, but after naming some of their unnamed states, they were then able to add input transitions.
21. Add the ability to select multiple states and drag and drop them together.
22. Fix the bug where in the game manager, the selection box for choosing a new game when creating a new instance overflows and does not display all of the games.

23. Enable users to be able to successfully join and play two games simultaneously.

24. Add a more thorough explanation of the different scopes and how they can be used.

6. Discussion

6.1 Overall Discussion

The goal of the first study performed was to obtain baseline data about the usability of the WLCP. From the survey responses of the nine participants, the WLCP scored an average 63.3 on the SUS which lands approximately in the 33rd percentile. This is an overall low score and is 4.3 below industry standard, suggesting that the developers needed to work more on making a more usable product. The data about the ease of performing certain tasks was unclear and it was possible that the responses did not reflect the thoughts of the participants. Due to the doubts and small sample size, no conclusions were drawn from that data. However, researchers improved the format for future studies. From the open ended questions and group discussion at the end of the survey and study experience, researchers gained a lot of specific usability feedback and created a list of development priorities based on the feedback. Some items discussed and prioritized included fixing bugs, creating more pop up messages, adding pictures to states, adding variables and randomization, and enabling copy and paste functionality.

The developers then took the specific feedback from the discussion and implemented changes to the WLCP. Some changes of note included fixing six bugs, implementing localization to support English and Spanish, adding pictures to game states, adding more pop ups with helpful messages, and implementing a QuickStart tutorial that appears at login.

The second study was more extensive and performed with six classes of sixth and seventh grade students in Argentina. The 143 participants also completed the SUS survey and gave the WLCP an average score of 62.2 which is in approximately the 32rd percentile and is considered

low. This is not a significant change from the score in the US study, but it does support the findings of the first study. The two studies cannot be compared too heavily beyond this point because they were performed with distinct populations and followed different procedures.

There was no significant difference between the average ease of use scores given to the various WLCP tasks. All had a mean score between 3 and 4 with a standard deviation around 1.1. However, the lowest scoring task was creating a new game, so developers should explore the accessibility and clarity of this task more. With regard to the QuickStart tutorial implemented after the first study, the majority of participants said they used and liked the tutorial, rating its helpfulness an average 3.69 out of a possible 5. The researchers should expand and develop this more.

According to the open ended feedback, users still thought the system was complex, difficult to understand, and did not have enough explanation. Given that the QuickStart is meant to be quick, the developers should build an additional more detailed tutorial or help center to address these concerns.. The open feedback also showed that users still want more features and functionality in the WLCP. Additionally, some of the existing functionality is not very visible to the users, such as how to add more space to the canvas in the Game Editor. However, there was also a significant amount of positive feedback and in general participants liked the concept of the software being able to create games and learn programming.

While not many participants used the design patterns information sheet given out on the second day of programming, over 50% of participants thought it was beneficial and over 80% of participants would like to see the information on the WLCP. However, when asked whether they would prefer to drag design patterns into their games or be able to copy and paste sections of

their games, nearly 60% preferred the copy and paste functionality. The developers should prioritize implementing copy and paste functionality, but it could be useful and beneficial to users to have a static web page with reference information about the design patterns.

Finally, the researchers performing the studies and interacting with the participants made many observations about how they used the system well and were the WLCP did not successfully support the users. From these observations, the developers can take action to mitigate the difficulties that users faced. Some impactful action items from the researcher observations include fixing the login bugs, adding more messages when users make mistakes, simplifying and clarifying existing messages, and implementing undo and redo buttons.

6.2 Limitations

As with any study, there were limitations. In the US study, the sample size was very small which could have affected the accuracy of the results. Additionally, the participants seemed to misinterpret the section about the ease of performing different tasks in the WLCP which combined with the small sample size rendered that data unusable. Finally, the US study was condensed into under two hours and skipped certain steps of the procedure followed in the Argentina study so the two data sets were not comparable.

In the Argentina study, one large limitation was the workshop style of the study. Since the participants were broken up into six different classes, the participants in the first class always experienced more logistical problems than those in the last class and the later classes got more practiced instruction and interaction with the researchers leading the workshop. The schedule of the workshop was a limitation because often the classes would have a day off which would randomly add large time gaps between some activities. One class missed sessions due to

scheduling problems which forced the researchers to speed up two classes worth of content to fit in one class period so the section would finish on time. Additionally, no specific script was followed though the researchers tried to convey the information to all the classes in the same way, so not all participants received the same information. The overall procedure was impossible to replicate identically for all participants.

Another limitation to the Argentina study was working in schools which tend to be more unpredictable and controlled environments . The culture of the schools affected how the students participated in the study and was difficult to account for. Each school had a different teacher that gave their students different instructions that were not accounted for and replicated across all participants. Additionally, there was no way for the researchers to monitor or account for instructions that the teacher gave about the study outside of the times that the study was scheduled to take place. Some teachers assigned work related to the study as homework. At least one teacher gave the students a grade for the work they performed in the study which affected how the students behaved.

6.3 Future Work

From a developer's perspective, both studies gave a significant amount of feedback on the WLCP's strengths and flaws. This feedback can be incorporated into the plan for future developments and improvements. The SUS scores showed that there is plenty of room for improvement on the usability front. From the researcher's perspective, these studies can be continuously repeated to measure the growth in usability and the impact of new features.

7. Conclusion

The goal of these studies were to determine how the users of the Wearable Learning Cloud Platform can be better supported. Through the survey data collected, I found that users can be supported by improving the usability of the WLCP since it's usability is ranked below standard. To do this, developers can implement some of the features requested by the users during the studies, they can fix the bugs observed while participants were using the WLCP, and they can continue to reevaluate the usability of the WLCP to track their progress and continue to understand the system's strengths and weaknesses. The QuickStart tutorial implemented for the Argentina study was considered helpful by the users, but it could be improved to be more clear and the WLCP is still lacking a more robust and interactive tutorial system or help center. Finally, the main functionality of the WLCP's Game Editor is relatively intuitive and easy to use, but participants found some of the additional features and nuanced options to be confusing and unclear. Additionally, users requested to add features for selecting sections of their games, copying and pasting, and clarifying confusing error messages. The developers and researchers can use the results of these studies to guide future development and research on the WLCP.

