Wu, Y.-T. et al. (Eds.) (2018). Proceedings of the 26th International Conference on Computers in Education. Philippines: Asia-Pacific Society for Computers in Education

Learning logic, search and cracking through a number guessing game

Tsui-Qin MOK, Wei-Ming MAU, Calvin Weng-Kit WONG, Kian-Wheng KOAY, Meng-Jeat CHENG, Chien-Sing LEE
Department of Computing and Information Systems, Sunway University, Malaysia.
*moktsuiqin@hotmail.com, *chiensingl@sunway.edu.my*

Abstract: This paper describes the development of a number guessing game aimed at playful learning of logic, search and cracking for the young. It can also be used to train memory and intelligent guessing for seniors. Theories and models utilized are Goals, Operators, Methods and Selection (GOMS), Norman's 7 stages, and Schneiderman et. al's design principles for user interface design. Findings among students indicate that the number guessing game is easy to use, and useful and that history and feedback are the most important design factors. This paper has managed to come out with an expected result where multitasking is required and player has to read and understand the meaning from the output to complete the game. We look forward to possibly designing for both children and seniors in the future by improving the game features without making the game become more complicated.

Keywords: number guessing game, logic, search, cracking, memory, guessing, young, seniors

1. Introduction

1.1 Problem to be addressed

Children with dyslexia often face difficulties reading and this becomes more pronounced with age. An example of the various difficulties is highlighted by the Mott Children Hospital (Table 1). It indicates that dyslexia is a complex language problem arising from inability to break down a word into the sounds for each syllable. As such, they are often not able to spell or recognize words correctly. Hence, dyslexia is not an intellectual deficiency and some have performed well in STEM such as Albert Einstein.

Table 1
Problems dyslexic children face from pre-school to 3rd grade

Preschool	Kindergarten	1st and 2nd grades	2nd and 3rd grades
• Cannot tell the difference between letters and doodles • Cannot recognize one's own name • Can only utter a small number of words • Avoid rhyming games and cannot fill in the rhyming word in familiar nursery rhymes. • Has difficulty pronouncing words (may be due to confusion with words that sound alike)	• Slow to identify familiar objects and colours • Cannot differentiate between the sounds that make up a word • Cannot remember the names and sounds of the letters • By the end of kindy, most of the consonant sounds in a word and vowels would be missing till later)	• Has difficulty pronouncing new words and remembering them • Has difficulty blending sounds together to say words • Cannot figure out unknown words • Avoids reading and reading aloud • Falls far behind classmates	• Starts to withdraw • Has some behaviour problems • Guess at unknown words • Does not understand reading

Source: Mott Children Hospital (2018)

Franceschini, Gori, Ruffino, Viola, Molteni, and Facoetti (2013) have discovered that the effect of playing video games such as "Rayman raving rabbids", could help dyslexic children aged between 7 to 13 years to read faster and accurately. They also find that fun, among other factors such as usability, leads to better results compared to the traditional reading interventions. Seniors also love to play games such as mahjong, bingo, Candy Crush. These require attention as well as the ability to focus and think logically. Therefore, we have decided to create a game, i.e., '*crack' the treasure box.*

A treasure box is intriguing to many. Furthermore, *'cracking,'* allows us to have fun and to teach young people logic and computer security concepts. With only three panels to look at, attention is focused and cognitive load more manageable. Hence, we hope to provide a good user experience to users by developing a human-machine player competitive game.

1.2 Objective

The aim of this research is to develop a game, for cognitive training while enjoying the game. Furthermore, multi-tasking by competing with the bot in cracking the code within a certain time limit may improve dual-processing ability as well as memory.

Our system generates a number that requires the user to solve or crack in order to win the computer. The game that we have developed is based on a format which helps guide the users along the way to help crack the code. A history panel is implemented to help provide information regarding their past inputted number and the number that is placed in the wrong or correct order. Implementing a none language-based system would help boost confidence and enjoyment towards the game. The game would also help users to socialize with other users as they can enjoy solving the code together. Other aspects that our system can benefit users, is improving their concentration level and memory, enhancing their reaction time, having a better understanding of things (logic and search).

