МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет Информационных Технологий

Кафедра Информационных систем и технологий

Специальность 1-98 01 03 «Программное обеспечение информационной безопасности мобильных систем»

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА КУРСОВОГО ПРОЕКТА**

по дисциплине «Компьютерные языки разметки»

Тема «GoodOldGames»

Исполнитель

студент 2 курса 7 группы Тихон Алексей Александрович

(Ф.И.О.)

Руководитель ассистент Кантарович В. С.

(учен. степень, звание, должность, подпись, Ф.И.О.)

Курсовой проект защищен с оценкой ­

Руководитель Кантарович В. С.

(подпись)

# Содержание

[Содержание 2](#_Toc90853342)

[Введение 3](#_Toc90853343)

[1 Обзор технических методов и программных средств разработки веб-сайта 4](#_Toc90853344)

[1.1 Обследование объекта и обоснование необходимости создания программного продукта. Анализ аналогичных решений 4](#_Toc90853345)

[1.2 Формирование требований к программному продукту 7](#_Toc90853346)

[1.3 Техническое задание 7](#_Toc90853347)

[1.4 Постановка задач 8](#_Toc90853348)

[2 Последовательность разработки содержания и структуры веб-сайта 9](#_Toc90853349)

[2.1 Построение прототипов интерфейсов и макетирование страниц 9](#_Toc90853350)

[2.2 Разработка дизайна элементов программного продукта 9](#_Toc90853351)

[2.3 Разработка решений по общесистемным вопросам 12](#_Toc90853352)

[2.4 Составление структуры программного продукта 13](#_Toc90853353)

[3 Программная реализация проекта 14](#_Toc90853354)

[3.1 Разработка и внедрение базы данных 14](#_Toc90853355)

[3.2 Верстка программного продукта 15](#_Toc90853356)

[3.3 Разработка программного кода 16](#_Toc90853357)

[4 Описание хода работы программного средства с копиями экрана пользовательского интерфейса основных элементов системы, а также ключевые функции разработанного программного кода 20](#_Toc90853358)

[4.1 Руководство пользователя 20](#_Toc90853359)

[4.2 Кроссбраузерное тестирование 23](#_Toc90853360)

[Заключение 28](#_Toc90853361)

[Список использованных литературных источников 29](#_Toc90853362)

# Введение

Игровая индустрия зародилась в 1970-х годах, как движение энтузиастов и за несколько десятилетий выросла в индустрию с миллиардными годовыми прибылями. Игровая индустрия также оказала влияние на развитие персональных компьютеров и их комплектующих: графические карты, которые изначально развивались в направлении увеличения количества цветов, позже стали развиваться для аппаратной поддержки GUI. Звуковые карты и вовсе изначально были созданы для внедрения качественного цифрового звука в компьютерные игры, а только потом адаптировались под другие нужды. CD и DVD изначально воспринимались как недорогой и относительно надежный способ хранения информации, когда эти технологии начали применяться в компьютерных играх началось развитие в сторону увеличения скорости чтения данных.

Однако на данный момент игровая индустрия погрязла в бездушных AAA проектах, которые ставят перед собой основной целью не привнести что-то новое, а просто заработать как можно больше денег. Каждая новая игра похожа на предыдущую, меняются только декорации, студии, которые когда-то делали качественно выкупаются гигантами индустрии и начинают участвовать в разработке проектов, которые уже давно встали на конвейер. Проекты, которые способны действительно удивить и дать глоток свежего воздуха в этом мире дешевых копий себя же настолько редки, что кажется: еще немного и действительно задохнешься.

Тем не менее отнять у людей старые полюбившиеся игры, которые были действительно уникальными, не может никто. Поэтому наше время идеально, чтобы обратиться к старым добрым играм, про которые все забыли.

В процессе работы над проектом были поставлены следующие задачи: разработать дизайн сайта; создать кроссбраузерную верстку для сайта; реализовать полноценную игру Tetris; реализовать полноценную игру Snake; реализовать полноценную игру Breakout; реализовать таблицу рекордов, хранящуюся в базе данных с возможностью занесения рекорда в таблицу.

Основная цель курсового проекта — создание полноценного сайта для того, чтобы посетители могли интересно провести время и познакомиться с истоками игровой индустрии.

Курсовой проект включает в себя 7 разделов: введение, содержание, обзор технических методов и программных средств разработки веб-сайта, последовательность разработки содержания и структуры веб-сайта, программная реализация проекта, описание хода работы программного средства с копиями экрана, заключение, список использованной литературы.

# 1 Обзор технических методов и программных средств разработки веб-сайта

# 1.1 Обследование объекта и обоснование необходимости создания программного продукта. Анализ аналогичных решений

В сети интернет есть множество ресурсов, предлагающих попробовать себя в классических играх. Пример одного из таких ресурсов:

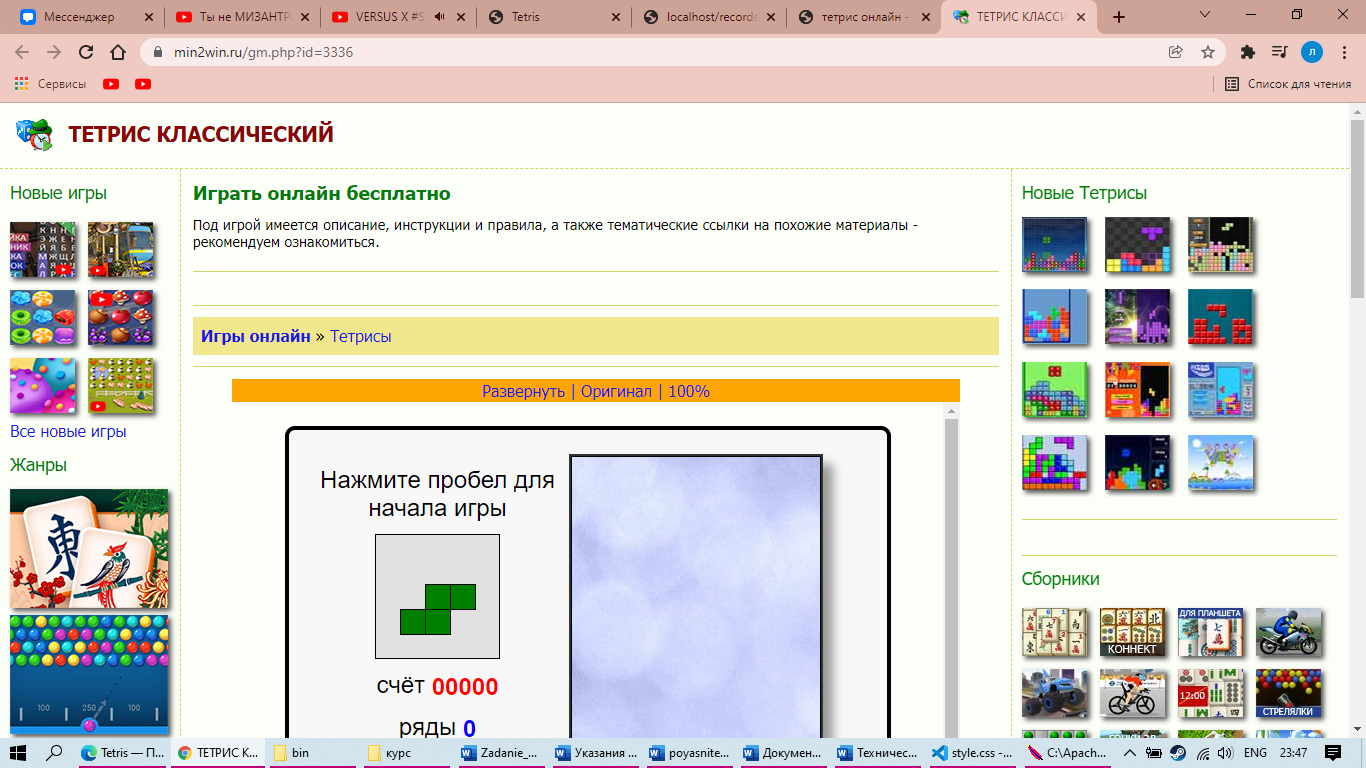


Рисунок 1.1 ­– Сайт с возможностью поиграть в Tetris

На данном сайте слишком много лишнего. Реклама других игр с сайта и т.д. Также отсутствует возможность заочно соревноваться с другими посетителями. То есть посетитель играет в Тетрис сам с собой. Также отсутствие таблицы рекордов уменьшает среднее время, проведенное на сайте, ведь возможность побить чужой рекорд люди могут рассматривать как вызов.

Примером хорошего сайта можно считать следующую страницу:

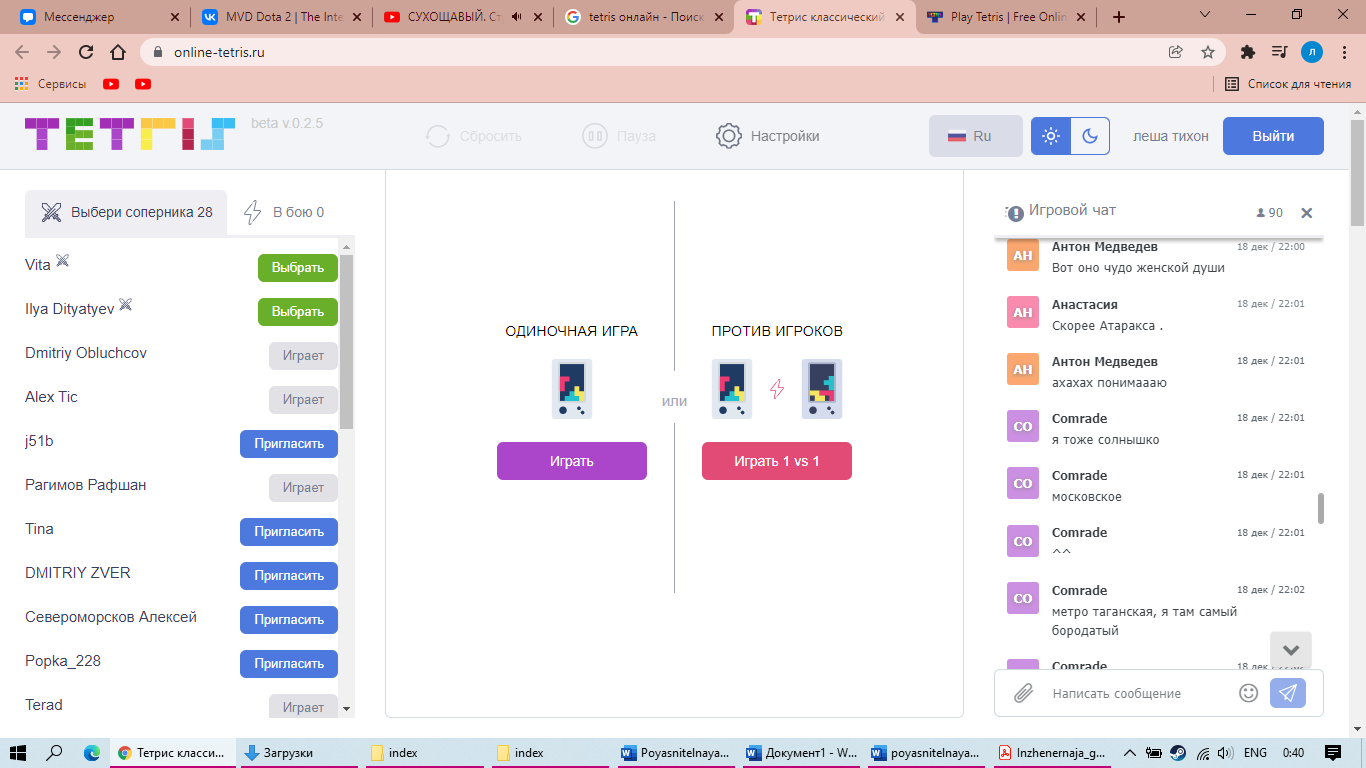


Рисунок 1.2 ­– Сайт online-tetris.ru

Приятный дизайн, возможность играть одному, играть 1 на 1, есть чат, в котором можно пообщаться с другими игроками. Также наличие таблицы рекордов (рис. 1.3) и даже рейтинга сезона (рис. 1.3):

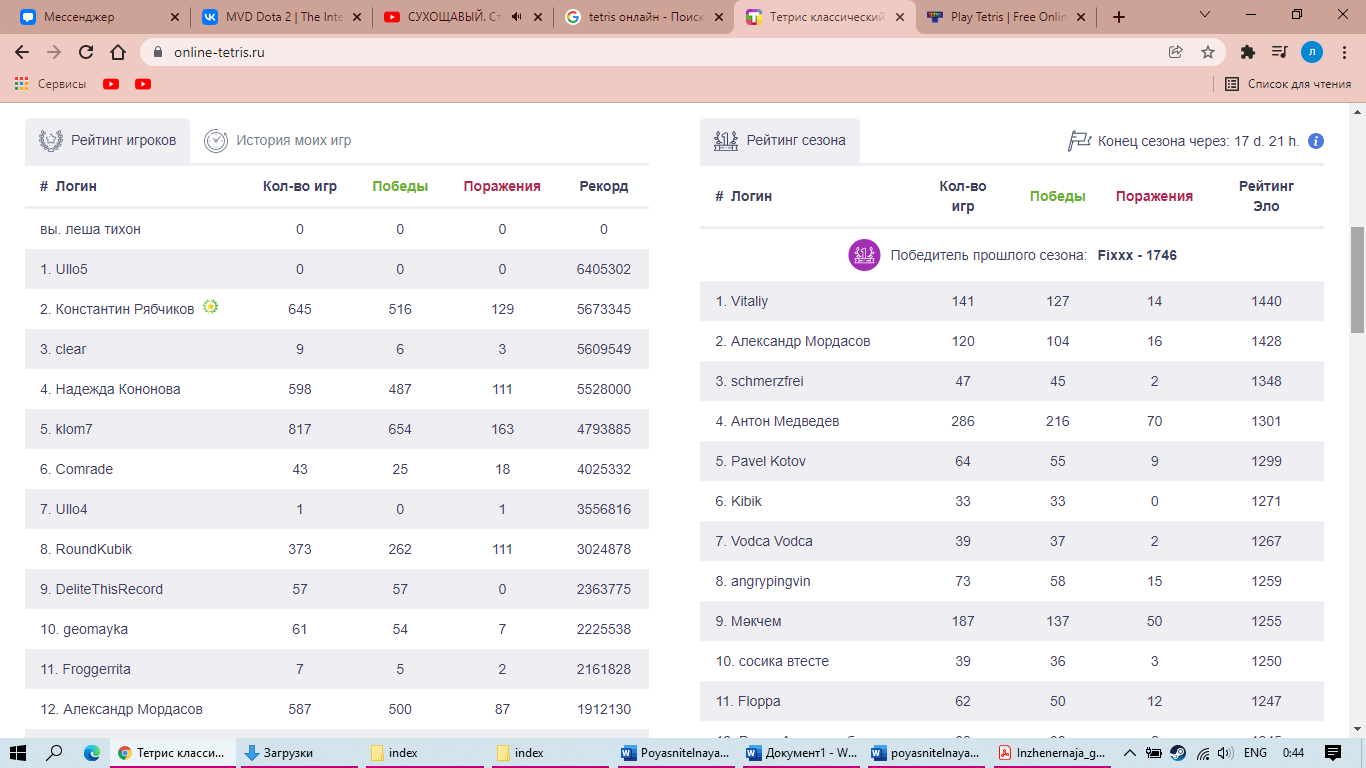


Рисунок 1.3 ­– Рейтинг игроков и Рейтинг сезона

История версий (рис. 1.4) и краткое описание правил игры (рис. 1.4):

