May 2012

Application of Artificial Intelligence in a Game-like Tutor

Hao Zhu
Worcester Polytechnic Institute

Follow this and additional works at: https://digitalcommons.wpi.edu/iqp-all

Repository Citation

This Unrestricted is brought to you for free and open access by the Interactive Qualifying Projects at Digital WPI. It has been accepted for inclusion in Interactive Qualifying Projects (All Years) by an authorized administrator of Digital WPI. For more information, please contact digitalwpi@wpi.edu.
APPLICATION OF ARTIFICIAL INTELLIGENCE
IN A GAME-LIKE TUTOR

Interactive Qualifying Project Report completed in
partial fulfillment
of the Bachelor of Science degree at
Worcester Polytechnic Institute, Worcester, MA

By:
Hao Zhu

Submitted to:
Professor Joseph E. Beck, Advisor

Date:
May 13, 2012
# Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ABSTRACT</strong></td>
<td>1</td>
</tr>
<tr>
<td><strong>1 INTRODUCTION</strong></td>
<td>2</td>
</tr>
<tr>
<td><strong>2 BACKGROUND</strong></td>
<td>8</td>
</tr>
<tr>
<td>2.1 CHALLENGE</td>
<td>8</td>
</tr>
<tr>
<td>2.2 THE “UNLIMITED CEILING” EFFECT</td>
<td>9</td>
</tr>
<tr>
<td>2.3 INSPIRATION AND STIMULATION</td>
<td>10</td>
</tr>
<tr>
<td>2.4 ATTENTION SPANS</td>
<td>11</td>
</tr>
<tr>
<td><strong>3 DESIGN 0</strong></td>
<td>13</td>
</tr>
<tr>
<td>3.1 INITIAL IDEA</td>
<td>13</td>
</tr>
<tr>
<td>3.2 DESIGN</td>
<td>17</td>
</tr>
<tr>
<td>3.2.1 Motivation</td>
<td>17</td>
</tr>
<tr>
<td>3.2.2 Flow</td>
<td>19</td>
</tr>
<tr>
<td>3.2.3 Play</td>
<td>21</td>
</tr>
<tr>
<td>3.2.4 Reflection</td>
<td>23</td>
</tr>
<tr>
<td><strong>4 DESIGN 1</strong></td>
<td>24</td>
</tr>
<tr>
<td>4.1 MAIN FOCUS</td>
<td>24</td>
</tr>
<tr>
<td>4.2 CHANGES</td>
<td>25</td>
</tr>
<tr>
<td>4.2.1 Roles Reversed</td>
<td>25</td>
</tr>
<tr>
<td>4.2.2 Degenderization</td>
<td>26</td>
</tr>
<tr>
<td>4.2.3 “Cheating” Dice</td>
<td>32</td>
</tr>
<tr>
<td>4.2.4 Modification of Difficulty Selection</td>
<td>33</td>
</tr>
<tr>
<td>4.2.5 Feedback System</td>
<td>34</td>
</tr>
<tr>
<td>4.3 DESIGN</td>
<td>36</td>
</tr>
<tr>
<td><strong>5 DESIGN 2</strong></td>
<td>39</td>
</tr>
<tr>
<td>5.1 CHANGES</td>
<td>39</td>
</tr>
<tr>
<td>5.2 DESIGN</td>
<td>41</td>
</tr>
<tr>
<td>5.2.1 Dynamic Difficulty Adjustment</td>
<td>41</td>
</tr>
<tr>
<td>5.2.2 Game Play</td>
<td>47</td>
</tr>
<tr>
<td>5.3 FEEDBACK</td>
<td>49</td>
</tr>
<tr>
<td>5.3.1 Setup for Testing</td>
<td>49</td>
</tr>
<tr>
<td>5.3.2 Feedback Received</td>
<td>51</td>
</tr>
<tr>
<td><strong>6 DESIGN 3</strong></td>
<td>54</td>
</tr>
<tr>
<td>6.1 CHANGES PROPOSED WITHIN THE GROUP</td>
<td>54</td>
</tr>
<tr>
<td>6.2 CHANGES PROPOSED BY STUDENTS</td>
<td>57</td>
</tr>
<tr>
<td><strong>7 FUTURE WORK</strong></td>
<td>61</td>
</tr>
<tr>
<td><strong>8 CONCLUSION</strong></td>
<td>64</td>
</tr>
<tr>
<td><strong>9 REFERENCES</strong></td>
<td>66</td>
</tr>
<tr>
<td><strong>APPENDIX: STUDENT FEEDBACK FORM</strong></td>
<td>68</td>
</tr>
</tbody>
</table>
Abstract

In this project, my team developed a guiding framework on the application of Artificial Intelligence in a game-like tutor, with focus on the creation of a computer opponent to stimulate interest in learning. We also managed to come up with a functional tutor prototype featuring human-computer competition based on this framework. By presenting the prototype to a group of students, we received feedback showing that most target users like having a computer rival in a tutor and will use our tutor as a study aid in the future. As a result, we were able to finally conclude that the proposed framework is both feasible and effective in guiding the development of tutors involving competition elements supported by Artificial Intelligence.
1 Introduction

Schools today have too many demands: high public expectations for education, different types of learners, technological challenges, and competition from other schools. In fact, these are all issues that schools and school leaders can by no means avoid. High expectations for students reflect the realities of the modern society. Students come from increasingly diverse homes and backgrounds, which indicates that they are less likely to be motivated by using the same method for all, as what teachers did traditionally. In addition, technology not only creates new learning opportunities for teachers, students, and parents, but also requires a certain level of understanding, especially from instructors, to use the right technology in education. Finally, the exploding education market is proposing new service delivery schemes, introducing more fierce competition, and expanding choices available to students, parents, teachers and administrators. However, if we look beyond appearance to the essence, motivation becomes the key to solving all the above problems: How to motivate distinctive leaners to study hard to help them meet expectation? In what way can modern technologies, such as Internet, be utilized to motivate students and thus gain more competitive advantages? However, these questions still lack clear and convincing answers.

Our group realized the potential of motivation to bring drastic changes to the current education system as well. Therefore, we decided to start finding out more about this topic in the context of educational software by researching how we could make educational software more motivating, while still retaining the attractiveness to students.

So what can make students want to study a subject outside of class time? In fact, this is where the idea of video games and the many different elements they employ come
into play. What are some techniques that game designers use to motivate players? Can these be used to motivate and teach students different subjects? From these questions we hope to find a solution in the form of a tutor that includes special features to entice and entertain students while still teaching them and honing their speed and comprehension.

The fact of the matter is that further melioration in the motivation aspect is called for in our current educational system. Between Attention Deficit Disorder (ADD) and generally shorter attention spans, students need something quick and entertaining to keep their interest. So how do we make something that will motivate students? Can we borrow elements from activities students enjoy? Using elements from video games to create something that will challenge pupils and keep their attention long enough for them to learn the material was what we considered to be a good idea. So our project finally invested in using the popular elements in games to create something both entertaining and educational at the same time.

First and foremost, what are game-like elements and why did we choose to implement them? In general, game-like elements refer to those elements that are utilized by commercial games, such as reward systems, Artificial Intelligence (AI), hints, achievements and so on, but can also be used in other software to enhance interactivity and entertainment. In the case of a tutor, these game-like elements can present different motivators and each has the potential to be used to create an interesting and educational experience. Games as a means of entertainment hold the attention of millions of people around the world. The idea to integrate the elements used in video games into educational tutors as a way to attract students to learn should work as well. The big questions thus became: What elements can lend themselves best to the educational side
of the equation? How should educational software designers properly adapt them to an educational environment?

As a result, choosing an element from games that would work best became a significant decision point of the project. There are so many successful ways that games appeal to multiple audiences and mind sets that choosing one is very difficult. The challenge was finding something interesting that could lend itself to teaching. Focusing on how to truly engage students, Artificial Intelligence was the element we kept coming back to when trying to make our decision.