2. Literature Review

Video games are increasingly popular. Many users are dedicating countless of hours of their time to play games such as First-Person Shooter (FPS) games, Multiplayer Online Battle Arena (MOBA) games, Role Playing Games (RPG), etc. Some of the games would host competitions with a prize money of a few thousands and even millions of dollars to the players. This would serve as motivation for them if they are to take up playing video games as a profession.

However, despite the immense popularity of video games nowadays, there are many game genres thus far not as popular to the general public such as guessing game. This is because most of the guessing games have similar characteristics and functionalities. After some time, for young people, similarity does not excite. As such, some websites offer guessing games as side content or as mini-games. This application falls under the mini-game category.

2.1 Guessing Game Systems

2.1.1 "Khan Academy" guessing games

In "Khan Academy" guessing games (Khan Academy, 2018), instructions on how the game works are provided. Players select a random integer from 1 to 300 within the limit of 9 guesses. The game is easy to understand as it provides a clear understandable layout and system functionality, similar to a normal guessing game. The players will have to keep guessing the numbers randomly generated by the system in order to win the game. The system will also provide hints if the selected number is too high or too low from the system's randomly generated number. For instance, if the player selects the number 64 from the range of 1 to 300 as the first number that they guess, if the number is a wrong answer, the wrong guessed number will not be able to be selected again. This game converts

search logic from linear to binary search. This game attracts as it is fun, easy and simple. An example is Figure 1.

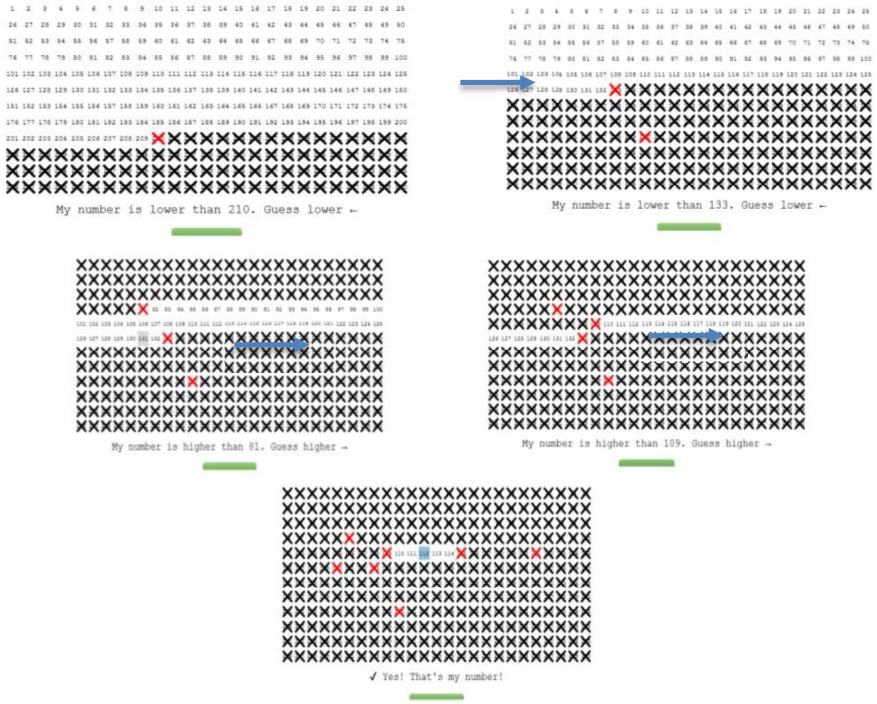

Figure 1. Example of incremental binary search

Source: Khan Academy's guessing game (2018)

2.1.1 Mozilla guessing quiz games

Mozilla guessing quiz (Mozilla, 2018) is another guessing game. Developed by Mozilla developers, the game is deployed into their website to allow players to enjoy the game while developing the ability to learn how to build a simple guessing game system. The Mozilla system is different when compared to the "Khan Academy" guessing game because instead of a list of numbers and random generation of a number based on the list of numbers, the Mozilla guessing game uses a key-in value system. For instance, the system will inform the player that a random number between 1 and 100 has been selected and he/she will be asked to guess the number in 10 turns or fewer. The player keys-in the guessed value based on the number set by the system and the answer to that question will be randomly generated. The system will also inform the player if his/her guess is too high or too low. Figure 2 shows an example of the game.