References

Cerruti, C., Leondires, P., McCarthy, D., Ramadan, Si., Silva, A., & Tapply, D. (2015). *Physical Games for Learning* (Undergraduate Interactive Qualifying Project No. E-project-100915-160926). Retrieved from Worcester Polytechnic Institute Electronic Projects Collection: http://www.wpi.edu/Pubs/E-project/Available/E-project-100915-160926/unrestricted/Physical_Games_For_Learning.pdf

Micciolo, Matthew (2017). *Physical Games for Learning II*. (Undergraduate Major Qualifying Project No. E-project-032417-144555). Retrieved from Worcester Polytechnic Institute Electronic Projects Collection: http://www.wpi.edu/Pubs/E-project/Available/E-project-032417-144555/unrestricted/PhysicalGameForLearningII_mmicciolo.pdf

Micciolo, Matthew, *"Designing a Visual Programming Language for the Creation of Multiplayer Embodied Games"* (2018). *Masters Theses (All Theses, All Years)*. 1272. https://digitalcommons.wpi.edu/etd-theses/1272

Nielsen, J. (2012). *Usability 101: Introduction to Usability*. [online] Nielsen Norman Group. Available at: https://www.nngroup.com/articles/usability-101-introduction-to-usability/ [Accessed 29 Sep. 2019].

Quesenbury, W. (2001). *What Does Usability Mean: Looking Beyond 'Ease of Use' - Whitney Interactive Design*. [online] Wqusability.com. Available at: http://www.wqusability.com/articles/more-than-ease-of-use.html [Accessed 29 Sep. 2019].

Rountree, Wendy Leigh, "*Redesigning Traditional Children's Games to Teach Number Sense and Reinforce Measurement Estimation Skills Using Wearable Technology*" (2015). *Masters Theses* (*All Theses, All Years*). 506. https://digitalcommons.wpi.edu/etd-theses/506

Sauro, J. (2011). *Measuring Usability with the System Usability Scale (SUS)*. [online] MeasuringU. Available at: https://measuringu.com/sus/ [Accessed 29 Sep. 2019].

Appendices

Appendix A: Study 1 Design a Game Instructions

Group Activity: Design a Math Game!

Today you are going to design a math game with your team. This game is for 3th-4th grade kids, so try to put yourself in the shoes of a younger student. We want you to:

1. **design a math game,**
2. **describe the game, and**
3. **draw a representation of the game on the paper pads.**

The game has to meet these criteria:

- A game that 3-4th grade kids can play in school over recess (it can be played in the classroom or outside in a playground/park or in the gym),
- The game has to teach (or allow students to practice) a particular math concept
- The game should have at most 4 players and if there are teams, at most 2 teams
- We want to get the students moving, so the game must be active; it should require physical movement by the students
- Ideally, the movement should be connected to the math in some way.
- The game should involve mobile technology (cell phones)
- We want you to specify the game (show us how your game works) on these pads on paper, and we will ask you to explain how it works later.

If time allows, prepare to give a 3 minute presentation of your game to the class!

Appendix B: Study 1 Finite State Machines Training Presentation

Redesigning Games with Finite State Machines

Ivon Arroyo, Erin Ottmar, Avery Harrison, Taylyn Hulse, Rich Valente

Implementing your Games

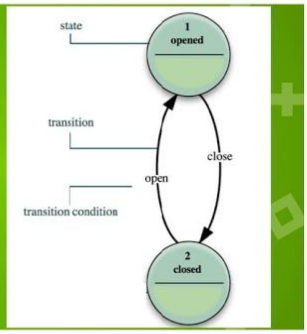
- Think what the technology does in your games, step by step.
- What should the screens say/show?
- What should the player do on the phones, at each step?

Today's Goals!

- Learn what a Finite State Machine (FSM) is
- Learn how to draw FSM diagrams
- Get a glimpse at the game editor tool you will be using to input your Games

Finite State Machine

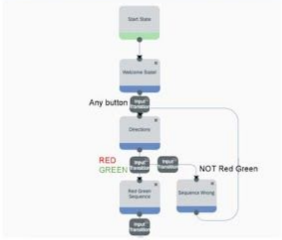
that describes the behavior of a … ?



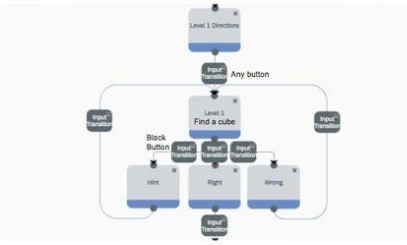
Estimate It FSM Example

- Mostly everyone played **EstimateIT!** on day 1
- **EstimateIT!** is specified as a Finite State Machine!
- **EstimateIT!** in the game editor of the "Wearable Learning Cloud Platform"
- http://wlcp.embodied.wpi.edu

EstimateIT! as a Finite State Machine

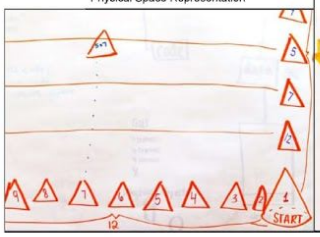


EstimateIT! as a Finite State Machine



MultiplyME!

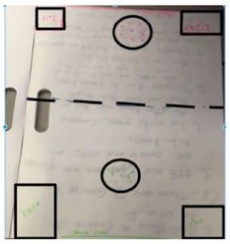
Physical Space Representation



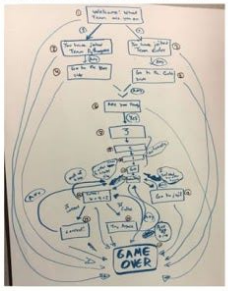
Finite-State machine representation for Phone

Math "Capture the Flag"

Physical Space Representation



Finite-State machine representation for Phone



Your Instructions for today

- You will Create **Finite State Machines** on whiteboards to describe **your games**
  - What will the cell phones do at each step of the way?
  - Then we will put them in the computer! In 15 minutes or so...

