**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное автономное образовательное

учреждение высшего образования

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

Инженерная школа информационных технологий и робототехники

Отделение информационных технологий

Направление информатика и вычислительная техника

Отчет по проектной работе в дисциплине

**«ТВОРЧЕСКИЙ ПРОЕКТ»**

**Игровое веб-приложение «Шахматы»**

Выполнили:

Студент группы 8В11 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Беликов Я.А.

Проверил:

Доцент ОИТ ИШИТР \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Е. А. Мыцко

Томск 2022

**Оглавление**

[Цель работы 1](#_Toc106592061)

[Задание 1](#_Toc106592062)

[1. Понятие Web-приложения 2](#_Toc106592063)

[2. Требования к игровому веб-приложению 4](#_Toc106592064)

[3. Библиотеки для разработки 5](#_Toc106592066)

[4. Язык разработки 6](#_Toc106592068)

[5. Эскиз страницы веб-приложения 9](#_Toc106592074)

[6. Результат вёрстки сайта 10](#_Toc106592075)

[Заключение 11](#_Toc106592077)

[Список литературы 12](#_Toc106592078)

[Приложение 13](#_Toc106592079)

# Цель работы

Создать игровое веб-приложение для настольной игры «Шахматы».

# Задание

Для достижения выполнения цели работы необходимо:

* Выбрать библиотеку и фреймворки, которые будут использоваться при создании приложения
* Выбрать языки разработки
* Разработать эскиз веб-страниц
* Реализовать вёрстку веб-страницы по имеющемуся эскизу

## Понятие Web-приложения

**Веб-приложение** представляет собой веб-сайт, на котором размещены страницы с частично либо полностью несформированным содержимым. Окончательное содержимое формируется только после того, как посетитель сайта запросит страницу с веб-сервера. В связи с тем что окончательное содержимое страницы зависит от запроса, созданного на основе действий посетителя, такая страница называется динамической.[1]

Любое веб-приложение представляет собой набор статических и динамических веб-страниц. **Статическая веб-страница** — это страница, которая всегда отображается перед пользователем в неизменном виде. Веб-сервер отправляет страницу по запросу веб-браузера без каких-либо изменений. В противоположность этому, сервер вносит изменения в **динамическую веб-страницу**перед отправкой ее браузеру. По причине того, что страница меняется, она называется динамической.

Например, можно создать страницу, на которой будут отображены результаты программы оздоровления. При этом некоторая информация (например, имя сотрудника и его результаты) будет определяться в момент запроса страницы сотрудником.

**Веб-сервер —** это программное обеспечение, которое предоставляет веб-страницы в ответ на запросы веб-браузеров. Обычно запрос страницы создается при щелчке ссылки на веб-странице, выборе закладки в браузере либо вводе URL-адреса в адресной строке браузера.

**Сервер приложений** **—** программное обеспечение, которое используется веб-сервером для обработки веб-страниц, содержащих серверные сценарии или теги. При запросе таких страниц веб-сервер сначала передает их серверу приложений для обработки, а затем отправляет клиентскому браузеру.

**База данных –** набор данных, хранящихся в таблицах. Каждая строка таблицы представляет собой одну запись, а каждый столбец — поле записи, как показано в следующем примере.

**Система управления базой данных –** СУБД, или система баз данных, представляет собой программное обеспечение, предназначенное для создания баз данных и управления ими.