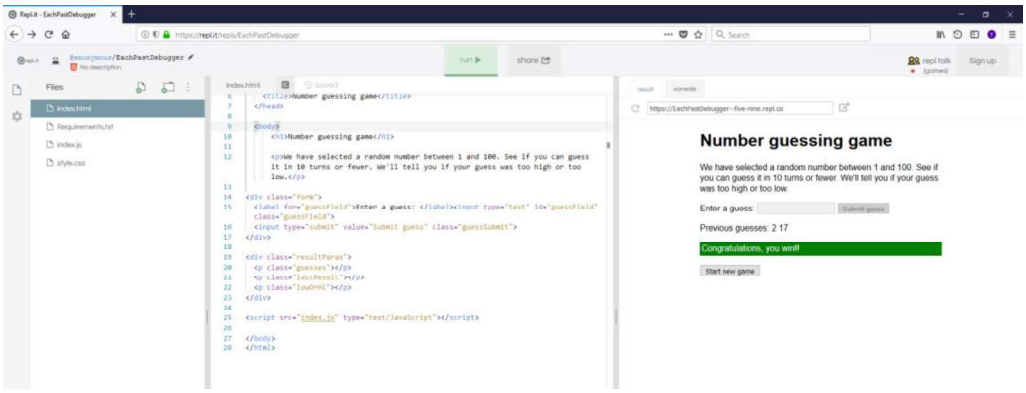

Figure 2. Mozilla's number guessing game

Source: Mozilla's MDN Web Docs (2018)

This guessing game may be difficult compared to the previous game as the elimination of answers, hint on whether the guessed number is more than or less than the actual answer is not shown. However, the history of previous guesses is shown. So the player knows which number not to guess next.

2.2 Improvements / Enhancement

Based on the above reviews, we are able to identify a list of features to be implemented into our guessing game. Keeping the game easy to understand and navigate should be the top priority of any system. This is because having a complicated system which the player does not understand will have a major impact on the growth of a system and affect the users' experiences.

Implementing a computer system to compete with the players is important as it allows the players to have a sense of accomplishment when they win the game against the computer. Learning from video games, instead of just implementing a record function where the system keeps record of previously keyed in data similar to the Mozilla guessing game, the three numbers keyed in, how many of these three numbers is in the correct sequence and how many in the wrong sequence is displayed. The history of each three numbers keyed in is also displayed so that the player can guess which of the numbers in the history list is the correct number and in the correct sequence. This would be beneficial to the users as guides and tips to improve their chances of winning.

Furthermore, the design of the system's user interface, functionalities and interactions are important as the design of the system correlates with the experience, engagement/enjoyment of the users. These are tied directly to the success of the game. Thus, the keyed in number history list for both the player and the system, background images, timers' count down, can help improve users' experiences in terms of engagement/enjoyment.

3. Methodology

The methodology for this paper is presented in Figure 3 below.

