# **Поисковик для покупки книг по лучшим ценам**

# **WannaReadSome**

# **Актуальность**

* Активное развитие интернет технологий.
* Люди, любящие читать, часто используют интернет для поиска книг.
* Удобство пользования интернет поисковиком книг.

# **Цель**

Создание удобного веб-сайта, который способен помочь пользователю найти книгу и предоставить ее цену и сайт где она продается. При этом пользователю не надо долго лазить по интернету, ведь наш сайт оперативно предоставит все доступные варианты на одной странице и поможет найти подходящий.

**Задачи**

1. Наличие удобного современного дизайна сайта;
2. Наличие обширной базы данных;
3. Возможность поиска книг по различным отличительным признакам (имя автора, издательство, название, жанр);
4. Возможность сравнивать одни и те же книги в разных интернет-магазинах по их цене или рейтингу покупателей;
5. Провести обзор технологий и фреймворков, которые мы можем использовать при создании сайта;
6. Определить целевую аудиторию и требования к разрабатываемой системе;
7. Разработать диаграмму вариантов использования;
8. Разработать эскизы веб-страниц;

**Целевая аудитория сайта:**

К целевой аудитории нашего сайта мы относим всех читающих людей. Для более точного выяснения целевой аудитории мы взяли результаты опроса Всероссийского центра изучения общественного мнения (ВЦИОМ), который проводился в 2019 году. По данным этого опроса книги читают 53 процента россиян, на основе этого мы можем сделать вывод, что разрабатываемый нами сайт может быть полезен примерно каждому второму россиянину, которые используют современные средства поиска информации.

**Требования к разрабатываемой системе**

1. Наличие удобного современного дизайна сайта без лишних элементов;
2. Наличие обширной базы данных, содержащей нужную для пользователей информация по книгам;
3. Возможность поиска книг по различным отличительным признакам (имя автора, издательство, название, жанр);
4. Возможность сравнивать одни и те же книги в разных интернет-магазинах по их цене или рейтингу покупателей;

В нашем проекте мы собираемся создать сайт, целью которого будет поиск книг. Для создания сайта нам потребуется разобраться, как мы это будем делать.

А в этом нам помогут фреймворки и библиотеки JavaScript, но для начала надо разобраться, что из себя представляет JavaScript и также что такое фреймворки и библиотеки.

**Обзор технологий**

Для начала нам стоит выбрать язык программирования, который мы будем в основном использовать, для этого мы решили сделать обзор самых популярных языков программирования, используемых для создания веб-сайтов. Обозревать мы будем PHP, JavaScript, CSS и HTML.

**PHP**

Это скриптовый серверный язык образованный от английского «PHP: Hypertext Preprocessor» - препроцессор гипертекста, служит для создания веб приложений. Язык не создает сайт, а лишь решает вычислительные и логические операции, выглядит это как тот же HTML только с внедрением PHP кода.  
PHP генерирует HTML код в нужном месте сайта, сам код отвечает за логику работы сайта, его функциональность и связью с базой данных. Язык PHP получил широкое распространение в среде веб разработки из-за своей простоты, гибкости и производительности.

**JavaScript**

JavaScript – это «клиентский» скриптовый интерпретируемый браузером, язык программирования. Его основная задача создания интерактивных веб-страниц анимационных эффектов и работы с элементами DOM. Ни один современный сайт не мыслим без JavaScript, так как с него может происходить полноценная загрузка страниц, так называемые AJAX сайт, что ускоряет работу сайта особенно если он обладает большим количеством данных и сложным функционалом.

*(AJAX — подход к построению интерактивных пользовательских интерфейсов веб-приложений, заключающийся в «фоновом» обмене данными браузера с веб-сервером.*

*Объектная Модель Документа (DOM) – это программный интерфейс (API) для HTML и XML документов. DOM предоставляет структурированное представление документа и определяет то, как эта структура может быть доступна из программ, которые могут изменять содержимое, стиль и структуру документа.)*

**CSS**

CSS «*Cascading Style Sheets*» в переводе с английского «каскадные таблицы стилей» можно назвать языком таблиц стилей, используемый для описания представления документа, написанного на языке разметки, таком как HTML.

**HTML**

Это язык гипертекстовой 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.  
Элементы HTML являются строительными блоками HTML страниц. Как правило, сайты созданные с применением только HTML это небольшие статические страницы которые натребуют частых изменений и динамического наполнения.

**Выбор языка программиования**

Мы решили использовать язык программирования JavaScript так как ни один современный браузер не обходится без поддержки JavaScript, а с использованием написанных на JavaScript плагинов и скриптов несложно разобраться. Так же сам JavaScript имеет полезные функциональные настройки и он может осуществлять взаимодействие с приложением даже через текстовые редакторы – Microsoft Office и Open Office.

**Обзор фреймоворков и библиотек JavaScript**

**Библиотека** является многоразовой частью кода, которая предлагает определенные функции. Это набор функций, объектов и классов, которые вы можете использовать в своем приложении. Вы можете вызвать библиотечную функцию и передать ей некоторые параметры, и библиотека выполнит ее и вернет вам элемент управления.

**Фреймворк** же это готовая модель в IT, заготовка, шаблон для программной платформы, на основе которого можно дописать собственный код.

В основном фреймворки используются из-за того, что в них есть предварительно написанный код для использования в стандартных функциях и задачах программирования.

У использования фреймворков есть ряд плюсов. Во-первых, с точки зрения бизнеса разработка на фреймворке экономически эффективнее и качественнее по результату, нежели написание проекта без использования каких-либо платформ.

