Ryerson University

Introduction to Software Engineering

(CCPS406 – Fall 2018)

Project Status Report#4

Project Requirements

Prepared for: Professor Nawar Hakeem

Prepared by: Fahad Jamil

Yashodhar Dave

Meet Pandya

Benjamin Travaglini

Contents

[I. System Specifications 3](#_Toc528951114)

[II. System and sub-system Functional specifications: 4](#_Toc528951115)

[III. Non-Functional Requirements 5](#_Toc528951116)

[IV. Technical Requirements 6](#_Toc528951117)

[V. Use case model 7](#_Toc528951118)

[VI. Activity Diagram 8](#_Toc528951119)

# System Specifications

When a user launches a webpage, the game board canvas will be painted with the welcome splash screen and brief instructions on how to run the game.

The user will have “Start Game”, “Pause Game”, “Stop Game” buttons control the game operations.

When the “Start Game” button is pushed.

The game board canvas will clear, and the colorful snake will be drawn in the centre of the screen with a randomly selected direction to start the snake movement in left, right, up or down direction slowly.

The user will then be required to control the snake’s movement maneuver using the keyboard arrow keys.

At the random time during the game play, a food object with distinct colour other than the snake colour would appear on the screen. It will stay on the screen until the user directs the snake to the food object at which point, the food object would be consumed by the snake and the food object would disappear from the screen. Optionally the snake’s length may increase.

* User should be able to control the snake using the keyboard arrow keys to move Up, Down, Left or Right
* A food object should be there on the game board canvas for the snake to eat and grow
* The snake should grow from the initial size after eating the food object
* The food object should be reachable by snake. In other words, the food object should be drawn a few pixels inside the game board canvas.
* The score should be displayed for the food consumed by the snake or the distance travelled.

# System and sub-system Functional specifications:

* Game board canvas attributes: will be maintained to set the initial size of the game board. The width, length and the colour will need to be maintained.

The size attributes may change depending upon the user action.

The game board canvas colour may be controlled internally depending upon game events.

* Snake Object attributes: will be maintained using an array object of variable length changing dynamically as game progresses.
  + - * Length
      * Snake head = First element
      * Snake tail = Last element
      * Snake body pieces = elements between head and tail. Initial number would start at 2, but can increase or decrease as game progresses
* Window event listener object: will be responsible for responding to following:
  + - * game board window size change,
      * game board window closing,
      * user pressing the arrow keys,
      * user selecting to pause the game
      * user selecting to restart the game
* Food object generation: This object should be generated at a random time to draw a food object on the canvas
  + - * One food object can be present at a time on the board canvas
      * Once the food object is consumed, the snake size can be adjusted, and the food object count should be reset to indicate that the new food object can be generated.
* Score object:
  + - * Current game score and
      * the current high score object.

Above will be used to display the score of the user in terms of food objects consumed by user’s snake. It may optionally display the distance travelled in the current game or the time of play in the current game.

# Non-Functional Requirements

* User should be able to run the game in any web browser
* Game board canvas should not be affected if the user changes the window size
* Snake should be confined to stay within the game board canvas. Alternatively, the snake could exit from right boundary and enter from left boundary or vice versa and could exit from top boundary to enter from the bottom boundary or vice versa.