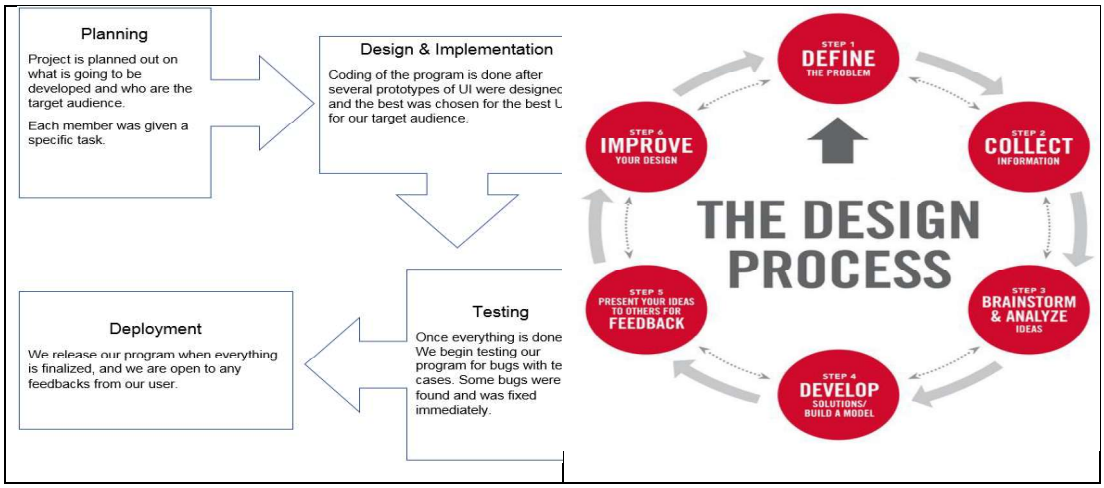

Figure 3. Design and development methodology

5.1.1 Define the problem

Children with dyslexia often face difficulties reading and this difficulty becomes more pronounced with age. The problems from dyslexic children face from pre-school to 3rd grade had been address in table 1. Furthermore, multitasking has become difficult to many as it requires retrieval from memory and parallel processing of information. Because of the problem been address, we implemented this games which could improve children handling multiple task memory and our system does not require any reading which could solve children with dyslexia problems.

5.1.2 Collect information

We have collected multiple designed game on number cracking which had been address in Literature Review. We found out that the game does not solve people's multitasking problems which cause memory load on guessing different number to win the game. Children with dyslexia shows no improvement on the game.

5.1.3 Brainstorm and Analyze Ideas

After gathering enough information and requirements, we come out with several ideas where the game must easy to be understand. Clear instructions should be given so that player is the one who keep control in the system. Other than that, any inputs enter by the player must receive a feedback to ensure all actions taken by player are not ignored. The game is ended when the player fulfills the requirement and managed to solve the game.

5.1.4 Develop Solutions

We have develop a history window at the left side to record any 3 number that user had entered to acknowledge them to not entering the same number and also allow user to aware of the time count. We also reduce reading load on dyslexia children so that they do not require to read a lot when interacting the game.

5.1.5 Gather Feedback

We presented our game to our friends and lecturers in our university. We receive most positive comments from them and minor disadvantages on the design part of the game which needed to improve so that the game meet most HCI aspects like colours, constrast and etc.

5.1.6 Improve

We will further improve our developed game meeting all negative comments from our friends and lecturers in the future to further receive positive view and comments to make the game a better platform for dyslexia childrens and seniors when interacting with our game.

Other methodologies and references we used are Card, Moran and Newell's (1983) Goals, Operators, Methods, Selection (GOMS), Shneiderman, Plaisant, Cohen, Jacobs, Elmqvist and Diakopoulos's (2016) 8 golden rules of interface design and Norman's seven stages of action (1988). The latter two are presented in Table 2 below.

Table 2
Design guidelines: Shneiderman et. al.'s 8 golden rules and Norman's seven stages of action

Shneiderman et al.'s 8 golden rules	**Norman's seven stages of action**
strive for consistency,	forming the goal
cater to universal usability	forming the intention
offer informative feedback	executing the action
design dialogs to yield closure	specifying the action
prevent errors	perceiving the system state
permit easy reversal of actions	interpreting the system state
keep users in control	evaluating the outcome
reduce short-term memory load	

For *universal usability*, it must provide clear instructions for the player before starting the game. The game should provide *clear feedback* so that the player is not confused while the game is processing. Any possible action taken by the player which includes error inputs must also be clearly stated to *prevent any further errors* and chances can be given to the player to re-enter inputs that follow the given instruction or warning. To *reduce short-term memory load*, the game will provide hints whether the number the player input is correct or wrong to let the player improve in future performance. A computer will compete with the player to make the game more challenging to improve the ability of the player while playing the game. Lastly, all possible outcomes should be evaluated for any further enhancement in the future so that the efficiency of playing the game can be increased.

4. Requirements gathering

We mainly focus on developing a system that helps address the needs of people who have learning disabilities or memory problems. Different stakeholders or users have different interpretation and opinion of what a system should do. The user or stakeholder needs to communicate with the developer their needs and at the same time the developer needs to be able to anticipate their needs and ask the correct questions during the requirement gathering phase of a project in order to further understand the user and problems that need to be solved. User requirements refer to the features or attributes that a product should have or how it should perform from the user perspectives. In short, the developer needs to gather requirements to be validated by every stakeholder that would use the system to prevent any misinterpretation or any requirement inconsistency.

