Engaging Students with Adaptive Learning (ESAL)

An adaptive e-learning tool to teach young students beginner computer science topics

Abstract: *Adaptive e-Learning is a field that is growing more prevalent yearly. It's an educational method that is focused on computers customizing the class material to individual students. With extra research and development time a teacher can produce a tool that can adapt to each student's individual learning styles. By focusing on how the student learns and retains information the system will offer intelligent hints, leading the user to a full understanding of the subject. Keeping a user from becoming overwhelmed by a subject will ensure their continued participation and learning throughout the process. This style of system is generally designed towards higher education and specialized skills, which often utilizes outside technology to allow for more user information. We can develop a system similar to this that targets young students who are interested in Computer Science. The system will be a lightweight adaptive application and will focus on how the user interacts with a puzzle, not on how they interact with the screen. The application will be developed for use in iOS and a mobile apps that will guide students through several Computer Science topics. The system will try to emulate the student's' zone of proximal development and merge it with the three models of adaptive learning. By following the model of "Hour of Code" and adding in adaptive e-learning features we will create a system to help young students engage with computer science such that they feel supported. This project aims to show the viability of lightweight adaptive systems in educational apps to aid students engage more thoroughly with the material.*

1. Introduction

Adaptive learning is a simple concept. Instead of treating every student the same, tailor how you present information to each student individually. This system allows students of different backgrounds, ages, and learning styles learn the same subject at similar paces. This system has been utilized from everything like simulating lab environments and teaching geometry.

Adaptive learning is clearly gaining more ground as classrooms around the world are gaining access to computers and tablets. There are three key parts to adaptive learning that must be fulfilled, but how you implement them is key to how you plan on tailoring your education system. First is the expert model, which is the total information that exists to draw from for your class. Second is the student model, which tracks the student through the class and tailors the expert model to the students needs. Finally is the instructional model, which is the part that actually conveys the information to the student in the way that is best suited to them. These three tools come together to present a constantly updating style of teaching for the individual student.

The goal of this paper is to examine different styles of feedback systems being used with the adaptive learning technique. We will look at several types of feedback and how they integrate into the three models of adaptive learning. Crucial to this was looking into existing systems. Knewton and Integrated Education Solutions (IES)[1] are two of these systems that helped inspire ESAL. Both of these companies are working on adaptive systems in classrooms. Knewton's system is very focused on the feedback given to the student and the teacher. Throughout this paper there will be references to a work in progress iOS application that is being developed in unison with this paper. This app will be implementing the adaptive learning models and the feedback systems discussed throughout the paper.

2. Related Works

Throughout my research I relied several papers to help me better understand the pre-existing research on this topic. There is plenty of existing research in the adaptive learning field, but the most helpful research for me came from companies who published white papers (a comprehensive report from the company on their research). One of these white papers, from Integrated Education Solutions helped me understand the basic systems of adaptive learning and the three models that work together to make a fully adaptive system. The other white paper is from Knewton, and talks on the feedback systems that go into an adaptive system. Most importantly what feedback is most helpful and when to present this feedback.

An Adaptive Educational System For Higher Education[2], covers the adaptive system from the classroom/archaic view and how technology can help improve this. *Is Adaptive Learning Effective? A Review of the Research*[3], is a look at other existing work and seeing if adaptive learning is a style worth continuing. Their take away was that mainstream adaptive learning is currently too difficult to pursue but instead focus on augmenting existing classrooms with assistive learning systems. *P - An Open Source Personalization Platform for Higher*

---

1 Oxman Steven, & Wong William. (2014). White paper: Adaptive learning systems. *DeVry Education Group,*

2 Martins, C., Faria, L., & Carrapatoso, E. (2009). An adaptive educational system for higher education. *Knowledge Engineering and Decision Support Research,*

3 Verdu, E., Regueras, L., Verdu, M., Castro, J., & Perez, M. (2008). Is adaptive learning effective? A reveiw of the research. *Conference on Applied Computer & Applied Computational Science,*

Education[4], talks more about the data tracking side of adaptive learning and how institutions can utilize this data to implement large scale adaptive learning. *The Evaluation of a Moodle Based Adaptive e-Learning System*[5], examines an implemented adaptive system utilizing the Moodle. Since, I have used Moodle this paper helped understand how improving a system I am familiar with is possible. *Web Intelligence and Artificial Intelligence in Education*[6], covers the artificial intelligence aspect of adaptive learning and how education will be affected by it. Finally, *SSCLS: A Smartphone-Supported Collaborative Learning System*[7], looks at the traditional lecture based system and how it can be improved by smartphone usage to increase engagement.

