1. **Abstract:**
   * Provide a concise summary of the project's objectives, methodologies, and outcomes.
2. **Introduction:**
   * Introduce the project's context and purpose.
   * State the problem or need your project addresses.
3. **Methodology:**
   * Describe the technologies and tools you used.
   * Explain the development process, including planning, coding, testing, and refinement.
4. **Design and Implementation:**
   * Detail the architecture and design of your project.
   * Discuss how you implemented key features and functionalities.
5. **Challenges and Solutions:**
   * Outline challenges you encountered and how you resolved them.
   * Share insights gained from overcoming obstacles.
6. **Results and Findings:**
   * Present the project's outcomes, including functionality achieved.
   * Include screenshots, diagrams, or data to support your findings.
7. **Discussion:**
   * Analyze the significance of your project's outcomes.
   * Reflect on lessons learned and areas for improvement.
8. **Conclusion:**
   * Summarize the key points of your report.
   * Discuss the impact of your project and its relevance.
9. **References:**
   * Cite sources or research materials used in your project.

**Snake Game**

Juan Ballas Marchant

Bellarmine University

CS 150 – 01

Dr. Sayani Sarkar

12/12/2023

**Abstract:**

Python has an unlimited world of options but for this opportunity the projects rely on simplicity. The point is for the user to enjoy and have fun with this project. The snake game will use some complex codes to provide a simple and fun time. The paper will go over what the codes are and the purpose of each of them. It will also have an overview of key features, challenges, and the actual output of the code, the game. It is important to understand that the purpose of this project is to have fun and enjoy this side of Python.

**Introduction:**

Python gives you unlimited options when it comes to codes and functions. It can go from a simple math problem to complex data analysis. In this occasion, the project will focus on a different area of Python. On the recreational side of it you can use codes to have fun and provide the user with some entertainment.

The project is about a game. The classic Snake games. It consists of a snake that moves through the screen while eating fruits positioned randomly in the screen to grow and grow. The more fruits the snake eats, the higher the points the user gets. That is the whole purpose of the game, and of course, do not lose. The user must avoid hitting the walls or hitting the snake’s body. We are using Python to come back to our basics and use some simple and some not so simple coding to provide the user a good time.

**Methodology**

This project is based on Pygame language. It was imported into the code and that allowed all the other codes to follow that path. The codes will be split into some parts like defining some conditions, providing the base of the game, the development of the game, and the end of the game. There were other libraries downloaded like time and random. All these three were the base for the rest of the coding.

The structure of the coding is simple. The code starts by defining and initiating some basics like displaying the window of the game. defining the size of it, the color, the shape are some of the basic features of this part. Also define some of the default aspects of the code like the size and position of the snake. These are some of the variables of the project. Most of them are lists, and all of them are related to the snake itself. The size of it, the growth method, the color, and the conditions of how to lose the game.

**Design and Implementation:**

The approach for this project was to make it as understandable as possible. There is some complex functions that are more advanced but at the end with some explanation they are understandable. The design is based on a rectangle window with a black background. On it, a snake of a list of size four appears on the top left side of the screen. This was strategically positioned as the snake will automatically move to the right, so that position gives the user time to start the game.

Then there are many “if” conditions that will define the movements of the snake of going up, down, right, or left. One of the keys of this functions was limiting the movements to two movements and not allowing the one that would move the snake to the opposite side of which is currently moving. This means that if the snake is going up, it can only change direction to the right or left. If the snake is moving left, it can only change direction up or down.

The other key part of the code were the “fruits”. The fruits are random dots that appear one at a time on the screen. The codes will position them randomly throughout the screen allowing one dot to be always displayed throughout the game. The purpose is for that snake to pass through the position of the dot and that is when the snake will increase its size as it will incorporate that dot into its body. In code language, the list is adding elements, so the list has more elements each time this happens.

Finally, there is “def” functions that allow the score to be displayed and the game over message to be displayed whenever the user loses. This was done like it was done under “class” functions. For the score display function, some new functions were used like “.font” which allowed to give the message some characteristics like font, size, and color. Another one that was crucial was the “.blit” function that will copy the contents of one surface into another one. In other words, first the block where the message was going to go was created and then the text was pasted in that block. The same process took place for game over with the only difference that there was a function that showed the message after losing the game and for only three seconds.