**Реляционная база данных –** база данных с одной или несколькими таблицами, в которых есть общие данные.

**Статическая страница –** веб-страница, которая отправляется браузеру без изменения ее сервером приложений.

**Динамическая страница –** веб-страница, настроенная сервером приложений перед отправкой ее браузеру.

# Требования к игровому веб-приложению

Для правильного функционирования и положительного опыта со стороны пользователя веб-приложения необходимо обозначить ряд требований к разрабатываемому веб-приложений, которые будет необходимо учесть во время его разработки:

1. Игровое веб-приложение «Шахматы» должно предоставлять возможность пользователю играть в настольную игру шахматы, используя браузер
2. Игровое веб-приложение «Шахматы» должно работать в соответствии с существующими правилами игры, веб-версией которой оно является.
3. Игровое веб-приложение Шахматы должно позволять пользователю играть в партию шахмат с ограниченным временем.
4. Игровое веб-приложение Шахматы должно позволять пользователю играть против другого игрока.
5. Интерфейс игрового веб-приложения Шахматы должен быть интуитивно понятен, комфортен пользователю и напоминать реальную настольную версию шахмат.

# Целевая аудитория

Целевая аудитория игрового веб-приложения Шахматы очень обширная. В неё входят люди каждого пола и возраста.

Игровым веб-приложением Шахматы будут пользоваться люди, которые хотят играть в шахматы, но не имеют возможности делать это в настольном варианте, либо предпочитают играть в них на своём компьютере, это могут быть, как любители, так и те, кто занимается шахматами более серьёзно.

# Библиотеки для разработки

## Python Django

Django описывают как «веб-фреймворк для перфекционистов с дедлайнами». Его создали, чтобы переходить от прототипов к готовым сервисам как можно быстрее.

Фреймворк поможет разработать CRUD приложение под ключ. С Django не придется изобретать велосипед. Он работает из коробки и позволит сосредоточиться на бизнес-логике и продуктах для обычных людей.

**Преимущества**

* Приложения Django.
* Безопасный по умолчанию.
* REST Framework для создания API.
* GraphQL фреймворк для создания API.
* Надежность

# Язык разработки

## Язык программирования – Python

Python — высокоуровневый язык программирования общего назначения. Он имеет минималистичный синтаксис и направлен на повышение читаемости кода (и в целом на повышение производительности разработчика).

Основной упор в Python делается на скорости написания кода (а не на скорости выполнения кода, как например в языках С и C++). Поэтому в первую очередь Python удобно использовать там, где нужно быстро написать что-то работающее.

Все чаще Python используется для анализа данных, как в науке, так и коммерческой сфере. Этому способствует простота языка и большое разнообразие открытых библиотек.

Другая область применения, для которой хорош Питон — системное администрирование и DevOps.

**Нам python пригодится для:**

1. Создания шахматной доски
2. Прописывания искусственного интеллекта
3. Реагировать на действия пользователя, принимать информацию, работать с ней.

**Плюсы:**

1. Низкий порог вхождения
2. Язык широкого применения
3. Минималистичный дизайн
4. Кроссплатформенность
5. Открытый исходный код интерпретатора CPython
6. Наличие дружелюбного, отзывчивого сообщества
7. Огромное количество библиотек
8. Входит в поставку большинства дистрибутивов Linux

## Среда разработки – Sublime Text 3

Sublime Text 3 — это кроссплатформенный текстовый редактор, разработанный для пользователей, которые ищут эффективный, но минималистский инструмент для редактирования кода. Редактор, конечно же, прост, в котором отсутствуют панели инструментов или диалоговые окна.

## В чем преимущества редактора Sublime Text 3

Мы выделили несколько полезных качеств редактора кода Sublime Text 3:

**1. Скорость и низкие требования к ресурсам компьютера**  
Редактор действительно быстро работает даже на “слабом” железе.  
  
**2. Работа в популярных операционных системах**  
Вы можете запустить редактор кода Sublime Text 3 в операционных системах Windows, Linux и Mac OS.  
  
**3. Сторонние плагины и дополнения**  
Если вас не устраивают базовые возможности Sublime Text, вы можете установить различные дополнения, начиная от красивых тем оформления, заканчивая удобными инструментами, например, Emmet.

Программирование и вёрстка – HTML и CSS**3**

Один из самых важных этапов работы при создании любого сайта – это его вёрстка. Для вёрстки веб-страницы игрового веб-приложения я буду использовать язык гипертекстовой разметки HTML и язык таблиц стилей CSS.

