

Design and Implementation of Shologuti: A Rural Game of Bangladesh

Samia Nawshin
Lecturer
Department of Computer
Science & Engineering
Daffodil International
University
Dhaka, Bangladesh

Mohd. Saifuzzaman
Lecturer
Department of Computer
Science & Engineering
Daffodil International
University
Dhaka, Bangladesh

Md. Abdullah Al
Mamun
Student
Department of Computer
Science & Engineering
Daffodil International
University
Dhaka, Bangladesh

Md. Al Muzahid
Lead Programmer
Software Technologies
Ltd, Tangail

ABSTRACT

Now a days, so much attention and concentration has been directed to the use of games for learning, entertaining and many more purposes because of their ability to attract player attention and hold it for lengthy periods of time as players learn to master game complexities and accomplish targets. This paper describes the initiatives to develop a game to make a small contribution to our own culture by saving an almost extinct rural game & create its virtual form, so that younger generation of our country may know about our own culture and games.

Keywords

Extinct, Framework, Game, Installation, Programming skills, Rural game, System architecture design.

1. INTRODUCTION

Shologuti is very famous and popular game in Bangladesh and some of the Indian regions. It actually a checker type game which basically played between two players. So from that view, it can be called as a rural version or innovation of chess. The local people play this game for passing their leisure period even sometimes they arrange tournament of this game. Generally the board of this game is very simple like the people can draw it in table or ground and use stones, dry branch or a small piece of cane as pawns. After drawing, two players take their place at the opposite direction of the board with their 32 guti's altogether of which everyone try to achieve opponent's sixteen beads. After placing 16 pawn from the edge of the board, the game starts by making a move on the empty space. The pawn can move right, left, forward, backward and crosswise where each opponents actually try to take or occupy others pawns. If any player's pawn can cross opponent's pawn, then the pawn will be out of the board. After that, only that pawn has the ability to have a bonus move if that pawn can find out opponents pawn can be crossed over. By this process, which player can take or capture all the pawns of opponents, he or she will be the winner.

Now a days, computer has become a part of our life and almost everyone like to play game in it. This led us to develop this attractive game of Shologuti. Due to lack of well-designed boards in market, city's people can't play it or don't even actually well known about the game. So, the problem can be solved by developing this game with an attractive UI.

2. OBJECTIVES OF THE RESEARCH

The objectives are actually fivefold. These are as follows:

- To learn how to design and develop computer based games;
- To learn how to use computer graphics and to create virtual reality;
- To learn how to apply concepts and techniques learned in other courses (such as C#, .Net framework, Algorithms, AI, computer graphics, mathematics) to the computer games;
- To improve programming skills.
- Finally to make a contribution for our country to save our rural culture by developing this desktop based game.

3. LITERATURE REVIEW

In general, this research cover the literature review from different sources which is targeted various aspect of designing game, installation & learning. Here it can be said that this research is an integrated research which combined two major aspect & make a unique proposal of effectiveness. Firstly, here is a discussion about the architectural part of the game like how to develop this game and secondly, how to save a rural game from extinction.

Whithy et.al [1] concerned in his research about the creation of a game installation and a reviewed a case study to test the game. *Coleman et.al [2]* described the initiatives they took to develop & design a game for improving their skills. *Symborski et.al [3]* & *Martey et.al. [5]* also showed the game development and design strategy guided by experimental test and evaluation. *Sethi et.al [4]* described "Don't just play a game, design a game & build a game and improve your programming skills". So, in this research, a computer based game is actually developed which will help to improve skills in game designing sectors.

Amory et.al [9], *Cheng et.al [10]* & *Dondlinger [12]* described in their paper that playing game is actually a part of social & mental development which will help us to learn an effective lesson. *Schrier et.al [11]* had the same opinion and he described about the learning, moral thinking and skill development by creating or designing a game which was filled with moral scenarios & moral choices. *Nergiz et.al [13]* & *Strzalkowski et.al [15]* described the experiences from the implementation of a computer game development that it could help a student to improve in problem solving, the application of previously learned knowledge, the utilization of independent learning and learning by doing. *Rhodes et.al [14]* described that serious games can be an efficient training engine, particularly for teaching procedural knowledge.

