Helium Hero

CIS 487 - 2D Game Pitch

Sean Croskey

Griffin McCormick

**1.0 Executive Summary**

**1.1 Game Background**  
A child bought a balloon from a balloon stand, and the balloon floated away from him. Game revolves around balloon’s adventures in the sky, where it avoids obstacles like crows, geese, acid-rain clouds, and planes. The obstacles in the sky are trying to pop the balloon because they do not like balloons.

**1.2 Character Background**

Balloon wanted to be purchased by a child and is smiling because he has finally been bought. However, the child loses him and he floats up into the sky. The balloon needs to evade obstacles in the sky for long enough to float back down to his new owner.

**1.3 Character Features**

* Balloon with a string
* Color is up to the player (Blue, Red, Green, Yellow)
* Durability (represented by number of lives)

**1.4 Setting**

Takes place in the skies above a balloon stand. White clouds and blue skies make up the background of gameplay scene.

**1.5 Situation**

The balloon is forced to survive against the evils of the sky. If the balloon is hit by one of its enemies it loses a life. If all lives are lost the balloon pops and the child is seen crying with the popped balloon. However, if the balloon makes it back to its owner both the balloon and the boy are happy.

**1.6 Objective/Goal**Player must maneuver the balloon to successfully avoid obstacles for the duration of the song in order to return the balloon to its owner.

**1.7 Possible Outcomes**

The player will either succeed or fail in fulfilling the game’s objective:

* Success Scenario: Player prevents the balloon from popping by making it to the end of the song.
* Failure Scenario: Player allows the balloon to pop.

**2.0 Gameplay Look and Feel**

**2.1 Player Roles and Actions**

Player controls the balloon and avoids the obstacles that are on the screen. Player uses the mouse to move the balloon around the screen.

**2.2 Design Characteristics**

2D sprites, cartoon-style. Balloon will be animated. Game elements will have either a static motion design or minimal animation.

**2.3 Music/Audio**

* Rhythm based, techno/EDM song (will crop out portions of the song)
  + - DJ Splash - This Is My Life
* Audio clips for various game elements (found online)
  + - Crow (raven noise)
    - Goose noise
    - Plane noise
    - Balloon getting hit by an obstacle (thud noise)
    - Player collecting helium pack (click noise)
    - Player collecting raindrops (raindrop noise)

**2.4 Mechanics**  
 Movement is based on mouse position. The balloon will follow the mouse movement. Avoid the negative-effect objects and collect the positive-effect objects. Difficulty will increase as you make it further into the song.

Some obstacles will target balloon based off of mouse position. Other obstacles will randomly spawn and move across the screen. If the balloon is hit by an obstacle, there will be a ~2 second cooldown where they are invincible from being hit by another obstacle. The balloon will have a fluctuating opacity during this time.

Player will start with three lives. Lives are decreased by one for each obstacle hit. The player gains lives by acquiring helium packs, and there is no upper limit on number of lives. If the player makes it to the end of the song, then the game ends with the player winning. If the player gets hit by an obstacle while at only one life, however, then the balloon pops and the game ends with the player losing.

**2.5 Game Elements**

* Crows: shot out of flying cannon that targets the balloon (mouse position)
* Geese: moves onto screen slowly from the left side at a random y coordinate. (do not target balloon afterwards)
* Planes: Cover half the playing area (top/bottom). Indicator before they emerge from edge of screen (hard-coded to always follow same pattern)
* Acid-rain clouds: form on screen and damage balloon. When the clouds are getting larger, the player cannot be damaged. When the clouds start to get smaller, they turn green and can damage the player only if they come into contact with one after the transition. Clouds fade in/out entirely in 2 seconds.
  + 1 second fade in, 1 second fade out where it can damage the balloon
* Helium Pack: Spawns in random position on screen for limited amount of time
* Raindrops: Falls from top of screen and increase player’s score. Spawns randomly on the top of the screen every time that another game object is instantiated.