Defined as "the study and design of intelligent agents" (Poole, Mackworth, & Goebel, 1998) where an intelligent agent is a system that perceives its environment and takes actions that maximize its chances of success (Russell & Norvig, 2003), AI is so broad than can be applied in numerous different areas and we’ll mainly discuss the application of AI in education here. In point of fact, the scope of AI in Education is not very clear-cut, so we’ll mostly focus on definitions in this section. One common way of defining AI in Education is” any application of AI techniques or methodologies to educational systems”, which summarizes educational AI in general. Other definitions focus more narrowly, for example: any computer-based learning system which has some degree of autonomous decision-making with respect to some aspect of its interaction with its users (Self, 1995). This second definition stresses the requirement that AI techniques should be able to reason at a certain level while interacting with the user. The reasoning might be about the subject being taught, about the best teaching approach, or about misconceptions or gaps in a student's knowledge. Additionally, there are wider ways of involving AI in teaching. For example, AI in education is seen to be defined as
“the use of AI methodologies and AI ways of thinking applied to discovering insights and methods for use in education, whether AI programs are involved at the point of delivery or not” (Naughton, 1986). Despite the seeming contradicting perspectives, these views in practice form a comprehensive description of AI in education: What all of them have in common is that the design principles of the systems are substantially derived from, and expressed in, the language of Artificial Intelligence (Self, 1995).

Consequently, to make our tutor more fun and more effective, we hoped to find an appropriate way in which AI can be used in an educational environment, possibly in the form a computer opponent common to commercial video games. The benefits of including an opponent in our project are that the opponent not only allows us to test competition and how users reacted to it, but also creates a sense of interaction not so prevalent in tutors and practice systems, although we still needed to experiment and see whether presenting an opponent would motivate students to keep using our tutor.

Using the AI element implies that we would be able to create competition, a motivating factor throughout time, in our tutor. Humans are competitive by nature. In terms of games, competitive elements can even become the determinant of enjoyment in playing computer games and factors that foster the selection of computer-games as offers of social competition, in which the user competes against an opponent that is controlled by the computer (Vorderer, Hartmann, & Klimmt, 2004). On a basic level, competitive elements can be incorporated into our proposed tutor system as the system is designed to be interactive, which allows for active engagement of the user in the playing process and for immediate feedback on user’s actions. On a broader level, the user’s feeling to play against an opponent likely evokes a competitive situation that should be especially
capable of both engaging and involving the user. Therefore, it appears reasonable to add competition as a factor in our tutor to help guarantee enjoyment and students’ preference of using our tutor.

In this particular case, we believe that by adding a computer opponent supported by AI, our users are provided with a simple yet consistent goal: beat the computer. We assume that students would enjoy competing against a computer opponent instead of simply answering question on a computer-based quiz outside of class. Students will not just answer question passively, but be stimulated while using the system and have the desire to win in the competition. This way, the ultimate goal of helping them to learn more in a more efficient way is also achieved. Nonetheless, it still requires more observation for us to answer the question about whether or not competition is a positive or negative influence on education. There have been connections between education and competition in the past. Students tend to compare themselves with their peers and judge their performance based on other students around them. Giving them an opponent that needs to be outsmarted might be a good way to involve them in a healthy competition with an emphasis on education, but requires a careful and delicate blending of gaming and learning.

Finally, as for what would the competition be and how would the students interact with their opponent during the problem solving process, we expected to find our answers through stages of designs. But since we were focusing on math as a subject for our tutor, we developed an initial idea of creating competition of speed and correctness, which specifically is about who (the player or the computer opponent) can solve the problems the faster. The idea of a speed competition is to push the players to be prepared for many
different problems and eventually prove their handle on the subject during the competition. Unlike a quiz or a test, our tutor offers them a fun environment in which they think of what they have learned in class within a short period of time and organize the knowledge in a meaningful way to catch the opponent, instead of to get a good grade. The computer opponent serves a measure of how well they complete the above task, but the measure is not the same for all players but a highly personalized one. We speculate the situation would be if users lose to their opponent, they will want to be faster so they can win next time. Unlike most games, there is no shortcut to victory and luck doesn’t help much either; to win the users need to hone their knowledge of the subject. We deem that students would be stimulated by failures in the tutor to study harder and be encouraged to learn more by the sense of success provided by the game.
2 Background

Recently, researchers have started to dive deeper into solving the issue of how to integrate the tried-and-true old-school teaching methods with modern technology. The idea of developing a system that implements game-like elements into the classroom could change education forever. But why games?

2.1 Challenge

First, a main driving factor in why using game-like elements could help modern education is that people respond to challenge. Students in the classroom are no exception. They certainly take action when presented with challenges. Consequently, games have the potential as an educational aid since games actually have the ability to present students with a vast variety of challenges and also entertainment.

True learning and understanding of material occurs when the level of challenge or difficulty is appropriately met with the level of attainability. What happens when a middle school child is in class is presented with a problem that he or she views as too challenging? We strongly suspect that the child will declare the problem impossible, decide the teacher has given them an unsolvable problem, and figuratively "check-out" without truly attempting the problem. An excellent way to overcome this roadblock is through the use of game-like elements. When presented with an enormously challenging task in a game, it is seen as much more attainable. In a game, if the students do not succeed in the first attempt, they have the ability to go back and try again. The bar may seem outside their grasp, but they usually feel that with some practice, they will be able to reach it.
Game-like elements are also a great idea due to the fact that they change the way a child views errors and mishaps. They are not viewed as failures, but instead they are viewed as opportunities to improve. Quizzes are meant for teachers and students to gauge the understanding of a subject in school. But that is not all that comes from quizzes and exams. Students also have an emotional reaction to these tests. If a student receives an "A" on their math exam, it provokes them to feel good and gain confidence in the subject. Now what happens when the student's friend receives a "D" or an "F" on the same exam? The student is filled with emotions telling him/her that they have failed and will not be successful moving forward. What if we could keep the positive reinforcement of receiving that "A" and get rid of the negatives that come along with being unsuccessful? The answer may lie within games. When a child is unsuccessful in a game, they do not right away feel like a failure, they almost always view it as an opportunity to try again, improve their skills, and ultimately be successful. Educators should start to consider implementing a game-like feel into their various evaluation techniques to encourage their students to feel like they have a chance to improve, rather than allow them to always view their errors as failures.

2.2 The “Unlimited Ceiling” Effect

Another element that could be taken from modern gaming is the "unlimited ceiling". Normally, students most probably finish the homework once and for all. If they end up with an imperfect or even a poor grade for the homework, most likely they will not be interested in doing it for a second time to confirm their understanding of the material. Of course, teachers can demand that those students should redo the assignment; however, as
students are not indeed motivated to do so, the ceilings of their learning results tend to be very limited.

This is where games can make a difference. Games have an unlimited ceiling when it comes to education because of the characteristic of replayability. Replayability refers to the fact that if students feel unsure on the matter, they simply go back and play again. The student gets a new set of problems on the same material. This in turn allows the student to gain maximum exposure to the material. They can play and replay the game until they feel they have a much stronger understanding of the subject. This whole idea of the unlimited-ceiling is expected to help students gain maximum proficiency within a subject.

2.3 Inspiration and Stimulation

An important factor that must be kept in mind is the ability to inspire and stimulate the students’ minds. What does this mean for an educational game though? A student should not sit down and feel bored or obliged to continue when playing the game. There must be an aspect of the game that makes the user feel good and want to continue the game. One way to do this is to have a system in the game that varies the difficulty of each question as the questions are presented. In simple terms, the game should provide some questions that are intended to be much easier to boost the student's confidence as well as provide difficult questions that force the student to challenge his or herself outside their comfort zone and inspire them. Students respond to both of these aspects, challenge and success. A good way to ensure the student receives this effect is to have the game provide simpler problems at the beginning of each new round. By providing simpler problems at the start of each new round, the game strives to boost the user's
confidence. This gets the user feeling good and puts them in a positive state of mind moving forward. The problems should vary in difficulty; however, the overall difficulty should be generally trending towards a more challenging approach. This allows students to experience the feeling of success after solving easy problems as well as challenge when facing hard ones. But what measures these questions as more or less difficult? How to quantify the varying proficiencies of students? Ideally, an educational game would keep track of each individual student's capability. This would then allow the game to provide tasks that involve varying levels of difficulty and incorporate proper challenges to account for each student's individual skill level.

### 2.4 Attention Spans

Children in the United States have relatively short attention spans compared to the length of a classroom lesson. Based on personal experiences, we assume our target age group's classes last for 40 to 50 minutes. This is much longer than the average middle school students’ attention span which varies from about 8 to 14 minutes (Pugliese, 2008). Attention spans of modern students, however, are generally elongated when using games. If games are such an obvious answer, it’s surprising huge successes are not seen in educational games. After some more research, we find the explanation lies in that either the game is too educational and students lose interest too quickly, or the game lacks enough educational content to be seen as valuable in the eyes of the educational community. Based on these findings, we believe that a good way to be successful in developing educational software is to design a tutor system that is not only educational, but also has well-designed and entertaining game-like elements. Therefore, we’d like to
establish a competitive advantage against other educational applications by especially focusing on the game-like aspect with the use of artificial intelligence.