You are be limited to:

- OUTPUT STATES:
  - States represented by boxes or circles (OUTPUT)
    - **Text Display only! Keep in mind where it falls short.**
- INPUT TRANSITIONS:
  - Transitions are represented as arrows (player INPUT)
    - **Single Button** Press (red, green, blue, black)
    - **Barcode** of Button Sequence (red, green, blue, black)
    - Typing a word/number with **keyboard**

You will need to ADAPT your games, then you will be asked where technology (WearableLearning.org) fell short.

Appendix C: Study 1 Usability Survey

WLCP Survey

With respect to the Wearable Learning Cloud Platform (WLCP), rate how much you agree with each statement where 1 = Strongly Disagree, 2 = Disagree, 3 = Neutral, 4 = Agree, and 5 = Strongly Agree.

I think that I would like to use this system frequently.	1	2	3	4	5
I found the system unnecessarily complex.	1	2	3	4	5
I thought the system was easy to use.	1	2	3	4	5
I think that I would need the support of a technical person to be able to use this system.	1	2	3	4	5
I found the various functions in this system were well integrated.	1	2	3	4	5
I thought there was too much inconsistency in this system.	1	2	3	4	5
I would imagine that most people would learn to use this system very quickly.	1	2	3	4	5
I found the system very cumbersome to use.	1	2	3	4	5
I felt very confident using the system.	1	2	3	4	5
I needed to learn a lot of things before I could get going with this system.	1	2	3	4	5

Rate the ease of use for the following tasks where 1 = Very easy, 2 = Somewhat easy, 3 = Neither easy nor hard, 4 = Somewhat hard, 5 = Very hard

Creating a new game	1	2	3	4	5
Creating a new state	1	2	3	4	5
Creating a new transition	1	2	3	4	5
Editing an existing state	1	2	3	4	5
Editing an existing transition	1	2	3	4	5
Testing and debugging a game	1	2	3	4	5

What did you like about using the Wearable Learning Cloud Platform, in general?

What did you NOT like about using the Wearable Learning Cloud Platform, in general?

Were there any features that you wished the Game Editor had, but it didn't?

Did you face any difficulties when using the Game Editor? If so, how did you overcome the difficulties?

Which do you consider yourself to be?

❑ Student ❑ Teacher ❑ Both ❑ Neither

How many times have you used the WLCP before this class?

❑ Never ❑ 1 time ❑ 2-4 times ❑ 5+ times

Which programmer were you?

❑ First Programmer ❑ Second Programmer

Appendix D: Study 2 Letter to Parents

WORCESTER POLYTECHNIC INSTITUTE

SOCIAL SCIENCE AND POLICY STUDIES DEPARTMENT

Estimados Padres, Junio del 2019

Nuestro equipo de investigadores del Instituto Politécnico de Worcester (WPI, Worcester Polytechnic Institute, Massachusetts, USA) viene a la escuela de su hijo a hacerlos usar un programa llamado JuegosMobiles.org, como parte de una experiencia piloto en Córdoba, una actividad de investigación dirigida por profesores y estudiantes de esta Universidad, en colaboración con la Universidad Blas Pascal y la escuela de su hijo.

Como parte de este proyecto de investigación, los maestros y los alumnos trabajarán juntos en actividades donde los chicos diseñan juegos de matemáticas (físicamente activos, para varios jugadores) y programan teléfonos celulares como parte del proceso, utilizando un software especial creado por investigadores del Instituto Politécnico de Worcester. Esta investigación es especial porque enseña a su hijo conceptos de ciencias de la computación que no se enseñan hasta la universidad, en particular el concepto de programación de dispositivos móviles como máquinas de estado finito.

Como parte de la actividad, los alumnos primero jugarán un juego de matemáticas usando dispositivos celulares como apoyo al jugador; después los chicos diseñarán sus propios juegos en papel, y por último, programarán los celulares con el software de programación de juegosmobiles.org durante varias semanas, durante las clases de matemáticas y computación. También contestarán un cuestionario antes y después de la experiencia. Su hijo podría ser elegido para una entrevista grabada en video para hablar en voz alta explicando los juegos que han diseñado. Los videos son solo para fines de investigación y para comprender el pensamiento computacional y el proceso de diseño de los alumnos, y se guardarán en la computadora de los investigadores. No se publicarán en la web ni se mostrará al público sin permiso extra.

La grabación de video es solo para fines de investigación y se guardará sólo en las computadoras de los investigadores. El nombre personal de su hijo NO se guarda, y los datos recopilados y las respuestas que proporcionan no se pueden vincular a ellos personalmente. Todos los datos se analizan en base a números anónimos, y cualquier referencia personal se destruye poco después de que los estudiantes usen el programa. También, su hijo puede optar por dejar de participar en cualquier momento.

No dude en contactarme si tiene alguna pregunta sobre esta investigación , que está regulada por la Junta de Revisión de Investigación Institucional de la Universidad (Worcester Polytechnic Institute) para proteger los derechos de los participantes humanos en las investigaciones, el Dr. Kent Rissmiller (correo electrónico: kjr@wpi.edu, +1 508-831-5019) si tiene alguna duda, o simplemente mándeme un email a mí personalmente.