3. Methods:

Adaptive learning is the idea of breaking up a curriculum into three models. The first being the expert model, the master location for all the material for the subject, the student model, the way of tracking the student's current skill level, and the instructional model, that takes the information from the student model and uses that to determine what information from the expert model is needed. The system used in ESAL is not a strict adaptive system, it is an light adaptive learning system. There is no strict expert model used and the student model utilized is focused on determining which puzzles the student is given. This is done by tracking their performance puzzle by puzzle compared to their overall performance.

Within each puzzle the adaptive system will track several pieces of user information. The most important is number of moves to complete the puzzle, this will let the puzzle know if the student is being efficient or not. Next is the number of attempts to complete the puzzle, which lets the adaptive system see if they are firmly grasping the concept or need some repeat puzzles. These types of input are helpful for the system to use, and make up the main feature of the adaptive system, but there are several other pieces that add to the students learning. With each puzzle there are several stages of help that the student can receive. This option will be hidden until the student has spent a certain amount of time on the puzzle and failed to complete the puzzle a couple of times (failing meaning the input they put in ended with the character bumping into the wall or not reaching the goal). Once they click the button there will be several general tips that will increase in helpfulness as they press the button and struggle with the puzzle.

The key to the adaptive system is to accurately simulate each students zone of proximal development. This ZPD is where the student is being challenged by the work, but only so much that they need a few tips to succeed. Each students ZPD will vary on their performance in the puzzle as it is compared to their overall performance. This system will not accurately be able to simulate different styles of learning, and that is one area where this research can be furthered once the initial goal has been met. Overall the adaptive system is aimed to make the student feel engaged and challenged so they can perform at their best.

---

4 Kellen, V., Recktenwald, A., & Bumgardner, C. (2010). P - an open source personalization platform for higher education. *University of Kentucky Academic Planning,*

5 Surjono, H. D. (2014). The evaluation of a moodle based adaptive e-learning system. *International Journal of Information and Education Technology, 4*(1)

6 Devedzic, V. (2004). Web intelligence and artificial intelligence in education. *Educational Technology & Society, 7*(7)

7 Chuang, Y. -. (2015). SSCLS: A smartphone-supported collaborative learning system. *Telematics and Informatics, 32*(3), 463-474. doi:10.1016/j.tele.2014.10.004 © 2014 Elsevier Ltd. All rights reserved.

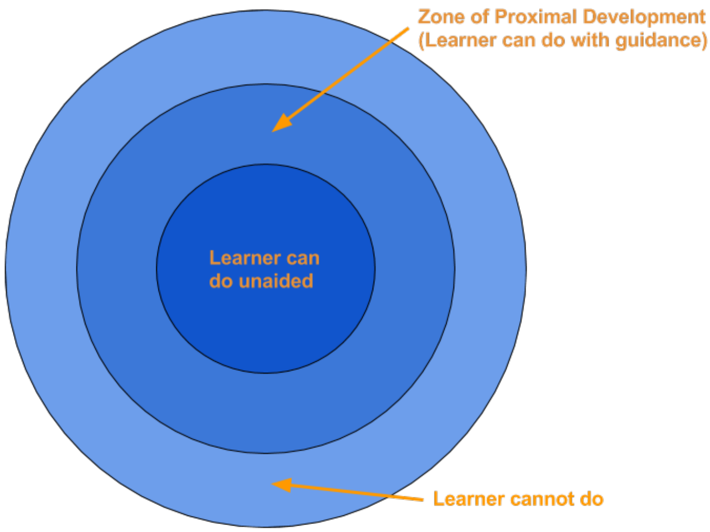

Diagram 1: This diagram demonstrates the principle of a student's ZPD. Finding the sweet spot between challenge and support.

