**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 8В12 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ М. В. Назаров

Проверил:

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

Оценка :

Томск 2022

**Оглавление**

[Цель работы 3](#_Toc106731181)

[1. Понятие Web-приложения 4](#_Toc106731182)

[2. Требования к игровому веб-приложению 6](#_Toc106731183)

[3. Обзор технологий и фреймворков 7](#_Toc106731184)

[3.1 Язык программирования JavaScript 7](#_Toc106731185)

[3.2 Фреймворки для JavaScript 8](#_Toc106731186)

[3.3 Среда разработки – Sublime Text 9](#_Toc106731187)

[4. Эскиз страницы веб-приложения 11](#_Toc106731188)

[5. Диаграмма вариантов использования 12](#_Toc106731189)

[Заключение 13](#_Toc106731190)

[Список литературы 14](#_Toc106731191)

# Цель работы

Создать игровое веб-приложение настольной игры «Шашки».

**Задание**

Для достижения выполнения цели работы необходимо:

* Выбрать фреймворк, который будет использоваться при создании приложения
* Выбрать языки разработки
* Разработать эскиз веб-страниц
* Реализовать вёрстку веб-страницы по имеющимся эскизам

## 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.

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. Широкое распространение получили системы управления баз данных Microsoft Access, Oracle 9i и MySQL.

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

**Целевая аудитория**

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

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

# Обзор технологий и фреймворков

* 1. **Язык программирования JavaScript**

**JavaScript** – это прототипно-ориентированный, мультипарадигменный язык с динамической типизацией, который поддерживает объектно-ориентированный, императивный и декларативный (например, функциональное программирование) стили программирования.

Современный JavaScript – это «безопасный» язык программирования. Он не предоставляет низкоуровневый доступ к памяти или процессору, потому что изначально был создан для браузеров, не требующих этого.

В браузере для JavaScript доступно всё, что связано с манипулированием веб-страницами, взаимодействием с пользователем и веб-сервером.

В браузере JavaScript может:

* Добавлять новый HTML-код на страницу, изменять существующее содержимое, модифицировать стили;
* Реагировать на действия пользователя, щелчки мыши, перемещения указателя, нажатия клавиш;
* Отправлять сетевые запросы на удалённые сервера, скачивать и загружать файлы (технологии AJAX и COMET);
* Получать и устанавливать куки, задавать вопросы посетителю, показывать сообщения;
* Запоминать данные на стороне клиента («local storage»);

JavaScript не может:

* JavaScript на веб-странице не может читать/записывать произвольные файлы на жёстком диске, копировать их или запускать программы. Он не имеет прямого доступа к системным функциям ОС;
* Различные окна/вкладки не знают друг о друге. Иногда одно окно, используя JavaScript, открывает другое окно. Но даже в этом случае JavaScript с одной страницы не имеет доступа к другой, если они пришли с разных сайтов (с другого домена, протокола или порта);
* JavaScript может легко взаимодействовать с сервером, с которого пришла текущая страница. Но его способность получать данные с других сайтов/доменов ограничена. Хотя это возможно в принципе, для чего требуется явное согласие (выраженное в заголовках HTTP) с удалённой стороной. Опять же, это ограничение безопасности.

**3.2 Фреймворки для JavaScript**

**Angular** — веб-фреймворк, который позволяет JavaScript интегрироваться с HTML и CSS. С его помощью построены более 400 тысяч сайтов по всему. С ним можно разрабатывать нативные и веб-приложения для ПК и мобильных устройств. И подходит для корпоративного софта. Его используют Google, Microsoft и YouTube.

Плюсы:

* помогает создавать прогрессивные приложения (PWA);
* удобно манипулировать DOM-элементами;
* высокая скорость и производительность;
* крутая кривая обучения;
* встроенный механизм внедрения зависимостей;
* поддержка Google и мощная экосистема.

Минусы:

* Сложный синтаксис, который исходит от первой версии Angular. Тем не менее, Angular 5 использует TypeScript 2.4, который изучить не так уж сложно;
* Проблемы с миграцией, которые могут возникнуть при переходе от старой версии к новой.