Flores et.al [6], Goude et.al [7] and Cameirao et.al [8] discussed in their research for the field of rehabilitation. Their research has shown that rehabilitation games and related virtual technology can provide significant motivation to patients, this will help them to produce a moment where patient is forced to do something unexpected which will help them to improve or motivate.

From above description, it can be said that many games exist for educational, training, medical, entertainment and so many other purposes. So this research is actually done for developing a game which will have all the functionality as mentioned. This developed game will be a helpful means for saving our culture from the extinction of a rural game.

In the coming chapters we will discuss about control flow diagram of the developed game in Chapter 4, Chapter 5 deals about requirement analysis, Chapter 6 shows us the physical implementation and results.

4. SYSTEM ARCHITECTURE

To demonstrate the whole system as well as the game, an effective architectural system actually needs to be designed. This section presents the design. The control flow of the program among various functions is shown using a single flowchart. Each function in the program is defined and their pseudo codes are given for better understanding of the program. This section contains Control flow diagram, Function description and Pseudo code.

4.1 Control flow diagram

Control flow diagram in fig 1 shows the execution sequence of functions in the program. In this program, it has an option which is PvP (Player vs. Player). In this game, two players can play continuously but there is no option for single player based on AI. AI part will be implemented in future development of this game. Currently this game is designed to play between friends.

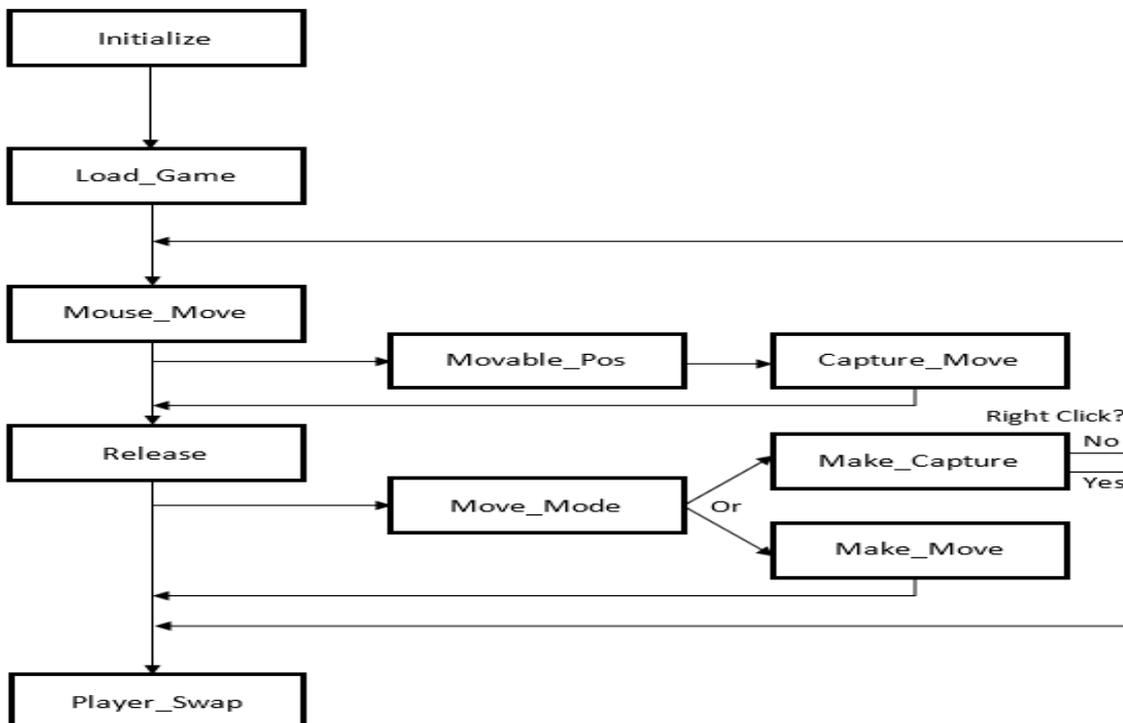


Fig 1: PvP (Player vs. Player) Control Flow Diagram

4.2 Function description

4.2.1 LoadGame

This function is used to check for previously saved games and if found, it generates a message if the player wants to play the saved game or not. If yes the game then loads the saved board and resumes the game. If not, it loads the default board with default pawn settings.