Requirements is a statement about an intended product that specifies what it should do or how it should perform. As such, requirements display the product or software capabilities to meet the users' needs and solve their problems. There are many different kinds of requirements. These include functional requirement, data requirement, usability requirement, user requirement, environmental requirements.

To gather user requirements, there are several techniques. but in our system we have chosen interview, focus groups, and questionnaires as our data gathering techniques. We have chosen interview as our data gathering techniques because interviewing allows developers to achieve a deeper level of engagement when talking to stakeholders or users. The interview is conducted with

a focus group. A focus group usually includes a group of people up to 6 to 12 people who will represent the project end users or customers. Focus group can be used to pinpoint users point of view or perception towards the concept or the system. Focus group will allow a developer to save time by interviewing multiple users or stakeholder at the same time as of compared to interview.

The list of requirements based on feedback from the focus group is as listed below: The system needs to:

1. display usable information and give feedback to help the user in further interacting with the system;
2. ensure the system is easily accessible and navigated by any user;
3. have the ability to help improve users' memory, especially useful for Alzheimer's patients;
4. support different platforms which include Windows, Linux, Mac OS, etc.;
5. be easily understandable regardless of age group;
6. be enjoyable to all age groups;
7. be visually appealing without having any harsh colors such as red or purple which may not be appealing for some people;
8. act as a source of entertainment for the users;
9. not contain any violent or illegal activities which may violate the rules and regulations;
10. provide a sense of accomplishment upon wining or completing the game.

5. Design

The use case diagram for the cracking the treasure box game is presented in Figure 4a below.

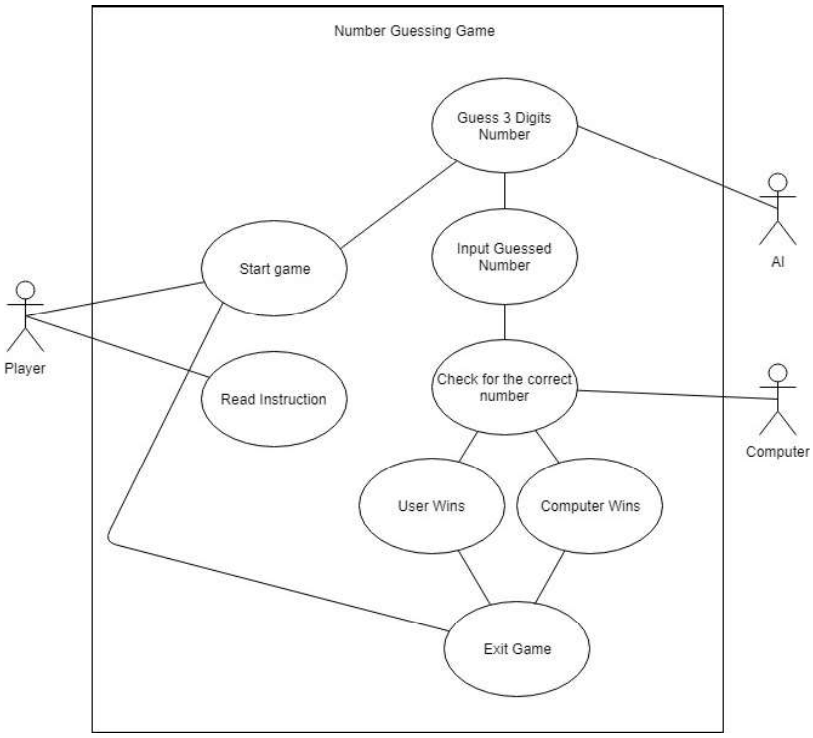

Figure 4. Use case for the cracking the treasure box application

6. Development

C# is chosen to create this game as this language can be used to create almost everything but is particularly strong at building Windows desktop application and games. Unity game engine is the most popular game engine today to create games and more than one third of the top games are made

with Unity. This makes it a great choice for any programmer to step in to the game development industry, especially with possibilities for virtual reality.

Hence, we choose C# to develop this game because C# is user friendly in terms of complexity and flexibility. By using C#, completing the number guessing game would be much easier. All the functions provided are used to create different game features such as user interfaces, functions to compete with the player and to provide feedback to the player based on the game result.

