Министерство образования и науки Российской Федерации

ФГБОУ ВО Ульяновский государственный технический университет

Кафедра «Вычислительная техника»

Лабораторная работа №5

«Игра»

Выполнила студентка

группы ИВТАСбд-11

Игнатьева Д.С.

Проверил:

Игонин А.Г.

Ульяновск, 2022

**Оглавление**

1. Цель работы………...…………….………………………………………...3
2. Описание метода выполнения……………………………….…………....4
3. Интерфейс…………………………………………………………………..5
4. Листинг кода программы……………………………………………………………….…6
5. Вывод…………………………………….………………………………...11
6. Список литературы…………………….…...……………………………...............12

**Цель работы**

Разработать на HTML, CSS, JavaScript, +- PHP игру. Игра – это пародия на известную игру «Flappy Bird», только упрощенная ее версия.

**Краткое описание метода выполнения**

Для начала мы создаем новые PHP, CSS, JS файлы, которые понадобятся при работе OpenServer в папке domains.

Суть игры состоит в том, чтобы летящий шарик прошел все блоки, не врезавшись в них и не коснувшись края линии.

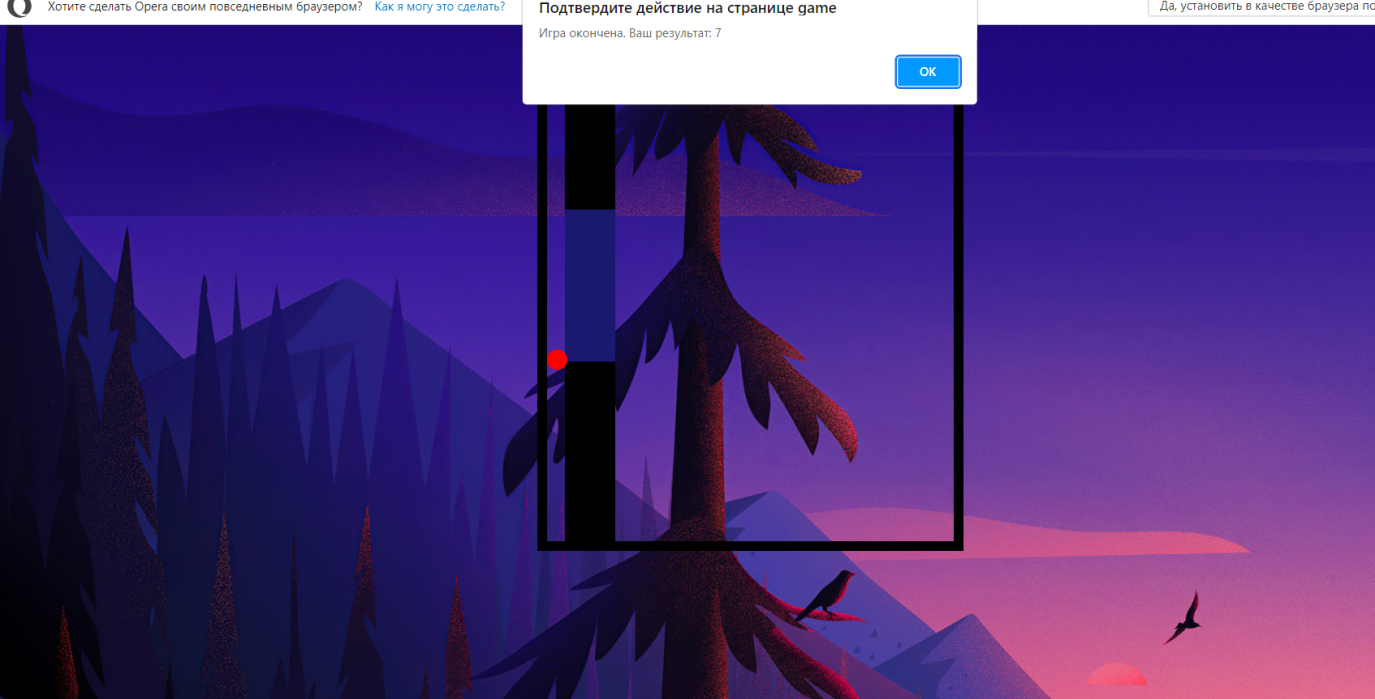
**Интерфейс**

Игра заключена в прямоугольник, и летающий шарик должен, избегая препятствия в виде черных блоков, пройти в отверстие. (Рис.1)

Рис.1.Игра

При проигрыше высвечивается окошко с надписью «Игра окончена» и так же высвечивается ваш результат, то есть сколько блоков удалось пройти.(Рис.2)

Рис.2.Результаты игры

****

**Листинг кода программы**

Сначала создаем файл index.php с разметкой.