**2.6 Gameplay layout:**

* Song starts (7 seconds)
* First segment: crows (7 seconds)
* Second segment: crows and geese(7 seconds)
* Third segment: geese (10 seconds)
* Fourth segment: planes (10 seconds)
* Fifth segment: planes, crows (10 seconds)
* Sixth segment: acid-rain clouds (10 seconds)
* Seventh segment: planes, crows, geese, acid-rain clouds (10 seconds)
* Song ends (7 seconds)

**2.7 Scenes:**

* Title
  + Instruction Screen (good/bad items and control description)
  + Color Selection(possible difficulty selector)
  + Quit - Quits game
  + Play- moves to intro cutscene
  + High score - shows the high score
* Story Intro Cutscene
  + Describes the story of why the balloon is in the sky.
* Gameplay
  + Where the game is located
* You Win Cutscene
  + High score - shows the high score
  + Score - shows the score from last gameplay
  + Replay- starts gameplay again
  + Main menu - returns to title scene
* You Lose Cutscene
  + High score - shows the high score
  + Score - shows the score from last gameplay
  + Replay- starts gameplay again
  + Main menu - returns to title scene

**2.8 Strategies and motivations**

Motivations: Player starts out with 3 lives-- if all lives are lost, the the game is over and the boy does not get the balloon back. There is motivation to avoid objects to prevent balloon from popping so that it can return to its owner. Strategies revolve around avoiding the obstacles. Each obstacle will have a hitbox, so the strategy will be understanding each obstacle’s path on the screen and avoiding them accordingly.

**2.9 Level summary/story progression**

The level consists of a screen with a sky backdrop and the balloon starting on the center of the screen. As the song progresses, obstacles will try and pop the balloon. The further into the song the player survives the more difficult the obstacles will become. Once the song ends, the player wins. If the balloon pops at any time, the player loses.

If the balloon gets hit a certain amount of times, then it will pop and fall to the earth leaving the boy sad.

**3.0 Development Specification**

**3.1 Hardware (per Unity)**

Generally, content developed with Unity can run on any modern system. This game will be runnable only on a desktop or laptop with Windows. Here are more in-depth system requirements:

* OS: Windows XP SP2+
* Graphics card: DX9 (shader model 3.0) or DX11 with feature level 9.3 capabilities.
* CPU: SSE2 instruction set support.