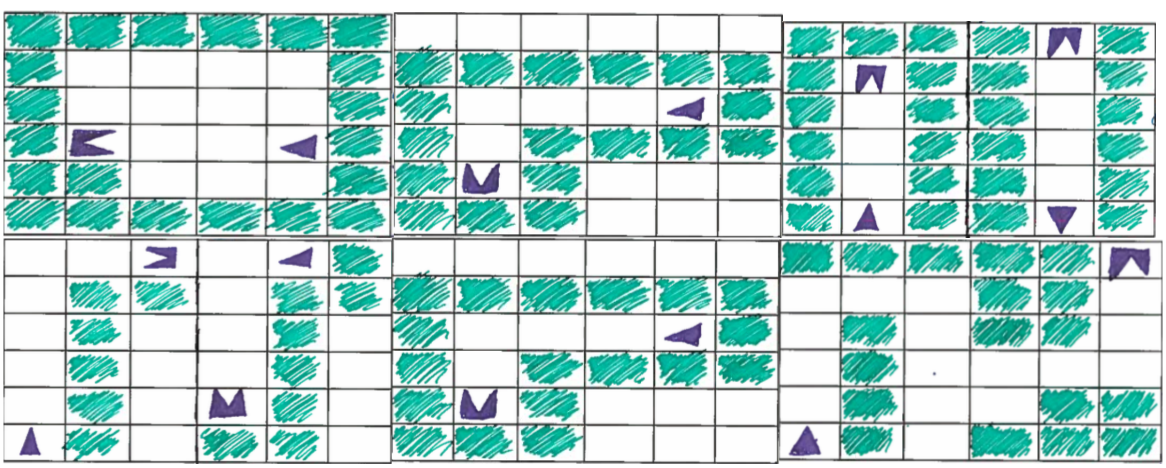
Diagram 2: Early designs of the levels to be used in ESAL to teach game rules and then for loop principles.

4. Development:

The adaptive system was developed in Unity 2D for iOS. The scripting was done in CSharp. The game is still in very early graphical shape, but the groundwork for improvement has been laid. Development started with creating a grid system that would allow the user to control the character within the confines of the grid. The next challenge was implementing the walls so that the player's character could not pass through them. After this came implementing the goal and recognizing when the player had completed their task. These were the basic game functionality that needed to be implemented before the adaptive system could be started.

To start the adaptive system was just tracking how many frames there were between each mouse click. This was chosen because when the user clicks on a button they have made a decision and are no longer wondering how to solve the problem. After generating the average, per level, button click time, generating the number of attempts was next. After this was the length of time on the whole level. These were all generated and used to decide which level the student was given next. The next adaptive feature was the ability to compare the user's path against several predetermined paths. This allows the system to see if the student is engaging with the puzzle in the way they are meant to. Next was a feedback system to help the student see how they are doing. This feedback system is the hints they are given throughout the puzzles. These hints are read in from a text file and displayed when the user clicks on the hint button. This is in place so that if a student is really struggling they will be able to be given a direction to think in, but it will slightly detract from their overall performance according to the adaptive system.

The students performance with the adaptive system is simplified down to an integer. This integer is created by comparing the student's performance against their average from previous levels, their path compared to the predetermined paths, and the number of hints they used. The students performance effects which level they are presented with next. If they have a high performance, used few or no hints and their path was the optimal path, then they will be given a level two higher in difficulty. This is the redundant lesson system that is designed to increase how well the students retain information.