Pseudo code:

- i. Check for previous saved game.
- ii. Load the game with saved values if saved game found.
- iii. If not load default board with default pawn settings.

4.2.2 Mouse_Move

This is the beginning function for PvP part of the application. Here the program detects a mouse click, if an eligible pawn is clicked on, it will take relative action and will wait for the right pawn to be clicked otherwise.

Pseudo code:

- i. Detect left click on the pawn
- ii. Check cpd value (Captured Value).
If cpd = 1 and EP (End Point) matches clicked pawn position, call movable_pos(EP)
Else
Send error message.
- iii. If cpd = 0 and clicked pawn belongs to current player call movable_pos(Position)
Else
Wait for next click.

4.2.3 Movable_Pos

Movable_pos function works with parameter passed from mouse_move function. This function receives the position sent and stores it in P. All adjacent positions are then calculated. Each adjacent position is stored in EP one by one, and their value is checked. If the value is 0 (represents open position) the positions color is set to brown. If 1 (represents enemy pawn) or 2 in case of player1, Capture_pos function is called with the parameter P and EP.

Pseudo code:

- i. Get selected pawn from Mouse_Move()
- ii. If cpd = 0, get adjacent positions
- iii. Select one position and set as EP
- iv. Check EP value (0 or 1 or 2)
If Value = 0, set color brown
If Value = 1 or 2 respectively to player, call Capture_Pos(P, EP)
- v. Do (iii)-(v) for every adjacent position

4.2.4 Capture_Pos

Based on the 1st and 2nd positions represented by P and EP accordingly, this function determines the 3rd position on the line and checks its value. If the value of 3rd position is 0 (represents open position) the color of the position is set to red and the control returns to Movable_pos function.

Pseudo code:

- i. Get P and EP from Movable_Pos()
- ii. Get third position on the line
- iii. If value of the position is = 0
Set color red.

4.2.5 Release

Release function is called when the left mouse button is released. This function detects the cursor location at the time of left button release and determines if and which position is on the board. This function then check the color of that position. If the color is red, MP becomes null and the position is stored in EP. If it's brown, position is stored in MP and EP becomes null. The function then calls Move_Mode function with MP and EP.

Pseudo code:

- i. Get released location
- ii. Determine the position on board
- iii. Check position color
If Orange, set EP = 0, MP = Released Position
If Red, set EP = Released Position, MP = 0
- iv. Call Move_Mode()

4.2.6 Move_Mode

Move_Mode function is used to determine the type of the move a player is making. If MP has value, then EP is 0 (zero). That means it's a move mode and Make_Move function is called with parameter P, MP. Otherwise it's a capture move and Make_Capture function is called with parameter P, EP. In both the cases before calling respective functions, mouse release position is checked and matched with either MP or EP to make sure everything is ok. If the location and the position

does not match, MP and EP is set to 0 (zero), game is loaded again, and the program waits for left click to happen.

Pseudo code:

- i. Check MP and EP
- ii. If MP != null and pawn release location matches MP's Value
Call Make_Move(SP,MP)
Else, MP = 0, LoadGame()
- iii. If EP != null and pawn release location matches EP's Value
Call Make_Capture(SP,MP)
Else, EP = 0, LoadGame()

4.2.7 Make_Move

This function replaces the property of moved position (MP) with the starting position (P) i.e. the pawn is moved to new position. The old position's value with default 0 i.e. old position is reset. If it's a PvP game, player swap function is called and Handover is called otherwise.