**3.2 Software**

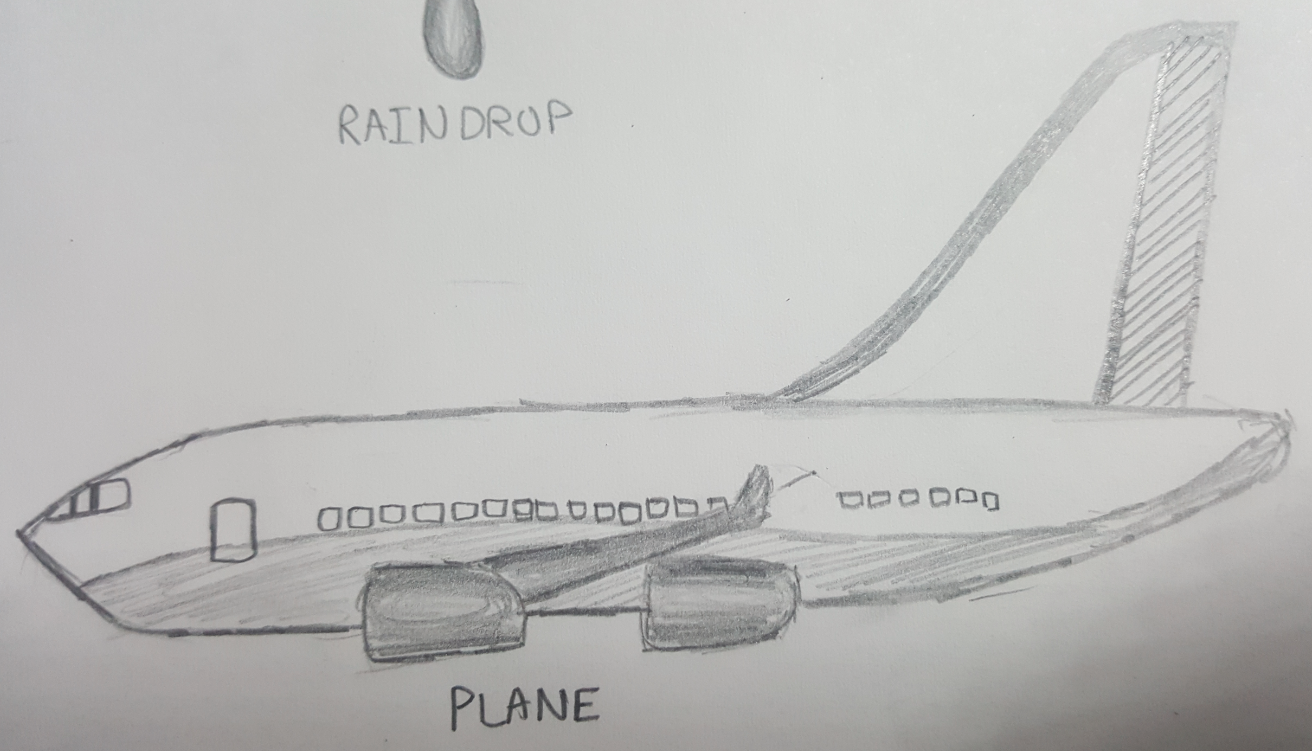
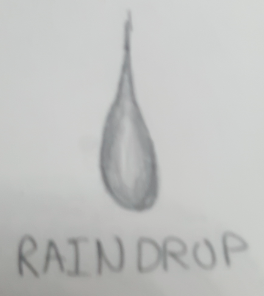
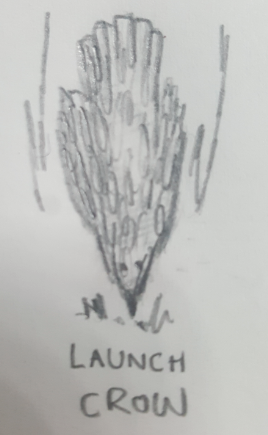
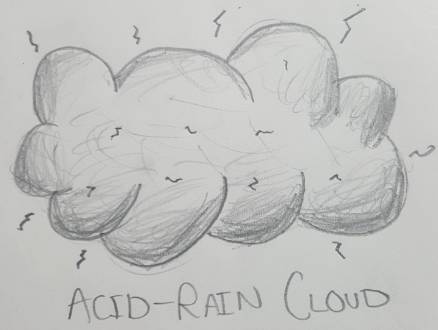
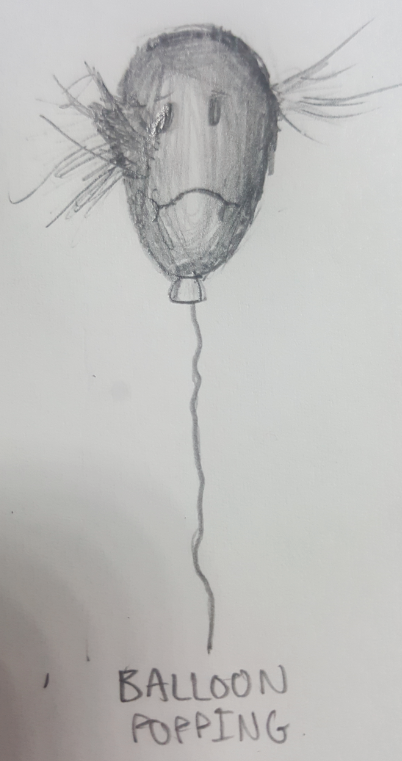
The game will be runnable through Unity, or as a windows x86 application. The game will be runnable only at 4:3 screen ratios. This is because the game plays better in more of a square screen. The recommended resolution is 800x600 windowed mode.