## Работа с базой данных – Open Server Panel

**Open Server Panel** — это портативная программная среда, созданная специально для веб-разработчиков с учётом их рекомендаций и пожеланий.

Данный программный комплекс включает в себя тщательно подобранный набор серверного программного обеспечения, а так же невероятно удобную и продуманную управляющую утилиту, которая обладает мощными возможностями по администрированию и настройке всех доступных компонентов.

**OSPanel** широко используется с целью разработки, отладки и тестирования веб-проектов, а так же для предоставления веб-сервисов в локальных сетях.[7]

В качестве системы управления базами данных я выбрал MySQL, т.к. уже имел опыт работы с ним, а также она имеет следующие преимущества:

* Простота. MySQL легко устанавливается, имеет понятный интерфейс, а разнообразие плагинов и дополнительных приложений упрощает работу с БД.
* Функционал. Включает в себя практически весь необходимый набор инструментов, который может пригодиться при разработке любого проекта.
* Безопасность. Многие системы безопасности уже встроены и работают по умолчанию.
* Масштабируемость. Может использоваться в работе как с малым, так и с большим объемом данных.
* Скорость. Является одной из самых быстрых среди имеющихся на современном рынке.[8]

# Эскиз страницы веб-приложения

При создании эскиза, будущего проекта, я использовал онлайн сервис figma, в силу его простоты и удобства.



Рис 1. Эскиз веб-приложения «Шахматы»

Выше представлен эскиз, с игровым полем, таймером для каждого игрока, с окном для записывания предыдущих ходов и новостным окном.

