

## Разработка игры T-Rex с помощью языка программирования JavaScript

*Стрельцова Марина Николаевна*

*Приамурский государственный университет им. Шолом-Алейхема*

*Студент*

### Аннотация

На данный момент существует огромное количество категорий игр, которые удовлетворяют различные интересы людей. Развлекательные, обучающие, познавательные – это далеко не весь список популярных жанров игровой индустрии. Но написание самих игр далеко не простая задача, которая может требовать усилия целых команд, состоящих из сотен людей. В данной статье описывается создание игры T-Rex с помощью языка программирования JavaScript.

**Ключевые слова:** игра, программирование, JavaScript, анимация.

## Development of the T-Rex game using the JavaScript programming language

*Streltsova Marina Nikolaevna*

*Sholom-Aleichem Priamursky State University*

*Student*

### Abstract

At the moment, there are a huge number of categories of games that satisfy different interests of people. Entertaining, educational, informative – this is not the whole list of popular genres of the gaming industry. But writing the games themselves is far from an easy task, which may require the efforts of entire teams consisting of hundreds of people. This article describes how to create a T-Rex game using the JavaScript programming language.

**Keywords:** game, programming, JavaScript, animation.

## 1. Введение

### 1.1 Актуальность исследования

На данный момент существует огромное количество категорий игр, которые удовлетворяют различные интересы людей. Развлекательные, обучающие, познавательные – это далеко не весь список популярных жанров игровой индустрии. Но написание самих игр далеко не простая задача, которая может требовать усилия целых команд, состоящих из сотен людей. В данной статье описывается создание игры T-Rex с помощью языка программирования JavaScript.

## 1.2 Обзор исследований

В своем исследовании Е. В. Соловьева и Т. А. Максимова описывают разработку развивающей компьютерной игры-головоломки с помощью функционального языка JavaScript [1]. П.В. Никитин и его соавторы в своей разработанной программе для ЭВМ представляют набор модулей, позволяющий пользователям подготовиться к соревнованиям по стандартам WorldSkills по компетенции «Веб-дизайн и разработка» по модулю «Разработка браузерной игры» [2]. В своей статье Е.Н. Поварничен рассматривает создание игры «Тетрис» на языке JavaScript, HTML и таблицей стилей CSS [3]. Nabhani S. и соавторы рассматривают сетевой прототип игры «Pharmacy Challenge» на студентах-фармацевтах для проведения исследования на оценку игры с точки зрения дизайна и оценивают влияние на успеваемость и уверенность учащихся [4]. N. Taketoshi, K. Yasushi предоставляют в своей научной работе механизм под названием «join token» в котором состояния игровых персонажей могут быть выражены в виде токенов, а взаимодействия могут быть описаны как обработчики, включающие несколько токенов. Для целей оценки они разработали язык сценариев игр под названием “Mogemoge” и написали несколько примеров игр на этом языке [5]. В работе Н. А. Базеевой и Д. С. Лебедева проведено исследование начала создания игровой индустрии, особенностей различных языков программирования, с целью выявления самого востребованного в разработке игр на различные существующие платформы [6].

## 1.3 Цель исследования

Целью данного исследования является разработка игры T-Rex с помощью языка программирования JavaScript.

## 2. Методы исследования

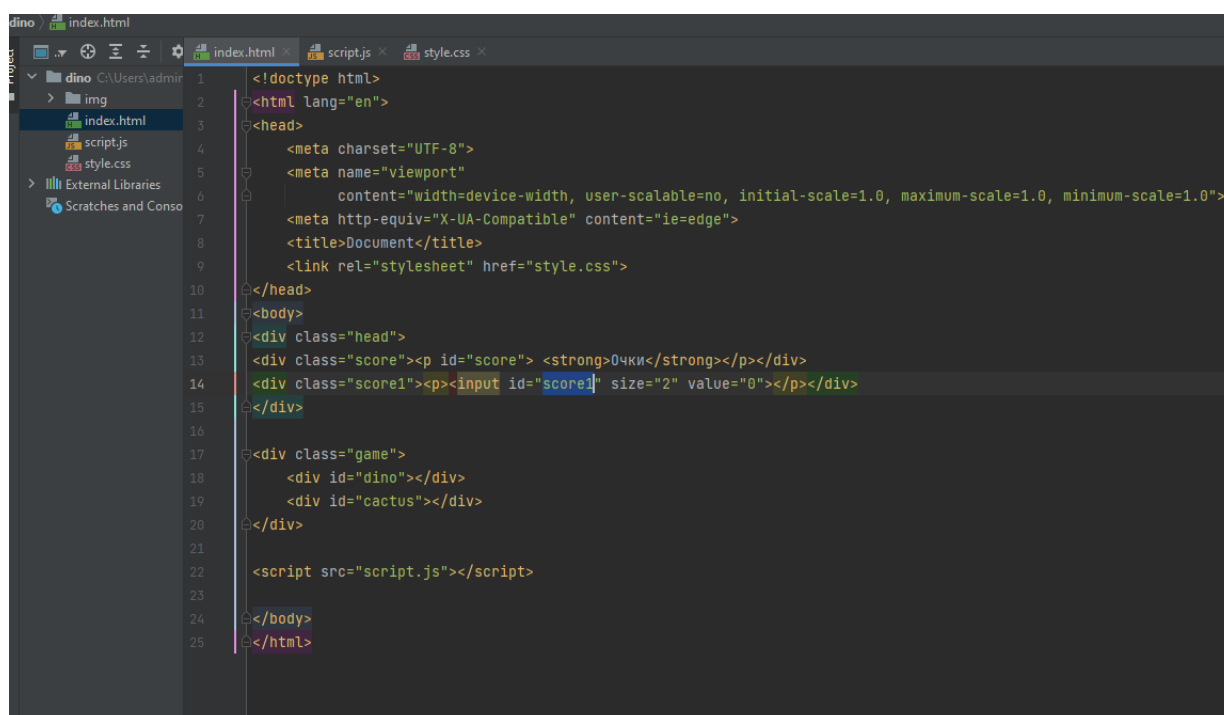
Для реализации игры T-Rex были выбраны следующие программные средства и языки:

- JavaScript - мультипарадигменный язык программирования. Поддерживает объектно-ориентированный, императивный и функциональный стили. Является реализацией спецификации ECMAScript [7].
- CSS - формальный язык описания внешнего вида документа (веб-страницы), написанного с использованием языка разметки (чаще всего HTML или XHTML). Также может применяться к любым XML-документам, например, к SVG или XUL [8].
- HTML - стандартизированный язык разметки документов для просмотра веб-страниц в браузере. Веб-браузеры получают HTML документ от сервера по протоколам HTTP/HTTPS или открывают с локального диска, далее интерпретируют код в интерфейс, который будет отображаться на экране монитора [9].

- PhpStorm - это интегрированная среда разработки на PHP с интеллектуальным редактором, которая глубоко понимает код, поддерживает PHP 5.3-7.3 для современных и классических проектов, обеспечивает лучшее в индустрии автодополнение кода, рефакторинги, предотвращение ошибок налету и поддерживает смешивание языков [10].

### 3. Результаты исследования

Для начала следует создать 3 файла: index.html, script.js и style.css. Затем создаем стандартную структуру html файла, прописываем блоки score и game для отображения счетчика очков и самой игры соответственно. Также подключаем файлы стилей и js (Рис. 1).



```
1 <!doctype html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport"
6     content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
7   <meta http-equiv="X-UA-Compatible" content="ie=edge">
8   <title>Document</title>
9   <link rel="stylesheet" href="style.css">
10 </head>
11 <body>
12 <div class="head">
13   <div class="score"><p id="score"> <strong>0чки</strong></p></div>
14   <div class="score1"><p><input id="score1" size="2" value="0"></p></div>
15 </div>
16
17 <div class="game">
18   <div id="dino"></div>
19   <div id="cactus"></div>
20 </div>
21
22 <script src="script.js"></script>
23
24 </body>
25 </html>
```

Рисунок 1 – Структура html файла

Далее скачаем изображения динозавра и кактуса которые будут служить фоном для созданных ранее блоков и поместим их в папку img (Рис. 2).

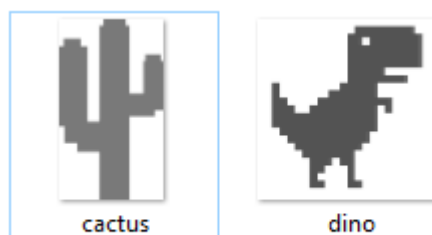


Рисунок 2 – Изображения игры

Перейдем в файл `style.css` и опишем свойства блоков игры, изображений и счетчика очков. Разместим блоки по середине, зададим их размер, установим фоновое изображение, а также добавим анимацию кактусу (Рис. 3).

```
index.html script.js style.css
1  .game{
2      width: 600px;
3      height: 200px;
4      border-bottom: 1px solid #000;
5      margin: auto;
6  }
7  #dino{
8      width: 50px;
9      height: 50px;
10     background-image: url(img/dino.png);
11     background-size: 50px 50px;
12     position: relative;
13     top: 150px;
14 }
15 #cactus{
16     width: 40px;
17     height: 40px;
18     background-image: url(img/cactus.png);
19     background-size: 20px 40px;
20     position: relative;
21     top: 110px;
22     left: 580px;
23     animation: cactusMov 1.5s infinite linear;
24 }
25 .score {
26     display: inline-block;
27 }
28 .score1 {
29     display: inline-block;
30 }
31 .head{
32     width: 100px;
33     height: 100px;
34     margin: auto;
35 }
```