Dra. Ivon Arroyo
Programa de Technologia Educativa y Ciencias del Aprendizaje
Laboratorio de Tecnologías Educativas Avanzadas
Worcester Polytechnic Institute, WPI, Worcester, MA
Email: iarroyo@wpi.edu

Hacer una cruz (X) en lo que corresponda:
___ SÍ autorizo a mi hija/o a ser grabado en video como parte de la investigación
___ NO autorizo a mi hija/o a ser grabado en video como parte de la investigación

Firma del padre, madre o tutor: _____

508-831-5296 (TEL) 508-831-5896 (FAX)

100 INSTITUTE ROAD, WORCESTER MA 01609-2280 USA
WPI.EDU

Appendix E: Study 2 Pre Test

Primer Nombre:_____Inicial del Apellido: __ Colegio:_____
Grado:_____ Edad:_____ Género: _____ Fecha:_____

Estamos interesados en los sentimientos de las personas con respecto a las computadoras y la programación. Leé atentamente cada enunciado y decidí con qué frecuencia te sentís así con respecto a las computadoras y la programación. Hacé un círculo alrededor de la respuesta.
***Nota**: Aquí no hay respuestas correctas o incorrectas, es simplemente cómo te sentís.*

---

1. **Puedo aprender cosas relacionadas con las computadoras y la programación, incluso si lo que hay que hacer es difícil.**

1	2	3	4	5	6
Nunca	Muy Raramente	Raramente	A veces	General-mente	Siempre

2. **Sé que puedo resolver incluso los desafíos más difíciles relacionados con la computadora.**

1	2	3	4	5	6
Nunca	Muy Raramente	Raramente	A veces	General-mente	Siempre

3. **Sé que puedo aprender todo lo que se me enseña en computación este año.**

1	2	3	4	5	6
Nunca	Muy Raramente	Raramente	A veces	General-mente	Siempre

4. **Puedo hacer todo el trabajo que nos dan en la clase de computación, si pongo esmero.**

1	2	3	4	5	6
Nunca	Muy Raramente	Raramente	A veces	General-mente	Siempre

5. **Puedo hacer incluso el trabajo más difícil de mis clases de computación, si lo intento.**

1	2	3	4	5	6
Nunca	Muy Raramente	Raramente	A veces	General-mente	Siempre

Primer Nombre:_____Inicial del Apellido: __ Colegio:_____
Grado:_____ Edad:_____ Género: _____ Fecha:_____

6. **Creo que soy alguien que sabe trabajar en actividades de programación.**

1	2	3	4	5	6
Nunca	Muy Raramente	Raramente	A veces	General-mente	Siempre

7. **En general, cuánto te gustan las computadoras y la programación?**

1	2	3	4	5	6
No me gustan para nada	General-mente No	No mucho	Un poco	General-mente Si	Me gustan mucho

8. **Vas a clases de computación afuera de la escuela?**

1	2	3	4	5	6
No, nunca hice computación afuera del colegio	Una sola vez tome una clase, pero nunca mas	Raramente tomo alguna clase	De vez en cuando tomo algunas clases	Si, voy una vez por semana	Si, voy varias veces por semana

9. **Hay algún tipo computadora en tu casa? Esto puede ser una computadora fija, o una laptop, o una Tablet (no una consola de video juegos o un celular).**

Si NO

10. **Si Hay, Qué tipo de computadora será? Explicá lo que sepas/puedas:**

11. **Cada cuanto la usás?**

1	2	3	4	5	6
Nunca	Casi Nunca	De vez en cuando	1-2 veces por semana	Casi todos los días	Todos los días

12. **SI la usás, para qué la usás?**

Primer Nombre:_____Inicial del Apellido: __ Colegio:_____
Grado:_____ Edad:_____ Género: _____ Fecha:_____

13. Hay algún tipo de consola de video juegos en tu casa?

Si NO

14. Si Hay, qué tipo de consola será? Explicá lo que sepas/puedas:

15. Cada cuanto la usás?

1	2	3	4	5	6
Nunca	Casi Nunca	De vez en cuando	1-2 veces por semana	Casi todos los días	Todos los días

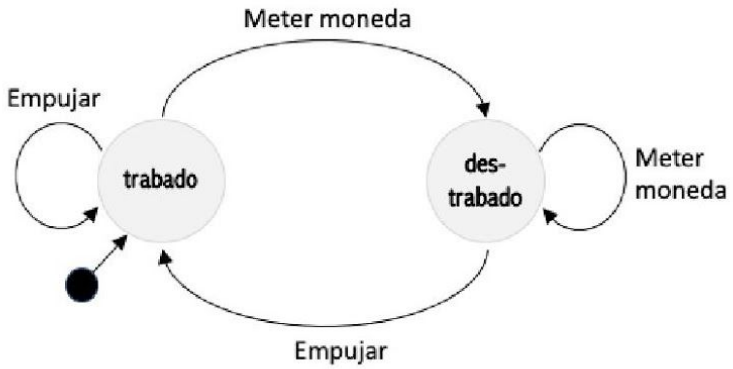
16. Tenés un teléfono celular de algún tipo (vos misma/mismo, no de tus padres)?

Si NO

¡Muchas gracias por contarnos de vos, y decir cómo te sentís!
Ahora, Respondé las preguntas de las páginas siguientes lo mejor que puedas.
Si no sabés, no te preocupes --estimá la respuesta lo mejor que puedas.

Primer Nombre:_____Inicial del Apellido: __ Colegio:_____
Grado:_____ Edad:_____ Género: _____ Fecha:_____

Este es un diagrama visual de cómo funciona una máquina.



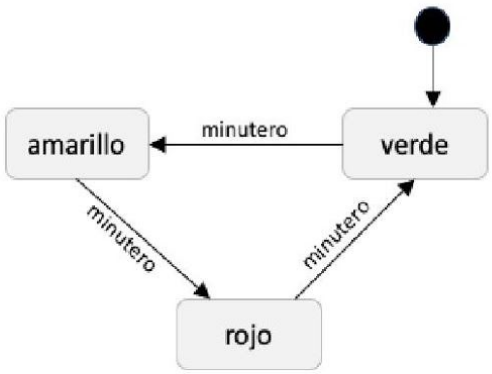
17a.) ¿Qué tipo de máquina podría ser esta? ¿A qué te recuerda y por qué?

17b.) De acuerdo a este modelo, ¿qué pasaría si pusieras tres (3) monedas? ¿Se puede?

17c.) Si la máquina está **trabada** en este momento... ¿Qué tendríamos que hacer para que esta máquina se **destrabe** y luego se vuelva a **trabar**?

Primer Nombre:_____**Inicial del Apellido:** __ **Colegio:**_____
Grado:_____ **Edad:**_____ **Género:** _____ **Fecha:**_____

18) Este es un modelo (un diagrama) de cómo funciona un semáforo. Este semáforo cambia las luces en función de un temporizador (un minutero): Cambia el color de las luces cada un minuto.