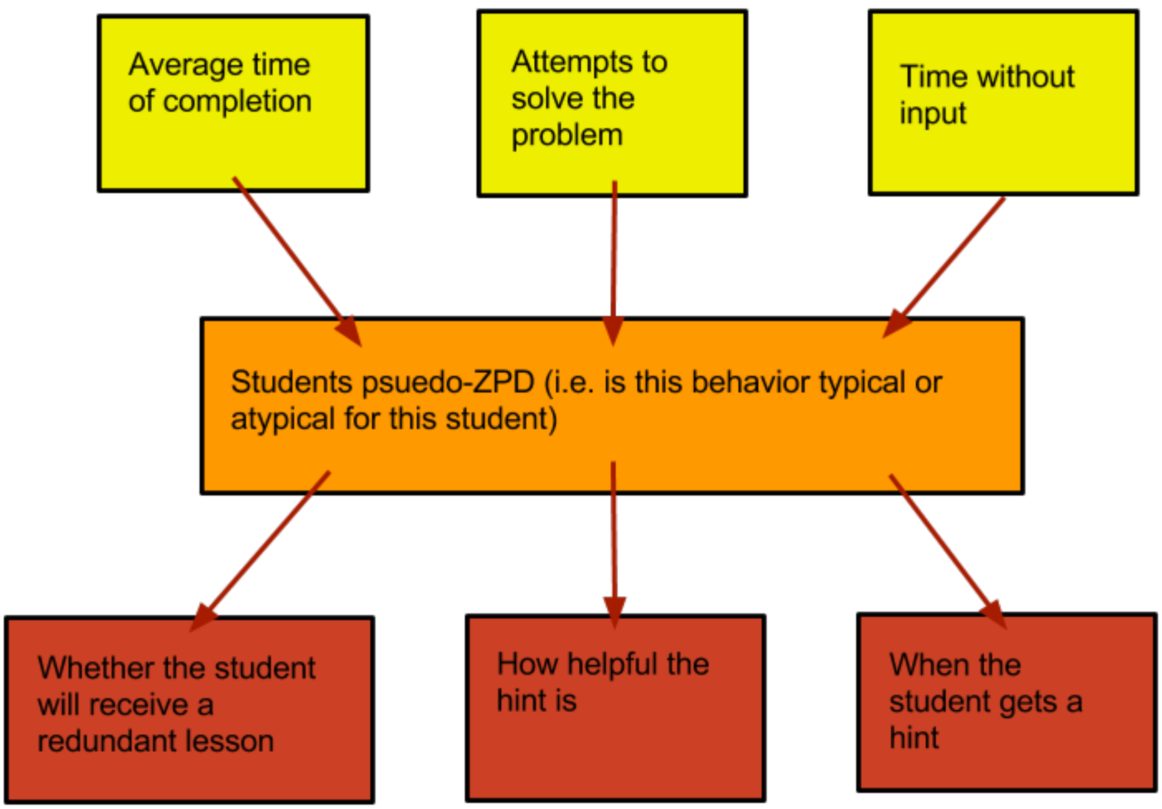

Diagram 3: Layout of how ESAL will simulate the student's ZPD using the lightweight adaptive learning principle

5. System Model:

How this adaptive system works over the course of the full lesson plan is simple. Upon completion of a puzzle the results will be written out to a text file. This allows for the user to update their performance every puzzle. What this means is that at the start of each puzzle the adaptive system will read in the text file. From the text file it will take in the average time between clicks, the average attempts per puzzle, the average time spent on a puzzle, etc so that as the system can have averages to compare against. Also at the beginning of each level the system will read in that level's hints and best paths, again so the system can compare the student against the proper system. These systems are designed to allow for easy scalability in the future. Allowing for more options and greater consistency in the system learning.

Students play the puzzles on their iOS compatible devices. Currently ESAT is only geared towards iPhones, so no iPad use is available. The buttons and text field are all mobile touch compatible and allow for students to play this whenever comfortable.