Рисунок 3 – Свойства блоков игры

Далее опишем поведение ключевых кадров при анимации кактуса и прыжка динозавра (Рис. 4).



```
index.html x script.js x style.css x
36 @keyframes cactusMov {
37   0%{
38     left: 580px;
39   }
40   100%{
41     left: -20px;
42   }
43 }
44 .jump{
45   animation: jump 0.3s linear;
46 }
47 @keyframes jump {
48   0%{
49     top: 150px;
50   }
51   30%{
52     top: 130px;
53   }
54   50%{
55     top: 80px;
56   }
57   80%{
58     top: 130px;
59   }
60   100%{
61     top: 150px;
62   }
63 }
64
65
```

Рисунок 4 – Описание ключевых кадров анимации

Теперь перейдем к файлу `script.js` и опишем возможные варианты поведения игры. Сначала следует прописать добавление класса `jump` блоку `dino` по нажатию клавиши пробел, которая подразумевает прыжок, а по истечению 300 миллисекунд будем удалять данный класс. Следующим шагом пропишем условия окончания игры. Если блок динозавра касается блока кактуса, то выводится сообщение о том, что игра окончена и набранные игроком очки. (Рис. 5).

```
index.html x script.js x style.css x
1  const dino = document.getElementById( elementId: "dino")
2  var score = 0;
3
4  document.addEventListener( type: "keydown", listener: function (event :KeyboardEvent ){
5      jump();
6  })
7  function jump(){
8      if( dino.classList != "jump"){
9          dino.classList.add("jump")
10     }
11     setTimeout( handler: function (){
12         dino.classList.remove( tokens: "jump")
13     }, timeout: 300)
14 }
15
16 let isAlive = setInterval( handler: function (){
17     let dinoTop = parseInt(window.getComputedStyle(dino).getPropertyValue( property: "top"));
18     let cactusLeft = parseInt(window.getComputedStyle(cactus).getPropertyValue( property: "left"));
19     score=++score;
20     document.getElementById( elementId: 'score1').value=score;
21
22     if(cactusLeft < 50 && cactusLeft > 0 && dinoTop>= 140){
23         alert ("Игра окончена. Набранные очки: "+(score))
24         document.getElementById( elementId: 'score1').value='0';
25         score=0;
26     }
27 }, timeout: 10)
28
```

Рисунок 5 – Описание ключевых кадров анимации

Следующим шагом посмотрим, как выглядит игра в браузере (Рис. 6-7).

Очки



Рисунок 6 – Вид игры

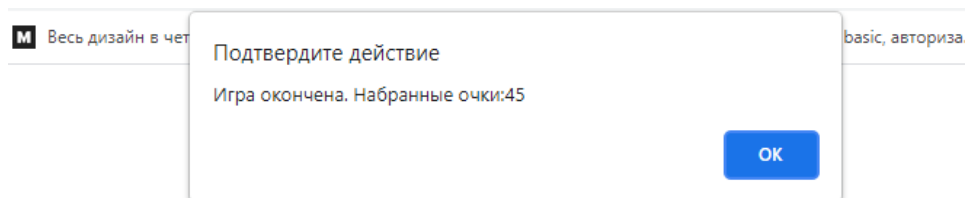


Рисунок 7 – Вид игры при проигрыше

#### 4. Выводы

В рамках данной статьи рассмотрены методы анимации блоков с описанием ключевых кадров, описано поведение блока при нажатии клавиши, обозначены условия окончания игры, а также создан счетчик очков. Результатом исследования является готовая игра T-Rex, разработанная с помощью языка программирования JavaScript.

#### Библиографический список

1. Соловьева Е. В., Максимова Т. А. Компьютерная игра-головоломка «Судоку» на языке JavaScript // Студенческие научные достижения. 2020. С. 21-25.
2. Никитин П. В. и др. Демонстрационные примеры по разработке браузерной игры по стандартам WorldSkills. 2019.
3. Поварницын Е. Н. Создание браузерной игры «Тетрис» // Форум молодых ученых. 2020. №. 2. С. 289-296.
4. Nabhani S. et al. Development and evaluation of an educational game to support pharmacy students // Currents in Pharmacy Teaching and Learning. 2020. Т. 12. №. 7. С. 786-803.
5. Taketoshi N., Yasushi K. Join token: A language mechanism for programming interactive games. 2012.
6. Базеева Н. А., Лебедев Д. С. Языки программирования для создания игр // E-Scio. 2019. №. 4 (31).
7. JavaScript URL: <https://ru.wikipedia.org/wiki/JavaScript> (дата обращения: 04.09.2021).
8. CSS URL <https://ru.wikipedia.org/wiki/CSS> (дата обращения: 04.09.2021).
9. HTML URL: <https://ru.wikipedia.org/wiki/HTML> (дата обращения: 04.09.2021).
10. PHPStorm URL: <https://www.jetbrains.com/ru-ru/phpstorm/> (дата обращения: 04.09.2021).