Another way of keeping the students' interests and elongating their attention spans is to have the game cheat. What kind of cheating could help students stay interested in an educational game? One form of this would be rubber-banding. Basically rubber banding ensures that the student does not gain an excessive lead on the CPU opponent. This technique is used in many of today video-games such as "Mario Kart" or "Burnout". When the player gets out to a large enough lead, the game has an internal algorithm that helps the computer controlled opponent catch up to the player. This promotes competition. Applying this concept to an educational game could improve the game's interest and encourage more students to use the game more often and for greater lengths of time, thus leading to more exposure to the material at hand.
3 Design 0

Design 0 was our initial design. This would be our starting point and the basis for all subsequent designs. After completing the design, Design 0 was presented to a boardroom of our peers and advisors.

3.1 Initial Idea

The idea of adding an AI opponent to a math tutor was inspired by a word-forming computer puzzle game named Bookworm Adventures, which combines the "create words from sets of letters" aspect with several elements of a typical computer role-playing game. It is certainly a source of entertainment, but includes an educational purpose of vocabulary expansion as well. These aspects are very similar to what we would like to embed in our proposed math tutor with game-like elements.

In the game, players guide Lex the Bookworm through a number of stages, battling creatures along the way. The stages are arranged in order of difficulty so that player can gradually adapt to the pace of the game while still feeling the joy and challenge. As for the gameplay, each battle consists of Lex squaring off against a given enemy. Both Lex and his adversary have health meters (represented by a number of hearts), which, when depleted, signal defeat. However, unlike more traditional role-playing games where players might injure their opponents with arms or magic, forming words damages enemies in Bookworm Adventures.
Words are formed from a grid of available letters. The longer the word that is formed, the more damage is done to opponents. Similarly, words generated using letters that are less common do more damage than those using only common letters. Each turn, letters in the grid are updated and players can form a single word by selecting them, while enemies use one of their available attacks to injure Lex, heal themselves, or otherwise make the battle more difficult. Lex automatically recovers all of his health between battles.
Figure 3-2 Boss battle


After a certain number of battles in the same chapter are won, a more difficult boss enemy is encountered. This makes the game even more exciting and challenging. And if players defeat the boss, they complete the stage and are rewarded with a treasure. Treasures provide special abilities to Lex, such as a reduction in damage inflicted to him, or more damage generated from words containing certain letters. In some cases, rather than receiving a new item, an existing item is upgraded. After the player has accumulated more than three items, Lex must then choose three of them to bring along on later chapters. Such a reward system not only gives players a sense of self fulfillment, but also
serves as a highly personalized difficulty adjusting mechanism since players always choose those treasures that match their playing styles best.

Figure 3-3 Treasure system

(http://1.bp.blogspot.com/-Yl8trt5uhYA/TmFgbkDm7yI/AAAAAAAARD4/LYDMZTeyly8/s640/Bookworm+Adventures+Volume+2+3.jpg)

So after a complete analysis of Bookworm Adventures, we find several features that can be deployed in our game design. First, we will modify the computer opponent in the above game setting to make it more applicable in an educational environment. We are considering using AI to dynamically adjust the intelligence of the enemy so that students of different math skill can experience similar levels of challenge. Second, we would like
to have “battles” in our tutor, but they should be simple and fast. The “battles” of Bookworm Adventures are stimulating and visually appealing for a developmental game with a focus on entertainment, but they might not be appropriate for an educational tutor with a pure goal of education. What we want to do is to reduce the entertainment portion, which might distract students from the educational purpose of our game, and to make the fighting mode more straightforward to the extent that math questions become the only highlighted part. Last but not least, we also borrow the idea of a reward system and “chapters” from Bookworm Adventures. These are not necessary, but they are expected to encourage players to spend more time in the tutor.

3.2 Design

In our initial tutor design, we defined the elements involved in a systematic way by referring to the integrated model for educational game design proposed by Brad Paras and Jim Bizzocchi (2005), which highlights motivation, flow, play and reflection.

According to Paras and Bizzocchi (2005), an effective learning environment should not only include reflection in the process of play but also produce an endogenous learning experience that is motivating.

3.2.1 Motivation

So we first start with describing what motivation means in the setting of a tutor. For tutor designers, to motivate a user means to simulate his or her interest in the system and thus engage in the intended process of knowledge gaining or skill development. In fact, this is mostly the same as what games want to achieve, except that the ultimate goal of a tutor is to educate users instead of to entertain them.
To be more specific, there are two kinds of motivation that can be used in a tutor, extrinsic and intrinsic. In general, an activity is said to be intrinsically motivated if there is no obvious external reward associated with the activity. Conversely, an activity is said to be extrinsically motivated if engaging in the activity leads to some external rewards like food, money or social reinforcement (Malone, 1980). In our tutor design, we decide to use both motivation methods together to maximize the motivation effect.

In terms of intrinsic motivation, we came up with a storyline and a clouded adventure map to complement the development of the story.

![Clouded adventure map](image)

Figure 3-4 Clouded adventure map
Based on Sharon DeVary’s work (DeVary, 2008), challenges and obstacles that are woven into a strong learning-related adventure develop higher-order thinking. Therefore, we plan to let students experience a fun story and at the same time solve math problems in hope of using the story to encourage learning and thinking. The story doesn’t need to be realistic, but should be related to math in some way so that we can naturally integrate problem solving into the whole process.

Moreover, in any given instructional situation, the learning task needs to be presented in a way that is engaging and meaningful to the student, and in a way that promotes positive expectations for the successful achievement of learning objectives (Small, 1997). In this case, we introduce the clouded adventure map component into our design. The map consists of several sites representing various chapters, which require different skills and knowledge. Users are supposed to enter the sites in a predetermined order. Each time they beat the opponent in a certain site, the cloud over a new site will be removed and then players will be allowed to explore it. We believe this can entice students to spend more time in the tutor and explore the “hidden world”.

3.2.2 Flow

Csikszentmihalyi (1975) introduced the flow state through the study of people involved in activities such as rock climbing, chess and dance. Flow describes a state of complete absorption or engagement in an activity and refers to the optimal experience (Csikszentmihalyi, 1991). And Chan et al. (1999) better illustrated Csikszentmihalyi’s idea by writing that flow explains a phenomenon that many people find themselves experiencing when they reach a state of feeling a perfect balance between challenge and frustration, and where the end goal becomes so clear that hindrances fall out of view. To
better balance the game experience, flow-like experience is something else that we take into consideration while designing the system. Past research has shown that the flow state has a positive impact on learning (Webster, 1993), and therefore should be taken into account in the design of digital learning materials, which in this case refers to tutors. In practice, flow-like experience has something to do with the degree of complexity of the tutor. If the challenge is significantly greater than player’s skill level, he or she may feel anxiety; in contrast, if the challenge is significantly lower than player’s skill level, the player may feel bored (Kiili, 2005).

To address this issue, we implemented the difficulty selection system which allows user to select a preferred difficulty level before entering the gaming process and the dynamic difficulty adjusting which modifies the degree of game difficulty based on the skill of a player determined by a predefined algorithm multiple times through the gaming process together in our tutor, to make the challenge that a player faces suitable given the skill set of the player. Basically players have a chance to adjust the difficulty setting each time before a new chapter starts, so that they don’t face the same level of challenge on materials of distinct familiarity levels. This also adds personalization to the user experience. In addition, once they begin to contest with the computer opponent, our system will continue to change the difficulty based on students’ actual performance and the difficulty level selected previously.
3.2.3 Play

As for the actual game play, our aim is to combine the arrangements mentioned above with a creative game play experience. To achieve this, we used the four key attributes of educational game Lepper and Malone (Lepper, 1987) came up with as a guideline for the coordination of the different systems. First, according to Lepper and Malone, games must introduce challenge. Through goal reaching and feedback, the learner should continually feel challenged as difficulty increases in concordance with increased skills. This corresponds to the dynamic difficulty setting mentioned before, which directly controls how the opponent performs in the game. Second, the game
should create sensory and cognitive curiosity within the learner. We try to achieve this goal by having the clouded map and developing the creative gaming mechanism of chasing by rolling dice. We think having part of the map covered with cloud can arouse players’ curiosity and the desire to explore. And making players try to chase and catch the computer opponent, namely the little monkey in the mock-up below, is also thought to be quite innovative compared with traditional tutors. These two characteristics are expected to increase users’ interest in using the tutor and thus make the tutor more effective. Third, the learner should feel a sense of control throughout the game. This is accomplished by having the user-controlled difficulty selection system, which is common in modern games and is a good way to enhance game experience. Fourth, games should use fantasy to reinforce the instructional goals and stimulate the prior interests of the learner. To do this, we design cartoon characters and have an imaginative storyline that links all the chapters together.