Pseudo code:

- i. Get value from Move_Mode()
- ii. Replace MP property with P Property
- iii. Replace P property = Default 0

4.2.8 Make_Capture

This function replaces the property of ending position (EP) with the starting position (P) i.e. the pawn is moved to new position and the old position's value with default 0 i.e. old position is reset. This function also detects the position between P and EP, and set its value to default 0. Also increases the number of captured pawn by 1. If it's a PvP game player swap function is called and Handover is called otherwise.

Pseudo code:

- i. Get value from Move_Mode()
- ii. Replace EP property with P Property
- iii. Replace P property = Default 0
- iv. Detect Position between SP and EP
Replace it's properties with default 0

4.2.9 Player_Swap

This function is used to swap between players i.e. player1 & 2. Also swaps between player name visibility and some other elements.

Pseudo code:

- i. Swap Between player1 and 2
- ii. Swap between player name visibility
- iii. Swap label background

5. REQUIREMENT ANALYSIS

Requirement specification is the most important to develop applications. The requirements of this project are divided into two different category:

- Software requirement
- Hardware requirement

5.1 Software requirements

Requirements for developing:

- Operating system : Windows 7, windows 8 (with .NET framework installed)
- Language : C#
- Tools : Visual Studio ultimate 2013
- IDE : Visual Studio IDE
- Debugger : Built in debugger
- Other editors : Notepad ++

Requirements for running the application:

- Operating system : Any windows OS with .NET support.

5.2 Hardware requirement

Requirements for developing:

- Processor: 1.6 GHz or faster processor
- RAM: 1 GB of RAM (1.5 GB if running on a virtual machine)
- HDD space: 20 GB of available hard disk space
- HDD Speed: 5400 RPM hard disk drive
- Direct X: DirectX 9-capable video card at 1024 x 768 or higher display resolution
-

Requirements for running the app:

- Processor : 1.6 GHz or faster processor
- RAM : 128 Mb min
- HDD : 20 MB
- Direct X : DirectX 9-capable video card at 1024 x 768 or higher display resolution

6. IMPLEMENTATION AND RESULTS

The Shologuti application is designed from a user's point of view. Its user friendly design helps user to use it with ease. A big effort was put to make the design very easy for user yet attractive. The application code was written in C# language. The front end of the application i.e. UI was designed using basic windows form designing tool provided with the visual studio software. The default debugger was used throughout the development process. No database was used in this application because fetching database takes longer increasing run time for the application compromising the smoothness of the game.

6.1 Graphical user interface

The application has actually two different but exactly the same looking UI. There is a menu button on the top left corner of the screen. Minimize and close button are on the right top corner. Bottom left corner shows how many enemy pawn player1 has captured while bottom right shows for player 2. Top middle section shows which type of game is currently running and bottom middle part shows how many more pawns to be captured to win the game.

6.2 Selecting the game type

When the game starts if it detects a previously saved game, the game is loaded with previously saved properties. If not, a window shows up like this and asks the player to select a game type, either a PvP or PvAI. Here PvAI option has been kept but it will be developed properly in future soon. Fig 2 is the screenshot of Selecting the Game Type.



Fig 2: Selecting the Game Type

6.3 Player vs. Player board

If the player selects to play a new game with PvP, the board is loaded with pawns set in their default starting position. Player1 gets to play the first move always. If it's the first run for this application on any computer or player names are never changed then the names for the players will be set as Blue for the player1 and Green for the player2. Blue and Green are default names for the player. The names can be changed any time. Fig 3 is the screenshot of Player vs. Player.