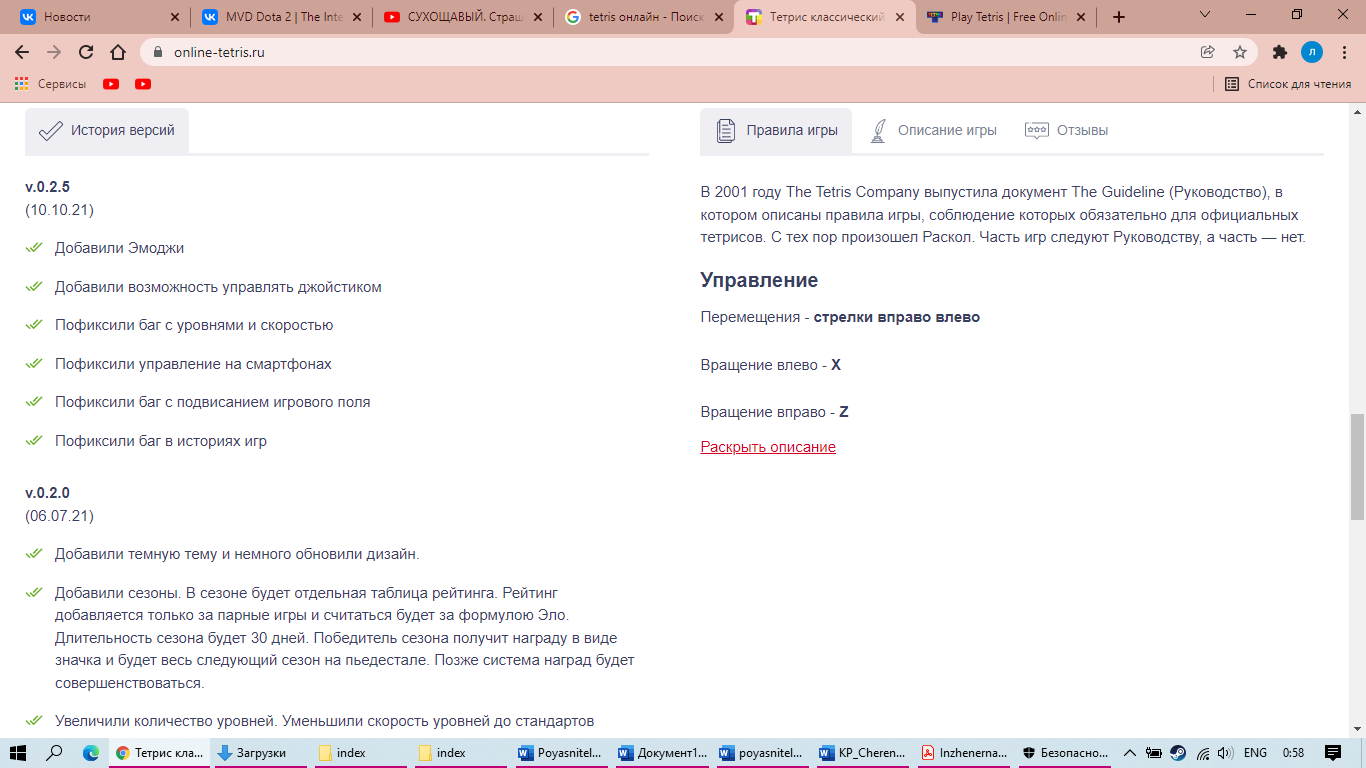


Рисунок 1.4 ­– История версий и Правила игры

Хорошо проработанный футер (рис. 1.5) выполнен в приятном дизайне и выглядит очень лаконично

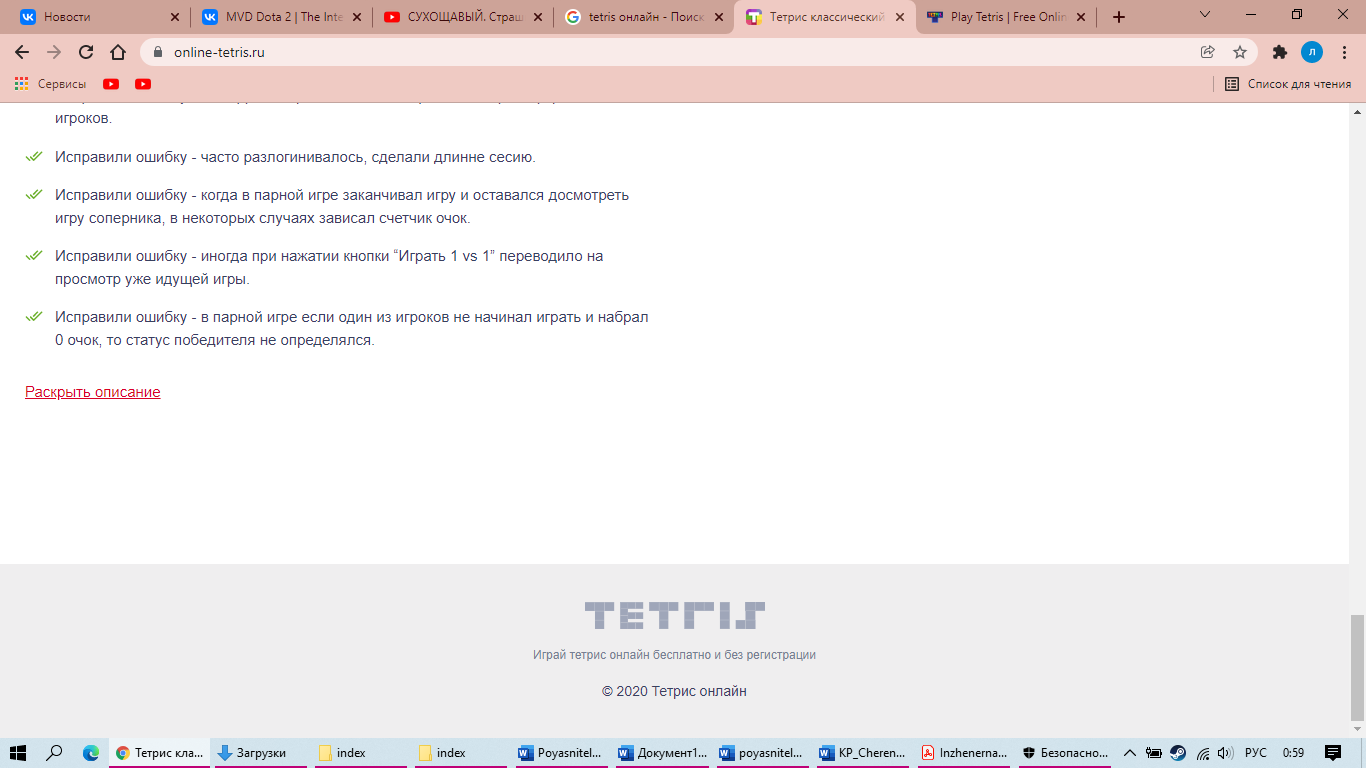


Рисунок 1.5 ­– Футер сайта

Сайт online-tetris.ru можно считать примером того, как должен выглядеть современный сайт. Приятный и негромоздкий дизайн, проработанные футер и хедер, отличный функционал, выражающийся в возможности играть 1 на 1 и наличии информации о лучших игроках.

# 1.2 Формирование требований к программному продукту

Основными требованиями к программному продукту являются комфортное использование и дизайн, отсылающий к тем самым играм, первые версии которых были созданы еще в прошлом веке. Сайт должен быть интуитивно понятным в использовании, удобным для просмотра и ознакомления с информацией. Сайт должен одинаково отображаться во всех популярных браузерах.

Основная задача сайта – развлечение посетителей, предоставление возможности ознакомиться с играми, с которых начиналась индустрия, дать возможность посоревноваться с другими пользователями и показать свои навыки.

Программный продукт должен быть выполнен в стиле минимализма, чтобы не отвлекать пользователей и позволить им полностью погрузиться в игровой процесс.

# 1.3 Техническое задание

Сайт дает возможность поиграть в классические игры: Tetris, Snake, Breakout. На сайте есть возможность узнать краткую историю, запустить любую из вышеперечисленных игр, а также занести свой результат в таблицу рекордов.

Целями проекта являются предоставление возможности интересного времяпрепровождения, ознакомление посетителей с историей и игровым процессом классических игр, предоставление возможности соревноваться и другими игроками.

Сайт предназначен для всех, желающих показать свои умения или ознакомиться с классикой игровой индустрии.

В проекте используются: HTML, CSS, PHP, JQuery и MySQL.

Дизайн сайта выполнен в трех основных цветах: желтый, белый и черный (рис. 1.6). Преобладающим цветом является черный

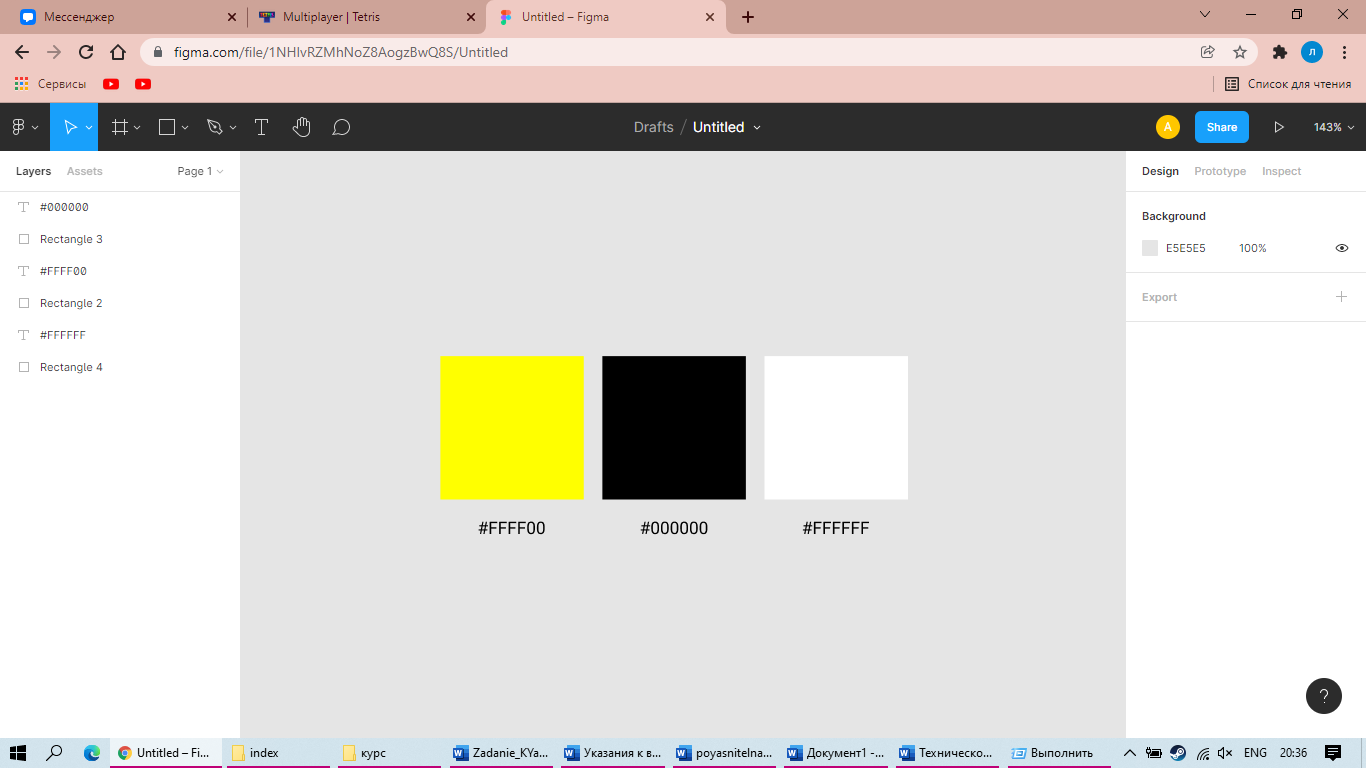


Рисунок 1.6 ­– Цветовая палитра сайта

Сайт состоит из 5 страниц: «Главная», «Tetris», «Snake», «Breakout», «Рекорды».

На главное странице расположить 6 блоков: 3 со скриншотами игрового процесса для каждой из игр и 3 текстовых блока с кратким описанием истории игры и кратким описанием игрового процесса.

На странице «Tetris» должно располагаться меню с возможностью перейти на любую из страниц, игровое поле размером 250 пикселей на 500 пикселей, справа от игрового поля должен быть расположен блок с информацией: количество очков, количество линий.

На странице «Snake» должно располагаться меню с возможностью перейти на любую из страниц, игровое поле размером 320 пикселей на 400 пикселей, справа от игрового поля должен быть расположен блок, отображающий количество очков и кнопка для начала игры.

На странице «Breakout» должно располагаться меню с возможностью перейти на любую из страниц, игровое поле размером 500 пикселей на 400 пикселей, справа от игрового поля должен быть расположен блок, отображающий количество очков и кнопка для начала игры.

Формат изображений JPG.

Сайт должен быть разработан при помощи технологий HTML5, CSS, JavaScript. Каждая страница должна содержать свой отдельный файл HTML, CSS и JS, если таковой требуется. Сайт должен корректно отображаться на всех актуальных браузерах. Основной язык русский. Структура сайта представлена на рисунке 1.7.

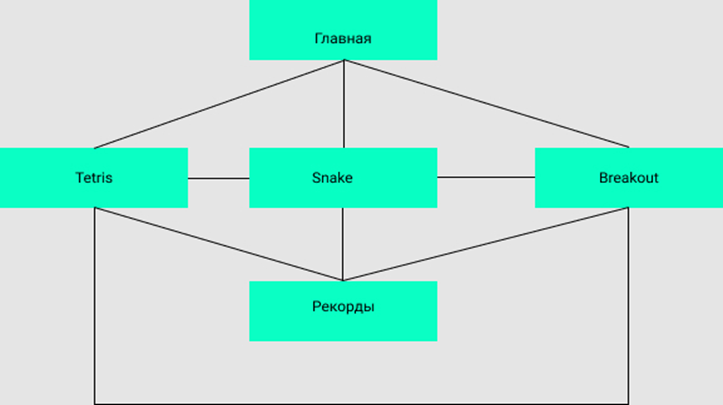


Рисунок 1.7 ­– Структура сайта

# 1.4 Постановка задач

Основными задачами курсового проекта являются:

* создание многостраничного сайта;
* реализация полноценной игры Tetris
* Реализация полноценной игры Snake
* Реализация полноценной игры Breakout
* Создание уникального дизайна;
* Создание страницы с таблицей рекордов
* Создание формы для занесения рекорда в таблицу
* Хранение данных о рекордах в базе данных

# 2 Последовательность разработки содержания и структуры веб-сайта

# 2.1 Построение прототипов интерфейсов и макетирование страниц

Прототип и макет сайта выполнены при помощи онлайн-сервиса для разработки интерфейсов и прототипирования Figma.