6.1 The system

The main page of the game contains the Play, Instruction and Exit button. By clicking the Play button, the main page will be closed and the player will enter into the game platform. By clicking the instruction button, the player will enter the instruction page (Figure 5) where the player can understand how to play the game. The last button which is the exit button allows the player to quit the game application.

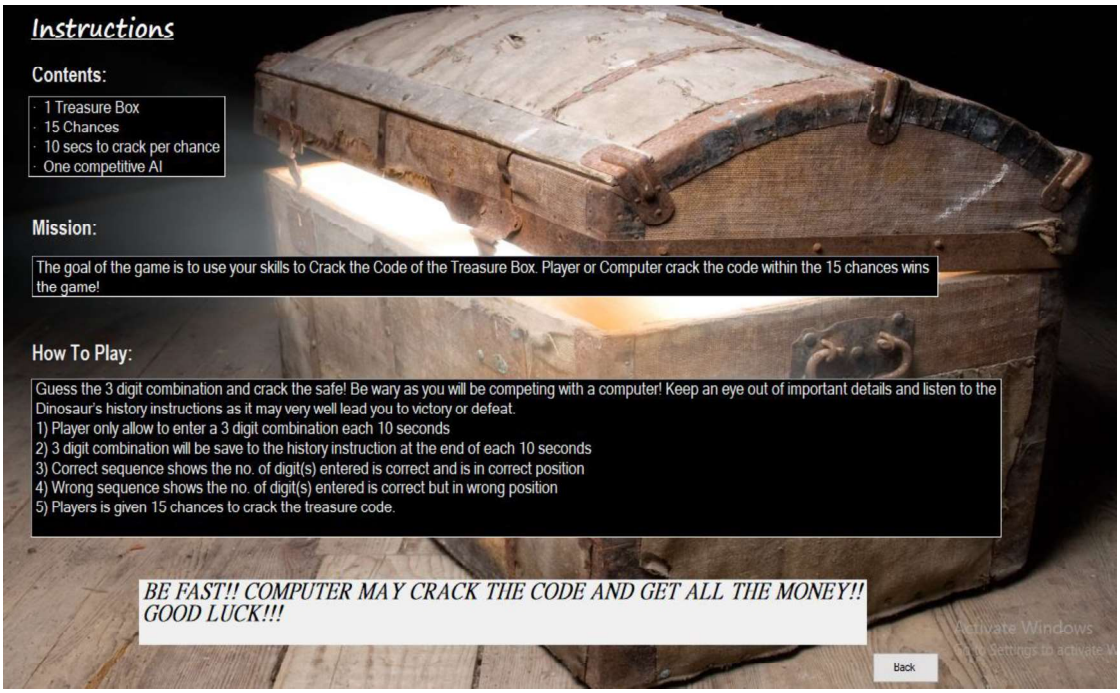

Figure 5. Instructions page for the cracking the treasure box application

When the game starts, players are required to guess a 3-digit number by pressing the number on the pad in the middle. For every 10 seconds, a chance will be deducted. If the player or computer enters a correct number, the game ends. If the player or computer enters a wrong number, the game continues and the previous numbers that both the computer and player input will be displayed on both sides of the panels. On the right column of the panel displaying the player's choice of 3-digit history, are feedback/hints on how many numbers are in the correct sequence and how many in the wrong sequence (Figure 6). If neither the computer or player wins, the application will force the game back to the main page, letting the player choose to play again or not.