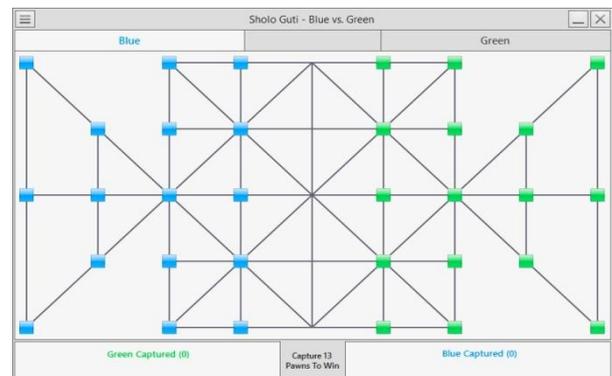


Fig 3: Default Board - Player vs. Player

6.4 Player pawn movements

If the player selects a pawn of his own, then possible moves are shown in Yellow and Red. Yellow color represents normal movable positions and Red color represents capturable positions on the board. Fig 4 is the screenshot of pawn movements of player pawn.

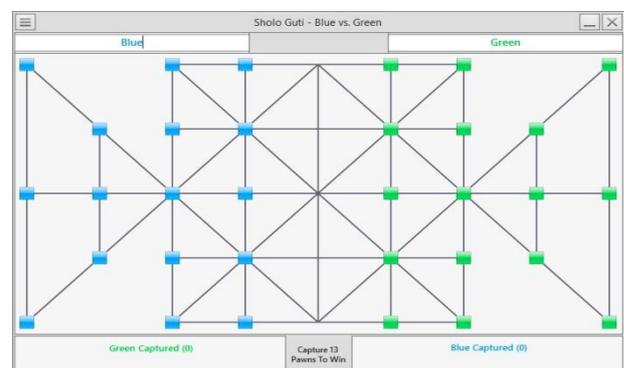


Fig 4: Pawn Movements - Player Pawn

6.5 Drop-Down menu

On top left of the UI there is a tab which shows a dropdown menu if clicked. The menu has different options and

submenus to customize game properties. Fig 5 is the screenshot of the game window with drop-down menu.

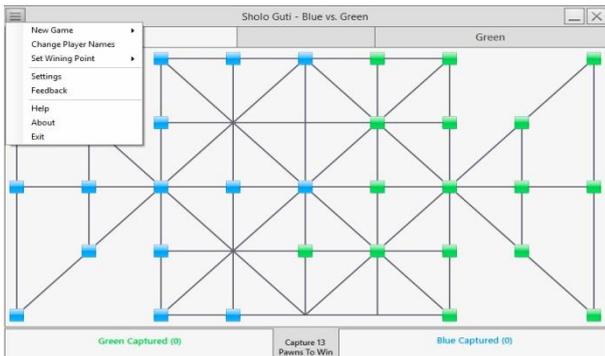


Fig 5: Drop-Down Menu

6.6 Changing name

On dropdown menu there is a submenu named as “Change Player Names”. On the click it shows two textboxes where player name can be entered. At first it focuses on the left player textbox. When Enter key is pressed, textbox of right player is focused. Fig 6 is the screenshot of how to change player names.

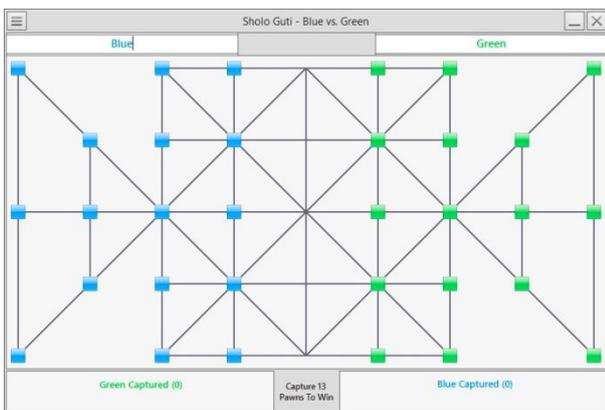


Fig 6: Changing Name

6.7 Setting wining point

On dropdown menu there is a submenu named as “Set Wining Point”. On the hover to this control, it shows five different numbers. Clicking on any of them, sets particular value. Which indicates how many pawns have to be captured by users or computer to win the current game. Fig 7 is the screenshot of how to set wining point on a game.