**3.3 Algorithm Style**  
Since the game is synced to the beat of the song, the times that each of the obstacles appear on the screen need to be hardcoded into the game. Each obstacle will have a array of times that correspond to an obstacle spawning on the screen. With that in mind, there will need to be a loop that runs the entire length of the song that checks if certain timestamps have passed. If they have, then the obstacle is spawned.

So, the algorithm style will be iterative instead of recursive. It will rely on a loop throughout the length of the song that calls functions to spawn game elements.

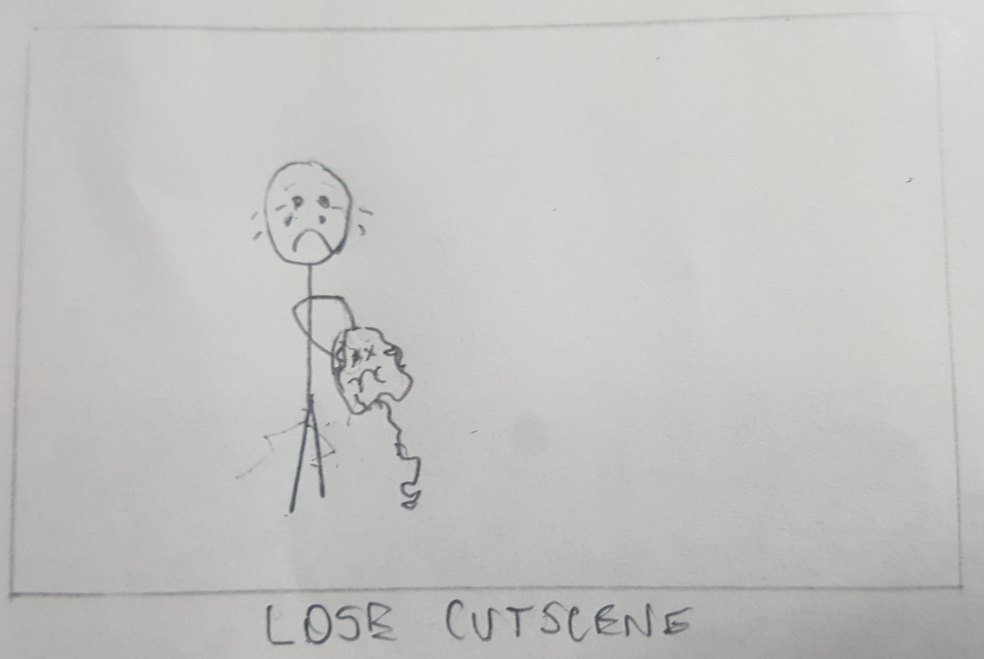
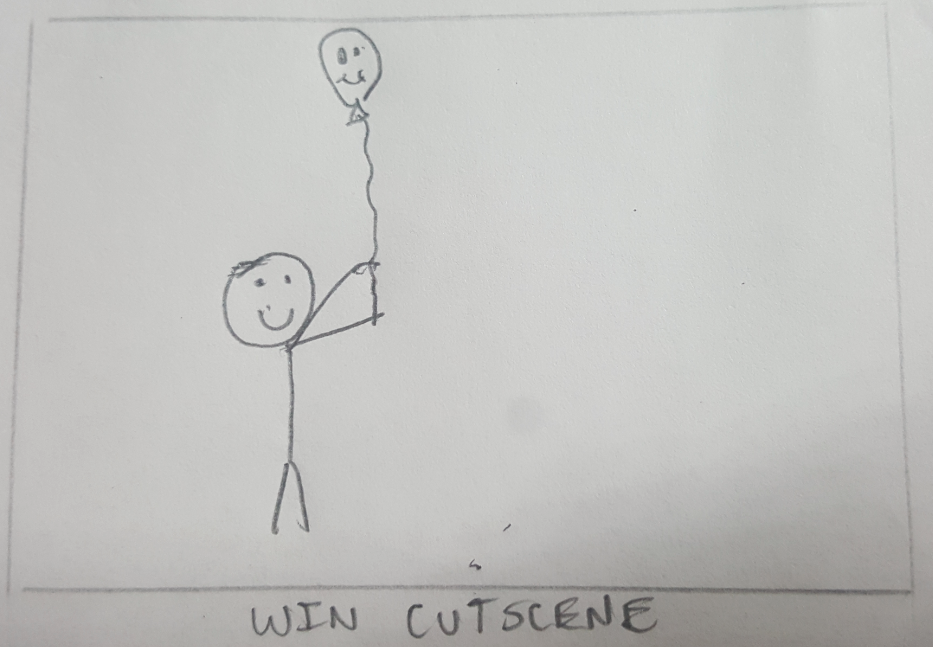