Figure 3-6 Game play
3.2.4 Reflection

Last but not least, we were thinking of developing an explanation system that gives the player a chance to reflect on points they don’t understand well and consequently better gain the intended knowledge or skills. Nevertheless, the problem of an explanation system lies in that it seems to have a break-up effect on the flow of the game.

After some careful research, we managed to find a solution with reference to Kristian Kiili’s theory (Kiili, 2005), which is the endogenous implementation of reflection. Specifically, in educational game design it is important to ensure that learning takes place within the realm of play, even if learning is only made possible through reflection. To obtain such effects, reflection must appear to the learner as one of the many in-game goals that drive the game-play. As a result, we decided to link the explanation system closely with the game play by making a cartoon teacher referee give the results and the explanation in a conversational manner. And in later phases, we would test players on similar problems to confirm understanding.
4 Design 1

Design 1 was the first Design that included our initial design and ideas presented to us through feedback from presenting Design 0.

4.1 Main Focus

The main idea that we looked into while updating our design was based on a gameplay style called the “Predator vs. Prey” style. The primal concept behind this style is that the user is controlling either the “Predator” or the “Prey” and is either trying to catch up to or escape from the opponent. This is a basic concept that has been used in games for many decades. Examples of such a concept can be dated back to the earliest games, such as “Pac-Man”. Even though “Pac-Man” is such a simple game, it still keeps players interested 30 years later. This could to a large extent be attributed to its general theme of “Predator vs. Prey”.

The primary goal or challenge within these games must not only seem challenging, but also must feel attainable. If the goal is too attainable and too easy, then the player will likely become bored and stop playing the game. However, if the game is too challenging, then it becomes less interesting and dissatisfying. These are pivotal concepts to keep in mind when designing games based on the “Predator vs. Prey” game-style; the player should feel the pressure and excitement whether playing predator or prey. Based on the above thoughts, our team decided to commit to the “Predator vs. Prey” game-play style. We believe that students will be more interested in an adventurous game involving a computer operated opponent with artificial intelligence instead of simply answering homework or quiz questions on a piece of paper.
4.2 Changes

After presenting our initial Design 0 to our peers and advisor, we decided to make some alterations to different aspects of our design. These changes are based on the feedback we receive from our presentation as well as research conducted subsequently.

4.2.1 Roles Reversed

Potentially the biggest change that was implemented when transitioning from Design 0 to Design 1 was the reversal of the roles within the game. As we stated in section 4.1, we decided to continue implementing the “Predator vs. Prey” game-play style, however, we decided to change the character’s roles in the game. In Design 0, the user would control a game character whose pet monkey was constantly escaping and running away. In this setup the user would be considered the “predator” while the opponent would be the “prey”. After having it brought to our attention, we realized the plot line would be very difficult to write and it might also be hard to create the desired level of pressure in the game. This prompted us to change this dynamic.

In Design 1, we would have the user control a monkey as their primary character. Larger, more intimidating creatures such as large monkeys or dragons would chase the monkey. Such role switch would provide a better base for later write a background story and plot line to the game. These opponents would constantly chase the user’s monkey and the user would have to use skills learned in the classroom, namely solving math problems, to advance their monkey and evade the creatures chasing the monkey. This in turn reversed the previous roles of “predator” and “pray”. In this updated version, the computer opponent would now be viewed as the “predator” while the user would control the monkey which would be considered the “pray”
4.2.2 Degenderization

4.2.2.1 Theory

Nowadays, interactive media and technologies, especially video games, are playing a more and more significant role in children’s life. According to a national survey conducted by the National Institute on Media and the Family (NIMF), 92% of children and adolescents age 2-17 play video games (National Institute on Media and the Family, 2002). However, a substantial gender difference in computer games has been observed regarding both involvement and preferences even with the use of digital games being on the rise. A report published by Kaiser Family Foundation claims that on any given day, 44% of boys report playing video games compared to 17% of girls. One possibility to explain this situation is the fact that the majority of commercial games are still being designed, developed, and marketed with the male player in mind, in spite of efforts to accommodate female gamers such as games by HerInteractive, Girl Games, Girltech and Purple Moon (Ibrahim, Wills, & Gilbert, 2010). In other words, most game developers and publishers don’t pay enough attention to how differently each gender plays and what their preferences are in games where there actually exists a wide gender gap. This could be elucidated by the fact that males are fond of games by natures and hence are thought to be more marketable.

An important factor causing the gender gap is violence. Whenever you look through a top sale list for video games, you can always find that most, if not all, of popular games targeted at teens and adults present an abundance of violent actions displayed a high degree of realism. Game producers might tend to think that most girls are not gamers by nature, which also means they’re not as marketable as boys. As a result, they begin to put
more and more element designed for boys in their games, making girls less and less likely to play them. Such vicious cycle makes the gender gap even larger. In reality, this kind of observation corresponds to research on media genre preference, which has demonstrated that males are more interested in violent entertainment than females are (Slater, 2003). On the other hand, females tend to display a very low preference for observing or participating in conflicts and their resolutions through violence (Bussey & Bandura, 1999) and find non-violent entertainment, such as comedy or sad films, more attractive (Oliver, Weaver, & Sargent, 2000). As is said previously, many computer games do not take females' preference for non-violent content into consideration. As a result, the lack of suitable non-violent games in the market might shed light on the reason why women and girls generally involve less with games.

Other than violence, boys and girls are found to treat competition in games distinctively. Research in sports psychology shows that girls find competitive activities less attractive than do boys and that girls are sometimes afraid of participating in competitive sports. It is reasonable to hypothesize, then, that some females would avoid competitive computer games as well. This hypothesis can also be confirmed by the observation that boys always feel the desire to “beat” the opponent and win the game, while girls often just take their time going through the game and experiencing the story, which seems more about establishing emotional attachment with the characters and events in the virtual world. In a word, boys generally expect competition in the real game play whereas description about conflict in the story telling might be enough for girls. Then it will be a rather difficult task for us to fine-tune the level of competition to make boys excited and girls comfortable. There are also similar findings informing us on how
each gender manages conflict both in and out of a game: while the male generally resolves “a problem by direct confrontation with a decisive win-or-lose result”, females usually “choose negotiation, diplomacy and compromise” (Ray, 2004).

With regard to specific game theme or topic choice, Kafai (1998) discovered that boys tend to design games themes that allow them to “get something” through a pursuit or adventure exploration. By contrast, girls create games that involve “doing something” without finding objects. The game is the activity itself.

As to the characters appearing in games, two-thirds of the characters were male (64%) and the other one-third were either nonhuman (19%) or female (17%). Males dominated as player-controlled characters (73%), and even nonhumans (15%) outnumbered female characters (12%) for players to control (Kaiser family Foundation, 1999). In a game designed for both genders such as our tutor, this could indicate problems: female users might have low attraction and find it hard to imagine themselves in the virtual word with the lack of female characters, and their enjoyment and playing motivation could be reduced in consequence. Therefore, we’re thinking of letting users choose the gender of game character.

Concerning visual stimuli, the generally considered panacea in games, also have different effects on both genders. It is observed that even though both genders do respond to visual stimuli, their reaction is different. Males tend to show a physiological reaction but females need an emotional or tactile stimulus to elicit the same response (Ray, 2004). In addition to that, girls prefer a “rich texture phenomenon” which includes audio and expressive graphics as well (Miller, Chaika, & Groppe, 1996).
Finally, in terms of game play mechanism; achieving or beating a score is certainly the most common one to proceed through a game. However, do girls really like such kind of mechanism? To answer this question, Miller et al. (1996) discovered that girls view winning as not as important as the “experience” of playing the game. Turkle (1986) wrote about similar result in his paper, saying that males are more likely to take risks and experiment while females tend to seek understanding before trying. Since many games simply employ “trial and error” mechanics, it is not surprising that we end up with a relatively low amount of female gamer. So obviously, in order to attract girls, a game should use more gender-neutral mechanisms.

4.2.2.2 Implementation

During the actual design stage, we plan to degenderize our tutor basically on the five aspects mentioned in Kafai (1998)’s article to mix and compromise factors catering to respective gender. The five aspects are game genre, game world, game characters, game feedback and game narrative.

