Санкт-Петербургский политехнический университет Петра Великого

Институт прикладно математики и механики

**Кафедра «Теоретическая механика»**

**КУРСОВОЙ ПРОЕКТ**

**Реализация и визуализация приложения «Flappy Bird»**

по дисциплине «Основы программирования и алгоритмизации»

Выполнил   
студент гр.3630103/00001 Ямпуров Лаврентий

Руководитель

«30» апреля 2021 г.

Санкт-Петербург

2021

**СОДЕРЖАНИЕ**

Введение………………………………………………………………3

Постановка задачи……………………………………………………3

Описание кода HTML………………………………………………...4

Описание кода JavaScript…………………………………………….5

Результаты…………………………………………………………...9

Вывод………………………………………………………………...10

Введение

# Актуальность темы.

Данное приложение (программа) обусловлено потребностью общества в отдыхе от ежедневной рутины с помощью мобильных устройств. Данная тема позволяет нам понять основные принципы создания простейших программ на языке программирования JavaScript.

# Объект исследования. Процесс использования языка программирования JavaScript для создания простого приложения.

# Предмет исследования. Разнообразный инструментарий языков JavaScript и HTML для создания простого приложения.

Постановка задачи

Главной задачей было создание программы «Paint».

Основными пунктами реализации приложения стали:

* канвас ;
* вид игры (2d);
* вставка объектов, которые используются в данном приложении(блоке , задний фон, нижний фон(“земля”), птичка);
* вставка аудио звука , который будет возникать , как только счет будет увеличиваться на единицу;
* позиция птички и ее передвижение в рамках игры ;
* счет в игре

# Технологии, использованные для написания приложения:

При написании «Flappy Bird» были использованы стандартные библиотеки и возможности языков программирования JavaScript, HTML.

Описание кода HTML

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<title>Flappy Bird!</title>

</head>

<body>

<canvas id="canvas" width="288" height="512"></canvas>

<script src="Flap.js"></script>

</body>

</html>

Описание кода JavaScript

//создаем канвас

var cvs = document.getElementById("canvas");

//вид игры:

var ctx = cvs.getContext("2d");

//создаем объекты

var bird = new Image();

var bg = new Image();

var fg = new Image();

var pipeUp = new Image();

var pipeBottom = new Image();

bird.src = "flappy\_bird\_bird.png";

bg.src = "flappy\_bird\_bg.png";

fg.src = "flappy\_bird\_fg.png";

pipeUp.src = "flappy\_bird\_pipeUp.png";

pipeBottom.src = "flappy\_bird\_pipeBottom.png";

// Звуковые файлы

//var fly = new Audio();

var score\_audio = new Audio();

//fly.src = "js\_game\_audio/fly.mp3";

score\_audio.src = "js\_game\_audio/score.mp3";

var gap = 90;

// При нажатии на какую-либо кнопку

document.addEventListener("keydown", moveUp); //отслеживаем какое-либо дейтвие

// за что отвечает функция moveup

function moveUp() {

yPos -= 26; // поднимает на 25 единиц выше

//fly.play();

}

// Создание блоков

var pipe = []; //создаем массив пустой

pipe[0] = {

x : cvs.width, //за экраном находится

y : 0

}

var score = 0;

// Позиция птицы

var xPos = 10;

var yPos = 150;

var grav = 1;

//рисуем объекты в канвасе

function draw() {

ctx.drawImage(bg, 0, 0); //рисуем картинку

//i идет от 0 до количесва эллементов в массиве

for(var i = 0; i < pipe.length; i++) {

//используем координаты из массива

ctx.drawImage(pipeUp, pipe[i].x, pipe[i].y);

ctx.drawImage(pipeBottom, pipe[i].x, pipe[i].y + pipeUp.height + gap);

pipe[i].x--;

// новые блоки - постоянное их создание + добавляем новый эллемент

if(pipe[i].x == 125) {

pipe.push({

x : cvs.width,//ширина

y : Math.floor(Math.random() \* pipeUp.height) - pipeUp.height //случайное число (случайное место появления блоков)

});

}

// Отслеживание прикосновений (столкновений)

//с блоками и землей

if(xPos + bird.width >= pipe[i].x

&& xPos <= pipe[i].x + pipeUp.width

&& (yPos <= pipe[i].y + pipeUp.height

|| yPos + bird.height >= pipe[i].y + pipeUp.height + gap) || yPos + bird.height >= cvs.height - fg.height) {

location.reload(); // Перезагрузка страницы

}

//количество очков

if(pipe[i].x == 5) {

score++;

score\_audio.play();

}

}

ctx.drawImage(fg, 0, cvs.height - fg.height);

ctx.drawImage(bird, xPos, yPos);

yPos += grav; //увелич на 1.5

ctx.fillStyle = "#red";

ctx.font = "27px Verdana";

ctx.fillText("Счет: " + score, 10, cvs.height - 20);

requestAnimationFrame(draw); //постоянно метод ctx.drawImage

}

pipeBottom.onload = draw;

//используется чтобы все изображения были загружены и только поcле мы видели риунок

Результаты

В результате получаем игру “Flappy Bird” которая удовлетворяет поставленным задачам.

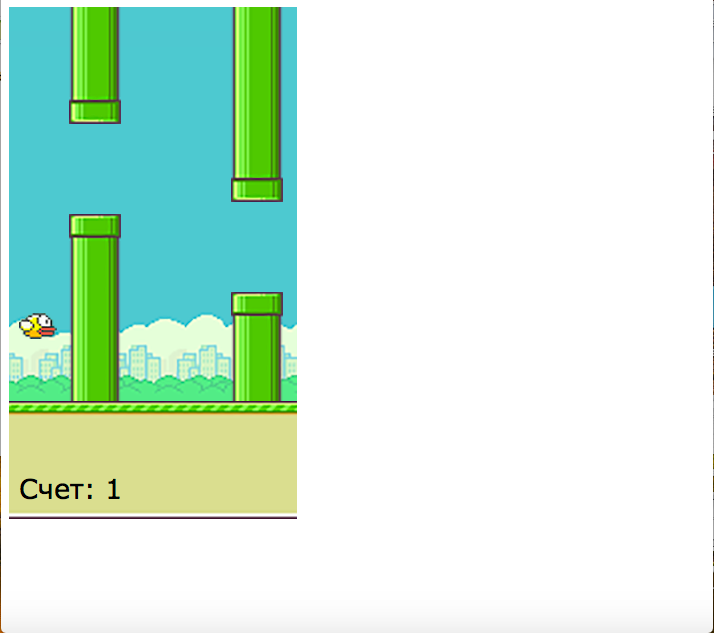


Рис.1 Внешний вид Flappy Bird

Вывод

Подводя итоги курсовой работы, мы можем сказать, что поставленная задача выполнена, а также достигнуты сопутствующие ей цели. В ходе выполнения данного проекта был получен ценный опыт работы с необходимым инструментарием языка JavaScript.

В заключение, можно упомянуть о возможности доработки данной программы с помощью добавления различных инструментов. Например, можно жобавитьбонусы , которые будут периодически появляться и давать “баффы” птичке , например возможность воскрешения.