Super Snake
Final Report

By: Maor Gornik, Luke Liu, Qirong Su, Rahim Somjee

Project Name	**ICS4U Performance Task**
Product Name	**Super Snake**
Product Release Version	**1.0.0**

Date: 14/06/2019

Project Scope Statement

Date: 18 May 2019

From: Maor Gornik
Luke Liu
Qirong Su
Rahim Somjee

Subject: [Super Snake] Project Scope Statement

Version	Date	Author	Description
	[DD-MM-YYYY]	[Your name]	Draft initiated
1.0	18-05-2019	All group members	Finished initial version of Project Scope Statement
1.1	14-05-2019	All group members	Updated after project completion

Project Overview

[Identify the name of the project and a simple project description.]

Project Name: Super Snake
This project is going to an extended version of the previously popular game known as “Snake”. The player will control a snake and go around eating different things including apple, cherry and enemy snake; each with its own advantages/disadvantages. The player will lose if the snake head collides with its body, and the goal is to become as big as possible. The score will be determined by the size of the snake before the player eventually loses. This version of snake will include a variety of additional gameplay features such as arena variations, obstacles, etc. There will also be a leaderboard to keep track of the high scores.

Project Objectives

[Identify overall project objectives in terms that is represented in some form of a measurable success criteria (which usually refers to the Triple Constraint: time, costs, scope, and optionally quality).]

Objectives
This version of Snake will look to add to the game in a way never done before. It will aim to satisfy and exceed player expectations will its additional features while still maintaining to preserve the original user experience of the vanilla version of Snake. The goal is to entertain players with this basic game in ways that they have previously not experienced before with the original Snake and hopefully enhance their enjoyment of the game and therefore increasing play time. Since most users were unable to spend large amounts of time on the original version of Snake and got bored easily, this project will hopefully change their opinion of how much leisure time they want to spend playing the game out of amusement.

Product Scope Description

[Overall definition and characteristics of the software product feature and function set.]

This project will be developed and run in a Greenfoot environment using Java. It can run on any operating system or platform that runs Java’s virtual machine (JVM) and supports Greenfoot and its native libraries. The player will be able to start, stop, pause, resume, etc whenever needed using the Greenfoot menu. The project is meant to be run as smoothly as possible on user devices and will be optimized for efficiency.

Users

[This is a list of the users of the software. As with the stakeholders, each user can be referred to either by name or by role (home website user, support rep, etc.). The needs of each stakeholder are described in a few sentences.]

Users/Players: To satisfy any person of any age that either want to enjoy the nostalgic atmosphere of this old yet highly-addictive game or is interested and finds amusement in playing the Snake game with new features added.
Stakeholders: to test out the game and ensure that it is playable and profitable

Project Assumptions

[This is the list of assumptions that the stakeholders, users, or project team have made. These assumptions often relate to the amount of time that a programmer expects to complete the tasks. Identify what you think are true.]

The players are expected to be familiar with the basics of the game Snake; controls, rules, and the objectives which will be provided in the main menu. They are also expected to read the instructions menu of the game since many new features will be added to the game and the player might be unfamiliar with them. The users are also expected to have Greenfoot downloaded and know how to run it in that environment. The stakeholders are assuming that our game will be profitable, and will serve a purpose to consumers and buyers. The team assumes that we would be able to finish the project and implement all features successfully in a 2 week time period.

Project Deliverables

[Project deliverables are the output of a team to guide towards the successful completion of that project. They can be anything from an end product, a process, an outcome, or they can also refer to the tasks within a project management plan]

- Game prototype (expected 1 week into programming)
- Test plan for prototype
- Test report for prototype
- Finalized game

Product Features

[This section contains a list of features. A feature fulfills a specific need by providing a service or capability. Each feature should be listed in a separate bullet point. It should be given a name, followed by a description.]