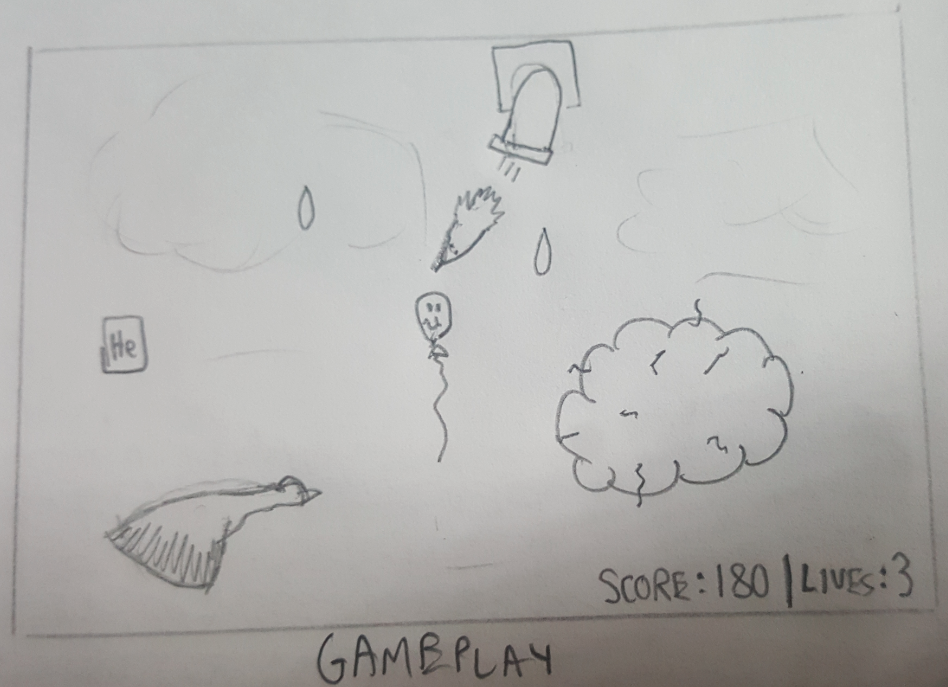
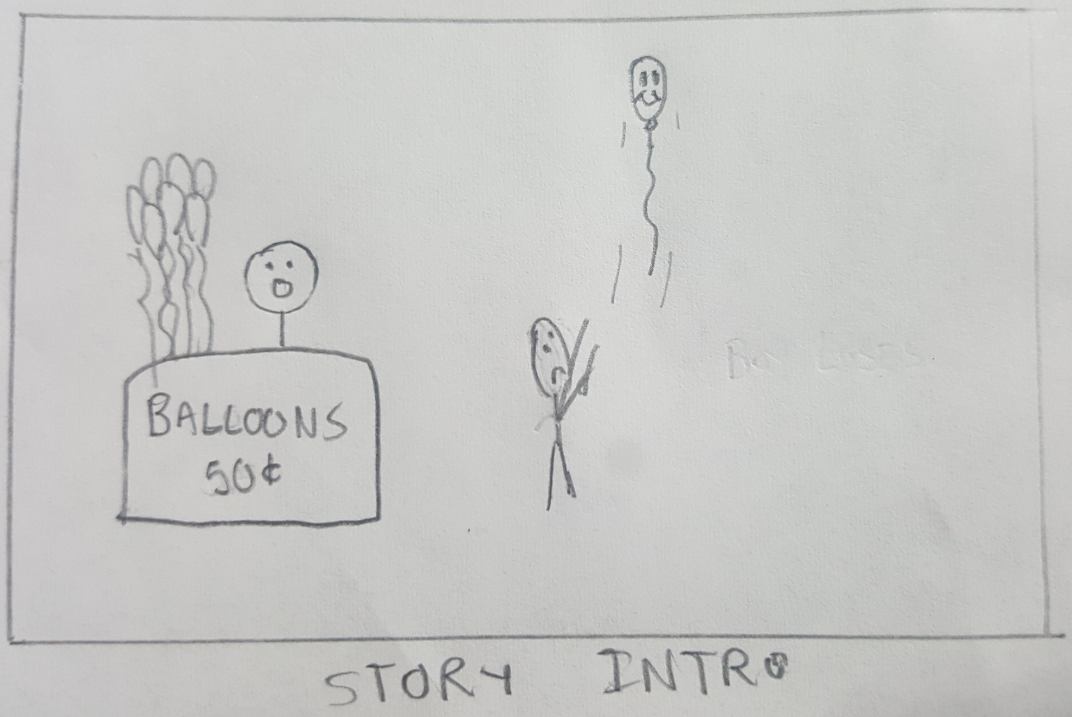
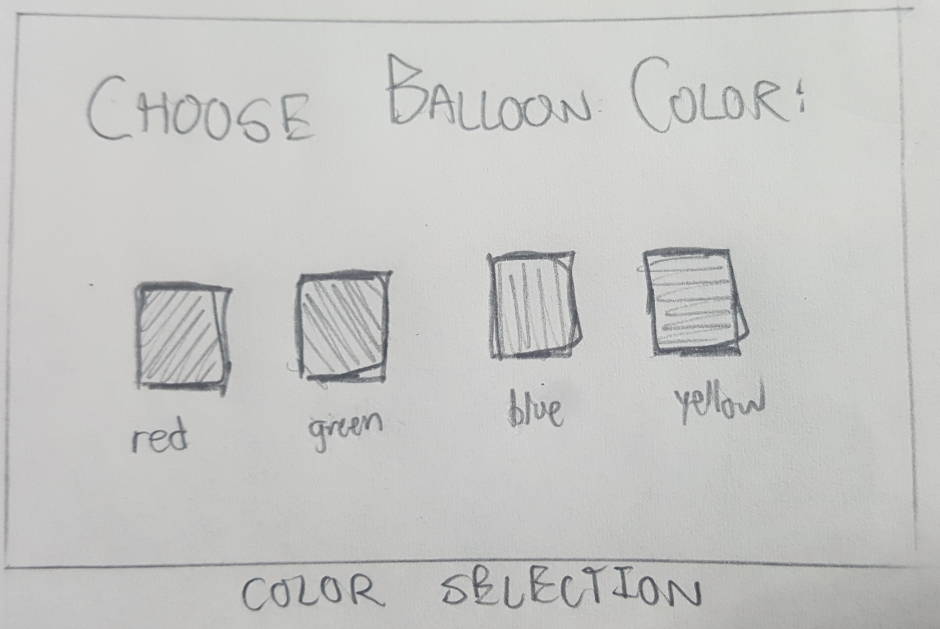
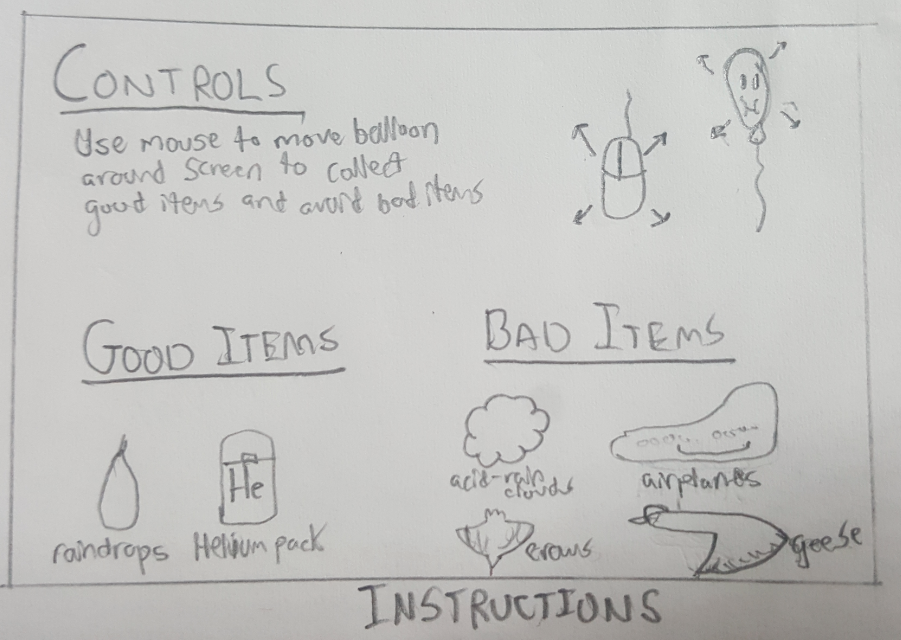
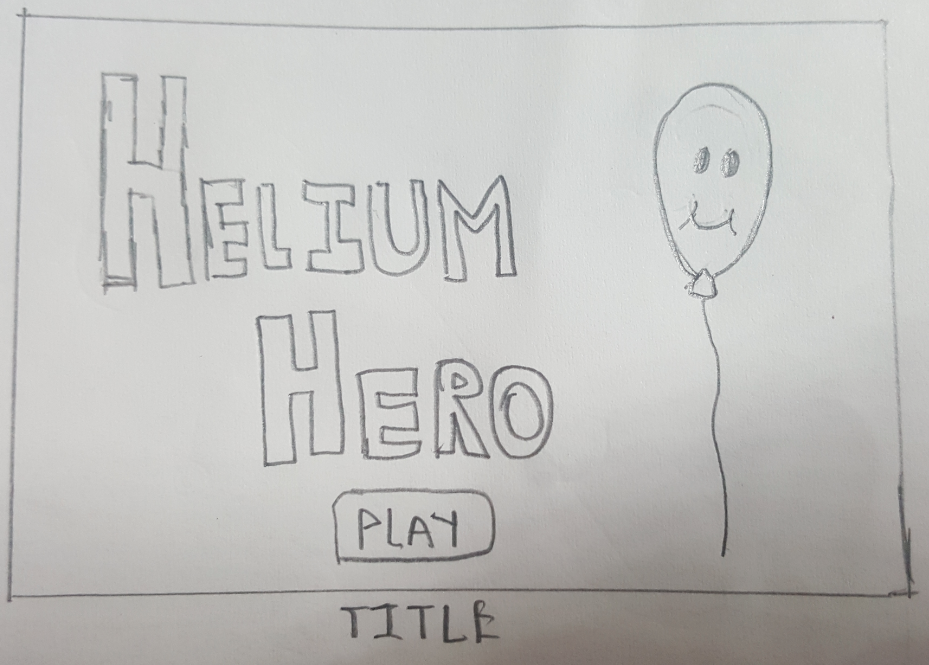
**4.0 Concept Art and Storyboard**

**4.1 Concept Art**



Note: Art will have color in the actual game. Each of the birds will have multiple sprites to animate their flying. Balloon will also have an string made of hinge joints.

**4.2 Storyboard**



**5.0 Appendix**

**5.1 Sources**

* <http://armorgames.com/play/60/streamline>
  + Game layout inspired by this game
* <https://unity3d.com/unity/system-requirements>
  + System requirements for Unity