В ходе разработки прототипов страниц было решено располагать контент блоками, что не вызовет никаких трудностей с расположением контента. Все страницы выполнены в сочетании трех цветов (рис. 1.6). С помощью макета было выявлено наиболее благополучное расположение элементов страницы, а также подобрана цветовая палитра.

# 2.2 Разработка дизайна элементов программного продукта

Данный программный продукт является сайтом, основная цель которого – развлечение посетителей. Так как игры, представленные на сайте, являются классикой, то было принято решение создавать дизайн в ретро-стиле.

Для каждой страницы был разработан отдельный прототип с целью получить максимально привлекательные положения элементов и сочетание цветов.

Страницы, на которых расположены игры состоят из меню и игрового поля чтобы обеспечить максимальное погружение в игровой процесс и не отвлекать пользователя от игры. Страница с рекордами и главная страницы имеют черный фон и также выполнены в трех цветах, представленных на рисунке 1.6.

На страницах данного курсового проекта используется 1 шрифт: Press Start 2 Play. Он является основным и абсолютно весь текст на сайте представлен в этом шрифте. Этот шрифт был выбран, потому что его стиль отлично соответствует основному ретро-стилю.

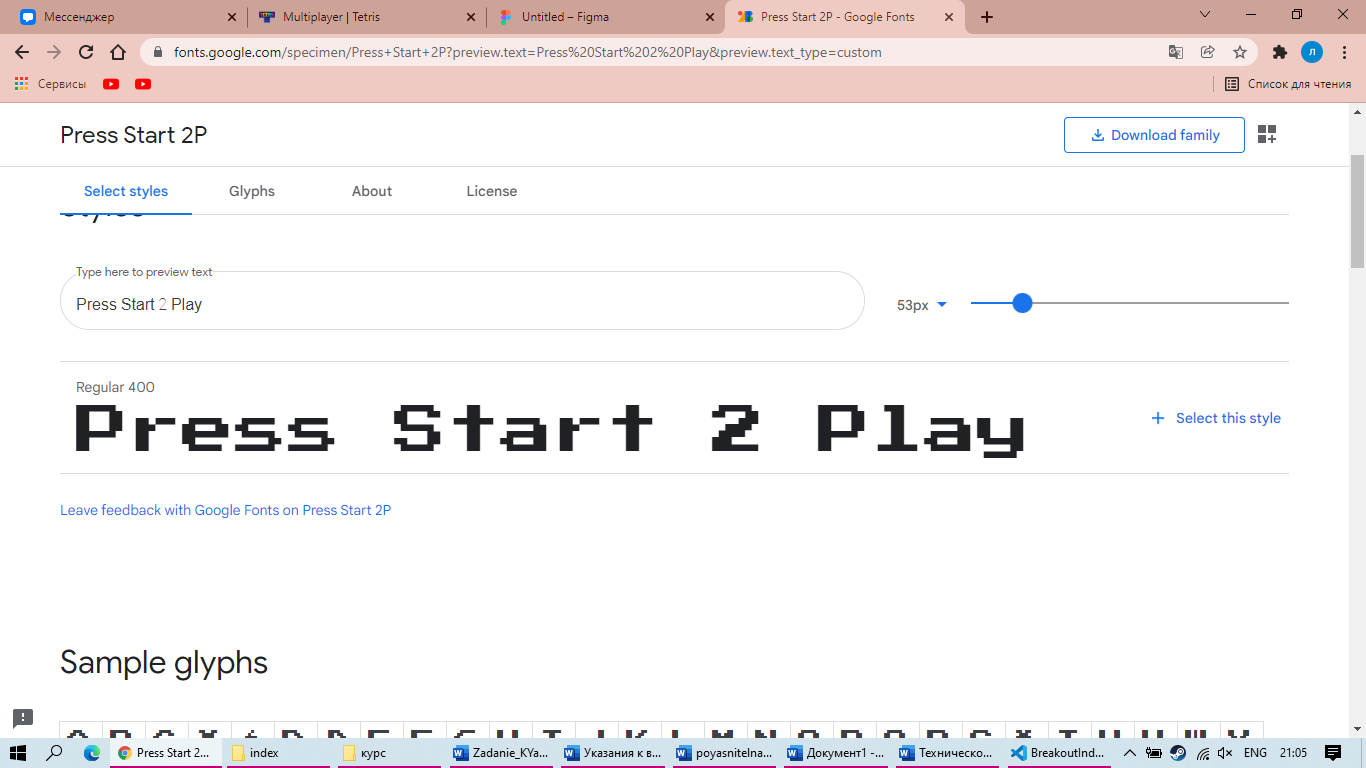


Рисунок 2.1 ­– Выбранный шрифт

Хедер выполнен в желтом и черном цветах. Хедер содержит название сайта и его слоган.



Рисунок 2.2 ­– Хедер сайта

Меню расположено на всех страницах кроме главной, так как с главной страницы можно перейти на любую страницу, нажав на соответствующую картинку.

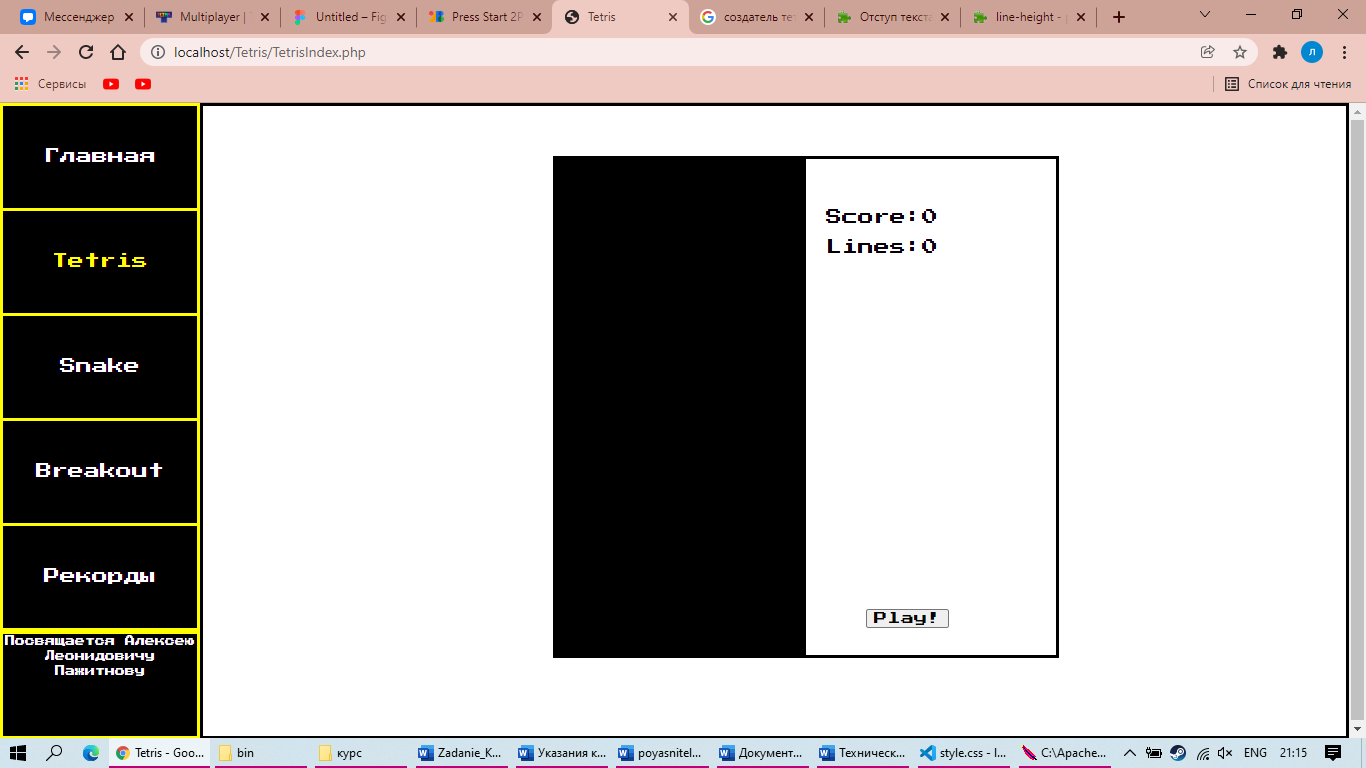


Рисунок 2.3 ­– Меню сайта

Игровые поля имеют одинаковый стиль, они различаются только размером и информацией, расположенной в блоках справа.

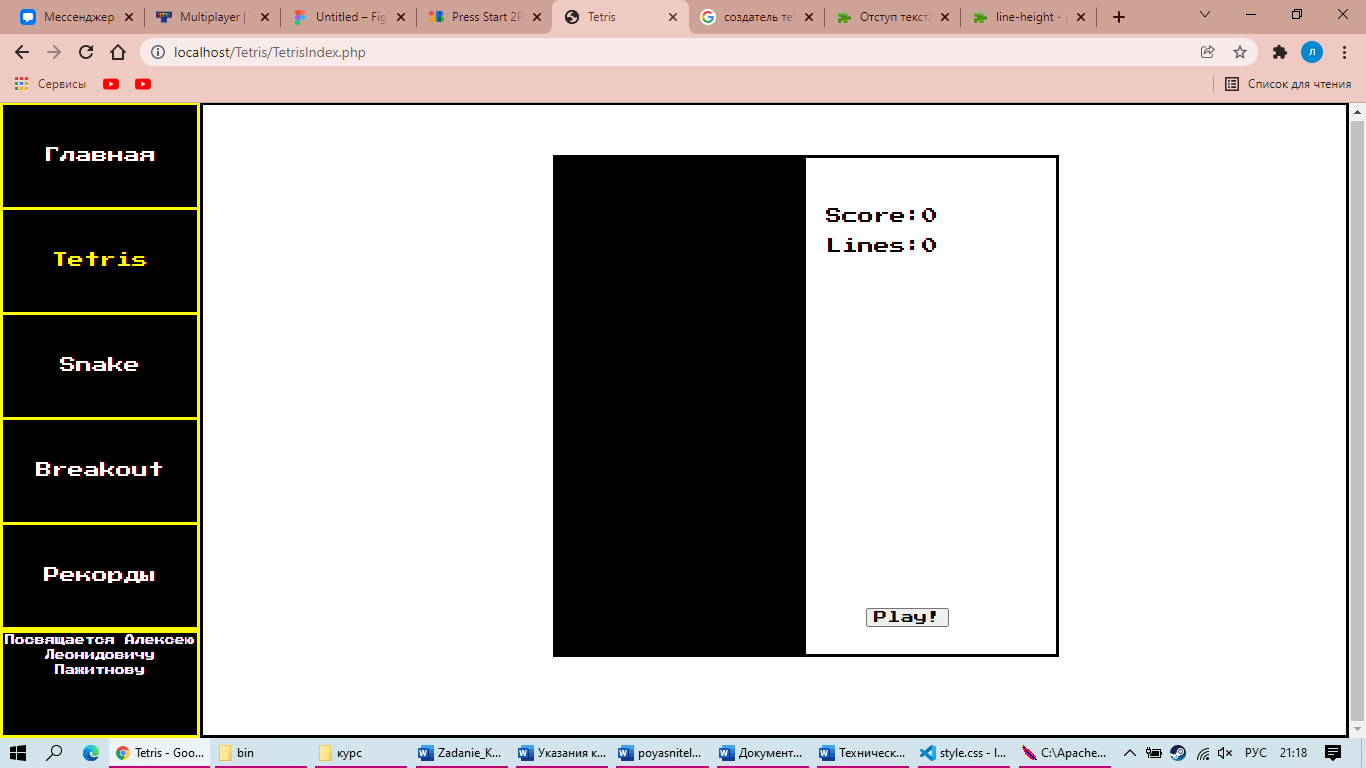


Рисунок 2.4 ­– Игровое поле Тетриса

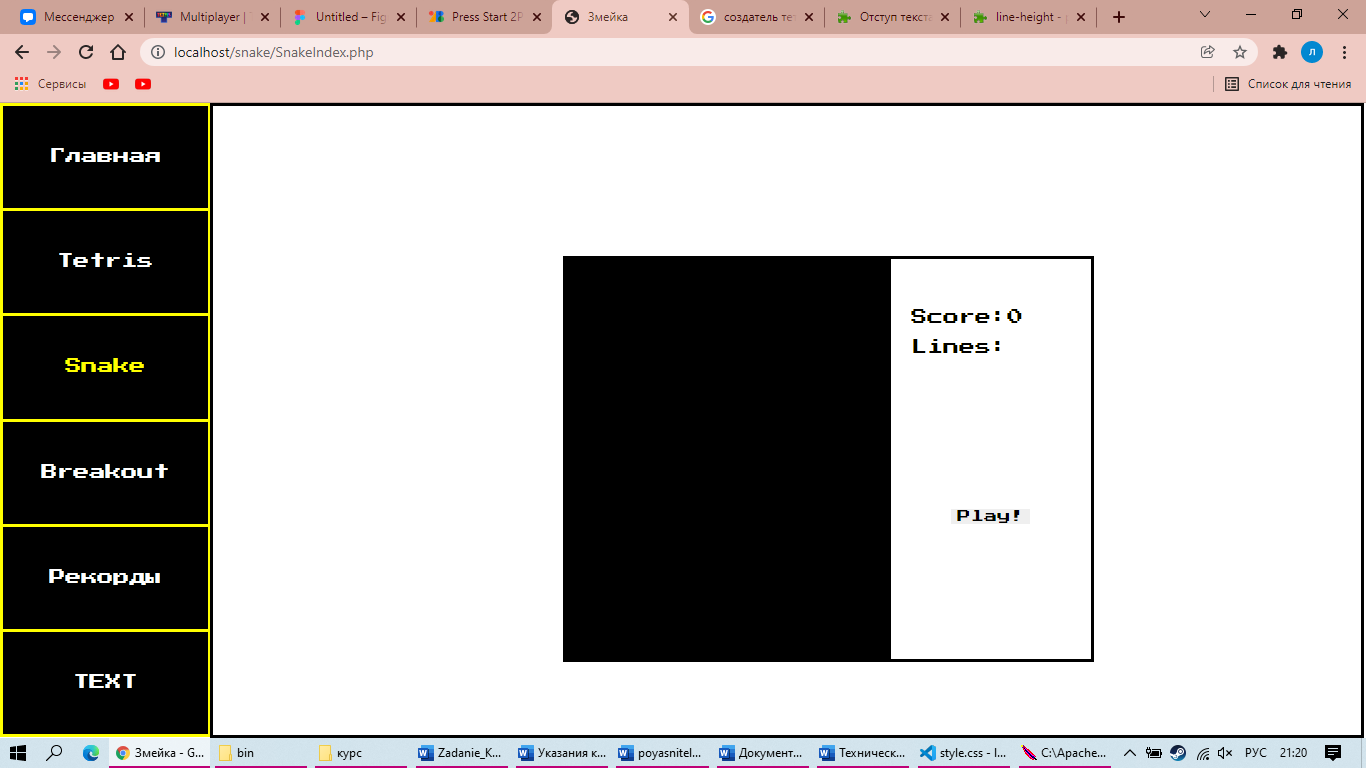


Рисунок 2.5 ­– Игровое поле Змейки



Рисунок 2.6 ­– Игровое поле Breakout

В случае, если игрок установит новый рекорд слева от игрового поля появится форма для занесения результата игрока в таблицу рекордов.

