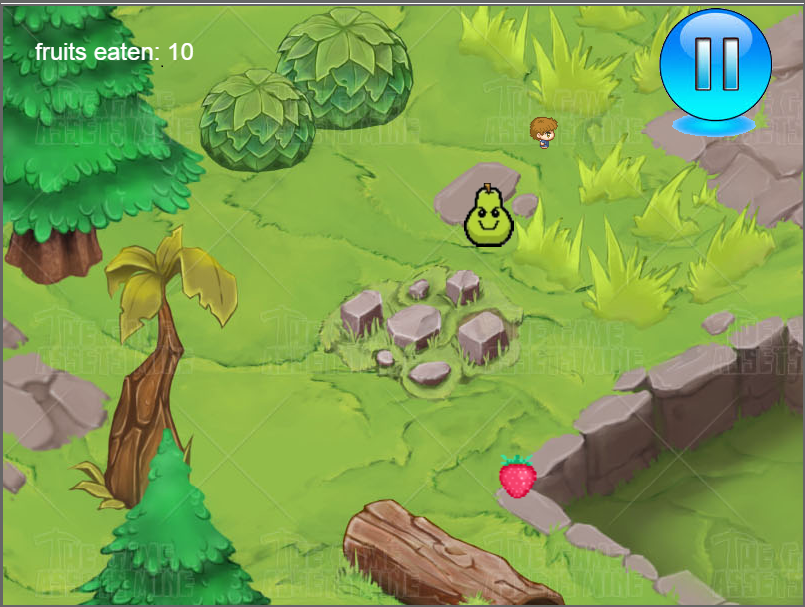
Fruit Chase



Robert Hansen

Barkha Mathur

Fruit chase is a fun game which lets you eat and get eaten by fruits. The game is light weighted and totally web based. It altogether uses different concepts of HTML and javascript .

Table of Contents

Overview ... 2

New and Complex ... 3

Blooms Taxonomy ... 4

Overview

Fruit chase is a lightweight application where users try to eat as many strawberries as they can while avoiding the evil pears. The user is first introduced to the main welcome screen where he has a choice to start a new game or view high score. The application use the server provided to keep track of ten highest scores. The application is set up so that a pear appears after every ten strawberries are collected. The pears are programmed to detect collisions with each other and try to move out of each other’s way while chasing the player.

The application also supports audio and has an option for the user to pause it if they like. Also, the main character is built from four different sprites to give a realistic effect. The sprites change according to the direction the character moves. On touching an evil pear, the game is over and the user is brought back to the main screen. The user is also able to see their session’s high score.

The main objective behind this project was to explore and exploit new features of HTML and JAVAScript and put them together in one application. The application use some complex concepts like using canvas object, and an audio object.

The same canvas is used for welcome screen, high score display, and the game itself, thus decreasing redundancy and increasing the code efficiency.

New and Complex

* The use of Audio object :

Before landing on our final choice we tried a bunch of different options for our background music. We learned how different situation demands different solutions. Our simplest option was to add an audio object in javascript like this:

var music = new Audio();

music.play();

* Continually drawing images

We had to learn how to continuously draw the images in the correct order and at the right time on the canvas.

* Javascript Objects:

We learned how to create objects in javascript to hold the positions/speed of the in-game characters.

* Event Listeners within a canvas:

In order to handle clicks on specific areas of the the canvas, we used event listeners that interacted with the game’s logic to start/quit the game and pause/play the audio. Listeners also checked for keypresses to control the player’s character.

* Game logic:

Much of the complexity for the project comes from the logic behind the fruit placement, collision detection between the hero and pears, as well as pears with each other, then deciding what to do after the collision is detected. The answer was to create the character objects and check every update whether the objects’ coordinates were close enough to be colliding.

**BLOOM'S TAXONOMY**

**Synthesis:**

For this project, we assembled some of the ideas learned from lab two as well as a few new javascript techniques and previous programming knowledge to create a functional game powered by javascript. Designing a game was a fun way to incorporate many different concepts we have learned through class and challenge ourselves to develop something that would require us to learn some new concepts to accommodate for features that we had not learned yet. For example, we took what we knew about html canvases and drawing images on them, learned how to use event listeners to receive inputs for keypresses and clicks, and learned how to store data in objects for each character we wanted in the game. With all of these concepts combined, we were able to move characters around on the screen based on their position stored in their object, with the player’s character manipulated by the arrow keys.

**Evaluation:**

Given the information learned from lecture and the first two labs, we judged that the best way to combine everything we had learned so far was to create a game. A game allowed us to creatively apply what we are learning in a way that is engaging for us and is somewhat familiar for us already, even if the language and medium is unfamiliar.

**Analysis:**

While learning new concepts for this project, we had to decide between ways to display and code different pieces of the program. For example, for the audio, we could have created an audio element in the html code on the page, but decided instead to just create an audio object in the javascript code because we were able to have more freedom and access to the audio in order to code the play/pause functionality.