4.2.2.2.1 Genre

First, we reinforce the decision of choosing adventure as the game genre. According to Kaiser Family Foundation (Kaiser family Foundation, 1999), among 8-18 year olds, the three genres that dominate children’s video game playing are action or combat (42%), sports (41%), and adventure (36%). Boys who play computer games are more than three times as likely to play action or combat games compared to girls who play computer games (27% v. 8%). They are also more likely than girls to play sports or competition
games (23% v. 9%). Hence, adventure becomes the optimal solution to mitigating gender difference due to game genre.

4.2.2.2 World-style

Second, based on Kaifa’s experiment, boys prefer fantasy or virtual world while girls prefer realistic settings such as the space of the home. In this facet, we’re considering using fantasy, but limiting the settings to those of a typical fairy tale, which we think will not make girls feel uncomfortable.

4.2.2.3 Player/Character

Third, Rachel Karniol et al. (2000) stated that children tend to anthropomorphize animal characters and develop affective reactions to them. This implies that an animal character, if properly designed, can be a good candidate for our tutor. Additionally, in a study with animal characters (Arthur & White, 1996), younger, but not older, children were found to assign their own gender to bear characters. Yet even though older children do not assign gender to animal characters in line with their own gender, their preferences do reflect their own gender. According, we finally decided on starring a little monkey in the game story and let the users decide its gender.
4.2.2.2.4 Feedback

Fourth, we want to match our feedback system with girls’ general liking by making it nonviolent. Namely, we would like it to be mild and cartooned (for example, see Figure 4-4). This kind of nonviolent modality does diminish the feeling of excitement within the game, but it’s expected to help keep female users.

4.2.2.2.5 Narrative

Fifth, narrative will be provided to support the storyline. Sarah Joy Bittick & Gregory K.W.K. Chung (2011)’s research confirmed that the use of narrative in educational video games has the potential to increase student engagement and learning outcomes, especially for males. Thus, we’ll make use of narrative to attract boys.
After all the above steps have been completed, we would have a gender inclusive tutor structure ready for more competitive elements.

4.2.3 “Cheating” Dice

In our original design, upon answering a question correctly, the player and/or opponent would be allowed to roll an animated die that would produce numbers between 1 and 6 at random. After deliberation and receiving feedback on this feature, we decided
to have the dice “cheat”. What does that actually mean though? Instead of the dice producing numbers at random as it was in our initial design, the dice would produce numbers to promote competition. If the player began to gain a substantial lead over the opponent, he/she would be more likely to roll a lower number. Conversely, the computer-controlled opponent would be more likely to produce a roll with a higher value.

On the other hand, the dice can also be used to elongate the game. Why would we want this? If the computer opponent begins to gain on the user too quickly then the game may end in the player being captured too quickly. We want to encourage the students to play regularly, so it is most likely in our interest to ensure that the character is not caught too soon as this might discourage the student from continuing to play. It may make the game feel too challenging and not so player-friendly. Overall we feel that giving the game an internal system to control the competitive nature of the game using “fixed dice” will strongly benefit the tutor system.

4.2.4 Modification of Difficulty Selection

As stated in Design 0, players could choose their own difficulty. This was our original idea to encourage the user to play at a difficulty they were comfortable with. This idea had some previously unconsidered issues. Students would have the option of choosing to face challenging opponents or much easier opponents. However, our eventual goal is to create a ‘smart’ tutor that presents different students with different goals that are both challenging and attainable. With this in mind, we decided we would now implement a hidden scoring system to evaluate a player’s performance. The system would then use the score to influence how challenging the opponent would be. Students who were struggling would see less challenging questions and opponents that would not
pressure them as aggressively. Students who were excelling would see many more challenging questions whilst his or her opponent applied added pressure. Using this system, all players would get a good distribution of challenge and attainability with respect to their proficiency level. Players might still be able to select the difficulty they would like to start with, but complexity of problems would be dynamically determined as game advanced.

4.2.5 Feedback System

First, we removed the concept of taunting from the game. In our previous design, upon answering a problem incorrectly, the challenger would playfully taunt the user. The main reason we decided to remove this was the fact that we had no clue whether our users would like it or not. We were very unsure how students would react to a computer opponent poking fun at their mistakes and were concerned that some students might become offended or discouraged, which was just an unnecessary risk for us. To fix the problem, we chose to remove the presence of taunting from within the game.

Along with the removal of taunting, we got rid of the presence of the “Wise Goat” character (see Figure 4-3). His role in the game was to provide the user with an explanation as to why they got a question wrong if both the opponent and player provided incorrect answers. He was removed for similar reasons as taunting. We were uncertain about students’ reactions to being lectured, especially in the context of a game. Some students might feel put down being lectured by a virtual goat in the game.
Another aspect of the feedback system that was changed for Design 1 was the reward system. We found that with our game changing so much, the reward system would likely have to be altered as well. However, upon completion of the general concepts of Design 1, we did not have any concrete ideas for the reward system in the game. Design 1 would be presented with no true reward system, but we had a plan to receive outside input concerning views on possible reward systems in the future.

A new concept introduced with this design was the use of a hint system. If the player was taking a long-time to answer a question, their monkey would have a sudden thought bubble containing a hint to assist the student (see Figure 4-4). This hint would try to give the student a basic starting place for the problem. We felt this would be a useful element within the game to assist struggling students with problems that were too
challenging. This hint would pop-up after a set time. This meant the game would need an internal clock to decide when to give the hint to the student.

![Image: The 2 lines are perpendicular, does that mean the angle is 90?](image)

**Figure 4-4 Using a hint**

### 4.3 Design

The actually game-play of Design 1 would not vary too far from Design 0. Design 0 would be updated using the changes previously discussed; a new mock-up was created
based on the role reversal of the opponent and the player (see Figure 4-5). The concept of a chased would still be the basic layout of the design. The other basic aspects, such as the mapping system, would remain unchanged from Design 0. We created a new mock-up for the mapping system (see Figure 4-6), but the overall function and concept of it would remain unchanged.

Figure 4-5 Design 1 mock-up
Figure 4-6 Design 1 map mock-up
5 Design 2

In our second design we focused our efforts on creating a working design for our target audience. We were given the opportunity to present this version to a group of students, and being able to present to a group of students gave us some really useful feedback and helped us hone our ideas.

On top of creating a working version, we had some really useful feedback on Design 1 that allowed us to better focus on the AI element of our project. In Design 1 one of our problems was not being able to fully focus on our initial concept of competition. We stripped some of the features and decided to take a serious look at what we could do with AI, which brought out the idea of dynamic difficulty adjustment.

5.1 Changes

Originally our concept was a player created character chasing his or her pet monkey. Though through feedback from Professor Beck, we were convinced to change to a predator versus prey approach, where the player is the monkey being chased by a larger predator of some kind. It seemed very trivial but it was a change that helped us better integrate AI elements with the game.

Specifically, this new idea created many upsides from a design perspective. The player feels a much more urgent need to remove their monkey from the danger of being harmed by this much larger creature. This sense of urgency generates tension as the player is trying to escape their opponent. It also creates a much more interesting story for us to create the game around. Players’ no longer running from a human but a large enemy instead allows us to change out the pursuer and create new worlds for the monkey to be traversing. There is also no confusion of which character the player should really be
helping; a more natural connection to the underdog is present. A simple swap of roles changes the game completely and lays the ground works for Design 2.

This change showed us just how many different elements were implemented into our original designs. We would have to adjust or simply drop some of them from the newest version. So we started thinking about what we needed and what was just getting in the way of our original goal. What we ended up realizing is that AI had become one of multiple focuses in our project when it should have been the only one. In our previous designs there were multiple game-like elements that our group focused on, such as rewards and feedback. Although these were interesting mechanics for the player, it caused the core idea to fall to the wayside. The core idea behind our project was to use artificial intelligence to teach Math skills in a game like environment.

We decided as a group to remove the feedback and rewards from the game. The feedback was a focus of a different IQP group making it not worth the effort for us to also try and create a system for it. If we wanted to return the feedback system in a later version, we would first talk to the feedback group and possibly implement their ideas first. The reward system, which we added as a way to motivate players to keep playing the game, seemed to be also beyond our scope. Much like the feedback system there were other groups working on a reward system that we could possibly use in the future. With all these helping to decide the main direction of our project, we started developing more ideas directed towards AI and what we could do with it.

We were pondering making the AI more applicable to different students in the form of an algorithm. Besides, we were thinking about writing a different storyline that would fit the new style to make sure that students keep coming back and always try to be faster
and beat the AI. The goal for the player and how we wanted to use our game-like element were all changed to reflect the new play style of the game. This small change pushed gave us greater ideas of how to use AI in a much broader way than we were before.