- ➔ Screens - User Friendly Interface
  - ◆ Many of the elements were uniquely designed for the game with user-in mind
- ➔ Different levels - Progression
  - ◆ More challenging maps (more walls)
- ➔ Different kinds of food for the snake to eat
  - ◆ Apple (gain 1 point)
  - ◆ Cherry (gain 2 points)
- ➔ Different obstacles to dodge
  - ◆ Enemy snake (if the snake head touches it, gain 5 points; if the snake body touches it, lose 5 points)
  - ◆ Randomly generated Pylons
- ➔ Timers and auto-score save
  - ◆ These elements were created to intensify the game and make it more challenging
- ➔ Sound effects
  - ◆ Carefully chosen soundtracks that reflect the game's state
  - ◆ Soundtracks change after some time
- ➔ Our own art (obstacles, food, screens)
  - ◆ The elements used in the game were carefully designed to immerse the player into a whole new world of snakes and challenging situations.
    - Buttons, screen visuals, and images were delicately modified specifically for the game

Project Constraints

[Identify key known limitations—usually time, cost, and other factors that have a direct impact on Scope. This often lists features that will not be developed. Features are often left out of a project on purpose. When a feature is explicitly left out of the software, it should be added to this section to tell the reader that a decision was made to exclude it. For example, one way to handle an unrealistic deadline is by removing one or more features from the software, in which case the removed features should be moved into this section.]

There may be a possible constraint of time since we only have two weeks to work on the project. As a result, some features that we have discussed initially may be left out. We do not expect to encounter any other limitations such as cost, as it is not a factor for this project.

Schedule Milestones

[Identify key milestones (most practically, these are ranges initially). This is NOT meant to be a detailed schedule, just a list of overarching deadlines.]

Software Scope Document	May 13 - May 19
Software Project Plan	May 20 - May 26
Implementation of Prototype	May 27 - June 2
Implementation of Finalized game	June 3 - June 9
Testing and Verification	June 10 - June 12
Finalization of Report	June 13 - June 14

Software Project Plan

Date: 25 May 2019

From: Maor Gornik
Luke Liu
Qirong Su
Rahim Somjee

Subject: [Super Snake] Software Project Plan

Version	Date	Author	Description
	[DD-MM-YYYY]	[Your name]	Draft initiated
1.0	25-05-2019	All group members	Finished initial version of Software Project Plan
1.1	14-05-2019	All group members	Updated after project completion

Architectural Design

Outline of Classes

World

- SnakeWorld

Actor

- Button
    - BackButton
    - InstructionsButton
    - LeaderBoardButton
    - PlayAgainButton
    - StartButton
- Labels
    - LevelLabel
    - ScoreLabel
    - Timer
    - Instructions
- Food
    - Apple
    - Cherry
- Obstacles
    - Wall
    - Pylon
- SnakeHead
    - EnemyHead
- SnakeTail

Methods in SnakeWorld

- Constructor
- Start Screen
- Start World
- End World
- Place Instructions Labels
- Place LeaderBoard Labels
- Display Level
- Get Game Frames
- Get Score
- Increase Score
- Decrease Score
- Add Enemy
- Add Pylon
- Add Food

- Generate Coordinates (random)
- Started (controls what happens after the run button is pressed)
- Stopped (controls what happens after the pause is pressed)
- Get Scores (from the document)
- Input Score (into the document)
- Act (controls what happens when the world is running)

Methods in SnakeHead

- Constructor
- Is Colliding (test for collision with obstacles)
- Colliding Food (test for collision with food)
- Move Tongue
- Teleport (wrap the snake around the edge)
- Munch Sound
- Snake Move (turning and movement)
- Act (controls what happens when the world is running)

Methods in SnakeTail

- Constructor
- Change Last Tail
- Set Life Duration
- Get Life Duration
- Get Tail Frames
- Add To List (sort snake tails based on how long they have been alive)
- Is Snake Turning
- Twist Image (Change image to reflect the snake's turn)
- Twist Body
- Act (controls what happens when the world is running)