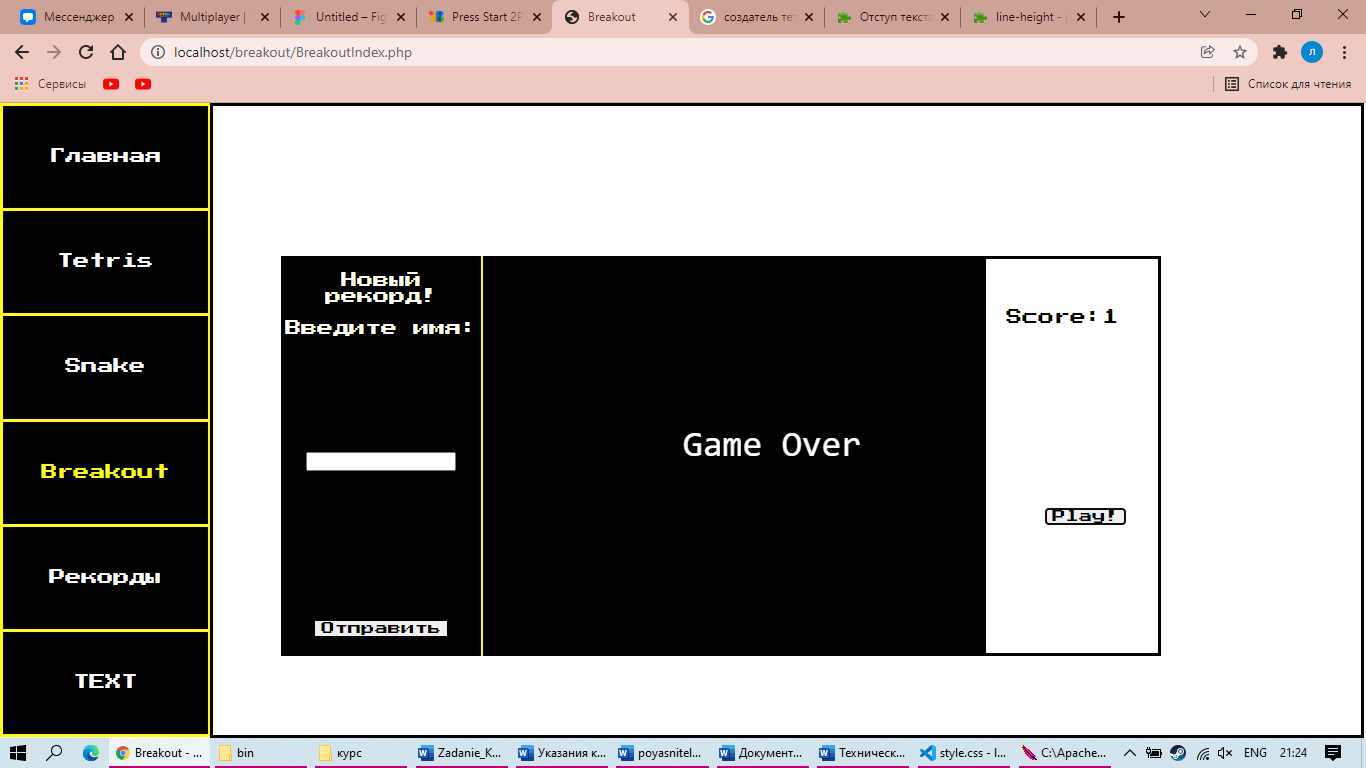


Рисунок 2.7 ­– Игровое поле Breakout

Каждый из пунктов меню при наведении на него меняет свой цвет с изначального на желтый.

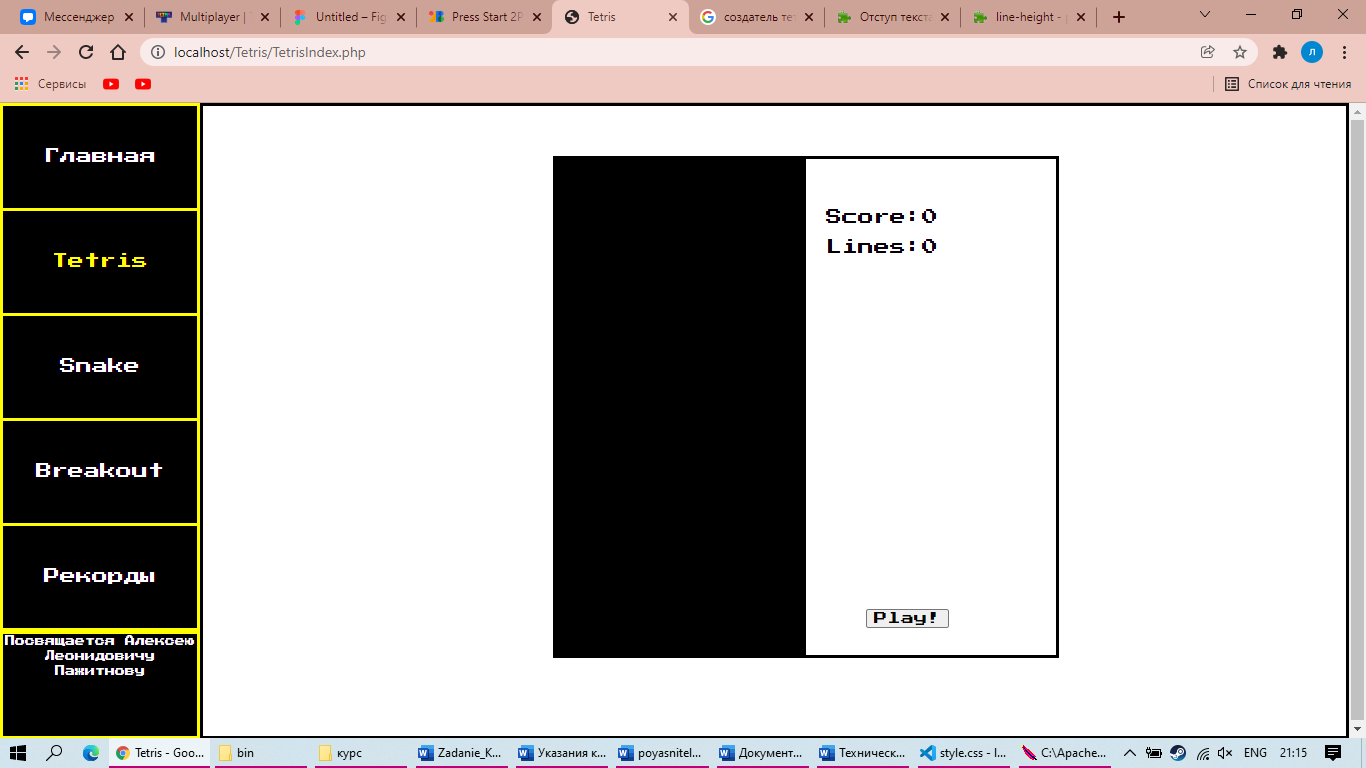


Рисунок 2.8 ­– Пункты меню до наведения

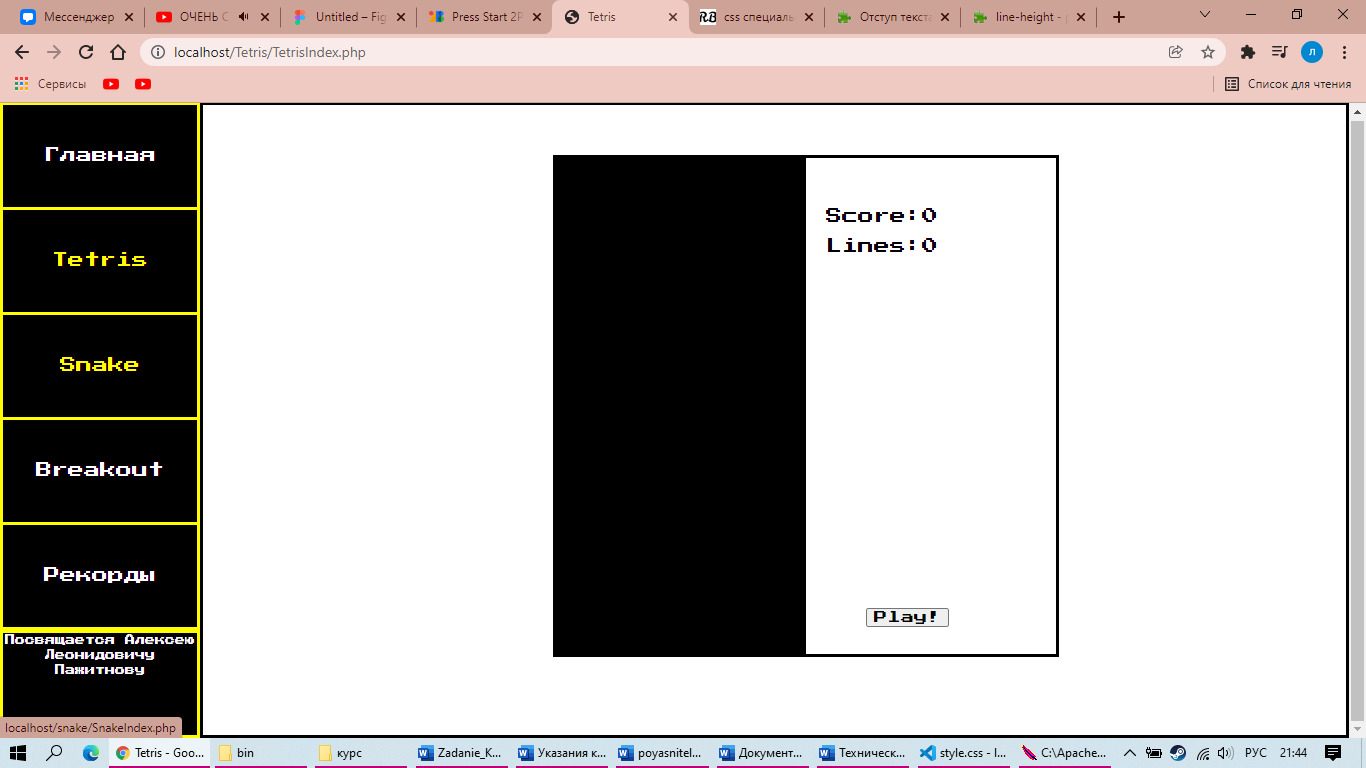


Рисунок 2.9 ­– Пункт меню после наведения

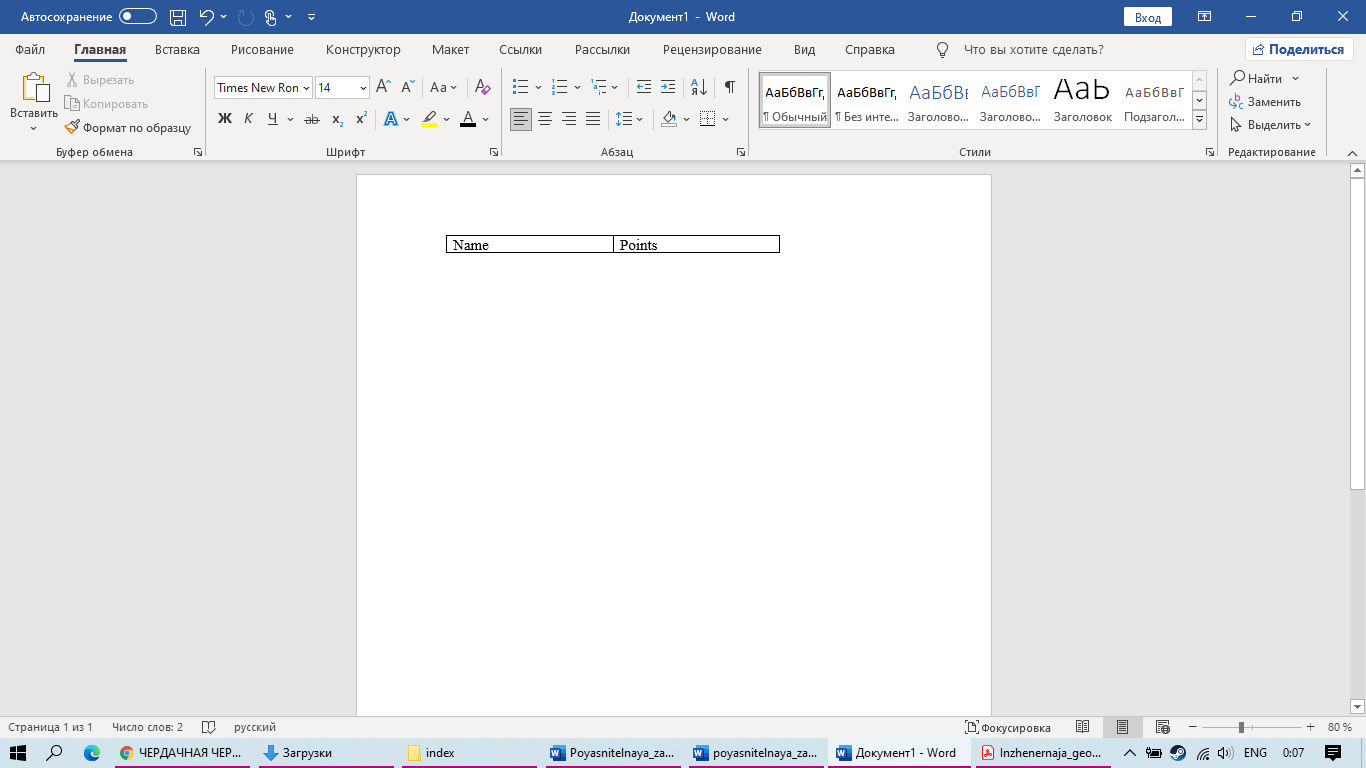
Футер выполнен в основных цветах сайта. Футер содержит информацию о месте создание сайта, а также о дате написания данной записки.