Looking at possible ideas for how the AI would compete with the player, we came up with the idea of using dynamic difficulty adjustment. This is a system where the AI will get more successful as the player does the same and vice versa. We used the AI in the Mario Kart series as an example when first trying to develop this. In those games, the computer player acts like a “rubber band” in that the AI will fall behind but then quickly shoot up based on better power ups and a slightly boosted speed.

The purpose of this system was to keep the player from losing easily but also from beating the opponent to quickly and just breezing far ahead of it. For example if the player were to do well and get a sizeable distance away from the opponent then the opponent would proceed to answer more and more questions faster and get better dice rolls than the player, allowing it to catch up. The opponent would essentially be cheating the dice making the players come up with worse outcomes so it could get closer. We wanted the opponent to behave based on the player’s performance, stressing that urgency and constantly creating tension between the player and opponent.

5.2 Design

5.2.1 Dynamic Difficulty Adjustment

5.2.1.1 Why is Dynamic Difficulty Adjustment needed?

Game balance is a concept in game design describing fairness or the balance of power in a game between multiple players or strategic options. A game is called
unbalanced if one or more players have an unfair advantage over the others (Newheiser, 2009). In actuality, if a game does not meet a player’s appropriate level of game difficulty, it cannot become a real success. However, in many games nowadays, the degree of game difficulty is very uniform or static. Static difficulties result in people easily feeling bored of the game when the game is too easy for superior players or put novice players in frustration if the game is too difficult for them. This also holds true for our tutor. As a result, to provide players with the appropriate level of difficulty in a game, we need to balance of game difficulty. And this is actually the AI component comes into play. By integrating an AI algorithm with our tutor’s game play, the degree of game difficulty can be modified based on the skill of a player without even letting the player realize it. Moreover, there are three advantages associated with adapting the proposed algorithm:

- Developers, or even teachers with adequate knowledge, can change the challenge level of the tutor easily. Students are either pushed harder in the gameplay or given more tolerance of errors, depending on what the actual expectation of them are. But students are not likely to know what in fact has been changed.

- Players enjoy a more engaging gaming process. There is less probability of their losing their interest in the tutor before they practice much and have a better understanding of the intended knowledge. Thus, the lifecycle of the tutor is extended and the effectiveness is also boosted.

- AI provides a useful viewpoint of the gameplay. As many levels of difficulty are included in order to match up the level of players, players enjoy richer game-playing. Although the AI sector is not as easy to notice as other sectors, such as graphics and
audio, it is essential in terms of user experience, which is to make the gameplay more interesting and abundant.

### 5.2.1.2 How will Dynamic Difficulty Adjustment be implemented?

Before embedding the AI algorithm into the tutor, we decided to first write a demo in Java. Below is the core of our source code regarding difficulty:

```java
final int TURN=10;
final int DISTANCE=4;//initial distance between player and opponent
Random randomInteger=new Random();
int arrive;
int escape;
int catchup;
int num;
in player=DISTANCE;
in ai;
in arr[]=new int[TURN];
in cat[]=new int[TURN];

private void jButton1MouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    int dice=randomInteger.nextInt(6)+1;
    if((dice<=3) && ((player-ai)/25<11/25))//We assume fitness level to be 11/25
    {
```
dice=randomInteger.nextInt(6)+1;
player+=dice;
}
else
player+=dice;

jLabel1.setText(Integer.toString(dice));
jLabel7.setText(Integer.toString(player));

if(num==9&&player>ai&&player<25)//Little monkey escapes
{
    escape++;
    jButton1.setEnabled(false);
    jButton2.setEnabled(false);
    JOptionPane.showMessageDialog(null, "Escape successfully!", "Congratulations", JOptionPane.PLAIN_MESSAGE);
}

if (ai>=player)//Little monkey is caught
{
    catchup++;
cat[num]++;
    jButton1.setEnabled(false);
    jButton2.setEnabled(false);
    JOptionPane.showMessageDialog(null, "You're caught!!", "Sorry", JOptionPane.PLAIN_MESSAGE);
}
if(player>=25)//Little monkey arrive at destination
{
    arrive++;
    arr[num]++;
    jButton1.setEnabled(false);
    jButton2.setEnabled(false);
    JOptionPane.showMessageDialog(null, "Arrive Successfully!", "Congratulations", JOptionPane.PLAIN_MESSAGE);
}
else
{
    num++;
    JLabel3.setText(Integer.toString(num+1));
}

What the above algorithm does is comparing the player’s fitness level, which is supposed to be the player’s expectation of his or her performance in the chase, with his or her real performance. Based on the comparison result, the algorithm determines whether to cheat on dice or not to create an appropriate level of confrontation. In more details, when the user’s performance level is smaller than the predefined fitness level, and the throwing of the dice in the first run results in a number that might make the user uncomfortable about the competition, the dice will be thrown for a second time. Even so,
the result will still be random and not guaranteed to be the desired one. This way, there
still exists a certain level of uncertainty that makes the gameplay fun.

And to foresee the outcome of implementing this algorithm, we also conducted a
number of simulations with a computer. We assume an average student can get 60% of
all the questions right and here is the result from a simulation:

number of trials: 10000
arrive: 3742 escape: 1315 catch: 4943
arrive turn 1: 0
catch turn 1: 1962
arrive turn 2: 0
catch turn 2: 866

We can clearly see that there is a problem here: too many players fail in the first 2
runs. This is definitely something we don’t want in our tutor. Users should not fail just
due to bad luck when they are presented with something they don’t know as the first
question; they might be simply frustrated about this and develop a dislikes towards the
system. Therefore, we would like to find a way to give them more chance of “survival”
in the first few runs.

Possible solutions to the mentioned problem are: decrease the predetermined fitness
value; increase the initial distance between player and opponent; cheat more on dice;;
shut timer off for the first few problems. Among them, we think making the first 2
problems easy might be the most effective, so we’ll like to try it and see how it works in
the presentation to middle school students later.
5.2.2 Game Play

We didn't change how the player interacts with the system. They still are given multiple choices for the question all displayed in the white rectangle on the right. This system is just how we believe the students will best use their skills without being overly distracted by the game side of it. We want them to focus on doing the problems to the best of their ability with an added flair of visuals to keep them motivated.

Figure 5-1 Concept art for chase behavior for Design 2
Figure 5-2 Possible visual effect for an incorrect response by the player

Figure 5-3 Possible visual effect for a correct response by the player
Visuals such as the monkey tripping when the answer is incorrect and the monkey kicking dust back at their opponent when the player is correct were also added in the Design 2. We plan to use these visuals as cues when the character advances or loses ground, instead of just the player getting further away or closer as the player answer questions. These visuals also replace the feedback screens from design 0 and 1, allowing us to display right or wrong in a fun way instead of an overly critical one.

5.3 Feedback

As previously stated, Design 2 was field-tested on a group of middle school students.

5.3.1 Setup for Testing

Design 2 was our only design given to a test audience. The audience in question was a classroom of 7th and 8th grade students at Sullivan Middle School in Worcester, MA. Partway through the second term of the project, we were informed of a chance to show what we had accomplished to these students, so we began preparing a demo of our design for them. One day after the end of the term, two representatives from our team, Hao Zhu and Ethan White, presented our progress to the class.

Our demo took the form of a PowerPoint presentation simulating the game we had designed thus far. We created 10 question slides, slides corresponding to correct and incorrect answer for each of these questions, and connected the slide transitions based on which answer the students selected. All of the questions were based on content from Monkey’s Revenge¹. To approximate the AI aspect of our game, we included a 45-second timer in each question slide. When the timer expired it would be assumed the AI

¹ [www.gltutors.com](http://www.gltutors.com), a tutor with game-like elements developed by Dovan Rai of Worcester Polytechnic Institute
got the question correct before the student could, and the student would be advanced to
an ‘incorrect’ slide. Also, our group’s representatives each brought a laptop with the
above algorithm used to adjust the degree of competition between the player’s monkey
and the hostile gorilla. During this stage, we still needed to input into our algorithm
whether the player got the question right or not. Consequently, our plan was to manually
run the algorithm question by question for each student. At the conclusion of the
presentation, the students were shown images of the victory, loss, and map screen slides,
and informed of their role in the hypothetical game.

Unfortunately, there were several limiting factors on this experiment which kept us
from getting as much out of it as we might have otherwise hoped. When we arrived, after
introducing ourselves to the students and moving to the computer lab, we were given less
than an hour to show our demo to about 20 students in groups of 4 or 5 (there were also
other groups like ours with their own game demos, so students’ time had to be divided
accordingly). As a result, we were allotted 15 minutes for each group, which turned out
to be too tight to test our design and then receive thorough feedback on the game. In the
end, we actually only had time to test our game on two groups instead of three, since the
feedback collection phase took us much more time than expected. In the end, there was a
larger feedback session with the entire class as well.