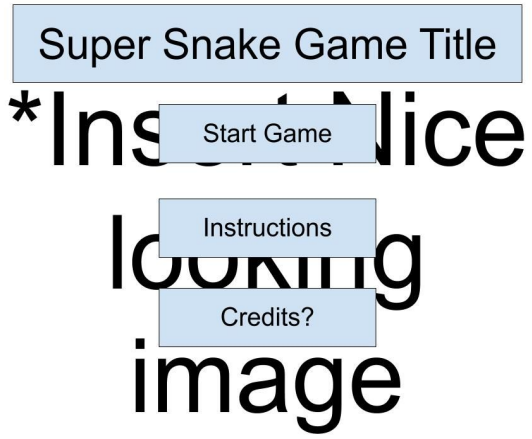
Methods in other classes

- Constructor
- Act (controls what happens when the world is running)

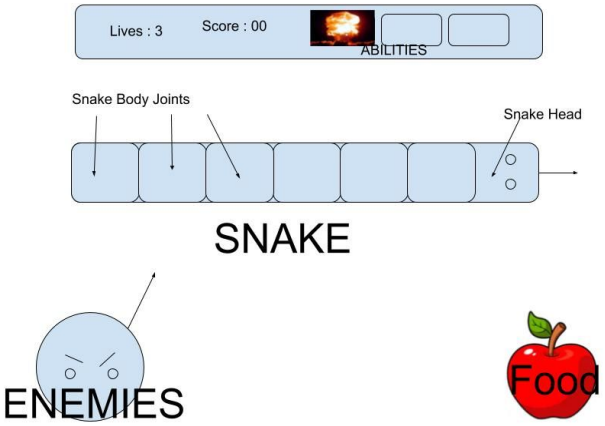
Storyboards

Rough Drafts

Title Screen



Game Screen



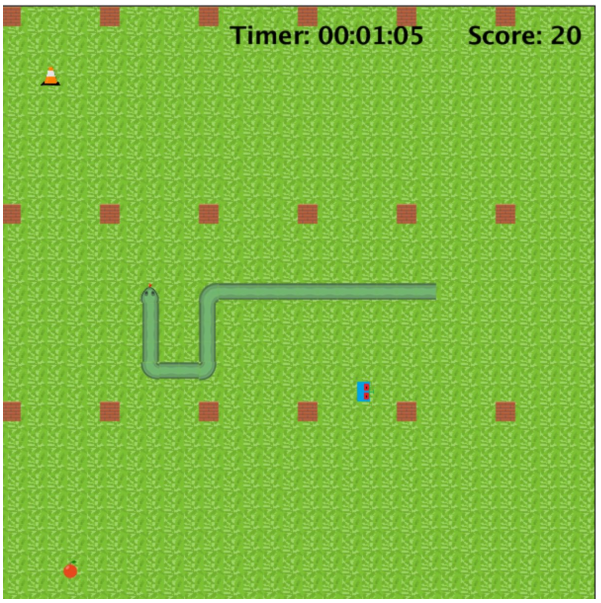
Final User Interface Designs

Start Screen

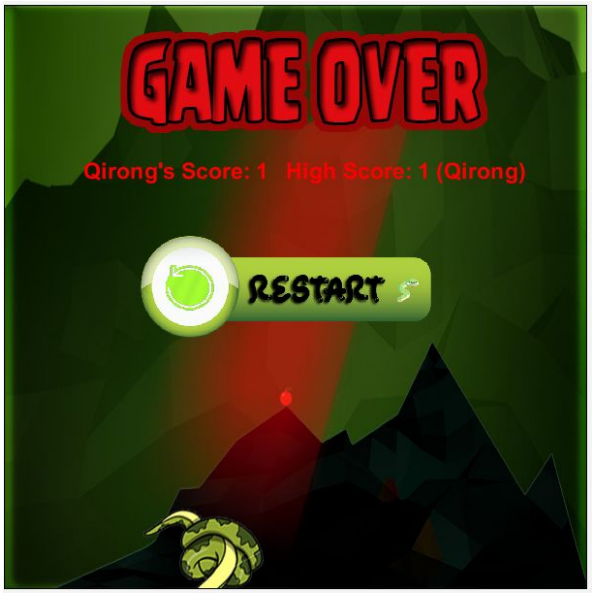
Instructions Screen

Leaderboard Screen

Game Screen



End Screen



Statement of Work

Names	Color Codes
Luke	XXX
Rahim	XXX
Qirong	XXX
Maor	XXX

Critical classes:

World

- SnakeWorld

Actor

- Button
- Labels
- Food
- Obstacles
- SnakeHead
- SnakeTail

Additional features:

- Screens - User Interface
- Different levels - progression
    - More challenging maps (more walls)
- Different food for the snake to eat
    - Apple (gain 1 point)
    - Cherry (gain 2 points)
- Enemy snake (if the snake head touches it, gain 5 points; if the snake body touches it, lose 5 points)
- Sound effect
- Our own art (snake, obstacles, food, screens)

Work Breakdown

Time estimates for each feature

Critical classes:

- SnakeWorld (8-10 hours)
- SnakeHead and SnakeTail (8-10 hours)
- Button and Labels (3-5 hours)
- Food and Obstacles (4-6 hours)

Additional features:

- Screens - User Interface (3-5 hours)
- Different levels - progression (2-3 hours)
- Different food for the snake to eat (1 hour)
- Enemy snake (2-3 hours)
- Sound effect (1-2 hours)
- Our own art (2-3 hours)

**Estimated time for each programmer: 10-12 hours*

Super Snake User Guide

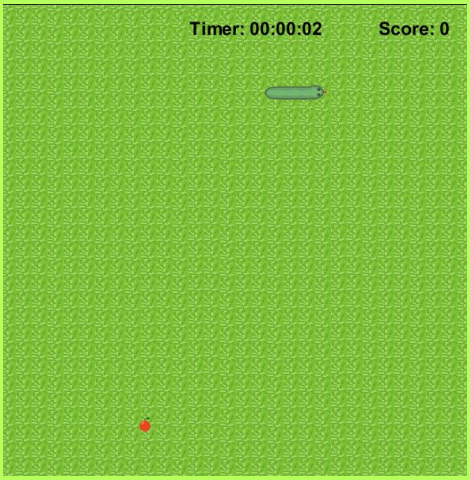
When you start up the game, you will be greeted with the start screen. Click the “How to Play” button to see instructions on how to control the snake and objectives during the game. Once familiar with the instructions, click on the “Back” button to return to the start screen.

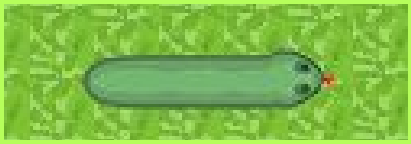
You can see current high scores by clicking the “Leaderboard” button and seeing the leaderboards. If there are no high scores yet, it will be displayed as empty. Click on the back button to return to the start screen.

To begin the game, click on the “Start” button. Before the game starts, you will be prompted to enter your name. Enter your name and press OK.

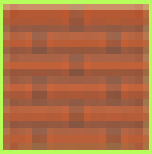
Super Snake User Guide

When the game begins, you will see your snake character moving across the screen. You can change the direction that the snake moves with the arrow keys. You will also see a food item in the world. The objective of the game is to eat foods and survive for as long as possible

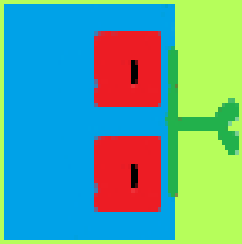




As you progress through the game, pylons walls will appear every 10 seconds. For every 10 points you gained, you would go onto the next level with more sophisticated designs of walls. These are obstacles that you need to avoid.



Another obstacle that you will encounter is enemy snakes, which are blue snakeheads that move in a single direction. If the player snake's head eats the enemy snake, the player snake will gain points. However if the enemy snake eats a part of the player snake's body, you will lose points.



Super Snake User Guide

When you collide into an obstacle or run out of points, you will be presented with the game over screen, where you can see your score and compare it to the high score. You also have an option to click on the restart button and begin the game again.