Queremos agregar un botón que un **peatón** pueda presionar para poder cruzar la calle. Al presionar el botón, la luz se pondría roja inmediatamente. Después de eso, reanudaría su funcionamiento normal. ¿Podés volver a dibujar el diagrama para representar esta nueva situación? ¿Qué sería necesario agregar?

Primer Nombre:_____**Inicial del Apellido:** __ **Colegio:**_____
Grado:_____ **Edad:**_____ **Género:** _____ **Fecha:**_____

19) En un ascensor, hay una pantalla y dos botones, como se muestra a continuación. Tu tarea es diseñar un diagrama (similar a los anteriores) para representar el comportamiento de la pantalla, y cómo cambia cuando se aprietan los botones mostrados.

Dibujá un diagrama (al estilo de los anteriores) que represente el funcionamiento de la pantalla digital de un ascensor, y cómo cambia esta pantalla dependiendo de cuál de los dos (2) botones se presionan, de esta manera:

- Al presionar el botón con este triángulo: △, la pantalla digital mostrará: "Arriba"
- Al presionar el botón con este triángulo: ▽ , la pantalla digital mostrará: "Abajo"
- El botón △ también se puede presionar cuando ya se está en el piso superior (en cuyo caso seguirá mostrando “Arriba”).
- El botón ▽ también se puede presionar cuando ya se está en el piso inferior (en cuyo caso seguirá mostrando “Abajo”).

Appendix F: Study 2 Student Workbooks

Nombre del grupo :

.....

Nombres de los alumnos :

.....

.....

.....

Observá un Juego

Observen a sus compañeros jugando un juego.

En esta etapa, serás observador de un juego, la Carrera de Tangramas. Les pedimos que registren sus observaciones en el espacio a continuación. Les sugerimos algunas cosas a observar, pero pueden escribir cualquier cosa que noten sobre el juego y sus jugadores.

Observaciones sobre el juego:
¿Qué les pide el juego que hagan a los jugadores?

¿Qué notan sobre los equipos? ¿Cómo interacsuan los jugadores de un equipo?

¿Cómo cambia el juego a medida que pasa el tiempo?

¿Qué tipo de estrategias usan los jugadores?

¿Qué tipo de movimientos (o gestos físicos) usan los jugadores?

¿ Alguna otra cosa más? ¿Qué más ven que pasa?

Dibujen, esbocen, describan su juego aquí.

Para comenzar el programa de Juegos Mobiles, tenemos que comprender la función y el propósito del programa.

- ¡El propósito de JuegosMobiles.org es hacer que los jugadores piensen mientras se mueven, y piensen de forma activa! Por lo tanto, el diseño de su juego debe basarse en el movimiento.
- Pueden hacer un juego para múltiples equipos, múltiples miembros de un equipo. También pueden hacer que su juego sea *público*, para compartirlo con otros chicos y maestros, dentro o fuera de la escuela.

¡Deben comenzar a imaginar, a generar ideas de juegos que incorporen tanto las **matemáticas** como el **movimiento**! Dibujen el juego, lo que hacen los jugadores, el espacio físico en el que jugarán, los pasos necesarios a seguir, las reglas, y todo lo que se les ocurra.

Lluvia de Ideas

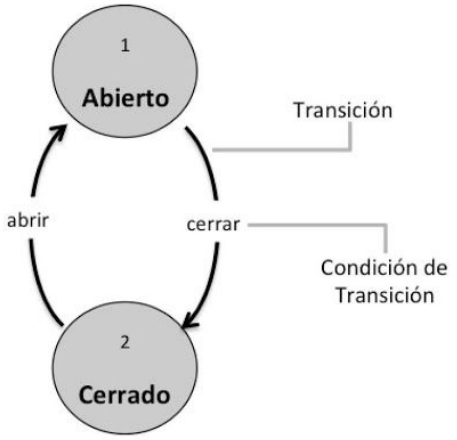
Dibujen, esbocen, describan su juego aquí.

Dibujen, esbocen, describan su juego aquí.

Máquina de Estado Finito

Usen esta página para definir el funcionamiento de los dispositivos móbiles en su juego

Esta es una máquina de Estado Finito de Ejemplo.



Máquina de Estado Finito

Usen esta página para definir el funcionamiento de los dispositivos móbiles en su juego

Máquina de Estado Finito

Usen esta página para definir el funcionamiento de los dispositivos móbiles en su juego

Ya Estás Listo Para Programar!

JuegosMobiles.org

Usa esta página para tomar notas sobre la programación, o cualquier otra cosa que necesites.

Instrucciones del Juego

Ahora que han terminado de programar su juego, descríbanlo aquí para que otra gente pueda seguir las instrucciones y jugar el juego.

Nombre del juego:

Nombre del juego en juegosmobiles.org:

Creadores:

Descripción del juego:

Contenido matemático del juego:

Para qué grado:

Número mínimo de jugadores y equipos:

Número máximo de jugadores y equipos:

Dónde están los materiales necesarios (incluidas las etiquetas) ¿en formato digital? :

¿Algún comentario para las personas que intenten jugar este juego?

Sigue en la próxima página →

¡Página adicional para materiales o cualquier cosa que necesiten!

Appendix G: Study 2 Finite State Machine Presentation

Rediseñar juegos con máquinas de estado finito

Ivon Arroyo, Erin Ottmar, Avery Harrison, Taylyn Hulse, Rich Valente

Implementar sus juegos

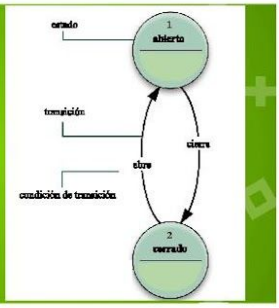
- Pensar de lo que hace la tecnología en sus juegos, paso a paso.
- ¿Qué deben decir/mostrar las pantallas?
- ¿Qué debe hacer los jugadores en los móviles, en cada paso?

¡Metas para hoy!