Other than what’s said above, there were also limitations concerning our demo itself.
Though we made the best use we could of PowerPoint’s capabilities, it simply wasn’t
made to be as flexible as games generally are. Some of our design’s aesthetics and
structure were left out of the demo, such as moving pictures and start menus. Our AI
simulation certainly suffered, since it was just a static time limit for each problem. After
the demonstration, we determined that the 45-second time limit we used to approximate the AI might be too long. In fact, we didn’t know exactly how well-suited the students would be for the problems we presented them with, so we decided to be generous in this regard. We later discovered that these students were all advanced in math from different grade levels, so the problems we gave them were too easy for them to get a proper feel for the pressure of an AI competitor. For the most part, they solved every problem in a quarter of the allotted time, or less.

5.3.2 Feedback Received

During the testing, we asked the students a predetermined list of questions (also attached in Appendix A) to encourage more useful feedback. The first thing we asked was whether the students were challenged, and whether they had fun with the game. Unfortunately, few of the students were challenged, since we’d underestimated our test group when making the questions. As a result, we got a similar answer to two of our other questions: “What do you think of the AI's difficulty?” and “Do you feel you had too little or too much time to solve the problems?” However, this would be much less of an issue for a more developed version of our design, which would benefit from much more thorough research of its target audience. On the other hand, the students did react positively to our design’s concept; most agreed that it was fun. Their reasons for liking our demo were made clearer by their answers to the rest of our questions.

We also asked the students how they felt after getting a question right, or wrong in the demo. This was mostly to gauge the emotional response to the screens we’d prepared for when students got a question right or wrong: the former displayed the player’s monkey leaving the gorilla in a large dust cloud, while the latter showed the monkey
falling flat on its face. The question of how harsh the feedback should be when the students got a question wrong had generated some discussion when we were working on the design. Anyway, the students seemed to enjoy the result screens for each question, whether they’d got it right or wrong. We also asked the students whether they would use our game as a study aid, and many replied that they would.

Our last two questions resulted in the most interesting answers, as they led different students to give conflicting opinions. One was “What kind of character or story would you like to see in the tutor?” When answering this question, students tended to focus on the nature of the competition. Some liked the scenario in the demo, in which the student helps a monkey escape a gorilla. But others said they’d prefer to play the role of the gorilla chasing the monkey. The main difference between these ideas, in terms of the source of entertainment, is that the first would generate more of a feeling of desperation in the player, and increase the urgency of the gameplay; in other words, it would provoke a prey-like mentality. The second, on the other hand, puts the player in the role of the predator. It gives them the upper hand, and the competition becomes a matter of running down a weaker opponent. Since some students expressed a preference for each, either could be said to be a valid source of entertainment.

Our last question was “Are there any ways for us to make this more fun and helpful to you?”, which was another way of phrasing an invitation for general criticism. One point that arose from this was that of the AI’s timer’s visibility. Some students enjoyed the extra pressure that arose from not knowing when they would run out of time to answer the question. It seems leaving the timer ambiguous augments the motivation based on competition that was our primary purpose in focusing our design on an AI
opponent. However, another student was concerned that not knowing the timer might put too much pressure on the player. If our design were carried through to completion, figuring out how much stress resulted in the maximum amount of motivation would be important, so this type of feedback would require a lot of scrutiny.

There were also a lot of suggestions we received in response to our last question that could increase the appeal of our design. Multiple students mentioned that they’d like to see bonus levels, with more difficult questions and multiple foes chasing the monkey. Such an idea might be useful to keep the interest of students who feel that the regular levels are too easy; it would be low priority, but it would have a good chance of finding its way into the hypothetical finished product. Some students also mentioned that they’d feel more engaged if the game’s characters reacted more dramatically whenever the player answered a question. In the demo, getting a question right or wrong would just result in the player’s monkey going faster or slower, respectively. But one student suggested that perhaps the player’s gain in advantage could be represented in other ways, such as by having the monkey throw things at the gorilla. A finalized version of our design would need to employ many such concepts to maximize its aesthetic appeal.

Though our setup was flawed, and some of our questions allowed for simple ‘yes’ or ‘no’ answers, we were able to glean some useful ideas from our test audience. The most useful pieces of feedback were the ones related to the nature of the competition (whether the player controlled the monkey or the gorilla) and the question of whether the timer should be visible, since these are both important parts of our design’s core goal of engaging students in educational software. It would take a lot more testing to refine our project to the point where it would be ready for the public, but this was a good start.
6 Design 3

Design 3 was a conceptual design phase, where our group members proposed possible changes that couldn’t be actually implemented in this project due to time constraints.

6.1 Changes Proposed within the Group

With all of the feedback and ideas we received from working with the students, we still have a lot to improve and refine based on Design 2. Our biggest challenge with Design 3 will be trying to adjust the systems that we already have in place to better accommodate our target audience. From our experience with the students, we understand that our current system has a flaw----the inability to predict how much of a head start the player needs and how much leeway the AI must provide so the player won't lose too early. These flaws draw us towards revamping our system of dynamic difficulty adjustment.

The main goal with a dynamic difficulty adjustment system is to give the player a challenge but to make it a balanced experience. The Psychologist Mihaly Csikszentmihalyi created a chart of the proper flow that makes things enjoyable. So to put it into the context of games, the player needs to feel challenged but still focus on what the game is about and not just what's challenging them. So to do this we want the player to stay within the "flow channel"; as displayed below, there is a balance that needs to be found in order to create a flowing experience.
We need to configure a nice level of challenge since our current system either ends up with the player winning or losing to easily. So to find the proper flow we need to run simulations using something like excel and tweaking the values for the AI so their dice rolls are not likely to be too high or too low depending on their distance from the player. For example, if the computer is 5 units away from the player, we wouldn't want him to roll a 5 if he has just reached the player within 3 rolls, so our system will influence the rolls to give the player a fighting chance. Adjusting like this can drastically help the experience because we can also scale the players’ rolls without their knowledge so they don't get too far from the computer. So the player is 20 units from the AI, we adjust their roles to allow for tension between the player and AI to come back. The goal then becomes to find the values that give us a nice 70%-75% win ratio with the player. The
big problem we see is if players can't get ahead of the AI right off the bat then they will lose almost instantly, so we want to see if they can win in 5 rolls. We also can use this to adjust the starting distance between the player and the AI opponent, which can directly influence how fast the player can lose or how easy it will be if they have a large head start. These are all of the variables we need to adjust for in our simulation so we can find a good balance for the final system.

To add additional motivating factors for students with different preferences we would like to start working with the other groups and implementing their ideas. Using our combined knowledge of our game-like elements we could expand each other's concepts and create two systems that complement each other. For example one group handled the idea of a reward system that covered customization items. If we are to implement a similar system on top of our current AI system to allow customization of the player’s monkey character, it could add a further motivator to continue playing after mastering the subject matter. This continued play would allow users to further develop their speed and understanding of the subject matter after mastering the basics. Further implementation of other elements like hints could also help students that struggle with the material and possibly allow for a crutch that they can turn on and off as they see fit. These are all possibilities that could help shape our system to help different types of students.

There are also smaller systems we need to better or fully implement into our design. We want to implement a complete version of the map system, through which we can divide up the different math based subjects to make each level a focus. Doing this will keep the problems from being cluttered and allow students to go to different levels to
enhance their skills in the different subjects instead of having to train different skills at the same time. We also would like to add other predators that the player is pursued by based on the different areas on the map. This will keep interest as the player progresses further through the game and adds a different feel to each area. Along with adding the ideas from the students stated earlier, we can truly gear our next design to appeal to our target audience.

### 6.2 Changes Proposed by Students

In the hypothetical next version of our project’s design, we would, for the first time, make modifications to our product based on comments from potential users.

There were a lot of suggestions that would likely be added to our future plans for the design. One was the idea of allowing the player to play as the gorilla instead of the monkey. This was somewhat similar to our original idea of having a player avatar chase the monkey, but that was less adaptable to multiple scenarios. Now we can make multiple contexts with similar gameplay more believable by changing the predator that chases the monkey every level. This idea would need more work in more developed versions of our design, but it should be workable. And as the students in our feedback group demonstrated, giving the player the appearance of having an advantage over their opponent is an attractive way of portraying the competition. Since other students seemed to like helping the monkey escape a larger animal, we would probably leave in an option to play the game that way as well, to appeal to a wider audience base.