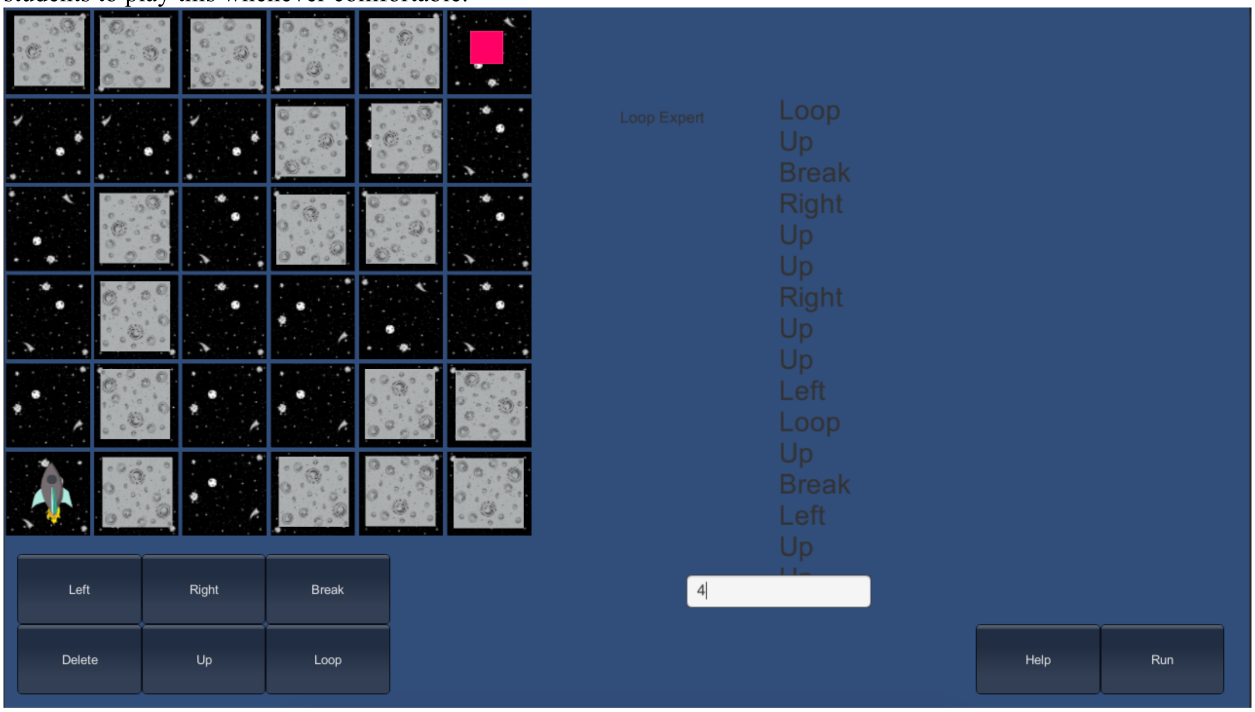

Diagram 4: Screenshot of the Unity in editor engine version of ESAL. The scaling of the GUI is off since the implementation of adaptive/feedback systems was tackled first

The game was designed to be accessible by all ages, primarily elementary school kids, so schools could use this tool with a variety of grades. The graphical design of the game is geared towards children who have a desire to explore. Space may not be the most neutral of themes, but the desire to explore space leads into the desire to explore computer science.