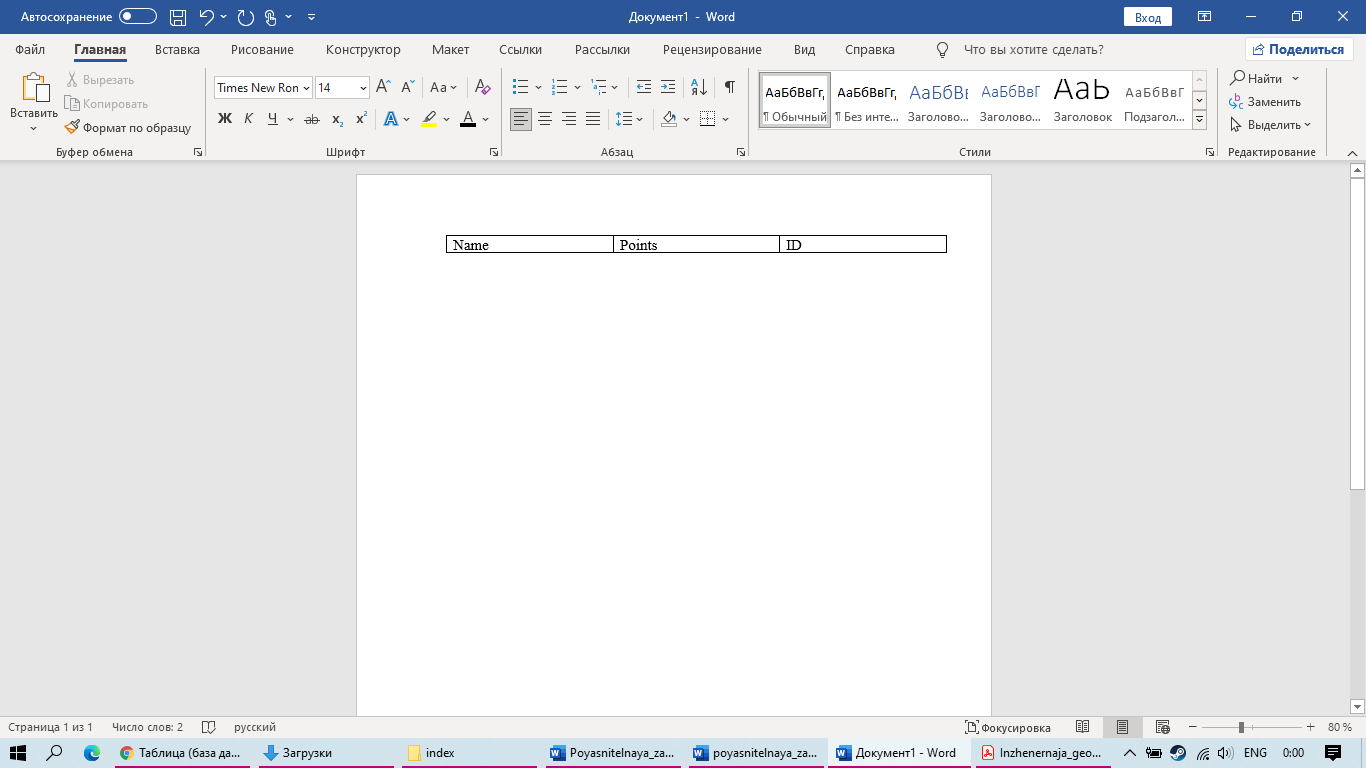
Рисунок 2.10 ­– Футер сайта

# 2.3 Разработка решений по общесистемным вопросам

В ходе работы надо курсовым проектом возникла проблема с занесением значений в базу данных. Изначальный вид строки базы данных:



Проблема с такой структурой заключалась в отсутствии уникального поля, используя которое можно было бы без проблем удалять строку для занесения новой. Например: в таблице находятся 2 столбца с одинаковыми значениями ячейки Points. Для занесения нового рекорда нам необходимо удалить один из столбцов. Для этого необходимо указать ячейку, по которой будет выполняться удаление, а также условие удаления. Так как значения ячеек Points одинаковые то использовать эту ячейку для удаления мы не можем. Но у нас все еще есть ячейка Name, с помощью которой можно удалить строку. Однако может возникнуть ситуация, в которой иметь одинаковые значения будут не только ячейки Points, но и Name. Следовательно, необходимо ввести еще один столбец:



Введение столбца ID помогло решить данную проблему. Эта ячейка изначально инициализируется уникальным значением для каждой строки и используется для удаления строки из таблицы. Значение ячейки ID новой строки равно значению удаленной строки, что позволяет избежать путаницы при управлении таблицей.

# 2.4 Составление структуры программного продукта

При создании сайта была выбрана блочная верстка, так как использование блоков позволяет с легкостью манипулировать расположением элементов на странице.

На главной странице расположены хедер, футер, и 6 блоков, содержащих информацию

На каждой странице с игрой расположены по 4 основных блока: меню, которое предоставляет возможность быстро переключаться между страницами сайта, игровое поле, форма для записи в таблицу рекордов, которая появляется если пользователь устанавливает новый рекорд, блок с информацией справа от игрового поля, который содержит информацию о текущей игре.

На странице с рекордами расположены два блока: один содержит меню, другой содержит 3 таблицы, которые хранят в себе информацию о достижениях посетителей.

Реализованы элементы отзывчивого дизайна: изменение цвета пункта меню при наведении с изначального на желтый (рисунки 2.8-2.9).

# 3 Программная реализация проекта

Сайт разрабатывался с помощью редактора кода Visual Studio Code. Visual Studio Code — редактор исходного кода, разработанный Microsoft для Windows, Linux и macOS. Позиционируется как «лёгкий» редактор кода для кроссплатформенной разработки веб- и облачных приложений. Приложение имеет все функции, которые можно ожидать от редактора кода, включая подсветку синтаксиса, автоопределение языков, автозаполнение текста, возможность использовать несколько панелей и т.д.

При создании сайта и его функционала были использованы HTML, CSS, jQuery, PHP и MYSQL.

HTML — 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.

Стилем, или CSS, называется набор параметров форматирования, который применяется к элементам документа, чтобы изменить их внешний вид. Зачастую возможностей HTML для оформления элементов сайта не хватает, поэтому CSS становится мощным инструментом для реализации дизайна.

JQuery — набор функций JavaScript, фокусирующийся на взаимодействии JavaScript и HTML. Библиотека jQuery помогает легко получать доступ к любому элементу DOM, обращаться к атрибутам и содержимому элементов DOM, манипулировать ими. Также библиотека jQuery предоставляет удобный API для работы с AJAX, который использовался для отправки данных на сервер.

PHP — скриптовый язык общего назначения, интенсивно применяемый для разработки веб-приложений. На данном сайте PHP применяется для взаимодействия с базой данных MYSQL.

MySQL-свободная реляционная система управления базами данных. Была выбрана именно эта СУБД, потому что она проста в установке и не требует дополнительной настройки. СУБД MySQL используется для хранения таблицы рекордов.

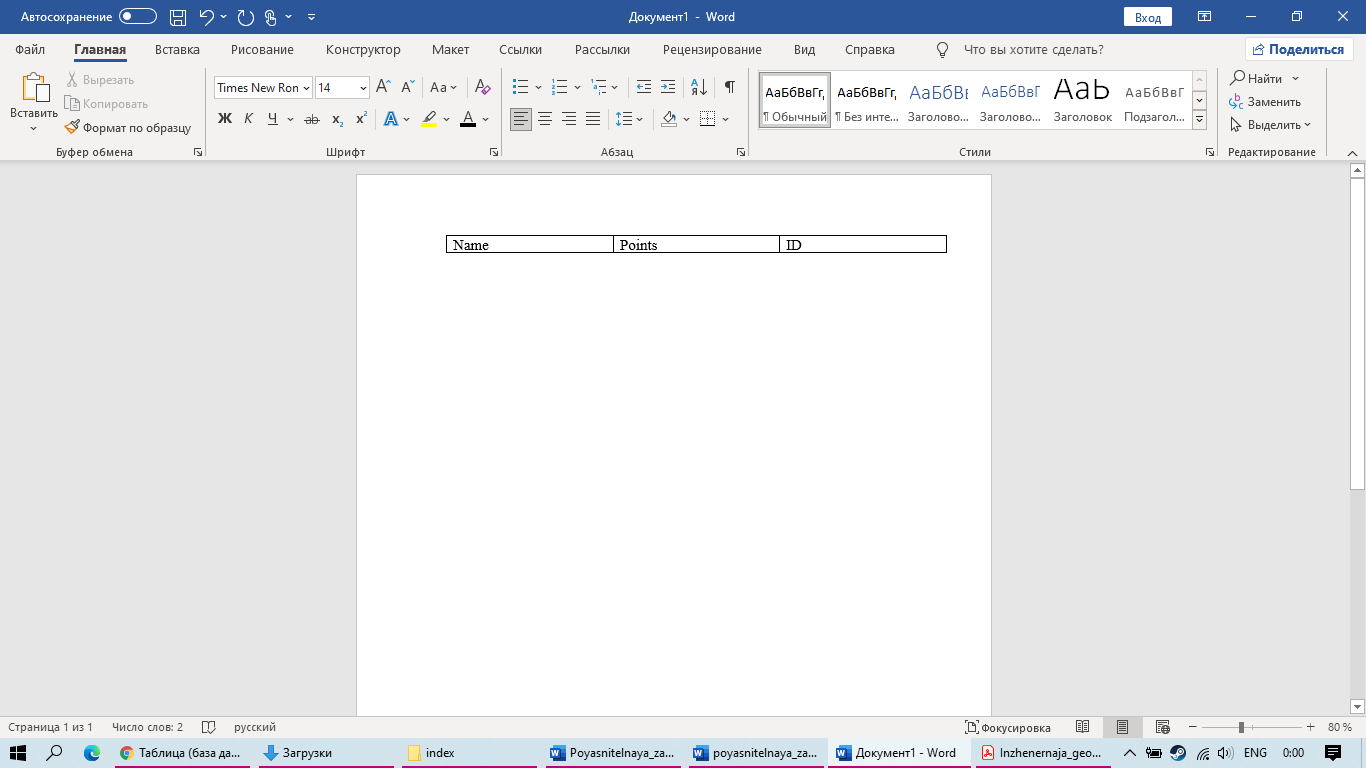
# 3.1 Разработка и внедрение базы данных

Для разработки базы данных был выбран СУБД MySQL. Эта система управления базами данных использует стандартную форму SQL. Утилиты для проектирования таблиц имеют интуитивно понятный интерфейс. MySQL поддерживает до 50 миллионов строк в таблице. Предельный размер файла для таблицы по умолчанию 4 ГБ, но его можно увеличить.

Преимущества выбранной СУБД:

* Масштабируемость
* Лёгкость использования
* Безопасность
* Скорость

База данных состоит из 3 таблиц: tetris, snake и breakout. Каждая таблица содержит 10 строк и 3 столбца.



Столбец Name хранит значение, которое пользователь вводит при установке нового рекорда.

Столбец Points хранит значение, которое равно количеству очков, набранных пользователем, установившим рекорд.

Столбец ID используется только на стороне сервера для упрощения удаления и занесения данных в таблицу.

# 3.2 Верстка программного продукта

Верстка была выполнена с использованием CSS Flexbox. Примеры использования CSS Flexbox представлены в листинге Г.3.

.head{

    position: absolute;

    top: 0;

    left: 0;

    width: 100%;

    height: 150px;

    background-color: black;

    display: flex;

    flex-direction: column;

    align-items: center;

    border-bottom: solid yellow;

}

Листинг 1 Г.3 ­– Использование CSS Flexbox в стилизации хедера

.field{

    left: 15.4%;

    position: absolute;

    top: 0;

    vertical-align: middle;

    text-align: center;

    border: solid black;

    width: 84%;

    height: 99%;

    display: inline-flex;

    color: white;

}

Листинг 2 Г.3 ­– Использование CSS Flexbox в блоке с информацией

.main{

    border-top: solid yellow;

    left: 0;

    background-color: black;

    height: 1366;

    width: 100%;

    top: 150;

    position: absolute;

    display: flex;

    flex-direction: row;

    justify-content: space-around;

    align-items: center;

    flex-wrap: wrap;

    align-content: space-around;

}

Листинг 3 Г.3 ­– Использование CSS Flexbox в стилизации главного блока

# 3.3 Разработка программного кода

Основной задачей данного проекта была разработка полноценных игр. Все игры были реализованы с помощью JavaScript. Далее приведены листинги наиболее важных функций для каждой из игр.

В основном игровом цикле игры Snake проверяется условие продолжения игры, а также происходит отрисовка змейки и ягоды.

function gameLoop() {

    if (flag === true) {

        requestAnimationFrame(gameLoop)

    }

    if ( ++config.step < config.maxStep) {

        return;

    }

    config.step = 0;

    context.clearRect(0, 0, canvas.width, canvas.height);

    drawBerry();

    drawSnake();

}

Листинг 4 Г.3 ­– Главный цикл игры Snake

В основном игровом цикле игры Tetris происходит отрисовка всех занятых ячеек при помощи цикла for, движение тетрамино при помощи функций CheckBottom() и CheckOther(), происходит вызов функции PlaceTetr() которая размещает тетрамино, если оно доходит до самого низа и функции DestroyLine(), которая отвечает за уничтожение полностью заполненных линий и начисление очков.

function Game(){

    if(flag===true){

      requestAnimationFrame(Game);

    }

    ctx.clearRect(0,0,250,500);

    for(i=0;i<20;i++){

        for(j=0;j<10;j++){

          if(playfield[i][j]===1){

            ctx.fillStyle="white";

            ctx.fillRect(j\*cell,i\*cell,cell-1,cell-1)

          }

        }

    }

    if(++count>40 && CheckBottom() && CheckOther()){

        tetromino.row++;

        count=0;

    }

    for(i=0;i<tetromino.matrix.length;i++){

        for(j=0;j<tetromino.matrix[i].length;j++){

            if(tetromino.matrix[i][j]===1){

            ctx.fillStyle="white"

            ctx.fillRect((tetromino.col+j)\*cell,(tetromino.row+i)\*cell,cell-1,cell-1)

            }

        }

    }

    PlaceTetr();

    DestroyLine();

    Over();

}

Листинг 5 Г.3 ­– Главный цикл игры Tetris

Функция rotate(matrix) принимает используется для поворота тетрамино на 90 градусов.

function rotate(matrix) {

    const N = matrix.length - 1;

    const result = matrix.map((row, i) =>

      row.map((val, j) => matrix[N - j][i])

    );

    // на входе матрица, и на выходе тоже отдаём матрицу

    return result;

}

Листинг 6 Г.3 ­– Функция поворота матрицы

В основной функции игры Breakout происходит отрисовка кирпичей функцией drawBricks(), отрисовка мячика функцией drawBall(), отрисовка ракетки функцией drawPaddle() и проверка столкновения мячика со стенками и изменение его направления в случае столкновения.

function draw(){

    if(flag===true){

        requestAnimationFrame(draw);

    }

    context.clearRect(0,0,canvas.width,canvas.height);

    drawbricks();

    drawBall();

    drawPaddle();

    colission();

    if(ball.x + ball.xSpeed > canvas.width-10 || ball.x + ball.xSpeed < 10) {

        ball.xSpeed = -ball.xSpeed;

    }

    if(ball.y+ball.ySpeed<10){

        ball.ySpeed=-ball.ySpeed;

    }

    else if(ball.y+ball.ySpeed>canvas.height-10){

        if(ball.x>paddle.px && ball.x<paddle.px+100){

            ball.ySpeed=-ball.ySpeed;

        }

        else{

            Over();

        }

    }

    ball.x += ball.xSpeed;

    ball.y += ball.ySpeed;

}

Листинг 7 Г.3 ­– Основной цикл игры Breakout

Также каждая страница с игрой получает два значения из базы данных: минимальное значение очков и ID записи с минимальным количеством очков.

<?php

     $link =new mysqli("localhost","root","naziwin","records");

     if($link->connect\_error){

      die("Ошибка: " . $link->connect\_error);

     }

     $value = $link->query(" SELECT \* FROM breakout");

     $min=1000000;

     $minID=0;

     foreach($value as $row){

         if($min>$row["Points"]){

             $min=$row["Points"];

             $minID=$row["ID"];

         }

     }

?>

Листинг 8 Г.3 ­– Получение данных из БД

Если пользователь установит новый рекорд, то слева от игрового поля появится форма для отправки данных. При нажатии на кнопку отправки произойдет вызов функции Ajax, которая передаст на сервер данные о количестве очков пользователя, имени, которое пользователь ввел в форму, а также ID записи с минимальным количеством очков.

$("#button").on("click", function(){

  $.ajax({

    url: 'TetrisAdd.php',

    type: "POST",

    data:{

      'Score' : scr,

      'Name' : document.getElementById("nname").value,

      'ID' : minID,

    },

    dataType:'html'

  })

})

Листинг 9 Г.3 ­– Отправка данных на сервер

Переданные данные обрабатываются на стороне сервера и заносятся в базу данных.