Another suggestion we received was to allow the monkey to throw things at the AI opponent. This was suggested as a more interesting way to represent the player getting a question right, but we might also implement it as an item system, either for students who
did well or those who needed extra help. The former will involve adding an item slot, which players can fill by answering a certain number of questions in a row, or answering a question very quickly. They can then use the item to gain an additional advantage against their opponent. Making various items with interesting effects would be a good way to keep players invested in the game. Another way we could use the item system, which would favor struggling students without making things duller for the others, is to give the player an item periodically, regardless of how well they perform. With some refinement, this kind of item mechanic would make our game much more engaging.

The students also expressed an interest in our game’s story, limited though it may be. Having more time to work on the design, we will flesh out the plot of our game beyond simply “a monkey trying to escape a gorilla.” We may try to add an explanation for why the monkey travels through a series of islands, and what it does to provoke a larger animal on each one. We wouldn’t want to expand the story too much, since that would risk upstaging the game’s educational purpose. But the students gave us the impression that they would enjoy the game more if the narrative was a little richer, so we would comply if we had the chance.

The above suggestions were received right after the rounds of testing from those students who trialed our tutor. Even though some of them still needed further consideration, we would certainly try to add them to our design either through changes made to our current systems or by working with the other groups. Moreover, we got more comments, which are mentioned below, in the post-testing session.

Our first run of tests came up with a majority of the students believing that the game was too challenging. The problem stemmed from the AI beating the students too quickly,
so they constantly had to try again. Yet during the second set of tests with the next group it was too easy for the students to win against the AI opponent. These two extremes show that our Dynamic Difficulty Adjustment system needs to be changed to keep this issue from occurring. Further balances and tests during the creation of Design 3 will allow us to tweak this system to provide an optimal win/loss ratio.

Another suggestion given by the students was the inclusion of a hint system. We removed this during an earlier design to focus in on our AI system. Now that our group has built the base system we can start expanding it without diluting our main idea. With Design 3 we can begin to work with the other groups to further improve and widen the appeal and depth of our system. By working together with the group that created a hint system, we can use their research and guidance to add an effective and unobtrusive hint system to our game.

Except for these features that we liked and thought would work well, there were some that just seemed impossible for us to implement even during Design 3, which were good in nature but went beyond our scope.

One of the ideas we were given to add was a system that would allow the player to change the grade or the knowledge level and thus change the questions they were given. Developing this system requires a well-defined problem bank and thus would be a large undertaking. Though it might be a good change, we can only make this change after we have developed a sound version of the tutor. In other words, this should definitely be considered when the development enters a mature stage, but in Design 3 we won’t be able to turn this idea into reality.
The students also suggested a bonus stage at the end of each of the levels, much like the old Sonic the Hedgehog and Mario games. For instance, after the monkey escapes the opponent they would be chased by all of the predators in the game and given rapid fire questions that are more challenging but are of the same subject of the level they have just completed. This idea is interesting, adding a lightning round basically would allow for use of more advanced material and a bonus stage would keep students from being punished for it. However, despite the fact that the students provided quite a few cool ideas for this system, adding something like this would require adding bonus levels for each of the levels. Since developing and balancing a rapid fire challenge round would take a lot changes to the Dynamic Difficulty Adjustment system, time constraints can be a great issue here.
7 Future Work

During the relatively short period of this project, we managed to complete a rough design of a tutor featuring AI elements. However, there’s still a lot we can do to better our design if our group want to continue this project in the future.

Given more time to work on this project, we would like to apply our AI to math topics other than line equations; perhaps we could even cover other subjects entirely. The adaptation will involve two main steps: design new chapters and contexts for the additional content, and devising new problems for the new subjects. The latter would require some research into other educational software, and perhaps the school materials as well, to generate a pool of problems for our tutor. The former seems to be easier, mostly just writing some new scenarios in which the player’s monkey is fleeing some other larger foe. We could even make the gaming experience more diverse by introducing some other, non-chase-scene situations to keep the player’s interest while keeping the element of AI and using it in different mechanisms. This will probably lead to a comparison between several styles of gameplay to give us a better understanding of how to best attract students to use a tutor. A fully implemented version of our project might also incorporate a more flexible difficulty system. The system should be able to control both the computer player’s behavior and the problems to give to the students. We are supposed to label each problem based on its perceived difficulty, possibly as “hard”, “intermediate” and “easy”. This way, the system will have the ability to simultaneously select of what difficulty every following problem would be for the student. This kind of selection would be based on such variables as how many questions the student get right or wrong in a row and a record of the student’s previous attempts. Problem selection
helps to overcome the limitations of indirectly determining the game difficulty by only changing the behavior of the computer opponent. We could even expand upon this by customizing the AI’s difficulty according to a database of each student’s scores in their classes. This means that students will have the AI and selection of problems exactly suited to him or her from the very first problem. Eventually, we would have a detailed comparison between our tutor and the original Monkey’s Revenge. We are going to also code our tutor with Action Script and publish it online. We intend to ask some students to try the tutors and tell us what they like and don’t like of the two, and gain a better idea of how to appeal to students in educational games and what aspects of Monkey’s Revenge we can learn from. Then we should eventually conduct a comparison experiment by making two groups of students each use one of the tutors and compare how they perform quantitatively in quizzes and/or test after a certain period of usage.

Another idea we plan to include in the tutor is bonus stages. This idea comes from students who participated in our design 2 demo mentioned previously. These bonus stages would be mini games in which the player competes against the AI after they have completed a stage. The big issue with this idea is: without a reward system or a system of points in some way, what would the prize be for completing these stages to really meet students’ expectations? For example, it could be an additional power-up of some kind, something that allows them to gain more space between the opponent and the player, or some sort of boost that doubled the player’s rolls after answering a question correctly. The addition of these power-ups should allow for further depth in the game play, but ought to be carefully designed in order to maintain the balance of the game. Other than this option, the reward can also be some special visual effects after our players win the
bonus stages. We need to make sure the effects are interesting and maybe also exciting to let players feel rewarded, which is the hard task here.

The bonus stages would also need games. What direction should these games take? Should the player be solving math problems from the chapter they have just completed or possibly a random different chapter so they are challenged with broader knowledge of the subject? The other idea is to make the bonus stages fun little mini games that don't necessarily teach the players anything but are pure bonuses for completing the level. We think of possibly implementing a single player game, such as a solitaire, and relating it to the monkey’s travel in each of the different areas.

All of these are ideas we consider meaningful to the current tutor design, though they will still need more consideration and polishing in the future.
8 Conclusion

To conclude what we accomplished, our group completed the framework for a math tutor with AI component in this project. We developed a proposal on how AI could be used to improve educational games, meliorated it based on readings, and tested it on a class of young volunteers.

Our initial outline of the tutor involved many elements to support our primary focus of an AI opponent; all of these non-AI elements, except the map screen, were shaved off in early iterations, to allow us to concentrate more on the AI itself. And we further honed our design with various readings, to maximize appeal to the various classroom demographics. We thought of including multiple difficulty levels to accommodate students of various abilities. We changed the nature of the competition, to one in which the player helped the monkey escape a larger animal, to increase tension as well as to make the scenarios easier to write. We also wrote an algorithm for shifting the AI and difficulty during the game, so as to increase when the opponent fell behind, and decrease if it got too close. This would further enforce the fierceness of the competition as well. After making these changes and some others, we brought a mock-up of our game before a sample class of students and received feedback.

If someone else were to continue our work, they should further test and polish the system. For example, they could take our bare bones story and expand it into a full game, with a different miniature plot for each level/school topic, and perhaps a proper overarching plotline, to maximize student engagement. Moreover, they could arrange sets of problems with more thorough topic coverage, and tailor them for specific grades and skill levels. The algorithm for the dynamic difficulty adjustment might still need
testing, but it had the potential to act as a basic structure and be further modified for similar educational software.

In summary, we explored how to properly implement an AI component in an educational environment in this project. We came up with a feasible solution to the problem of balancing entertainment and education with the use of AI. Our experience and findings can serve as a useful resource for others interested in the subject of increasing the attractiveness and effectiveness of educational software. We hope that by referring to our work, designers of educational software can be inspired and thus be able to better make use of AI in their designs.
9 References


Kaiser family Foundation. (1999). *Kids & media @ the new millennium*.


Appendix: Student Feedback Form

1. What do you think of the AI's difficulty?

2. Do you feel you had too little or too much time to solve the problems?

3. How do you feel after getting a question right, or wrong?

4. Will you use our game as a study aid?

5. What kind of character or story would you like to see in a math tutor?

6. Are there any ways for us to make our tutor more fun and helpful to you?