**Vue.js** — прогрессивный фреймворк, который можно интегрировать с уже готовыми проектами и библиотеками JS. В 2020 году Vue стал самым звездным фреймворком на GitHub, обойдя Angular и React. К тому же за последние 3-4 года интерес к фреймворку вырос на 18-20%. Его используют Stack Overflow, GitLab, Adobe.

Плюсы:

* высокая степень настраиваемости;
* легко учить;
* поддержка CSS переходов и анимации;
* гибкость и модульность.

Минусы:

* Нехватка ресурсов;
* Риск чрезмерной гибкости. Иногда у Vue.js могут возникать проблемы при интеграции в огромные проекты, а опыта о возможных решениях до сих пор нет.

**React** — JavaScript-библиотека с открытым исходным кодом для создания пользовательских интерфейсов. Она позволяет создавать компоненты интерфейса для мобильного и ПК-софта. Ещё его рекомендуют использовать для разработки SPA и корпоративных приложений. На GitHub React — это второй по популярности фреймворк. Его используют Facebook, Instagram, WhatsApp.

Плюсы:

* лёгкое объединение JavaScript и HTML;
* просто разрабатывать динамические веб-приложения;
* простая отладка;
* поддержка сообщества.

Минусы:

* Нехватка официальной документации;
* Долгое время для освоения.

**PhaserJS** — кросс-браузерный HTML5 фреймворк с открытым исходным кодом для создания браузерных игр с использованием WebGL и Canvas. В отличии от других фреймворков, Phaser в первую очередь целится на мобильные платформы и оптимизирован под них.

Плюсы

* Большая популярность
* Игровой фреймворк
* Большое активное сообщество
* Высокая произовдительность
* Использование WebGL и Canvas

Минусы

* Используется для создания 2D игр
* Не подходит для реализации сложных проектов

Я считаю, что лучшим решением для моих задач является PhaserJS. Потому что, это - игровой фреймворк, который призван помочь разработать кросс-браузерные HTML5 игры в короткие сроки.

3.3 Среда разработки – Sublime Text

Sublime Text Editor – это полнофункциональный текстовый редактор для редактирования локальных файлов или базы кода. Он включает в себя различные функции для редактирования базы кода, которая помогает разработчикам отслеживать изменения.

# Эскиз страницы веб-приложения

Для создания эскиза я использовал веб-приложение Figma, т.к. оно простое для понимания и использования, также я уже имел опыт работы с ним.

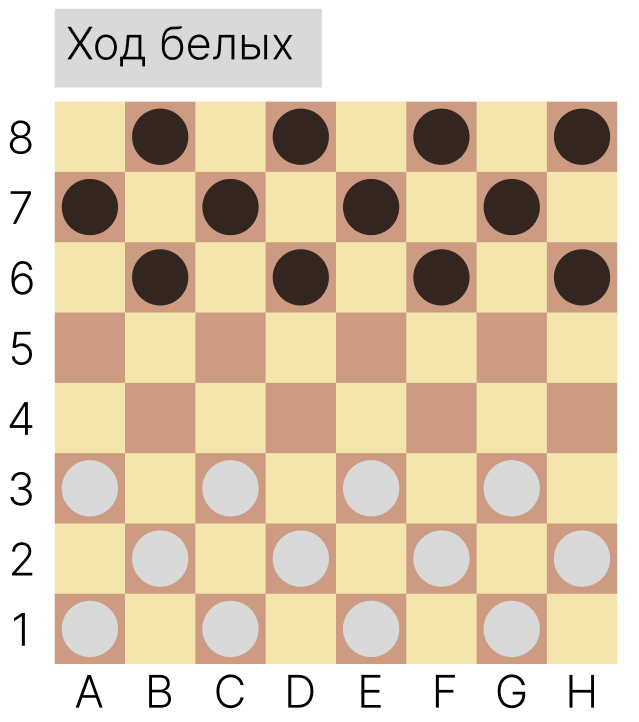


Рис 1. Эскиз шахматной доски

На рис. 1 представлен эскиз веб-приложения Шашки. На эскизе видно, что приложение имеет поле в виде шахматной доски, на которой расположено 12 шашек белого цвета и 12 шашек черного цвета. Также на эскизе представлена в левом верхнем углу очередность хода.