- Aprender que es una máquina de estado finito (MEF)
- Aprender cómo dibujar un diagrama de MEF
- Empezar de aprender cómo usar el editor de juegos para crear su juego

Máquina de estado finito

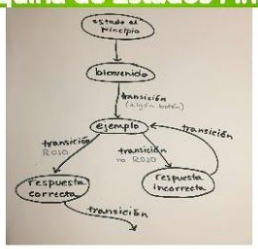
Este diagrama describe el funcionamiento de ... qué?



Carrera de Tangramas MEF Ejemplo

- Jugamos carrera de tangramas el primer día
- **Carrera de tangramas** es especificado en una máquina de estados finitos
- **Carrera de tangramas** en el editor de juegos de juegosmobiles.org

Carrera de tangramas como una Máquina de Estados Finitos



Carrera de tangramas como una Máquina de Estados Finitos

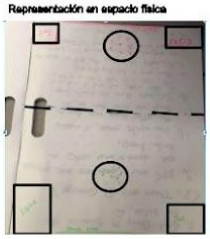
MultiplicaME!

Representación de estados finitos para los móviles

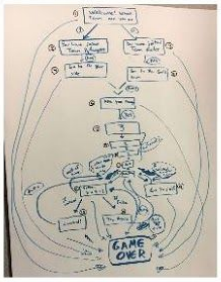
Representación en espacio físico

"Capturar la bandera" con Matemáticas

Representación en espacio físico



Representación de estados finitos para los móviles



Sus instrucciones para hoy

- Van a crear **máquinas de estados finito** en papel para describir **sus juegos**
  - ¿Qué harán los móviles en cada paso del juego?

Sus limitaciones son:

- Estados
  - Representados por rectángulos o círculos
    - ¡Solo texto!
- Transiciones
  - Representados por flechas
    - Pulsación de un solo botón (rojo, verde, azúl, negro)
    - Código de barras o secuencia de botones (rojo, verde, azúl, negro)
    - Teclar una palabra o número con un teclado

Appendix H: Study 2 Survey

Nombre:

Encuesta de JuegosMobiles.org

Con respecto al sistema JuegosMobiles.org, califica cuánto estás de acuerdo con cada oración donde 1 = totalmente en desacuerdo, 2 = en desacuerdo, 3 = neutral, 4 = en acuerdo, and 5 = totalmente en acuerdo.

Pienso que me gustaría usar el sistema a menudo.	1	2	3	4	5
Encuentro que el sistema es innecesariamente complejo.	1	2	3	4	5
Pienso que el sistema es fácil para usar.	1	2	3	4	5
Pienso que necesitaría el apoyo de un técnico para poder usar el sistema.	1	2	3	4	5
Encuentro que las funciones variadas del sistema estaban bien integradas.	1	2	3	4	5
Pienso que hay demasiada inconsistencia en el sistema.	1	2	3	4	5
Imaginaría que mucha gente aprendería usar el sistema muy rápido.	1	2	3	4	5
Encuentro el sistema muy engorroso para usar.	1	2	3	4	5
Me siento seguro/a cuando uso el sistema.	1	2	3	4	5
Necesité aprender muchas cosas antes de poder comenzar a usar el sistema.	1	2	3	4	5

Califica la facilidad del uso para las siguientes tareas, donde 1 = muy difícil, 2 = un poco difícil, 3 = Ni difícil ni fácil, 4 = un poco fácil, y 5 = muy fácil

Crear un juego nuevo	1	2	3	4	5
Crear un estado nuevo	1	2	3	4	5
Crear una transición nueva	1	2	3	4	5
Editar un estado existente	1	2	3	4	5
Editar una transición existente	1	2	3	4	5
Probar y depurar un juego	1	2	3	4	5

Nombre:

¿Usaste la tutoría que apareció cuando iniciaste la sesión? Sí No

¿Te ayudó ver la tutoría mientras estaba programando su juego? Sí No

¿Te gustó la tutoría? Sí No

En una escala de uno (para nada provechoso) a cinco (muy muy provechoso), ¿cuánto te ayudó la tutoría en JuegosMobiles?

1 2 3 4 5

¿Qué te gusta sobre usar el software JuegosMobiles, en general?

¿Qué NO te gusta sobre usar el software JuegosMobiles, en general?

¿Cuántos años tienes? 10 11 12 13

¿Eres varón o mujer? Varón Mujer

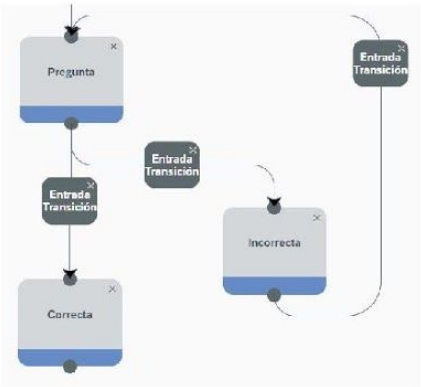
¿Has programado anteriormente? Sí No No sé

¿Por cuánto tiempo (aproximadamente) has sido el programador principal (la persona usando el teclado, el ratón/mouse, y panel táctil/trackpad)?

Menos que 5min 6min-15min 15min-30min 30min-1hora Más que 1hr

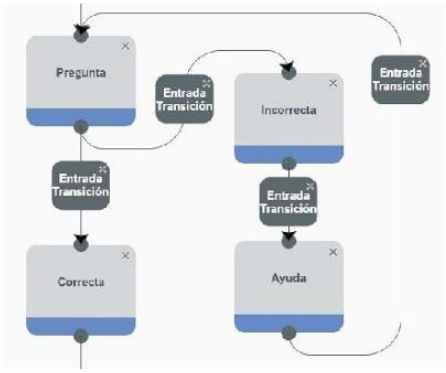
Appendix I: Study 2 Design Patterns Handout