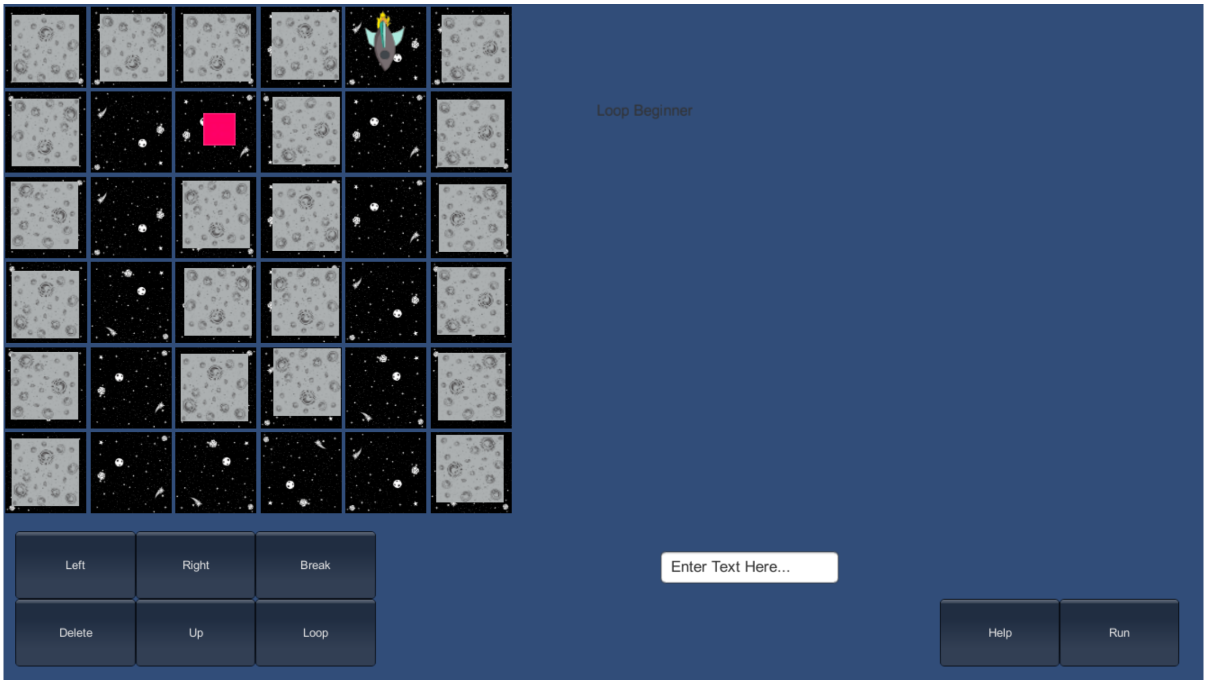

Diagram 5: Screenshot of ESAL on an iPhone 6

6. Conclusion:

This research does not currently have any definitive results attached to it. This paper serves as the recording of an implementation of a lightweight adaptive e-learning system. In the future this research can be expanded to include a user study, but as it stands there needs to be more front end development to polish the game. The goal of this research was to show the viability of lightweight adaptive e-learning and demonstrate an implementation of it. Above is the process that ended in ESAL, which hopefully is a tool that serves as proof of the objective. Lightweight adaptive e-learning is not a complicated tool to use and many systems would be improved with it. There is no "one size fits all" adaptive system, so any group wanting to create an adaptive system needs to plan out their system and what features can be improved by studying the user.

During implementation of ESAL there were several features that were more challenging to implement than other. Ignoring the development struggles with Unity and CSharp, most of the features were rather simple to implement. Designing a system to account for all the features and using that information properly was the biggest challenge. In the end the idea of a simple value to represent the students performance seemed simplest. This code can be seen in appendix 1. All of these values come together to decide if the student jumps ahead a lesson, moves forward one, stay on the same level, or regresses one. Sadly there was not enough time to implement several lessons of the same difficulty, and so for this version of ESAL, the adaptive system will simply have the student replay the level. The principles designed in ESAL are simple yet effective, showing that a little effort into designing a system like this can pay off immensely.

With education and computer science, there is a growing need for younger students to be exposed to this subject. Many schools are now adding computer science classes, but families may want to let their kids learn at a younger age, or may not be in an area that has such a program. Tools like ESAL are the answer to this problem. A fully fledged adaptive e-learning system would be able to ignore a student's race, gender, or age to give that student the help they need. Lessons that a student can carry around in their pocket and will remember what this student struggles on the most are the future of adaptive e-learning. Mobile tutors that supplement teachers or tutors, so that the student feels they are supported in and out of the classroom. Adaptive learning is just beginning to be a widely utilized idea and the applications for it are immense. Hopefully in the coming years there will be several tools similar to ESAL created so that this idea of private tutors in your pocket takes off.

Research by Philip Winchester
Advisor: Professor James Lee
Connecticut College Computer Science