# Диаграмма вариантов использования

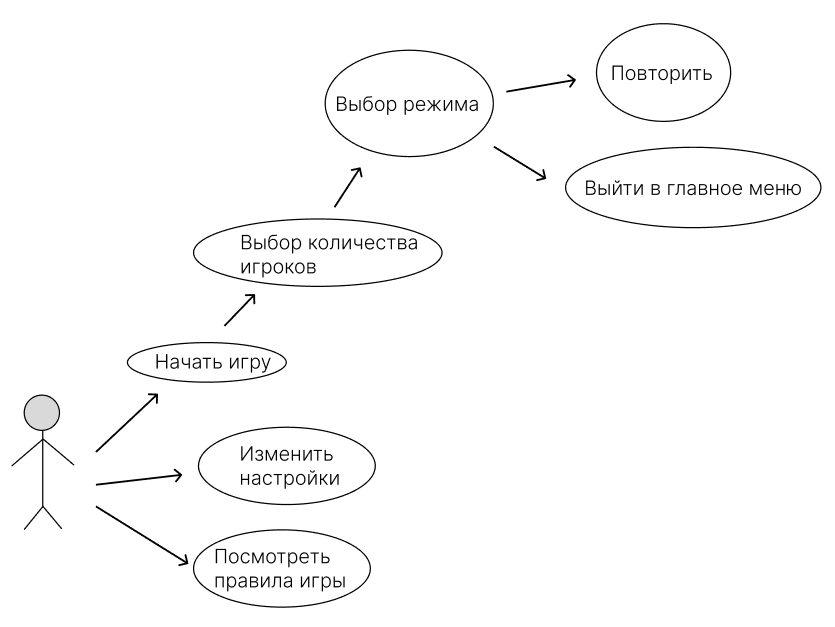


Рис 2. Диаграмма вариантов использования сайта

Пользователю предлагается начать игру, выбрав один из режимов, узнать правила или изменить базовые настройки игры. После начала игры пользователь может выйти в главное меня. А после игры пользователь может начать ее заново.

# Заключение

За текущий семестр удалось установить основные задачи проекта, провести обзор фреймворков, рассмотреть их преимущества и недостатки, изучить технологии разработки веб-приложений. Также сформированы требования к проекту и обозначена целевая аудитория. Разработан начальный вариант внешнего вида и функционала игрового веб-приложения. Выбран язык разработки, подходящий фреймворк и среда разработки.

В следующем семестре планируется:

* Верстка веб-страниц
* Реализация front-end функций работы веб-сайта
* Реализовать алгоритм игры в шашки
* Реализовать серверную часть проекта

# Список литературы

1. Общие сведения о веб-приложениях [Электронный ресурс]. - URL: https://helpx.adobe.com/ru/dreamweaver/using/web-applications.html#how\_a\_web\_application\_works (Дата обращения: 20.06.2022).
2. JavaScript [Электронный ресурс]. - URL: https://developer.mozilla.org/ru/docs/Web/JavaScript (Дата обращения: 20.06.2022)
3. Введение в JavaScript [Электронный ресурс]. – URL: https://learn.javascript.ru/intro (Дата обращения 20.06.2022)
4. 10 JavaScript-фреймворков, которые стоит выучить в 2021 году [Электронный ресурс]. – URL: https://tproger.ru/articles/10-javascript-frejmvorkov-kotorye-stoit-vyuchit-v-2021-godu/#:~:text=Angular,для%20ПК%20и%20мобильных%20устройств. (Дата обращения 21.06.2022)
5. ReactJS, Angular5 и Vue.js — какой фреймворк выбрать в 2018 году? [Электронный ресурс]. – URL: https://nuancesprog.ru/p/1490/ (Дата обращения 21.06.2022)
6. Sublime Text – Краткое руководство [Электронный ресурс]. – URL: https://coderlessons.com/tutorials/raznoe/vyuchit-vozvyshennyi-tekst/sublime-text-kratkoe-rukovodstvo (Дата обращения 21.06.2022)