# Результат вёрстки сайта

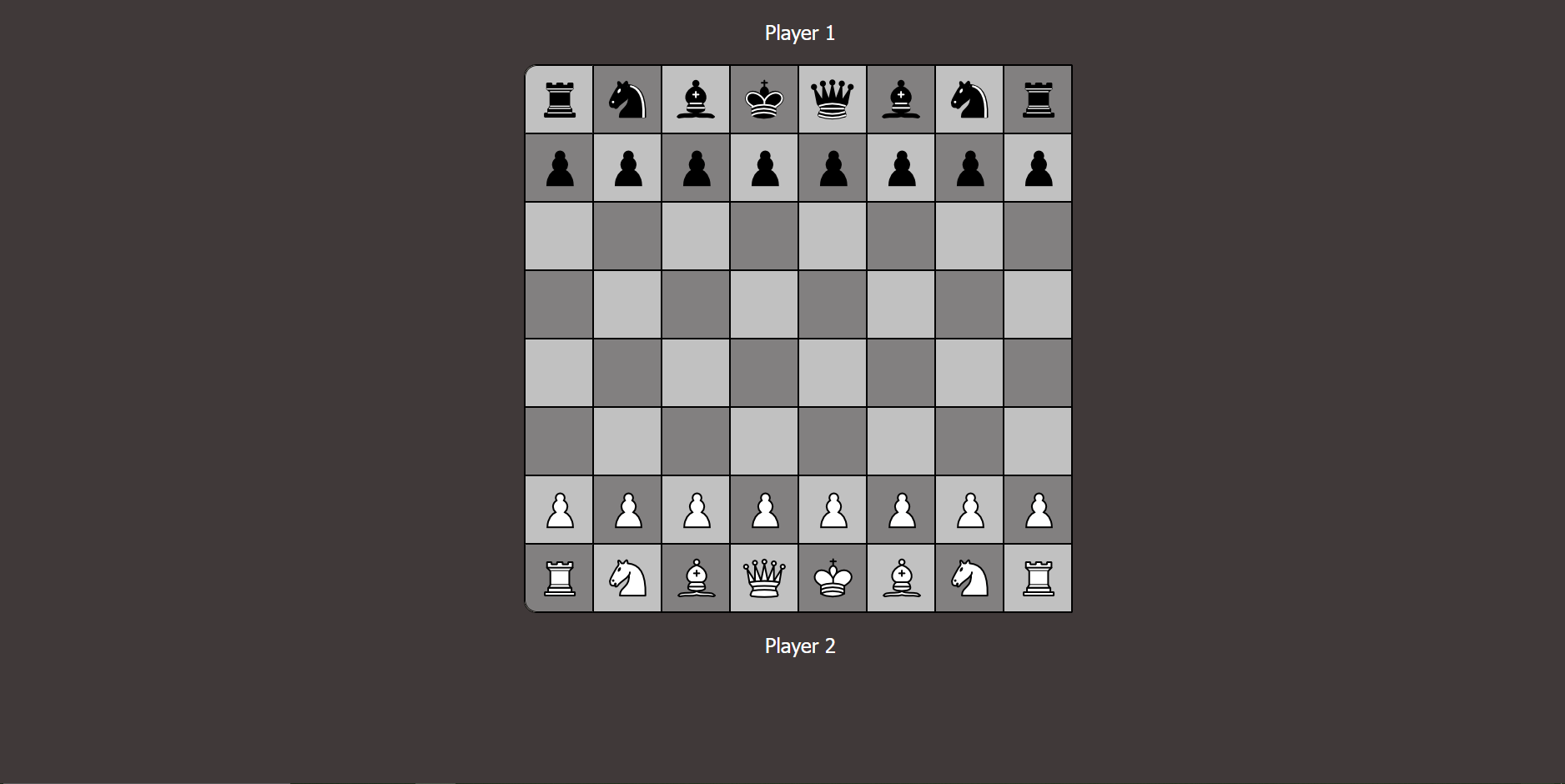


Рис.2 Результат вёрстки сайта

На рисунке два, представлен проект, в текущем его состоянии, на данный момент множество функций не реализовано.

# Заключение

При выполнении данной задачи, я изучил, новые языки разработки (HTML, CCS). Создал эскиз для своего проекта, а также сверстал начальную страницу.

Обозначим планы на следующий этап выполнения работы:

* Разработать алгоритм игры в шахматы.
* Внедрить искусственный интеллект, для возможность игры против него.
* Реализовать таймеры для игроков, новостное окно, окно с историей ходов.
* Интегрировать в сайт язык программирования Python и библиотеку Python Django.

# Список литературы

1. Sublime Text – краткое руководство [Электронный ресурс]. – URL: <https://coderlessons.com/tutorials/raznoe/vyuchit-vozvyshennyi-tekst/sublime-text-kratkoe-rukovodstvo>
2. Pygame – библиотека python[Электронный ресурс]. – URL: <https://www.pygame.org>
3. Python Django – The web framework for perfectionists with deadlines [Электронный ресурс]. – URL: <https://www.djangoproject.com>
4. HTML и CSS - HTML Academy: интерактивные онлайн курсы по HTML, CSS и JavaScript [Электронный ресурс]. – URL: <https://htmlacademy.ru>
5. Python для Web[Электронный ресурс] **–** URL: <https://pythonworld.ru/web>
6. Creating a Chess Engine in Python [Электронный ресурс] – URL: https://www.youtube.com/watch?v=EnYui0e73Rs&list=PLBwF487qi8MGU81nDGaeNE1EnNEPYWKY\_&ab\_channel=EddieSharic