<?php

    $Name=$\_POST['Name'];

    $Score=$\_POST['Score'];

    $ID=$\_POST['ID'];

    echo $Score;

    echo $Name;

    $link =new mysqli("localhost","root","naziwin","records");

    if($link->connect\_error){

     die("Ошибка: " . $link->connect\_error);

    }

    $DEL="DELETE FROM tetris where ID='$ID'";

    $link->query($DEL);

    $INSERT="INSERT INTO tetris(name,points,ID)values('$Name',$Score,$ID)";

    $link->query($INSERT);

?>

Листинг 10 Г.3 ­– Запись данных в таблицу данных

Для отображения таблицы рекордов также происходит обращение в базе данных с помощью PHP с использованием подключения в объектно-ориентированном стиле.

<?php

            $link =new mysqli("localhost","root","naziwin","records");

            if($link->connect\_error){

             die("Ошибка: " . $link->connect\_error);

            }

            $value = $link->query(" SELECT \* FROM breakout ORDER BY points DESC");

            foreach($value as $row){

                echo "<tr>";

                    echo"<td>" . $row["Name"] . "</td>";

                    echo"<td>" . $row["Points"] . "</td>";

                echo "</tr>";

            }

        ?>

Листинг 11 Г.3 ­– Получение данных из БД и вывод данных в таблицу

# 4 Описание хода работы программного средства с копиями экрана пользовательского интерфейса основных элементов системы, а также ключевые функции разработанного программного кода

# 4.1 Руководство пользователя

На главной странице находится Хедер, содержащий название сайта, Футер, 3 картинки, которые являются скриншотами игрового процесса и 3 блока с текстом, которые представляют из себя краткую историю и правила для каждой из игр.

Каждая картинка является ссылкой на страницу с соответствующей игрой. Для того чтобы перейти на страницу достаточно просто нажать на картинку.



Рисунок 4.1 ­– Главная страница

На каждой из страниц: Tetris, Snake, Breakout в левой части расположено меню с пятью пунктами: Главная, Tetris, Snake, Breakout, Рекорды. По центру каждой страницы расположено игровое поле и поле с информацией о текущей игре. Для того чтобы начать игру необходимо начать на кнопку “Play!”, расположенную внизу блока с информацией о текущей игре.

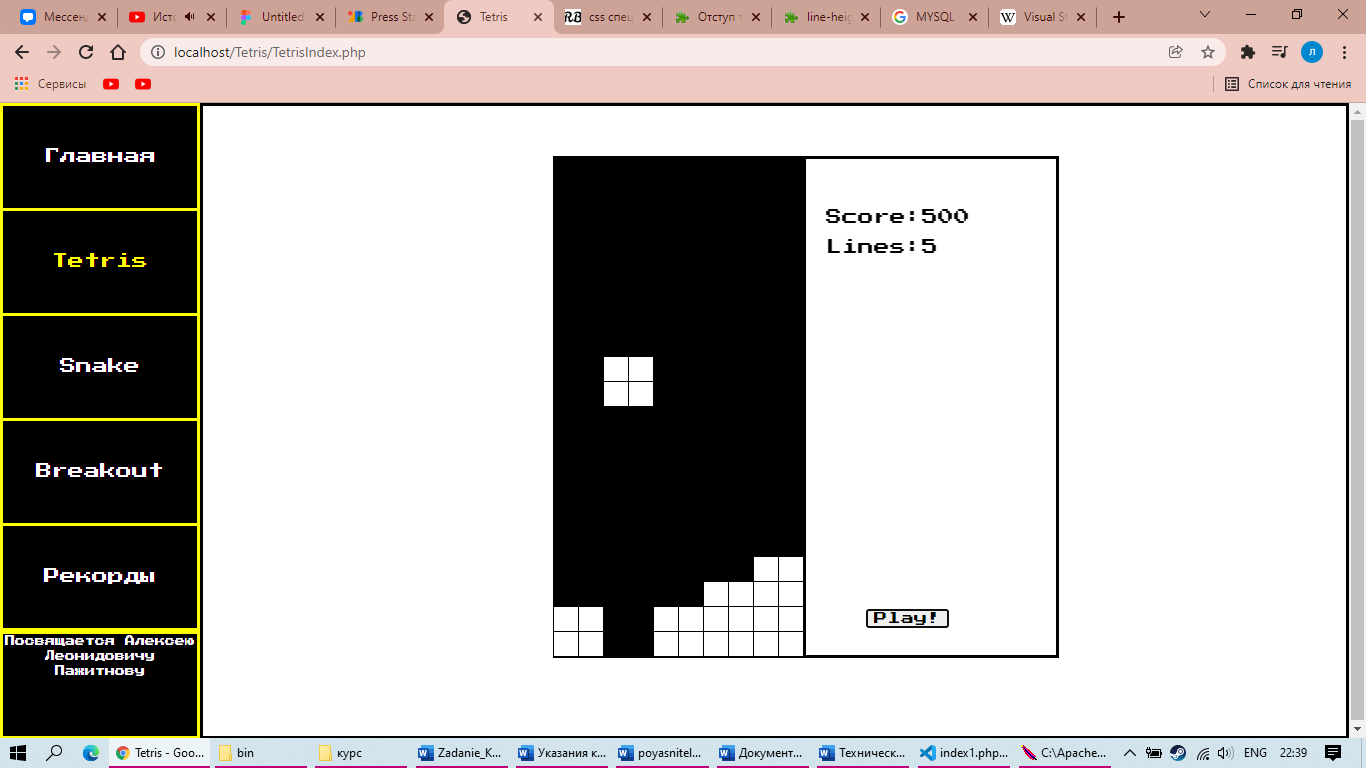


Рисунок 4.2 ­– Страница Tetris

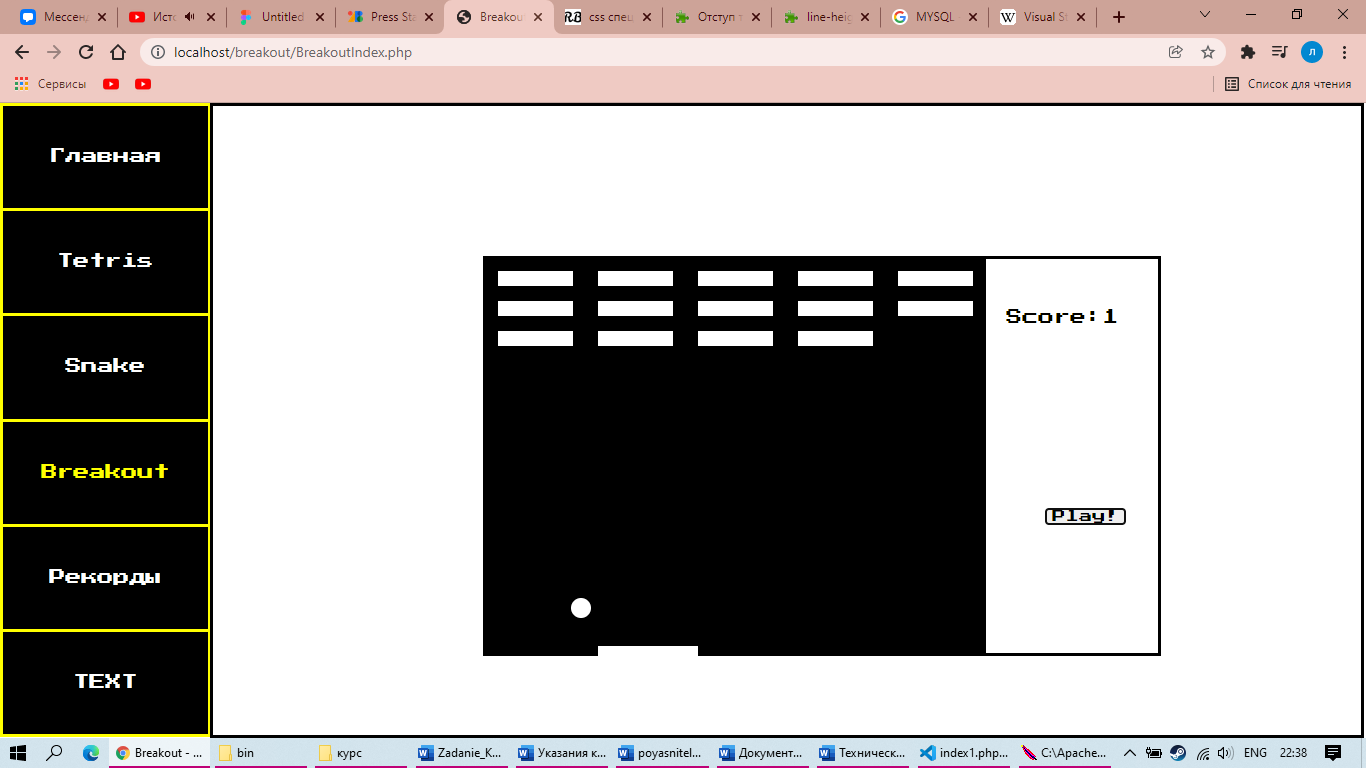


Рисунок 4.3 ­– Страница Breakout

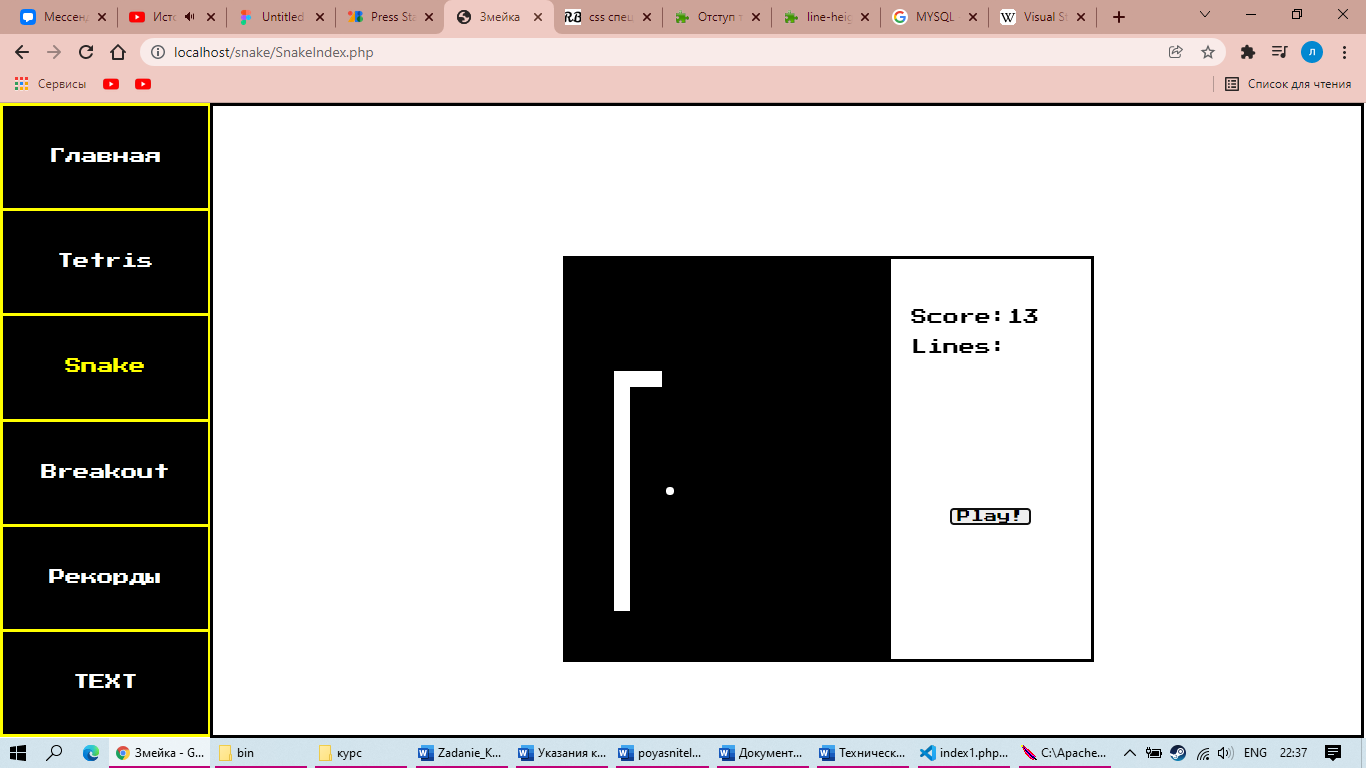


Рисунок 4.4 ­– Страница Snake

В случае установки нового рекорда слева от игрового поля появляется форма для записи рекорда. Для того чтобы отправить рекорд необходимо ввести имя и нажать на кнопку Отправить. В случае если пользователь не желает отправлять свой рекорд, ему достаточно просто нажать на кнопку “Play!”, и форма пропадет.

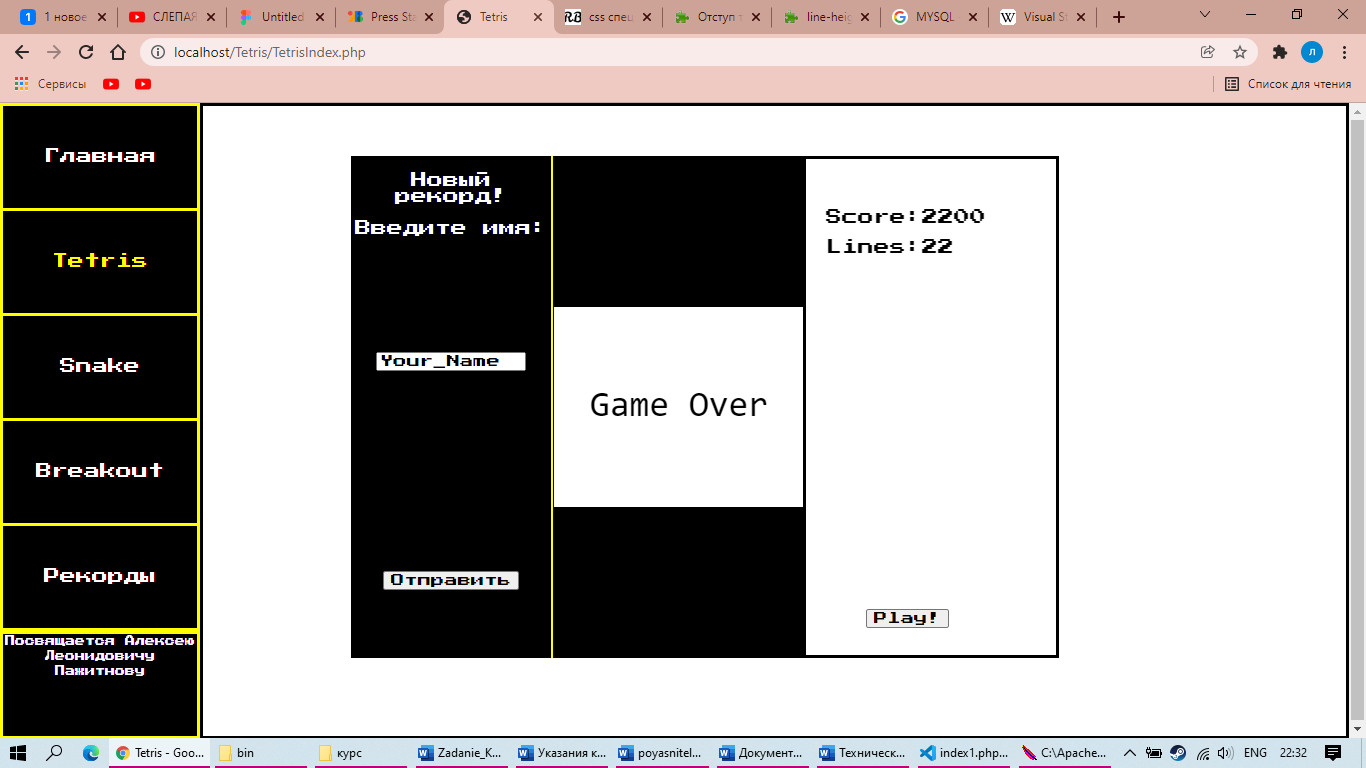


Рисунок 4.5 ­– Форма для отправки данных на сервер

На странице рекорды в левой части также расположено меню, с помощью которого можно быстро перемещаться по сайту, далее идет блок, содержащий в себе три таблицы, которые получают данные из базы и выводят наилучшие результаты посетителей за все время.

