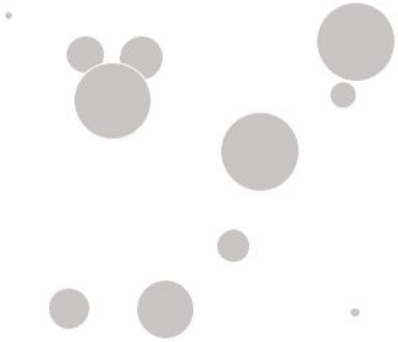
Game Description

Overview

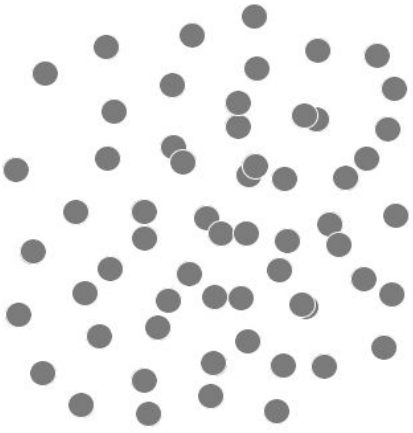
This game was inspired by Agar.io. In the game, you play as a bubble (instead of a cell!) that absorbs others cells to gain mass. Your goal is to reach a certain size and win the game.

There are 3 types of bubbles in the game. The player, an orange bubble:

This is what the player controls. It always steers towards the mouse. Its color fades away as time passes. When it becomes completely transparent, it dies and the game is over. The second type of bubble is the wanderer:



They are the grey bubbles randomly moves around. They can absorb each other and the player (in which case the game is over and the player loses) and they can be absorbed by the player bubble. Wanderer bubbles are constantly refilled to the screen with growing size. Lastly, there are flocker bubbles:



The flocker bubbles move around together like a flock of birds. They cannot absorb bubbles nor can they be absorbed. They have no effects on wanderer bubbles.

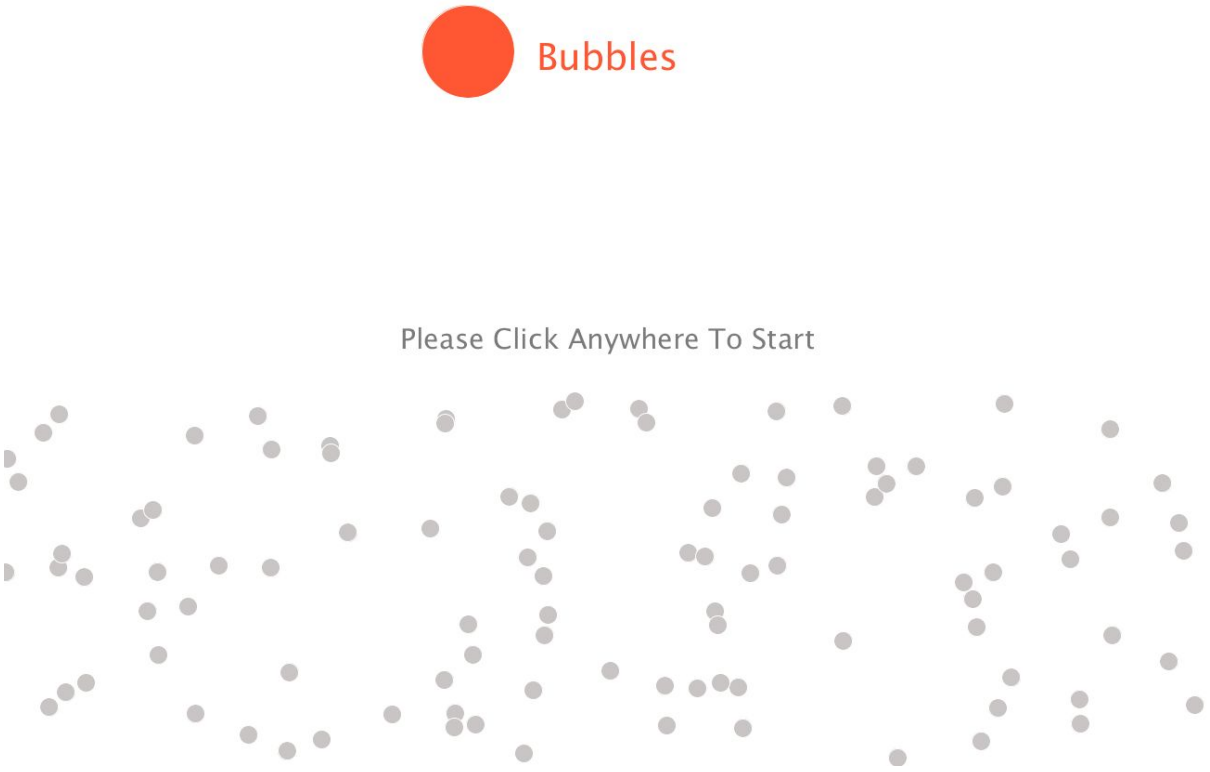
However, if the player bubble comes into contact with one of the flocker bubbles, it dies and the game is lost.