Testing Instructions

1. Open the program in Greenfoot and click run.
2. Check the user interface for ease of use and professionalism.
3. Test the instructions button. Pictures explaining the controls and rules of the game should be displayed once the instructions button is clicked. Check whether the rules are easy to understand.
4. Test the back button. This should return to the home screen with a start button, an instructions button and a high scores button.
5. Click on the start button to start the game.
6. Utilize a variety of number, character, and symbols when an input box appears. Test inputting empty names and names that exceed 6 characters.
7. Test the arrow keys to make sure they change the direction of the snake in the corresponding direction as the snake begins to move.
8. Direct the snake to eat a fruit. Check to see that the fruit disappears and a new one appears somewhere else in the world.
9. Verify that the timer and the score labels are displayed and updated correctly throughout the game.
10. Verify that the pylons are popping up every 10 seconds and enemy snakes are popping up every 30 seconds.
11. Verify that after every 10 points, there is an increase in level with walls popping up.
12. When the game is over, verify that the output (score and name), is presented correctly and appropriately
13. Test the restart button. This should return to the home screen with a start button, an instructions button and a high scores button.
14. Start the game. Make sure all game-related variables are reset and the game starts again normally.
15. After playing the game again, return to the home screen and click on the leaderboard button to verify that the scores are recorded
16. Continue playing for as long as you wish to check for user experience.
17. Complete post-testing questions.

Test Record Template

*Tester, feel free to use this template to keep track of your results during the testing to help you complete the post testing questions.

Testing Step	**Result**	**Details and Explanation (if necessary)**
1. Does the Game Run?		
2. Is the user interface easy to read and use?		
3a. Does the “Instructions” button work?		
3b. Are the instructions easy to understand?		
4. Does the “Back” button bring you back to the home screen?		
5. Does the “Start” button work?		
6a. When more than 6 characters are inputted for the name, are there any errors thrown?		
6b. When a variety of numbers, characters, and symbols are inputted for the name, are there any errors thrown?		
7. Do the arrow keys change the direction of the snake in the corresponding direction?		
8a. When fruits are eaten, do they disappear?		
8b. When fruits are eaten, does a new food appear somewhere else in the world?		

9a. Are the timer and score labels displayed properly		
9b. Do both labels update their information as score and time changes?		
10a. After 10 seconds does a pylon appear somewhere in the world?		
10b. After about 30 seconds, does an enemy snake appear somewhere in the world?		
11a. After achieving a score of over 10 points, does a label appear indicating in increase in level?		
11b. After reaching the next level, are multiple wall objects placed somewhere in the world?		
12a. When the game is over, is the inputted name properly displayed?		
12b. When the game is over, is the score properly displayed?		
13. When the restart button is pressed, does the home screen appear again?		
14. When the game is started after pressing the restart button, is the name, score, and world fully reset?		
15. After clicking on the “Leaderboard” button on the home screen, are the proper high scores displayed?		

Name: Maor Gornik **Date: 6/13/2019**

Post Testing Questions:

Please answer the questions accurately and to the best of your ability

1. Does the game work well?

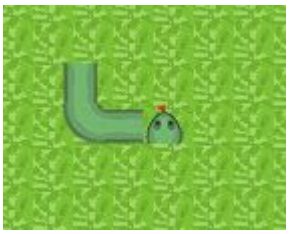
The final result of the game is working great and exceptionally well with all the added features. The game is light, challenging and easy to use, containing modern looking pictures, attractive and responsive. Along with that, it has clear instructions that cover all the game's mechanics and keeps the player focused and engaged at all times.

