Sky Wan

IGME-330 (Section 01)

March 26, 2018

Project link: <https://people.rit.edu/sxw8136/330/Smol-Pet-Sim/code/>

GitHub link: <https://github.com/peachhichew/Smol-Pet-Sim>

**Changes made from original design:**

* Changed code from ES5 to ES6 classes and modules.
* Increased the size and improved the quality of cat and dog sprites.
* Changed the Resize button on the Rabbit page to have more subtle size changes.
* Implemented click and drag behavior on Rabbit page.
* Implemented keyboard controls (arrow keys) for the Cat page.
* Added fish into the Cat page.
* Added background music and sound effects for each page.
* Used HowlerJS for sound effects and music.

**Meeting requirements:**

* Media requirements:
  + Sound: Added sound effects and background music (loops throughout the different pages).
  + Images: Consistent theme of style and picked my own graphics.
  + Fonts: Appropriate font chosen and is easy to read.
  + Canvas drawing/animation: Image drawn on Pixi canvas and is animated using PixiJS. They are also smoothed using deltaTime.
* Interaction requirements:
  + Controls: Allows the user to control the sprites/animals using mouse clicks or keyboard controls. One of the pages also includes buttons to alter the appearance of the sprite and alter what is happening on the screen.
* Usability requirements:
  + Window.onblur and window.onfocus are implemented.
  + Instruction page exists.
  + Mouse and onscreen UI buttons are more obvius/preferred over keyboard.
* Experience/game design requirements:
  + Meeting your plan: Met most of the “other” section in the Game Treatment, with the exception of some sound effects, cat poses, and turning buttons into sprites.
  + Experience specific requirements: Experience allows the user to interaction with three types of animals through mouse clicks and keyboard controls. The experience has a pixel art, 8-bit game sort of style. Hopefully the user will think the simulator is cute since they can interact with animals.
* Coding requirements:
  + Meeting all requirements of “Tech Stack” chosen.
  + Using a pre-approved JS library (PixiJS and HowlerJS).
  + Preloading images in separate files.
  + Uses module pattern.
  + Has 2 new function constructors created by me.
* Above and beyond:
  + Learning to use HowlerJS.
  + Learning to implement keyboard controls in Pixi.
  + Learning to improve the sprite sheet quality (since some of them were pixelated when you scaled up the size).
  + Learning to implement dragging behavior in Pixi.
  + Implementing collision detection in the Cat page.
  + Rewrote the Resize function, so that the resizing was more gradual and less abrupt.
  + Taking the time to find suitable music and sound effects.
  + Making the cat face the proper direction when moving (e.g. facing up when the UP arrow key is pressed).
  + Added Instructions page instead of keeping the Notes page.

**What went right:**

* Was able to learn HowlerJS fairly easily and implement sound effects.
* Going from ES5 to ES6 with classes and modules.
* Improving the quality of the sprites.
* Adding collision detection, dragging, and keyboard controls.
* See above and beyond for rest.

**What went wrong:**

* Wanted to include munching sounds for when cat ate the fish, but wasn’t sure how to do it without the sound playing every 60 seconds.
* Was not able to stop the barking sound whenever the barking dogs were in the sitting pose.

**Features I would have liked to add:**

* Everything that went wrong.
* Adding a counter for the fish collected in the Cat page, or maybe making a mini Snake game out of it.
* Potentially changing the Dog page so that you would only have one dog and the animations would be randomized every couple seconds. Possibly changing it so that you could feed the dog and maybe even have a love/affection meter.

**Resources used:**

* Sprites:
  + Fish: <http://i.imgur.com/KjDDO.gif>
  + Cat: <https://biofunk95.deviantart.com/art/Cat-Sprites-541217734>
  + Dog: <https://www.spriters-resource.com/fullview/82035/>
  + Rabbit: <http://rebloggy.com/post/cute-animal-bunny-rabbit-pixel-art-transparent-pixl/67708885863>
* Sound effects:
  + Background music: <https://opengameart.org/content/jump-and-run-8-bit>
  + Rabbit jump: <https://www.zapsplat.com/music/cartoon-spring-bounce-or-jump-8-2/>
  + Cat meow: <https://www.zapsplat.com/music/cat-meow-hungry/>
  + Dog bark: <https://www.zapsplat.com/music/dog-bark-springer-spaniel-3/>
  + Rabbit thump: <https://www.zapsplat.com/music/feet-jump-land-on-carpeted-wooden-floor-hard-2/>
  + Button clicks: <https://www.zapsplat.com/music/plastic-switch-toggle-2/>
* Coding references:
  + <https://github.com/kittykatattack/learningPixi>
  + Circle Blast

**Grading:** 95%, I definitely put in a lot more time and effort for this, especially when I was confused about how to change my code to the inheritance/ES6 classes and modules. I hit all the basic requirements and went in for office hours to resolve any issues I couldn’t figure out myself. I also tried to change two of the pages or implement more changes throughout so I could improve what I had originally.