As mentioned earlier, your main goal in this game is to control the player bubble to absorb wanderer bubbles to become larger. You should also try not to be absorbed by other bubble and avoid touching the flocker bubbles. You should also keep an eye on the color of the player bubble, as it will slowly fade away. If the player bubble does not reach the target size by the time it becomes totally transparent, you lose the game. The last thing to note is that a bubble can only absorb another when the size difference between them is big enough.

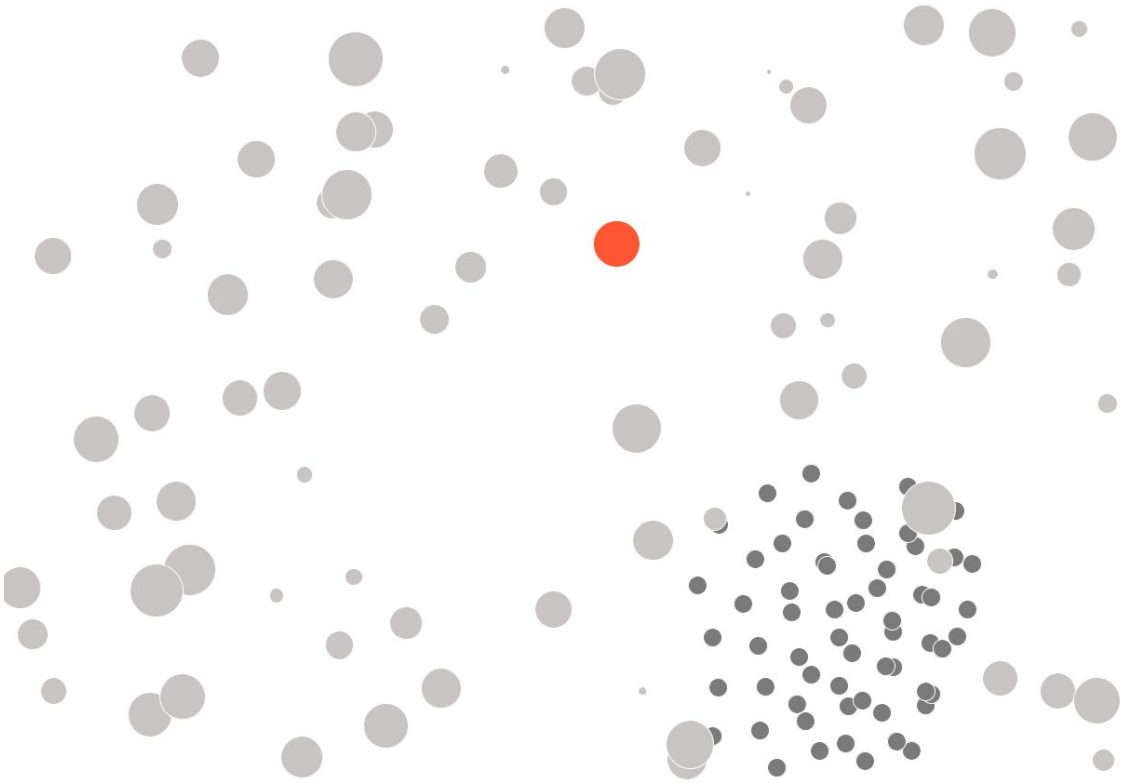
Control

The player uses the mouse to navigate the player bubble.

Screenshots



(The title screen)



(gameplay)

You Lose...