*<!DOCTYPE html>*

*<html lang="en" onclick="jump()">*

*<head>*

*<meta charset="UTF-8">*

*<title>Ssendem</title>*

*<link rel="stylesheet" href="css/style.css">*

*</head>*

*<body>*

*<style>*

*body{*

*background: url("image/1.jpg");*

*}*

*</style>*

*<div id="game">*

*<div id="block"></div>*

*<div id="hole"></div>*

*<div id="character"></div>*

*</div>*

*</body>*

*<script src="js/game.js"></script>*

*</html>*

**С помощью CSS мы задаем свойства**

*\*{*

*padding: 0;*

*margin: 0;*

*}*

*#game{ //Задаем позиции полю*

*width: 400px;*

*height: 500px;*

*border: 10px solid black;*

*margin: auto;*

*overflow: hidden;*

*}*

*#block{*

*width: 50px;*

*height: 500px;*

*background-color: black;*

*position: relative;*

*left: 400px;*

*animation: block 2s infinite linear; //анимация бесконечного перемещающегося влево блока*

*}*

*@keyframes block{ //* *устанавливаем ключевые кадры при анимации элемента*.

*0%{left:400px}*

*100%{left:-50px}*

*}*

*#hole{*

*width: 50px;*

*height: 150px;*

*background-color: midnightblue;*

*position: relative;*

*left: 400px;*

*top: -500px;*

*animation: block 2s infinite linear; //анимация бесконечного перемещающегося влево блока*

*}*

*#character{*

*width: 20px;*

*height: 20px;*

*background-color: red;*

*position: absolute;*

*top: 100px;*

*border-radius: 50%;*

*}*

С помощью JS задаем рандомное положение препятствиям в виде черных блоков:

*hole.addEventListener('animationiteration', () => {*

*var random = -((Math.random()\*300)+150);[[1]](#footnote-1)*

*hole.style.top = random + "px";*

*counter++;*

*});*

Метод **setInterval** позволяет вызывать функцию **регулярно**, повторяя вызов через определённый интервал времени.

*setInterval(function(){*

*var characterTop = parseInt(window.getComputedStyle(character).getPropertyValue("top"));*

Для того чтобы мячик смог прыгать не нужно продолжать прыгать постоянно, нужно чтобы мяч немного подпрыгивал, а затем опускался

*function jump(){*

*jumping = 1;*

*let jumpCount = 0;*

*var jumpInterval = setInterval(function(){*

*var characterTop = parseInt[[2]](#footnote-2)(window.getComputedStyle(character).getPropertyValue("top"));*

*if((characterTop>6)&&(jumpCount<15)){*

*character.style.top = (characterTop-5)+"px";//Мячик прыгает выше гравитации, поэтому значение ставится отрицательное*

*}*

*if(jumpCount>20){//ограничиваем количество интервалов*

*clearInterval(jumpInterval);*

*jumping=0;*

*jumpCount=0;*

*}*

*jumpCount++;*

*},10);*

*}*

Чтобы мячик мог видеть блоки и, соответственно, врезаться в них, создается условие if

*if((characterTop>480)||((blockLeft<20)&&(blockLeft>-50)&&((cTop<holeTop)||(cTop>holeTop+130)))){*

*alert("Game over. Score: "+(counter-1));*

*character.style.top = 100 + "px";*

*counter=0;*

*}*

**Вывод**

Цель работы была выполнена. В ходе лабораторной работы была написана и разработана игра. Игра протестирована и работает без помех. Удалось узнать новые функции, которые понадобились для разработки небольшой игры.

**Список литературы**

1. <https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Global_Objects/parseInt>
2. <https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Global_Objects/Math/random>

1. <https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Global_Objects/Math/random> [↑](#footnote-ref-1)
2. https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Global\_Objects/parseInt [↑](#footnote-ref-2)