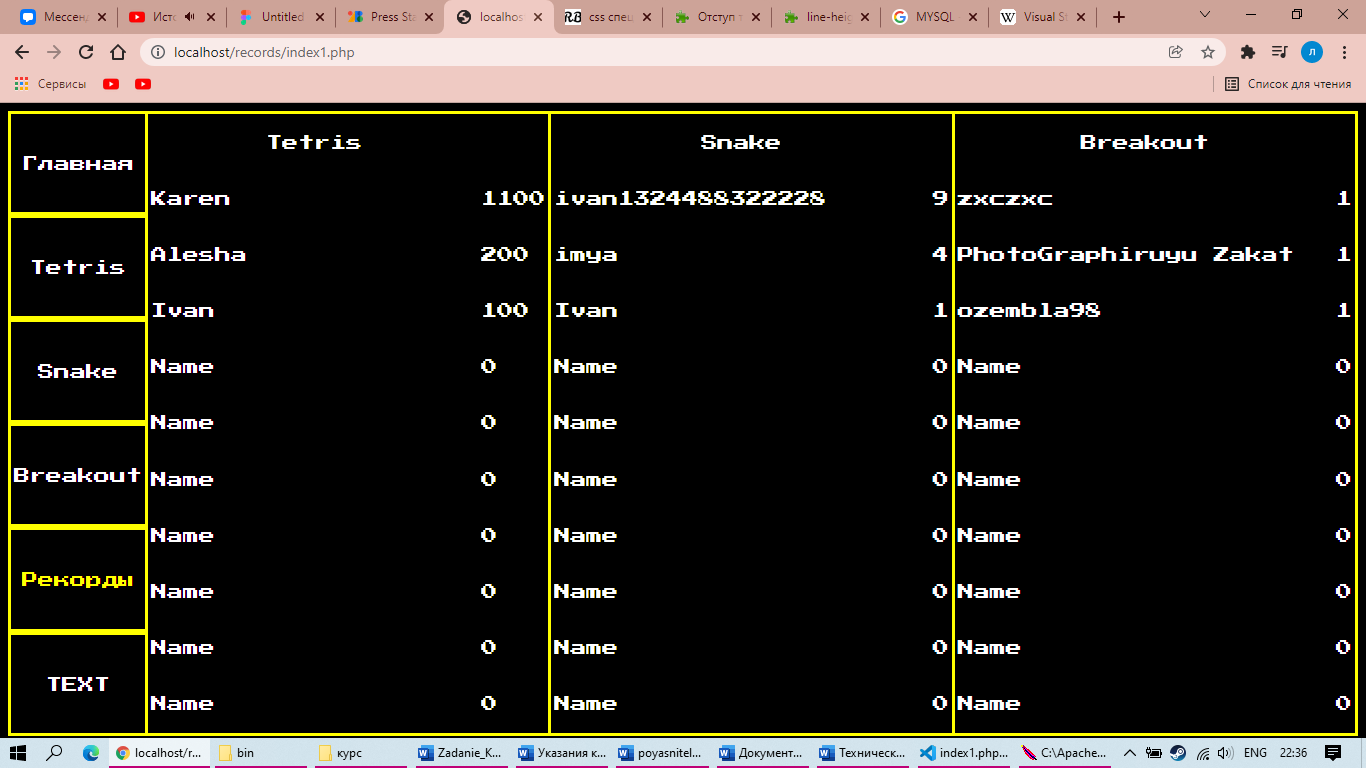


Рисунок 4.6 ­– Страница Рекорды

# 4.2 Кроссбраузерное тестирование

Одной из задач было создание сайта, который будет одинаково отображаться на разных браузерах.