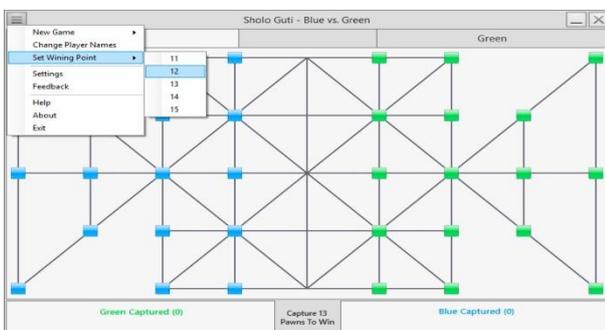


Fig 7: Setting Wining Point

6.8 Winning the game

When the specified numbers of pawn is captured by player or computer, a message box appears on the screen showing the winners name on it as shown in Fig 8.

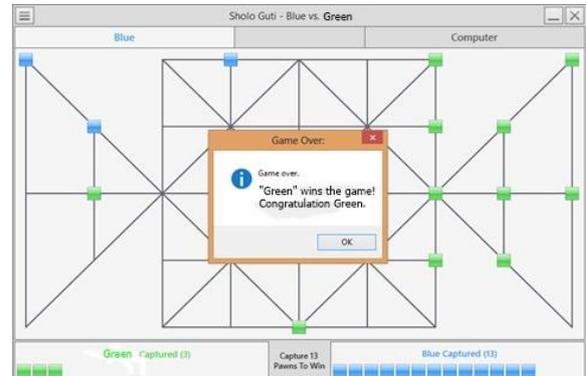


Fig 8: Wining the Game – Game Over

6.9 Save & Exit

When the close button on the top right is pressed or “Exit” is clicked from the dropdown menu, a message box appears to ask player if he or she wants to save the current game to play it later from the current state.

If the button “Yes” is clicked then the game will be saved with current properties. User can play the game from that saved state when he or she will run the game next time. Fig 9 is the screenshot of what happens when a close button is pressed during the gameplay.

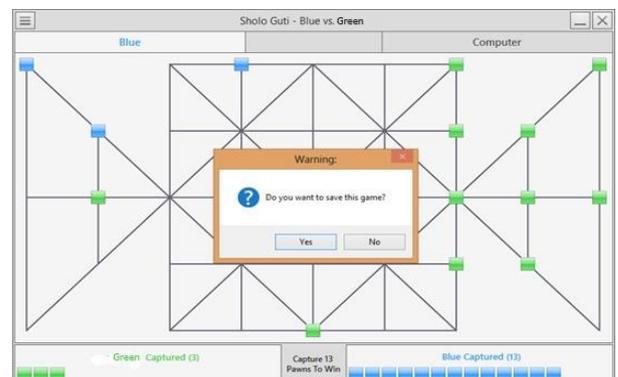


Fig 9: Save & Exit

7. CONCLUSION

Different person thinks in different way. This game is developed from a specific point of view and analysis. For developing or designing this game, many problems or difficulties actually came out. So there is a chance that the problems can be solved in a much more easy way by using advanced techniques. As it is a rural game which is going to extinct, so a smart looking playing board or UI is developed to save this rural game.

7.1 Challenges to overcome

It is true that the development environment is totally unpredictable. Many things go wrong while working on the real project. Firstly, any well-defined structure of the game haven't found. Secondly, Some problems arose while developing the system like creating database cost us more time fetching data from database for this a three dimensional array is used instead of using database. Finally, faced some problem in .NET platform but managed to fix it all.