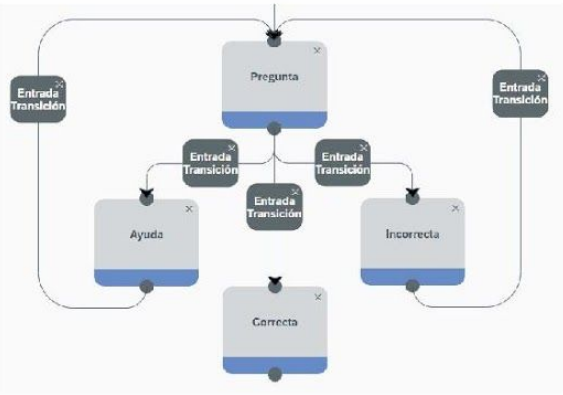
Patrones de diseño



Este diseño es uno de los más simples para definir las preguntas del juego. Observa que hay un estado con una pregunta. Después de apretar un botón, ingresar un código o un texto, el jugador puede llegar al estado correcto o al estado incorrecto (depende de lo que el jugador ingrese). Si llega al estado incorrecto, el jugador vuelve a ver la pregunta de nuevo y tiene que ingresar la respuesta correcta para continuar a la próxima parte del juego.

Este patrón es un poco más complicado que el anterior, porque añade un estado de ayuda. Fijate que después del estado incorrecto, en lugar de inmediatamente regresar a la pregunta, este diseño ofrece una ayuda, y luego recién regresa a la pregunta.





Similar al diseño anterior, este tiene una pregunta, un estado correcto, un estado incorrecto, y una ayuda. Sin embargo, en este puedes acceder a la ayuda desde la pregunta, en lugar de responder incorrectamente primero para poder ver la ayuda.

También, puedes programar algo diferente a estos patrones de diseño. Por ejemplo, si no quieres que los jugadores regresen a la misma pregunta cuando responden mal, puedes transicionar a un estado nuevo de la respuesta incorrecta. Hay muchas maneras y mecanismos distintos para construir las preguntas de sus juegos.

Appendix J: Study 2 Exit Ticket

Tu Nombre:
Tu Grado:

Design Patterns Survey

¿Usaste la hoja con los patrones de diseño?	Sí	No
¿Te fueron útiles/provechosos los patrones?	Sí	No
¿Te gustaría ver estos diseños posibles desde JuegosMobiles.org?	Sí	No
¿Te gustaría poder arrastrar los diseños hasta tus juegos en JuegosMobiles.org en lugar de hacerlos manualmente? (Recuerda el video mostrado anteriormente)	Sí	No

¿Preferirías poder arrastrar los diseños hasta tus juegos (el primer video) o copiar y pegar partes de tus juegos (el segundo video)?

Quiero arrastrar los diseños Quiero copiar y pegar

Tu Nombre:
Tu Grado:

Design Patterns Survey

¿Usaste la hoja con los patrones de diseño?	Sí	No
¿Te fueron útiles/provechosos los patrones?	Sí	No
¿Te gustaría ver estos diseños posibles desde JuegosMobiles.org?	Sí	No
¿Te gustaría poder arrastrar los diseños hasta tus juegos en JuegosMobiles.org en lugar de hacerlos manualmente? (Recuerda el video mostrado anteriormente)	Sí	No

¿Preferirías poder arrastrar los diseños hasta tus juegos (el primer video) o copiar y pegar partes de tus juegos (el segundo video)?

Quiero arrastrar los diseños Quiero copiar y pegar

Appendix K: Study 2 Post Test

Primer Nombre:_____Inicial del Apellido: __ Colegio:_____
Grado:_____ Edad:_____ Varon?/Mujer? Fecha:_____

Estamos interesados en los sentimientos de las personas con respecto a las computadoras y la programación. Leé atentamente cada enunciado y decidí con qué frecuencia te sentís así con respecto a las computadoras y la programación. Hacé un círculo alrededor de la respuesta.
***Nota**: Aquí no hay respuestas correctas o incorrectas, es simplemente cómo te sentís.*

---

1. **Puedo aprender cosas relacionadas con las computadoras y la programación, incluso si lo que hay que hacer es difícil.**

1	2	3	4	5	6
Nunca	Muy Raramente	Raramente	A veces	General-mente	Siempre

2. **Sé que puedo resolver incluso los desafíos más difíciles relacionados con la computadora.**

1	2	3	4	5	6
Nunca	Muy Raramente	Raramente	A veces	General-mente	Siempre

3. **Sé que puedo aprender todo lo que se me enseña en computación este año.**

1	2	3	4	5	6
Nunca	Muy Raramente	Raramente	A veces	General-mente	Siempre

4. **Puedo hacer todo el trabajo que nos dan en la clase de computación, si pongo esmero.**

1	2	3	4	5	6
Nunca	Muy Raramente	Raramente	A veces	General-mente	Siempre

5. **Puedo hacer incluso el trabajo más difícil de mis clases de computación, si lo intento.**

1	2	3	4	5	6
Nunca	Muy Raramente	Raramente	A veces	General-mente	Siempre

Primer Nombre:_____Inicial del Apellido: __ Colegio:_____
Grado:_____ Edad:_____ Varon?/Mujer? Fecha:_____

6. **Creo que soy alguien que sabe trabajar en actividades de programación.**

1	2	3	4	5	6
Nunca	Muy Raramente	Raramente	A veces	General-mente	Siempre

7. **En general, cuánto te gustan las computadoras y la programación?**

1	2	3	4	5	6
No me gustan para nada	General-mente No	No mucho	Un poco	General-mente Si	Me gustan mucho

8. **Gracias a que creamos nuestro juego de matemáticas en juegosmobiles.org, creo que _____ sobre cómo programar.**

1	2	3	4
No aprendí nada Nuevo	Capaz que aprendí algo	Aprendí un poco	Aprendí Mucho

9. **Las preguntas que siguen tienen que ver con el tema de matemáticas del juego que diseñaron ¿Nos recordás qué tema de matemáticas incluyeron en el juego que diseñaron? Escribilo aca:**

10. **Gracias a que imaginamos y creamos nuestro juego de matemáticas, creo que _____ sobre el tema de matemáticas de mi juego.**

1	2	3	4
No aprendí nada Nuevo	Capaz que aprendí algo	Aprendí un poco	Aprendí Mucho