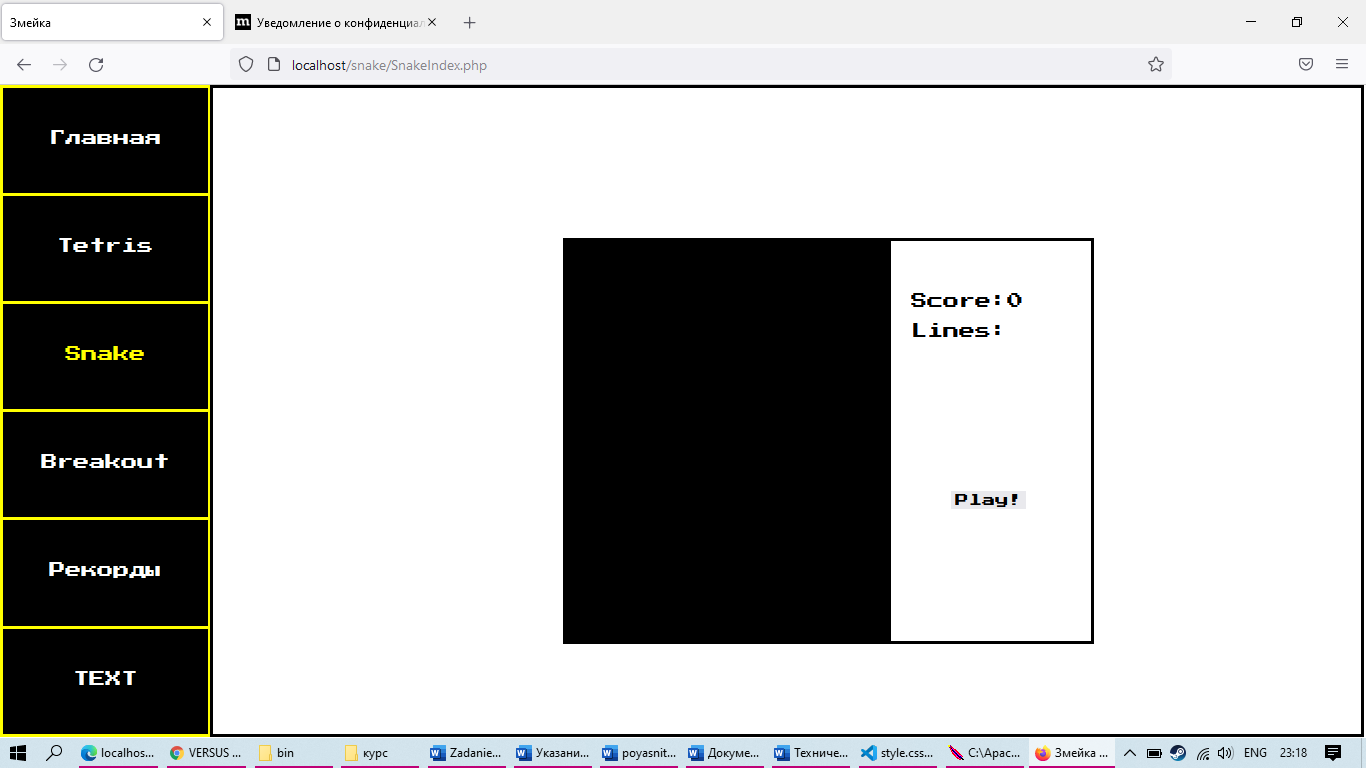


Рисунок 4.7 ­– Страница Snake в Mozilla Firefox



Рисунок 4.8 ­– Страница Breakout в Mozilla Firefox

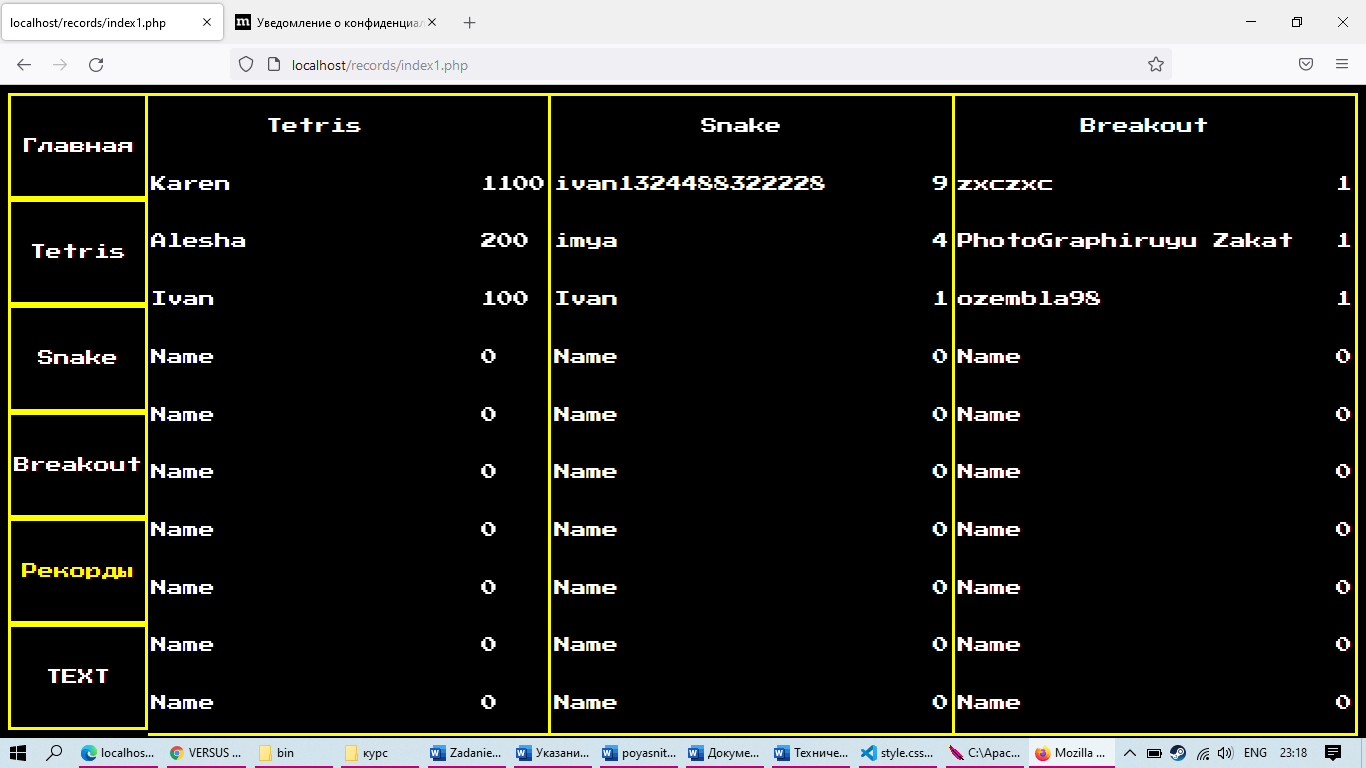


Рисунок 4.9 ­– Страница Рекорды в Mozilla Firefox



Рисунок 4.10 ­– Страница Breakout в Google Chrome

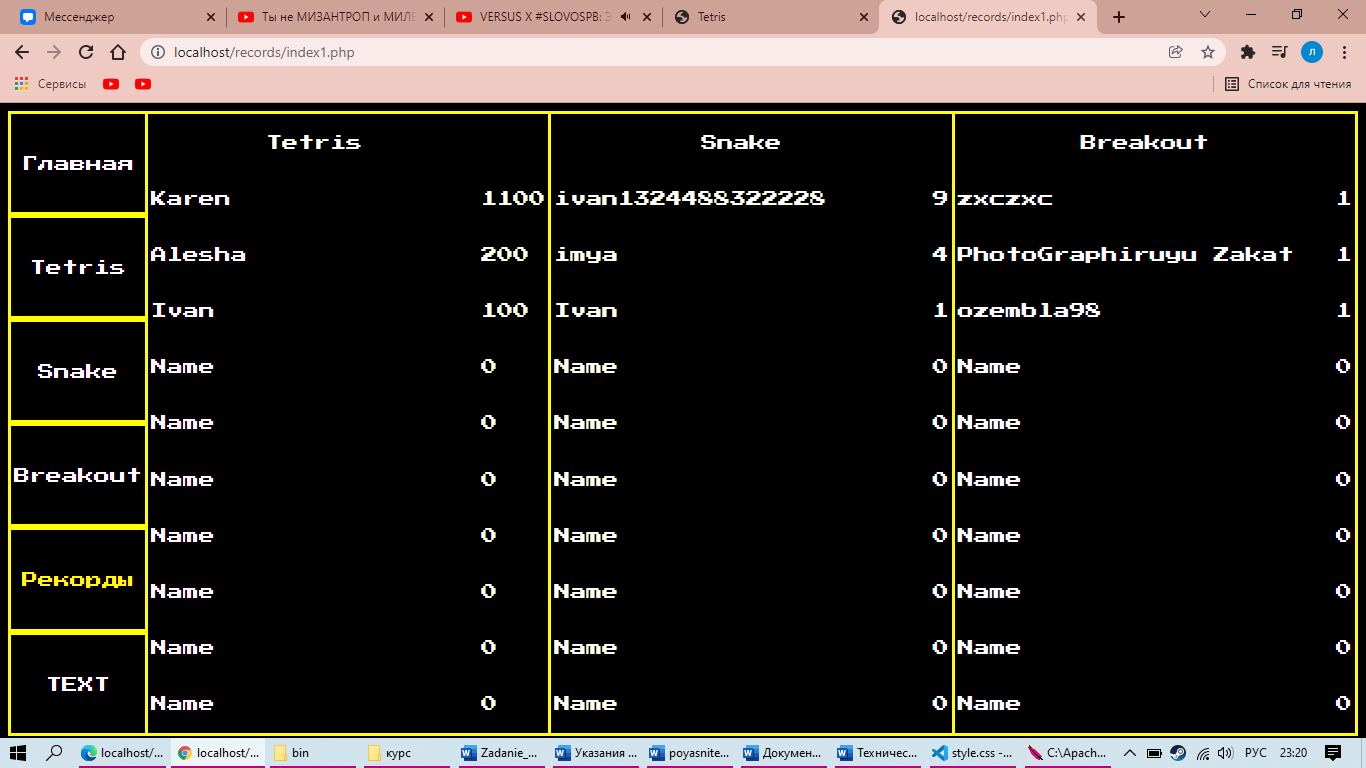


Рисунок 4.11 ­– Страница рекорды в Google Chrome

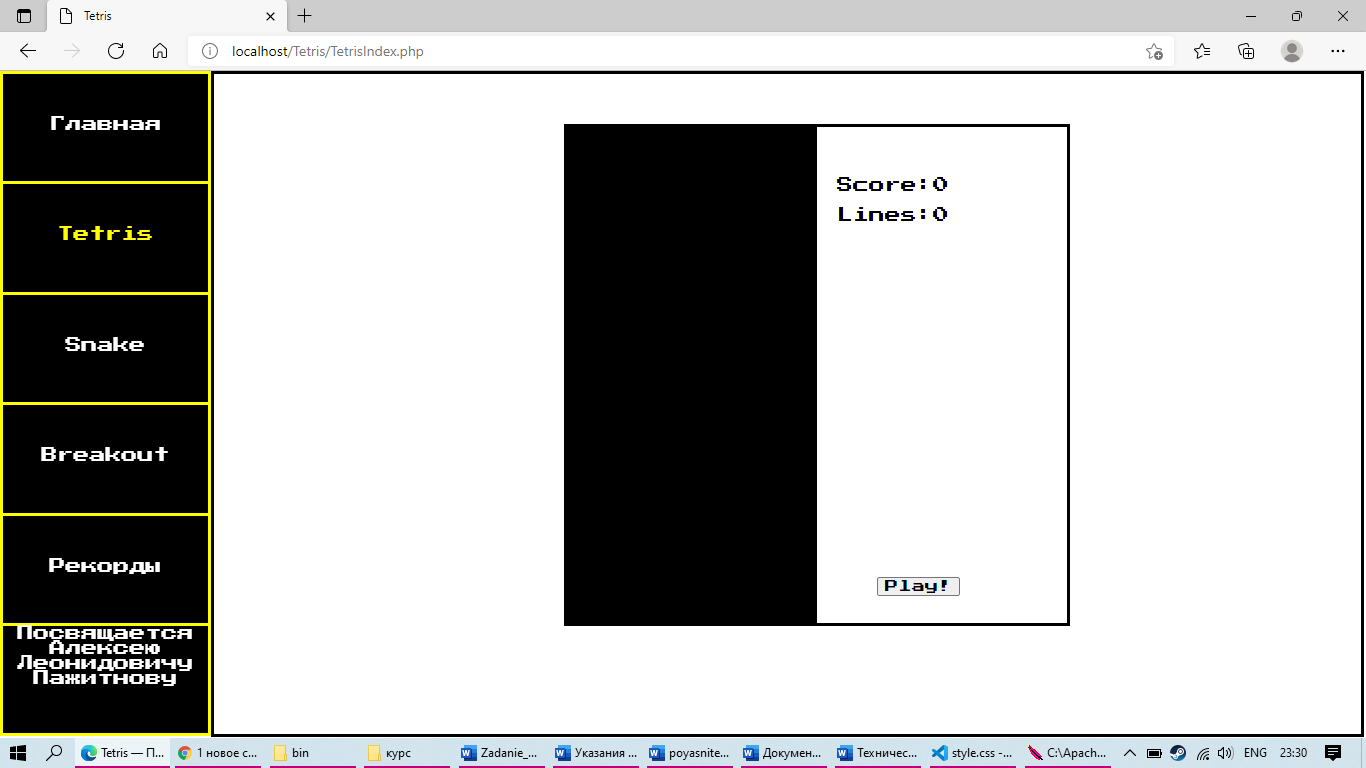


Рисунок 4.12 ­– Страница Tetris в Microsoft Edge

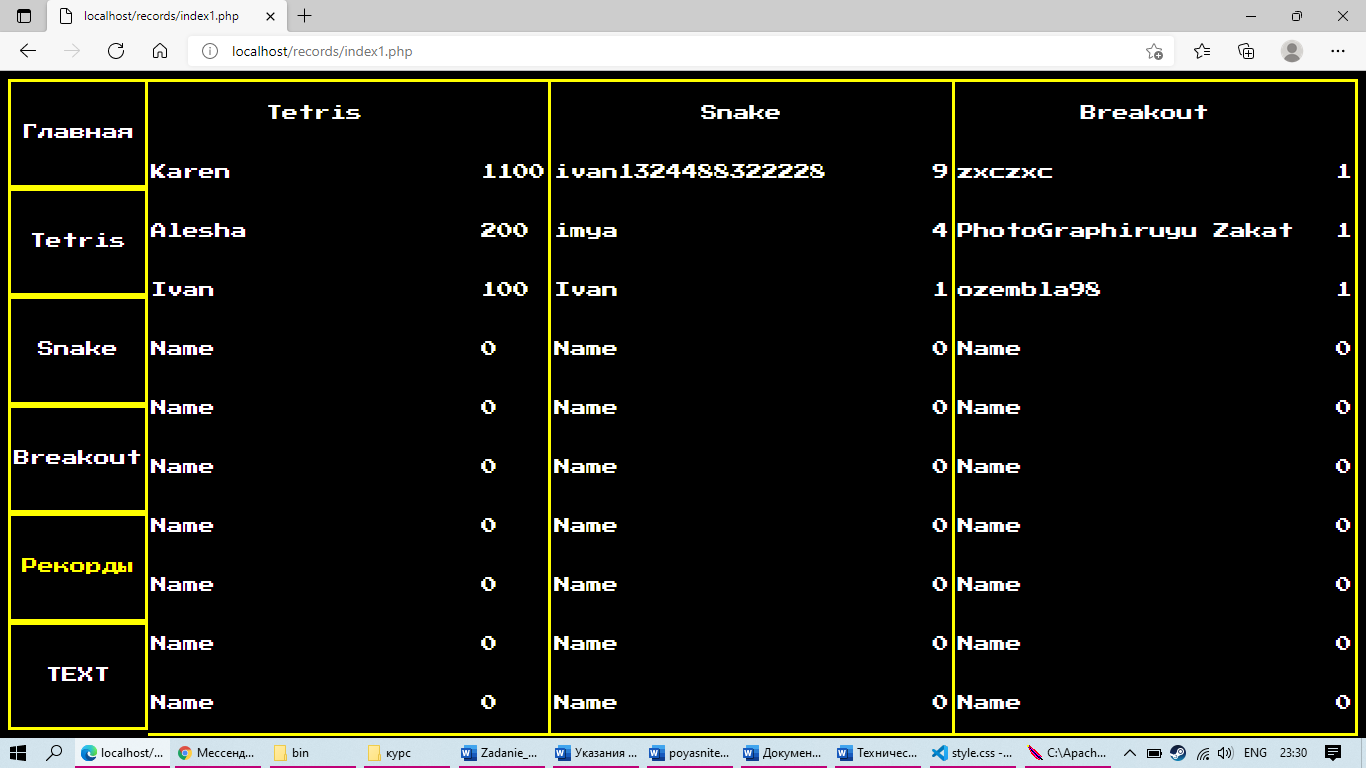


Рисунок 4.13 ­– Страница Рекорды в Microsoft Edge

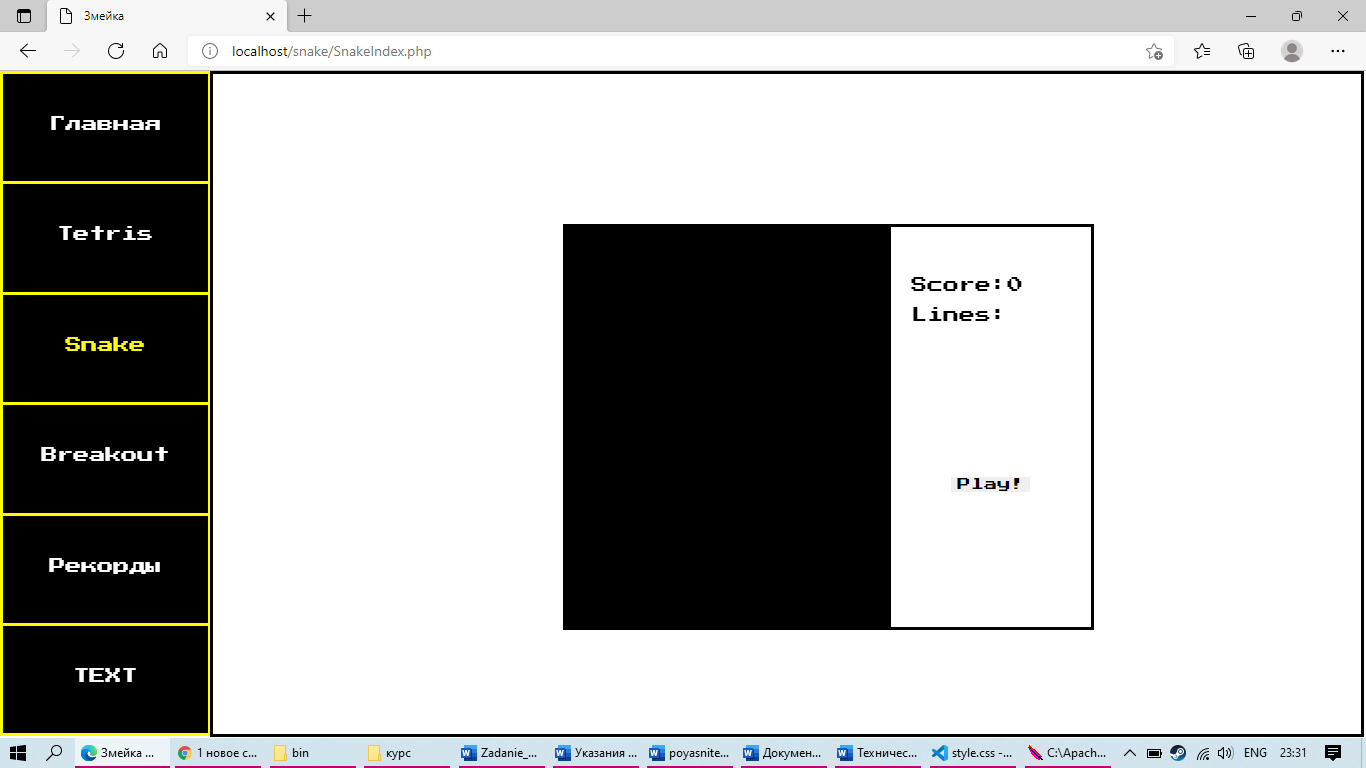


Рисунок 4.14 ­– Страница Snake в Microsoft Edge

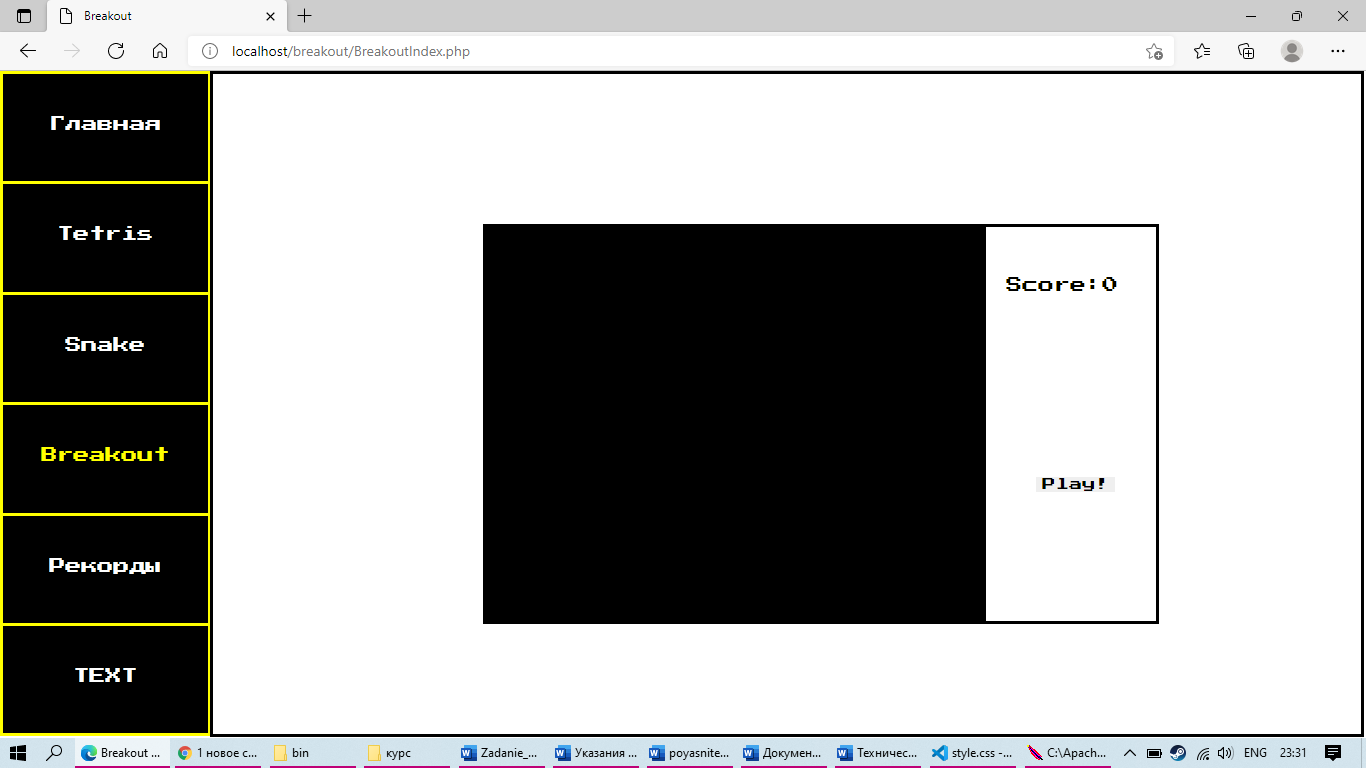


Рисунок 4.15 ­– Страница Breakout в Microsoft Edge

Как видно из скриншотов, данный сайт корректно отображается во всех актуальных браузерах, таких как Google Chrome, Microsoft Edge, Mozilla Firefox. Отображение непосредственно игровых элементов также не изменятся при изменении используемого браузера. Изменений во внешнем виде сайта не наблюдается.

# Заключение

В процессе создания курсового проекта были изучены такие технологии как CSS Flexbox, AJAX-запросы, PHP и СУБД MySQL. Были отточены умения верстки с использованием HTML и CSS. Были усвоены основы использования СУБД MySQL, скриптового языка PHP, использования связки PHP+MySQL. Также были получены базовые знания в сфере разработки игр с использованием элемента HTML5 canvas и JavaScript.

По завершению работы над программным продуктом был создан кроссбраузерный сайт, состоящий из 5 страниц. Были реализованы полноценные игры Tetris, Snake и Breakout. Была реализована база данных с использованием СУБД MySQL и возможность занести рекорд в таблицу, чтобы соревноваться с другими посетителями.

# Список использованных литературных источников

* + - 1. [Introduction to game development for the Web](https://developer.mozilla.org/en-US/docs/Games/Introduction) [Электронный ресурс] / Справочник Режим доступа: <https://developer.mozilla.org/en-US/docs/Games/Introduction>. – Дата доступа: 25.11.2020.
      2. Flexbox – изучение веб-разработки [Электронный ресурс] / Справочник Режим дступа:<https://developer.mozilla.org/ru/docs/Learn/CSS/CSS_layout/Flexbox> – Дата доступа: 9.11.2020.
      3. Введение в MySQL [Электронный ресурс] / Справочник Режим доступа: <https://metanit.com/sql/mysql/1.1.php> – Дата доступа: 10.12.2020.
      4. Руководство по PHP [Электронный ресурс] / Справочник Режим доступа: https://metanit.com/php/tutorial/ – Дата доступа: 11.12.2020.
      5. PHP и MySQL [Электронный ресурс] / Справочник Режим доступа: <https://metanit.com/php/mysql/> – Дата доступа: 14.11.2020.