**Challenges and Solutions**

The main challenge was getting familiar with the functions I did not know. For example, defining the colors. At first, I had to define the variables and give them a color by balancing the three primary colors. Red, green, and blue were easy as they are the primary ones. I decided not to do mixes and go with black and white. White being all three colors topped out and black being lack of color.

The other one that was challenging was the functions like blit, rect, fill, pygame.KEYDOWN. There was nothing that was too complicated for me not to understand it. What made it easier to understand was that the name of the functions where saying what they are for example, “.rect” is the functions of giving a rectangle block to put a text on so it was not too complicated not to succeed.

The last challenge was to personalize the code and provide the #descriptions to each function. Everything must be as simple as possible, and at times it is hard to simplify what is already simple. There is exactly where complications take place. But at the end, the parts that took me the long to understand were the ones that I had the most excitement to edit.

**Results and Findings**

The result is very straight forward. The window with the caption “Snake Game by Juan Ballas” and the window with black background with a green snake on top. The “fruits” are white dots. The score board and the game over message are both displayed in red.A screenshot of a computer

Description automatically generatedA screen shot of a computer

Description automatically generated

**Discussion**

The project was good as it took me out of my comfort zone. It was interesting to go a little deeper into Python and have a ending results that is out of the actual cell. I was used to the classic output that comes out after the code, but it was satisfying to create something rather than just have an output.

It was also satisfying to create something with a real purpose, create something real. I can show this game to people, and I am truly proud of what I have made and when they ask how I made it, I can respond to all questions. This project gave me a boost of confidence regarding Python. Yes, the snake game did not come out from my creativity, but in the end, I was able to make some adjustments and make it my own. Later in life, I will get some information from the internet, and I will mix it up with my own to create amazing projects.

**Conclusion**

In conclusion, I really enjoyed doing this project as I learnt a lot. I joined this class to learn Python, and I think after 4 months of classes, assignments, and projects I can say that I have learnt a lot. The snake game seems simple at sight, but the coding process has to be methodic. What matters the most is to understand what you are doing. For me, I decided to go with something easier, but I was able to understand what I was doing instead of going for something harder without knowing what was happening. If I had to do this project again, I would choose the snake game due to what I am taking from it. I think one day I will be able to do a more advanced project that is related to my field.

**References**

Edureka. (n.d.). Snake Game with Pygame. Retrieved December 12, 2023, from <https://www.edureka.co/blog/snake-game-with-pygame/>

GeeksforGeeks. (2021, June 17). Snake Game in Python using Pygame module. Retrieved December 12, 2023, from <https://www.geeksforgeeks.org/snake-game-in-python-using-pygame-module/>

Sharma, P. A. (2021). Snake Game in Python Using Pygame module [Final]. Retrieved December 12, 2023, from <https://www.scribd.com/document/647838144/Snake-Game-in-Python-Using-Pygame-module-final-docx>

Sharma, P. A. (2021). SnakeGame.py. Retrieved December 12, 2023, from <https://github.com/PavanAnanthSharma/Snake-Game-using-Python/blob/main/SnakeGame.py>

Art of Problem Solving. (n.d.). Getting Started With Pygame. Retrieved December 12, 2023, from <https://artofproblemsolving.com/wiki/index.php/Getting_Started_With_Pygame#:~:text=Open%20a%20terminal%2C%20and%20type,prompt%20enter%20%27import%20pygame%27>.

Pygame Community. (n.d.). Pygame Module Index. Retrieved December 12, 2023, from <https://www.pygame.org/docs/py-modindex.html>

Stack Overflow. (2016, June 16). & symbols in Python. Retrieved December 12, 2023, from <https://stackoverflow.com/questions/37845445/and-symbols-in-python>

Stack Overflow. (2016, June 14). What is the surface.blit() function in Pygame? What does it do? How does it work? Retrieved December 12, 2023, from <https://stackoverflow.com/questions/37800894/what-is-the-surface-blit-function-in-pygame-what-does-it-do-how-does-it-work>