11. **Creo que _____ sobre el tema de matemáticas de mi juego gracias a que programamos el juego en JuegosMobiles.org.**

1	2	3	4
No aprendí nada Nuevo	Capaz que aprendí algo	Aprendí un poco	Aprendí Mucho

Primer Nombre:_____Inicial del Apellido: __ Colegio:_____
Grado:_____ Edad:_____ Varon?/Mujer? Fecha:_____

12. **Cuando que diseñamos la ayuda para los jugadores, creo que _____ sobre el tema de matemáticas de mi juego.**

1	2	3	4
No aprendí nada Nuevo	Capaz que aprendí algo	Aprendí un poco	Aprendí Mucho

13. **Cuando le explicamos el juego a otra gente para que entienda cómo jugar, _____ sobre el tema de matemáticas de mi juego.**

1	2	3	4
No aprendí nada Nuevo	Capaz que aprendí algo	Aprendí un poco	Aprendí Mucho

14. **Cuando hicimos que otra gente jugara nuestro juego, aprendí _____ sobre el tema de matemáticas de mi juego.**

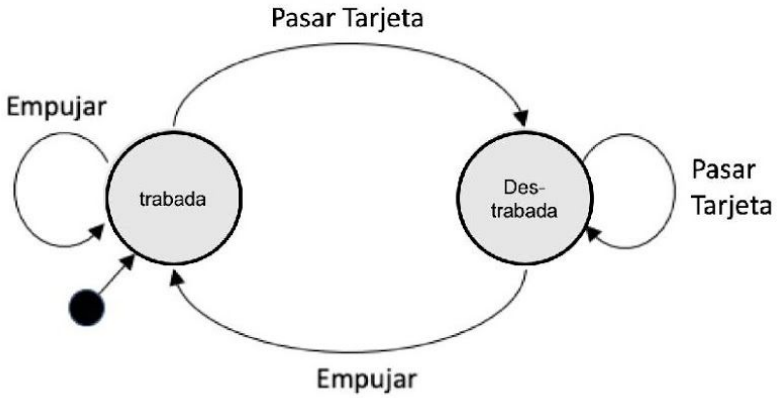
1	2	3	4
No aprendí nada Nuevo	Capaz que aprendí algo	Aprendí un poco	Aprendí Mucho

Algun otro comentario para los creadores de juegosmobiles.org? Ivon, Grace, Olivia y el equipo de WPI:

¡Muchas gracias por contarnos de vos, y contarnos de tu juego!
Ahora, Respondé las preguntas que siguen lo mejor que puedas.
Si no sabés, no te preocupes --estimá la respuesta lo mejor que puedas.

Primer Nombre:_____Inicial del Apellido: __ Colegio:_____
Grado:_____ Edad:_____ Varon?/Mujer? Fecha:_____

Este es un diagrama visual de cómo funciona una máquina, que involucra un lector de tarjeta (como para pasar una tarjeta de crédito).



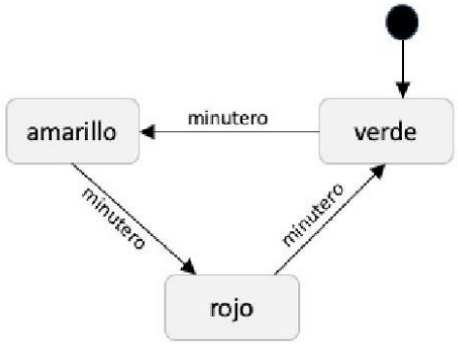
15a.) ¿Qué tipo de máquina podría ser esta? ¿A qué te recuerda y por qué?

15b.) De acuerdo a este modelo, ¿qué pasaría si pasaras tres veces (3) la tarjeta? ¿Se puede?

15c.) Si la máquina está **trabada** en este momento... ¿Qué tendríamos que hacer para que esta máquina se **destrabe** y luego se vuelva a **trabar**?

Primer Nombre:_____**Inicial del Apellido:** __ **Colegio:**_____
Grado:_____ **Edad:**_____ **Varon?/Mujer?** **Fecha:**_____

16) Este es un modelo (un diagrama) de cómo funciona un semáforo. Este semáforo cambia las luces en función de un sensor de tiempo: un minutero, que cambia el color de las luces cada un minuto.



Un **sensor de movimiento** es parecido a un minutero, pero en vez de producir un evento a cada minuto, sólo produce un evento (se activa) cuando el sensor percibe que algo se mueve. Algunos semáforos pasan de rojo a verde cuando un sensor de movimiento detecta que llega un auto cerca del semáforo.

Queremos cambiar el funcionamiento de este semáforo para agregar la posibilidad de que, cuando un sensor de movimiento detecte que llegó un auto, inmediatamente se ponga en verde. Después de eso, reanudaría su funcionamiento normal. ¿Podés volver a dibujar el diagrama para representar este nuevo comportamiento del semaforo? ¿Qué sería necesario agregar?

Primer Nombre:_____**Inicial del Apellido:** __ **Colegio:**_____
Grado:_____ **Edad:**_____ **Varon?/Mujer?** **Fecha:**_____

17) En la plaza de juegos de un shopping hay un trencito para chicos. En la cabina del trencito hay dos botones para manejarlo y una pantalla, como se muestra a continuación. Tu tarea es diseñar un diagrama (similar a los anteriores) para representar el comportamiento de la pantalla del tren, y cómo cambia cuando se aprietan los botones mostrados.

Dibujá un diagrama (al estilo de los anteriores) que represente el funcionamiento de la pantalla digital del trencito, y cómo cambia esta pantalla dependiendo de cuál de los dos (2) botones se presionan, de esta manera:

- Al presionar el botón con esta flecha:  , la pantalla digital mostrará: "Marcha Adelante".
- Al presionar el botón con el cuadrado:  , la pantalla digital mostrará: "Parado"
- El botón  también se puede presionar cuando ya el tren esté andando (en cuyo caso seguirá mostrando “Marcha adelante”).
- El botón  también se puede presionar cuando el trencito ya está parado (en cuyo caso la pantalla seguirá mostrando “Parado”).