# Приложение

# HTML-код:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

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

<title>Chess game</title>

<style>

html { overflow: hidden; }

div { height: 2000px; }

</style>

</head>

<body>

<p>Player 1</p>

<div class = "chess\_board">

<div class = "row" >

<div class="white\_square"id="round1"><img src="images/bR.png"></div>

<div class="black\_square"><img src="images/bP.png"></div>

<div class="white\_square"></div>

<div class="black\_square"></div>

<div class="white\_square"></div>

<div class="black\_square"></div>

<div class="white\_square"><img src="images/wP.png"></div>

<div class="black\_square" id="round2"><img src="images/wR.png"></div>

</div>

<div class = "row" >

<div class="black\_square"><img src="images/bN.png"></div>

<div class="white\_square"><img src="images/bP.png"></div>

<div class="black\_square"></div>

<div class="white\_square"></div>

<div class="black\_square"></div>

<div class="white\_square"></div>

<div class="black\_square"><img src="images/wP.png"></div>

<div class="white\_square"><img src="images/wN.png"></div>

</div>

<div class = "row" >

<div class="white\_square"><img src="images/bB.png"></div>

<div class="black\_square"><img src="images/bP.png"></div>

<div class="white\_square"></div>

<div class="black\_square"></div>

<div class="white\_square"></div>

<div class="black\_square"></div>

<div class="white\_square"><img src="images/wP.png"></div>

<div class="black\_square"><img src="images/wB.png"></div>

</div>

<div class = "row" >

<div class="black\_square"><img src="images/bK.png"></div>

<div class="white\_square"><img src="images/bP.png"></div>

<div class="black\_square"></div>

<div class="white\_square"></div>

<div class="black\_square"></div>

<div class="white\_square"></div>

<div class="black\_square"><img src="images/wP.png"></div>

<div class="white\_square"><img src="images/wQ.png"></div>

</div>

<div class = "row" >

<div class="white\_square"><img src="images/bQ.png"></div>

<div class="black\_square"><img src="images/bP.png"></div>

<div class="white\_square"></div>

<div class="black\_square"></div>

<div class="white\_square"></div>

<div class="black\_square"></div>

<div class="white\_square"><img src="images/wP.png"></div>

<div class="black\_square"><img src="images/wK.png"></div>

</div>

<div class = "row" >

<div class="black\_square"><img src="images/bB.png"></div>

<div class="white\_square"><img src="images/bP.png"></div>

<div class="black\_square"></div>

<div class="white\_square"></div>

<div class="black\_square"></div>

<div class="white\_square"></div>

<div class="black\_square"><img src="images/wP.png"></div>

<div class="white\_square"><img src="images/wB.png"></div>

</div>

<div class = "row" >

<div class="white\_square"><img src="images/bN.png"></div>

<div class="black\_square"><img src="images/bP.png"></div>

<div class="white\_square"></div>

<div class="black\_square"></div>

<div class="white\_square"></div>

<div class="black\_square"></div>

<div class="white\_square"><img src="images/wP.png"></div>

<div class="black\_square"><img src="images/wN.png"></div>

</div>

<div class = "row" >

<div class="black\_square"><img src="images/bR.png"></div>

<div class="white\_square"><img src="images/bP.png"></div>

<div class="black\_square"></div>

<div class="white\_square"></div>

<div class="black\_square"></div>

<div class="white\_square"></div>

<div class="black\_square"><img src="images/wP.png"></div>

<div class="white\_square"><img src="images/wR.png"></div>

</div>

</div>

<p>Player 2</p>

</body>

</html>

# CSS-код:

body{

width: 1920px;

height: 1080px;

background-color: #403939;

}

.chess\_board{

display: flex;

flex-wrap: wrap;

width: 656px;

height: 656px;

border: 1px solid #000000;

background-color: #F9F0E1;

border-radius: 15px 0 0 15px;

margin: auto;

}

.black\_square{

width: 80px;

height: 80px;

background-color: #828080;

border: 1px solid #000000;

text-align: center;

}

.white\_square{

width: 80px;

height: 80px;

background-color: #C1C1C1;

border: 1px solid #000000;

text-align: center;

}

p{

color: white;

font-size: 24px;

font-family: Tahoma;

margin-left: 5px;

text-align: center;

}

#round1{

border-radius: 15px 0 0 0;

}

#round2{

border-radius: 0 0 0 15px;

}

img {

margin-top: 10px;

}