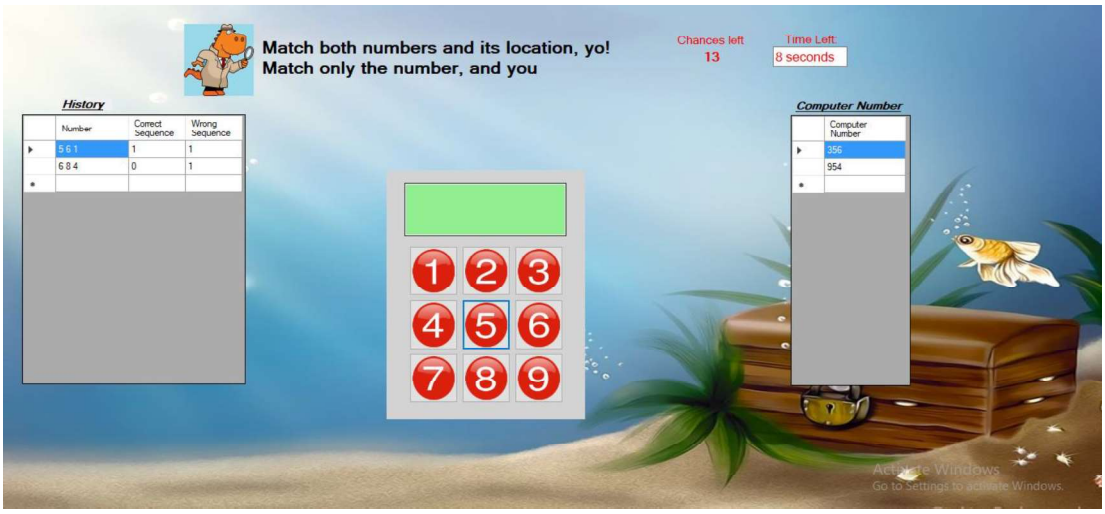

Figure 6. The player's history, system's feedback and system's attempts at cracking the code for the cracking the treasure box application

7. Evaluation

The Technology Acceptance Model (TAM) is originally created in the 1980s to predict the usefulness and acceptance of systems by users. TAM (Figure 7) assesses users' perception towards perceived ease of use and perceived usefulness towards the system tested. Perceived usefulness is how users find the system in helping them to achieve their goal whereas the perceived ease of use refers to how easy the users find it easy to use the system. These are the two most important factor when using the system but are often affected by external factors such as social factor, cultural factor and political factor (Davis, 1989). Holden and Karsh (2011) have also viewed the TAM positively in healthcare as they find that it is able to predict a significant portion of the use or acceptance of health systems. Suggestions from the focus group are positive and indicate that feedback and history are the most important in this game.

8. Conclusion

The aim of this project is to create a game that has the potential to help people to improve the performance of their memory, and logic to create a great user experience for the players in a human-machine player game. Media richness has been designed through the two design guidelines principles. History and feedback are found to be the most important design factors among these guidelines/principles. The findings are preliminary and so future work will involve user testing on dyslexic children and others who may benefit from such training. We may also develop a more interesting game that can make people feel happy/challenged; allowing the user to play multi-player online or specially design a game for children.

Acknowledgements

We would like to thank Sunway University. This paper is linked to the Malaysian Ministry of Higher Education's Fundamental Research Grant Scheme grant number FRGS/1/2016/ICT04/SYUC/01/1 and Sunway University's internal grant INT-2018-SST-DCIS-03.

References

Card, S., Moran, T. P. & Newell, A. (1983). *The Psychology of Human Computer Interaction.* Lawrence Erlbaum.

Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology, *MIS Quarterly*, *13* (3), 319-340.

Dyslexia and Reading Problems. Visclosky, T., Laule, S. (Eds). [Online]. Available: https://www.mottchildren.org/posts/your-child/dyslexia-reading-problems

Franceschini, S., Gori, S., Ruffino, M., Viola, S., Molteni, M., Facoetti, A. (2013). Action Video Games Make dyslexic children read better. *current biology, 23* (6), 462-466. [Online] available at: https://www.ncbi.nlm.nih.gov/pubmed/23453956

Holden, R. J. & Karsh, B.T. (2011). The technology acceptance model: Its past and its future in health care. *Journal of Biomedical Informatics, 43* (1): 159. doi: 10.1016/j.jbi.2009.07.002. [Online]. Available At: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2814963/ [Accessed 21 Jun. 2018].

Khan Academy. *A guessing game*. [Online] Available at https://www.khanacademy.org/computing/computer-science/algorithms/intro-to-algorithms/a/a-guessing-game [Accessed 20 Jun. 2018].

MDN Web Docs. *A first splash into JavaScript*. [Online] Available at: https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/A_first_splash [Accessed 20 Jun. 2018].

Norman, D. A. (1988). Psychology of Everyday Action. *The Design of Everyday Things*. New York: Basic Book.

Shneiderman, B., Plaisant, C., Cohen, M., Jacobs, S., & Elmqvist, N. (2016) *Designing the User Interface: Strategies for Effective Human-Computer Interaction:* Sixth Edition, Pearson.