2. Are there any problem areas? If so, what are they?

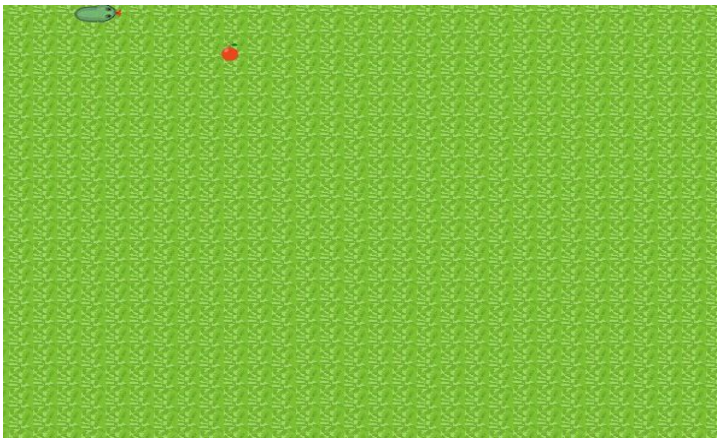
Yes, there are certainly a couple of problematic areas in the game that could have been fixed given they were identified earlier. When the player, for instance, makes a sudden turn, the snake's head part would disconnect from its body and would look like as if the snake is turning towards a different direction (C1). On top of that, when the player decides to go in one direction and then abruptly turn back, the snake collides its own body (C2). Another unexpected bug that may occur when the player levels up is that more obstacles are spawned onto the world, which might be spawned directly on top of the snake.

Cases:

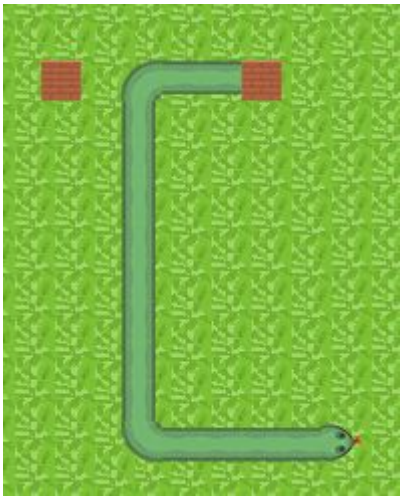
C1:



C2:



C3:



3. Is the program useful to you or to the target audience, and why?

This program is a game that is useful because it can be played to kill time, allow one to escape their problems, improve reflexes and quick thinking, and entertain

absolutely everybody regardless of their age. This game is something that everyone can easily get attached to, particularly because it is highly addictive and simple to use. It is thanks to its simplicity that it provides such an exceptional excitement to its users.

4. Is the program easy to use?

This program is absolutely easy to use since it includes instructions that clarify all the requirements to play this game, it is built upon a basic and simple, yet addictive concept that can easily be picked up by utterly anybody who would be willing to give it a try.

5. Do you have any recommendations for improvement to the product?

Yes, absolutely. Like other games, there is always room for improvement. First of all, I would try and tackle all the bugs, and fix them before I continue any further. Then, depending on the amount of time I am given, I will brainstorm new ideas and create a structured sketch or a diagram (i.e flowchart) and carefully plan and analyze the logic that must be used. To be a little more specific, I would most likely try to add different skills that the player would be able to acquire and use to intensify the game. As an example, skills such as shooting at obstacles, temporary shrinking the snake, and things of that nature would be considered and added to the game.

Name: Luke Liu **Date: 6/13/2019**

Post Testing Questions:

Please answer the questions accurately and to the best of your ability

1. Does the game work well?

The game works very well and contains many features that go beyond the simple snake game. The user interface is nicely designed with an easy-to-understand instruction screen and a high scores screen.

2. Are there any problem areas? If so, what are they?

There are minor problems that could be worked on and improved. The obstacles such as the pylon and wall may pop up right in front of the snake, so the user might die without knowing what occured. When the snake turns 180 degrees really quickly (e.g. the snake is travelling downwards, go left and then go up), the snake collides with its only body and dies. There is a very slight animation issue when the snake turns twice very quickly where the body does not seem to be connected.

3. Is the program useful to you or to the target audience, and why?

This game is extremely fun to play and keeps me addicted. It is a good recreational game during spare time where I could just relax and have fun. It builds on the existing snake game with many extra features (enemy snake, levels, different food). It is a useful program that has great potential.

4. Is the program easy to use?

The program is very easy to use with clear user interface that contains buttons and instructions on different features. Most people already know how to play the simple snake game so it would not be difficult to play this game.

5. Do you have any recommendations for improvement to the product?

I would fix the “obstacle popping up right in front of the snake” issue as well as the snake turning issue. Otherwise the game is well done.