7.2 Future scope

This game will be enhanced:

- i. To AI version
- ii. To More customization and animation
- iii. To Android version

8. REFERENCES

- [1] Whitby, M. (2017) "Designing and Creating a Game Installation", *The Computer Games Journal*, 6(3), 85–109. Doi: 10.1007/s40869-017-0039-2
- [2] Coleman,R., Krembs,M., Labouseur,A. & Weir, J. (2005) "Game design & programming concentration within the computer science curriculum", *ACM SIGCSE Bulletin*,37(1), 545-550, doi:10.1145/1047124.1047514
- [3] Symborski,C., Barton,M., Quinn,M.M., Korriss,J.H., Kassam,K.S., & Morewedge, C.K. (2017), "The Design and Development of Serious Games Using Iterative Evaluation", *SAGE Journals: Games and Culture*, 12(3), 252-264. doi: <https://doi.org/10.1177/1555412016673262>
- [4] Sethi,M. (2008), "Game Programming for Teens, 3rd Edition (Computer Game and Simulation Programming)". Retrieved from: <https://www.amazon.com/Game-Programming-Teens-ComputerSimulation/dp/1598635182>.
- [5] Martey,R.M., Galley,J.S., Shaw,A., McKernan,B., Saulnier,T., McLaren,E., Rhodes,M., Folkestad,J., Taylor,S.M., Kenski,K., Clegg, B., & Strzalkowski,T. (2016), "Balancing Play and Formal Training in the Design of Serious Games", *SAGE Journals: Games and Culture*,12(3),269-291. Doi: <https://doi.org/10.1177/1555412016674809>
- [6] Flores,E., Tobon,G., Cavallaro,E., Cavallaro,F.I., Perry,J.C. & Keller,T. (2008), "Improving patient motivation in game development for motor deficit rehabilitation", *ACE '08 Proceedings of the 2008 International Conference on Advances in Computer Entertainment Technology*, 381-384. Doi: 10.1145/1501750.1501839
- [7] Goude, D., Björk, S., Rydmark, M. (2007). "Game Design in Virtual Reality Systems for Stroke Rehabilitation", *Stud Health Technol. and Inform.* 125:146-148.
- [8] Cameirao, M. S., Badia, S. B., Zimmerli, L., Oller, E. D., Verschure, P. F. M. J., (2007), "The Rehabilitation Gaming System: a Virtual Reality Based System for the Evaluation and Rehabilitation of Motor Deficits." *Virtual Rehabilitation*, 2007 (September 27-29) 29-33. doi: <http://doi.acm.org/10.1109/ICVR.2007.4362125>
- [9] Amory, A., Naicker, K., Vincent, J., & Adams, C. (1999), "The use of computer games as an educational tool: Identification of appropriate game types and game elements", *British Journal of Educational Technology* 30(4), 311 – 321. doi: 10.1111/1467-8535.00121
- [10] Cheng, C.H., Ho Su, C. (2012), "A Game-based learning system for improving student's learning effectiveness in system analysis course", *Procedia - Social and Behavioral Sciences*, 31,669-675. doi: <https://doi.org/10.1016/j.sbspro.2011.12.122>
- [11] Schrier,K. (2017), "Designing Games for Moral Learning and Knowledge Building", *SAGE Journals: Games and Culture*. Doi: <https://doi.org/10.1177/1555412017711514>
- [12] Dondlinger, M.J. (2007), "Educational Video Game Design: A Review of the Literature", *Journal of Applied Educational Technology*, 4(1). Retrieved from: https://www.researchgate.net/publication/238444705_Educational_Video_Game_Design_A_Review_of_the_Literature.
- [13] Cagiltay, N.E. (2007), "Teaching software engineering by means of computer-game development: Challenges and opportunities", *British Journal of Educational Technology*, 38(3), 405–415. doi:10.1111/j.1467-8535.2007.00705.x
- [14] Rhodes,R.E., Kopecky,J., Bos,N., McKneely,J., Gertner, A., Zaromb,F., Perrone,A. & Spitaletta,J. (2017), "Teaching Decision Making With Serious Games: An Independent Evaluation", *SAGE Journals: Games and Culture*, 12(3), 233-251.doi: <https://doi.org/10.1177/1555412016686642>.
- [15] Strzalkowski,T., & Symborski, C.(2016), "Lessons Learned About Serious Game Design and Development", *SAGE Journals: Games and Culture*,12(3), 292-298. Doi: <https://doi.org/10.1177/1555412016673524>