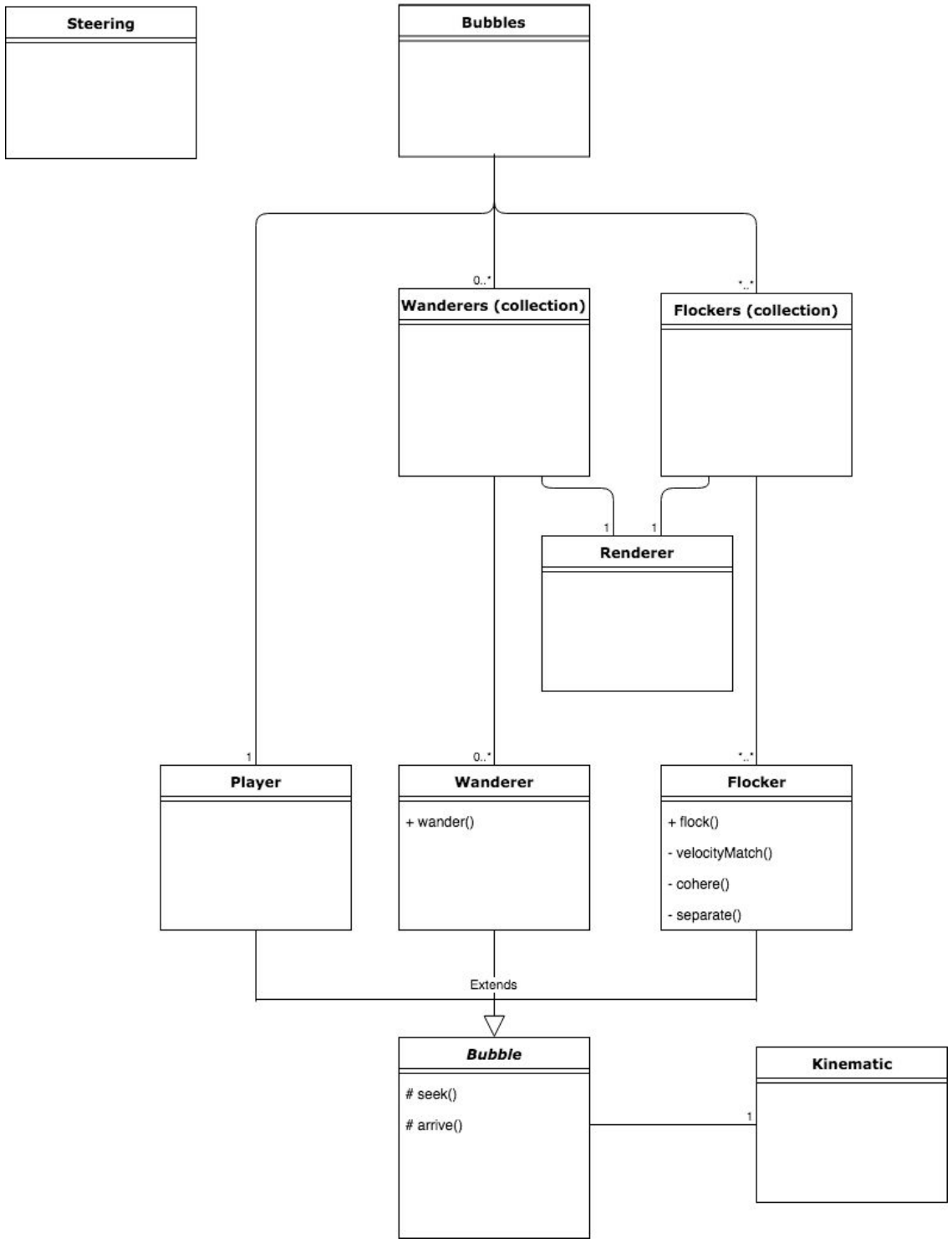
Please Click Anywhere To Replay

(game over screen)

Code Structure

Class Diagram

Below is a simple class diagram showing the structure of the code:



Please note not every method is in the diagram but only the steering behavior ones. And collection classes are indicated too.

Class Descriptions

- Bubble:
This is an abstract class containing some basic cosmetic properties, as well as basic steering methods (i.e. **seek** and **arrive**).

- Player:
This extends Bubble class. The player only has one steering behavior - **arrive** (at the position of the mouse).

- Wanderer:
This extends Bubble class. Wanderers only has one steering behavior - **wander**.

- Flocker:
This extends Bubble class. Flockers has 4 basic steering behaviors - **seek**, **cohere**, **separate** and **velocity matching**. Each of these behaviors is then weighted and combined to simulate **flocking**.

- Wanderers:
This is a collection class for wanderers. It contains some logic for updating wanderers.

- Flockers:
This is a collection class for flockers. It contains some logic for updating flockers.

- Kinematic:
This class contains kinematic information for bubbles.

- Renderer:
This class contains information and logic for rendering bubbles.

- Steering:
This class contains information of steering behavior results (i.e. the acceleration).

- Bubbles:
This class contains all non-game-life-cycle data and logic.

Running game & View source code

Running the game

(Assuming you have Java installed on your system)

- For Windows systems, navigate to “bubbles.windows32/” directory and double click the file named “main” to start the game.

- For Linux systems, use terminal to navigate to “bubbles.linux32/” directory and run “./main” command.

Source code

To view the code, go to “source/” directory. Files with “.pde” extension are the original code, which can be view with any text editor or IDE. Processing uses Java syntax so if you are using Notepad++, you can select “Java” as language to highlight keywords.

The “.java” file was generated from the original code.

Development Environment

- Operating system: Windows 10 64-bit
- Language: Processing
- IDE: Processing 2.2.1