# Technical Requirements

**Language used:**

* HTML Version 5 Graphic Objects
* JavaScript

**IDE:**

* Notepad++,
* MS-Visual Studio Editor

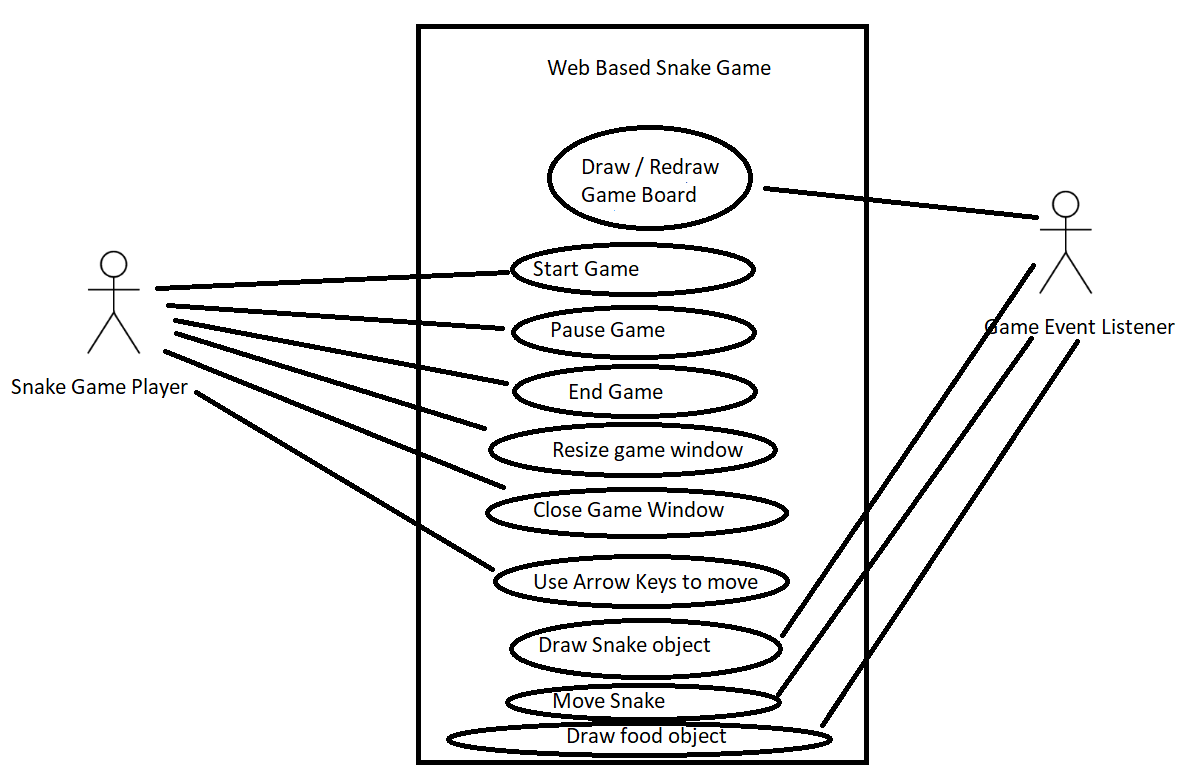
**SW Patterns:**

* Windows Event Listener for following:
  + listen for Keystroke events
  + Window closing event
  + Window resizing event
  + Button clicked event

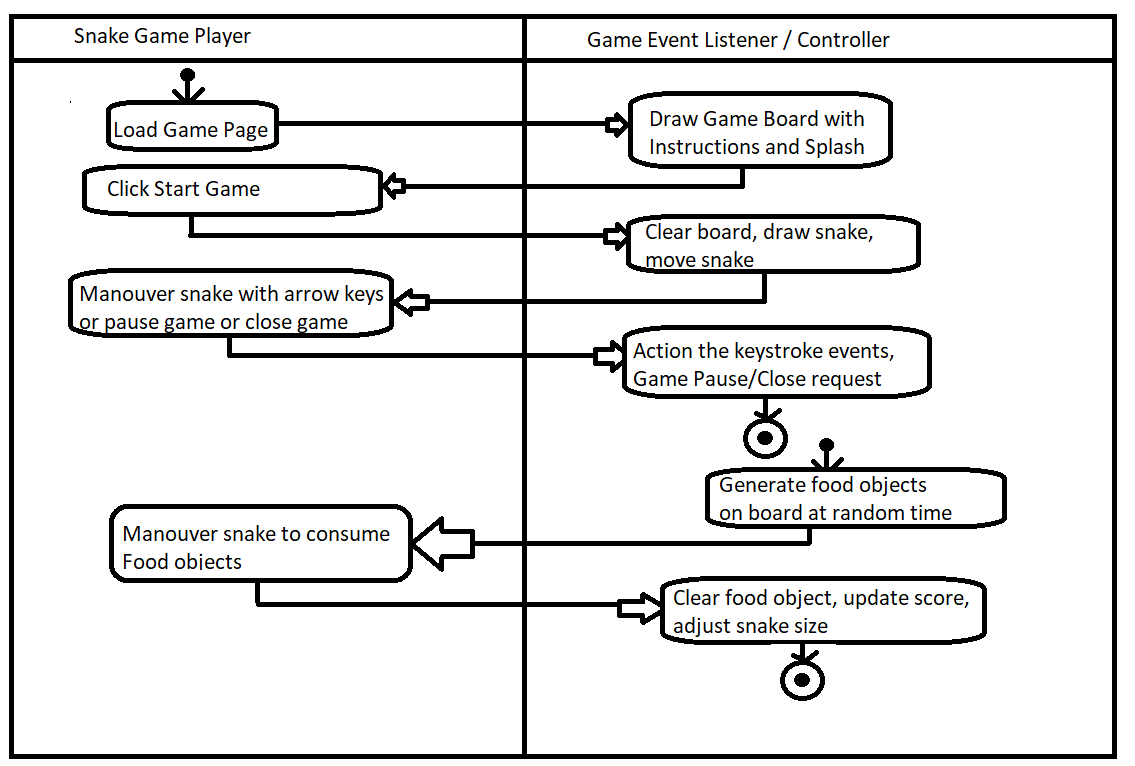
**HW:** Any hardware supported by the MS-Windows and Mac Operating systems.

A hardware supported by X-Windows environment on Main-frame or super-minis may also work, but it would be tested in future releases.

# Use case model



# Activity Diagram



-----\*\*\*-----