Name: Qirong Su **Date: 6/14/2019**

Post Testing Questions:

Please answer the questions accurately and to the best of your ability

1. Does the game work well?

Yes, the game is fun and satisfying to play. The game is the simple snake game, but with more features and depth added to it. The animations, background, and screens are nice and colourful and the buttons are straightforward and intuitive. The game also does not require prior knowledge or skills to play, and anyone can jump into.

2. Are there any problem areas? If so, what are they?

There are a few issues that could be fixed with the game. One large issue is that as the game progresses, pylons are added randomly and walls are placed as levels advance. When obstacles are placed, they do not take into consideration the position of the snake, so it is entirely possible for a pylon to be placed right in front of the snake, making the player instantly lose without having a chance to avoid it.
Another issue can be observed when players turn the head 180 degrees and run into their own bodies within one tick of gametime. This interaction was not an intended one and with more time, we would look into fixing it.

3. Is the program useful to you or to the target audience, and why?

Our target audience is people of all ages who have time to play simple games. With that in mind, this program is useful and can serve as a fun pastime. This game can be replayed as many times as the player wants and they can aim to achieve higher scores each time. The game is also very simple, allowing for anyone to get into.

4. Is the program easy to use?

Yes the program is easy to use. This is due to the clear instructions provided after clicking on the instructions button with pictures and icons to represent the controls and interactions and the fact that “snake” is a well known simple game already.

5. Do you have any recommendations for improvement to the product?

Apart from fixing the bugs, I would recommend adding new backgrounds as the levels change. Different music and more enemies could also make the game more interesting as it progresses.

Name: Rahim Somjee **Date: 6/11/2019**

Post Testing Questions:

Please answer the questions accurately and to the best of your ability

1. Does the game work well?
   The game works well. The graphics and animations are nicely done, and the overall experience is smooth, and easy. Some features were inconsistent such as the user interface. The back button was different than the front display and replay buttons but overall, everything was mostly consistent.

2. Are there any problem areas? If so, what are they?
   I'd say the only problem would be that the levels aren't explained in the instructions so that when the score gets high enough and the snake is somewhere near a wall, and a random wall pops up, the game just ends because the snake is technically colliding with a wall which is sort of unfair.

3. Is the program useful to you or to the target audience, and why?
   Yes the program fulfils its purpose as a fun little getaway game. It doesn't last too long, its fun, quick, easy, and helps to pass time when bored, which is why it is definitely a useful small game

4. Is the program easy to use?
   The program is fairly easy to use and understand. There are no things that are difficult about it.

5. Do you have any recommendations for improvement to the product?
   I'd say more information about the levels and timings of different things popping up would've been useful. Also some readability on the start screen could be improved but overall, this game is well done and fulfils its purpose.

Testing Summary

The project was tested by 4 four different users. Mostly, the responses were positive and appreciated the new additions to the vanilla version of Snake but also mentioned some very apparent bugs that would have been easily fixed. These included, inconsistency of button positioning, unfair wall/pylon/enemy spawning, and some minor animation issues with the snake's movement. Most tests mentioned how the game was a fun little outlet that was easy to use and hit a vast target audience (basically all ages unless they are unfamiliar with how a keyboard works). The games playability and ease of use was admired by all testers but some were not happy that the level changing mechanism was not explained either in the instructions or before the game started. There were many recommendations for improving this product and one of the major ones that kept being mentioned was the unfair spawns. Every time something was being spawned such as a wall or a pylon or even an enemy, the game would not check to see if the snake was already positioned in the spawn position and placed the object there either way. This caused the player to die since they would be touching the object that was placed on top of them even though there was no possible way for them to avoid their fate. This was and is one of the biggest issues with the game and there needs to be an update that tries to resolve this. Some good suggestions were made about how different maps for different levels along with different sound tracks would have given the game a new aspect and possibly made it more interesting and eye catching. Different sized enemies that had different capabilities such as shooting were also recommended. Some issues such as very minor readability and animation issues were also pointed out as potential bugs in the future but overall, the project was well received and we definitely hit our target audience.