Appendix 1: The function that compares the user's performance on a specific level to their overall average performance.

```
//compare average number of button clicks against level number of button clicks
    if (clicks < adaptClicks) {
      performance += 2;
    } else if (clicks == adaptClicks) {
      performance += 1;
    } else if (clicks > adaptClicks) {
      performance -= 2;
    }
    adaptClicks = adaptClicks * currLevel; //make it psuedo total
    adaptClicks = adaptClicks + clicks; //add new value
    adaptClicks = adaptClicks / currLevel; //reaverage it
    rewriteStr = string.Concat(rewriteStr, "Click: ", adaptClicks.ToString(), "\n");

    //compare number of attempts made to complete this level against average
    //comparing float and int
    if (numAttempts < adaptAttempts) {
      performance += 2;
    } else if (numAttempts == adaptAttempts) {
      performance += 1;
    } else if (numAttempts > adaptAttempts) {
      performance -= 2;
    }
    adaptAttempts = adaptAttempts * currLevel; //make it psuedo total
    adaptAttempts = adaptAttempts + numAttempts; //add new value
    adaptAttempts = adaptAttempts / currLevel; //reaverage it
    rewriteStr = string.Concat(rewriteStr, "Attempts: ", adaptAttempts.ToString(), "\n");

    //take in number of hints used comapre against average?
    if (numHelp == 0) {
      performance += 2;
    } else if (numHelp == 1) {
      performance += 1;
    } else if (numHelp > 2) {
      performance -= 2;
    }
    rewriteStr = string.Concat(rewriteStr, "Hints: 0\n");

    //compare average time to complete this puzzle against level completion time
    if (Time.frameCount < adaptTime) {
      performance += 2;
    } else if (Time.frameCount == adaptTime) {
      performance += 1;
    } else if (Time.frameCount > adaptTime) {
      performance -= 2;
    }
    adaptTime = adaptTime * currLevel; //make it psuedo total
    adaptTime = adaptTime + Time.frameCount; //add new value
    adaptTime = adaptTime / currLevel; //reaverage it
    rewriteStr = string.Concat(rewriteStr, "Time: ", adaptTime.ToString(), "\n");

    //comparing average button click time against level button click
    if (averageTime < adaptButton) {
      performance += 2;
    } else if (averageTime == adaptButton) {
      performance += 1;
```

```
} else if (averageTime > adaptButton) {
   performance -= 2;
}
adaptButton = adaptButton * currLevel; //make it psuedo total
adaptButton = adaptButton + averageTime; //add new value
adaptButton = adaptButton / currLevel; //reaverage it
rewriteStr = string.Concat(rewriteStr, "Button Time: ", adaptButton.ToString(), "\n");

rewriteStr = string.Concat(rewriteStr, "Level Attempts: 0\n");

System.IO.File.WriteAllText ("Assets/Resources/averages.txt", rewriteStr);

if (performance >= 9) {
   print ("Jumping Ahead!!");
   //Load 2 levels ahead
   UnityEngine.SceneManagement.SceneManager.LoadScene (currLevel + 2);
} else if (performance > 5) {
   print ("Moving Ahead!!");
   //Load next level
   UnityEngine.SceneManagement.SceneManager.LoadScene (currLevel + 1);
} else if (performance > 2) {
   print ("Let's Retry!!");
   //reload same level
   UnityEngine.SceneManagement.SceneManager.LoadScene (currLevel);
} else {
   print ("Let's go back and relearn!!");
   //load prevoius level
   UnityEngine.SceneManagement.SceneManager.LoadScene (currLevel - 1);
}
```

References

Adaptive Learning, K. (2011). Knewton adaptive learning, building the world's most powerful education recommendation engine. *Knewton White Paper,*

Chuang, Y. -. (2015). SSCLS: A smartphone-supported collaborative learning system. *Telematics and Informatics, 32*(3), 463-474. doi:10.1016/j.tele.2014.10.004 © 2014 Elsevier Ltd. All rights reserved.

Devedzic, V. (2004). Web intelligence and artificial intelligence in education. *Educational Technology & Society, 7*(7)

Kellen, V., Recktenwald, A., & Bumgardner, C. (2010). P - an open source personalization platform for higher education. *University of Kentucky Academic Planning,*

Martins, C., Faria, L., & Carrapatoso, E. (2009). An adaptive educational system for higher education. *Knowledge Engineering and Decision Support Research,*

Oxman Steven, & Wong William. (2014). White paper: Adaptive learning systems. *DeVry Education Group,*

Surjono, H. D. (2014). The evaluation of a moodle based adaptive e-learning system. *International Journal of Information and Education Technology, 4*(1)

Verdu, E., Regueras, L., Verdu, M., Castro, J., & Perez, M. (2008). Is adaptive learning effective? A reveiw of the research. *Conference on Applied Computer & Applied Computational Science,*

Research by Philip Winchester
Advisor: Professor James Lee
Connecticut College Computer Science