MOM	Minutes of the Meeting	Date	18-May-19

4:00 PM - 6:15 PM

Project Name	***Super Snake***
Participants	*Maor Gornik, Luke Liu, Qirong Su, Rahim Somjee*
Remarks	1st Meeting - Project Scope Statement and Software Project Plan

Points Discussed	Remarks
1	Feasbility of project
2	Features to implement
3	Priority of features
4	Discussed constraints and limitations
5	Discussed Project Deliverables
6	Finished the scope document
7	Started working on Software Project Plan
8	Discussed classes and subclasses (Architectural Design)
9	Discussed the storyboard
10	Discussed methods under each class

Points Agreed	Remarks	Responsible	Completion Date
1	RPG element in game		
2	Multi-level progression		
3	Possibility of multiplayer		
4	Possible inclusion of creative artwork		
5	Different skill elements and maps		
6	Powerups and different abilities		
7	Create storyboard in Photoshop (starting and ending screens)	Maor Gornik	25-May-19
8	Will meet again May 25 at 4pm to finish software project plan		
9			
10			

Prepared by	*All participants*

MOM	Minutes of the Meeting	Date	25-May-19
			4:00pm-6:15pm

Project Name	***Super Snake***
Participants	*Maor Gornik, Luke Liu, Rahim Somjee*
Remarks	2nd Meeting - Software Project Plan

Points Discussed	Remarks
1	*Completed storyboards*
2	*Thought about the algorithm to make the game*
3	*Discussed how different classes interact together*
4	*Divided up work*
5	
6	
7	
8	
9	
10	

Points Agreed	Remarks	Responsible	Completion Date
1	Arena	Rahim	
2	Snake	Maor	
3	Obstacles/food	Luke	
4	Text buttons/display ui	Qirong	
5			
6			
7			
8			
9			
10			

Prepared by	*All participants*

MOM	Minutes of the Meeting	Date	27-May-19

4:30-6:00pm

Project Name	***Super Snake***
Participants	*Maor Gornik, Luke Liu, Rahim Somjee*
Remarks	Started working on code

Points Discussed	Remarks
1	*Started working on the code*
2	*Completed very basic features of the game*
3	*Everyone now has a copy of base game*
4	
5	
6	
7	
8	
9	
10	

Points Agreed	Remarks	Responsible	Completion Date
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			

Prepared by	*All participants*

MOM	Minutes of the Meeting	Date	28-May-19
			4:30-6:30pm

Project Name	***Super Snake***
Participants	*Maor Gornik, Luke Liu, Rahim Somjee*
Remarks	Continued working on code

Points Discussed	Remarks
1	*Completed code for the wall and pylon obstacles*
2	*Completed code for keeping track of the score*
3	*Completed code for wrapping the snake around the world*
4	*Fixed bugs on the movement of the snake and consuming food*
5	
6	
7	
8	
9	
10	

Points Agreed	Remarks	Responsible	Completion Date
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			

Prepared by	*All participants*

MOM	Minutes of the Meeting	Date	8-Jun-19

4:00 PM - 8:00 PM

Project Name	***Super Snake***
Participants	*Maor Gornik, Luke Liu, Qirong Su, Rahim Somjee*
Remarks	Coordination Meeting

Points Discussed	Remarks
1	coordinated changes
2	improved instructions
3	fixed bugs relating to restart button
4	added new snake graphics
5	began code commentation
6	began working on report
7	
8	
9	
10	

Points Agreed	Remarks	Responsible	Completion Date
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			

Prepared by	*All participants*

MOM	Minutes of the Meeting	Date	14-Jun-19

6:30pm-7:30pm

Project Name	***Super Snake***
Participants	*Maor Gornik, Luke Liu, Rahim Somjee*
Remarks	Finalized Report

Points Discussed	Remarks
1	*Finished individual testing*
2	*Wrote testing summary*
3	*Updated project scope document*
4	*Updated software project plan*
5	*Wrote the user guide*
6	
7	
8	
9	
10	

Points Agreed	Remarks	Responsible	Completion Date
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			

Prepared by	*All participants*