Во-вторых, приложения на фреймворках значительно проще сопровождать и дорабатывать, ведь их структура понятна всем разработчикам, основывающимся на этой платформе.

Фреймворки эффективны в проектах со сложной бизнес-логикой и высокими требованиями к скорости работы, надёжности и безопасности. В простых, типовых проектах лучше использовать другие решения.

Существенный минус фреймворков — открытый код большинства из них. Это значит, что нужно отдельно решать вопросы безопасности проекта. Также фреймворки обычно не схожи друг с другом, поэтому при переходе на новый фреймворк надо заново осваивать среду разработки.

Фреймворки дают стабильность и удобство разработки, но своей архитектурой они накладывают ограничения для программистов.

«Фреймворк» отличается от понятия библиотеки тем, что библиотека может быть использована в программном продукте просто как набор подпрограмм близкой функциональности, не влияя на архитектуру программного продукта. В то время как «фреймворк» диктует правила построения архитектуры приложения.

Также, в отличие от библиотеки, которая объединяет в себе набор близкой функциональности, — «фреймворк» может содержать в себе большое число разных по тематике библиотек.

Ну вот теперь, когда мы узнали, что же это такое, нам стоит выбрать тот фреймворк или библиотеку, что мы собираемся использовать при создании нашего сайта. Сперва мы решили найти список самых популярных фреймворков JavaScript и выбрать несколько из них, так как для таких новичков, как мы, лучше всего использовать те технологии, по которым можно найти больше всего информации в сети интернет. Исходя из этого, мы выяснили, что самые подходящие для нас: фреймворки это Angular и Vue.js, а также библиотеки JQuery, Reaсt и Backbone.js. Для того, чтобы выбрать что-то из этого, мы решили рассмотреть каждый из них поближе.

**React**

Для начала рассмотрим React, который может использоваться для разработки одностраничных и мобильных приложений. Его цель — предоставить высокую скорость, простоту и масштабируемость.

React основан на компоненте многократного использования. Проще говоря, это блоки кода, которые можно классифицировать как классы или функции. Каждый компонент представляет определенную часть страницы, такую как логотип, кнопка или поле ввода. Используемые ими параметры называются реквизитами, что означает свойства. Говоря о синтаксисе, большинство разработчиков сходятся во мнении, что React легко освоить, когда вы уже знаете JavaScript. React использует JSX, синтаксис XML, который сочетает в себе JavaScript и HTML.

**Angular**

Теперь перейдем к Angular.

Angular — одна из самых мощных сред JavaScript. Эта среда разработки известна прежде всего потому, что она предоставляет разработчикам лучшие условия для объединения JavaScript с HTML и CSS. Более полумиллиона сайтов, таких как google.com, youtube.com и т.д., используют Angular.

Angular имеет компонентную структуру, как и React. Вам нужно будет использовать TypeScript, чтобы написать приложение в Angular. Это расширенный набор JavaScript, который использует тот же синтаксис, но также поддерживает статическую типизацию и классы. В TypeScript вы получаете модификаторы доступа, перечисления, обобщения, гибридные типы и многое другое.

**Vue.js**

Vue — это JavaScript-фреймворк с открытым исходным кодом для создания креативного интерфейса. Интеграция с Vue в проектах, использующих другие библиотеки JavaScript, упрощена, поскольку она разработана для адаптации. Более 36 000 веб-сайтов в настоящее время используют Vue. Такие компании, как Stackoverflow, PlayStation и т.д., полагаются на Vue для своих сайтов пользовательского интерфейса.

Vue.js также довольно прост в освоении: все что нужно, это JavaScript и HTML. Другой сильной стороной Vue.js является его интерфейс командной строки (CLI). Это базовый инструмент, который ускоряет разработку, предлагая массу плагинов, пресетов, мгновенного прототипирования и интерактивного инструмента разработки проектов. Некоторые из его функций включают компоненты, шаблоны, переходы и двустороннее связывание данных, а также фокус реактивности. Реактивность возникает при изменении или обновлении любого из объектов JavaScript в Vue. Vue.js использует то, что называется Shadow DOM, что делает рендеринг страницы быстрым.

**JQuery**

JQuery, пожалуй, самая популярная библиотека JavaScript с таким количеством функций для современной разработки. JQuery — это быстрая и лаконичная библиотека JavaScript, созданная Джоном Резигом в 2006 году. Это кроссплатформенная библиотека JavaScript, предназначенная для упрощения HTML-скриптинга на стороне клиента. Более 19 миллионов веб-сайтов в настоящее время используют jQuery! Такие компании, как WordPress, Facebook, Google, IBM и многие другие, полагаются на jQuery для обеспечения своего рода просмотра веб-страниц.

Вы можете использовать API jQuery для обработки, анимации и манипулирования событием в HTML-документе, также известном как DOM. Кроме того, jQuery используется со строительными инструментами Angular и React App. Одним словом, одна из самых важных библиотек JavaScript для веб-разработки.

**Backbone.js**

Это одна из самых популярных платформ JavaScript. Его можно использовать для создания одностраничного приложения. Разработка этого фреймворка предполагает, что все функции на стороне сервера должны проходить через API, что поможет достичь сложной функциональности за счет написания меньшего количества кода.

BackboneJS — это легкая библиотека JavaScript, которая позволяет разрабатывать и структурировать клиентские приложения, работающие в веб-браузере. В отличие от других сред, Backbone поручает разработчику выбрать правильный инструмент, который лучше всего подходит для данного проекта. Более полумиллиона сайтов в настоящее время используют Backbone, включая tumblr.com, espn.com, soundcloud.com и многие другие.

**Выбор фрейморка**

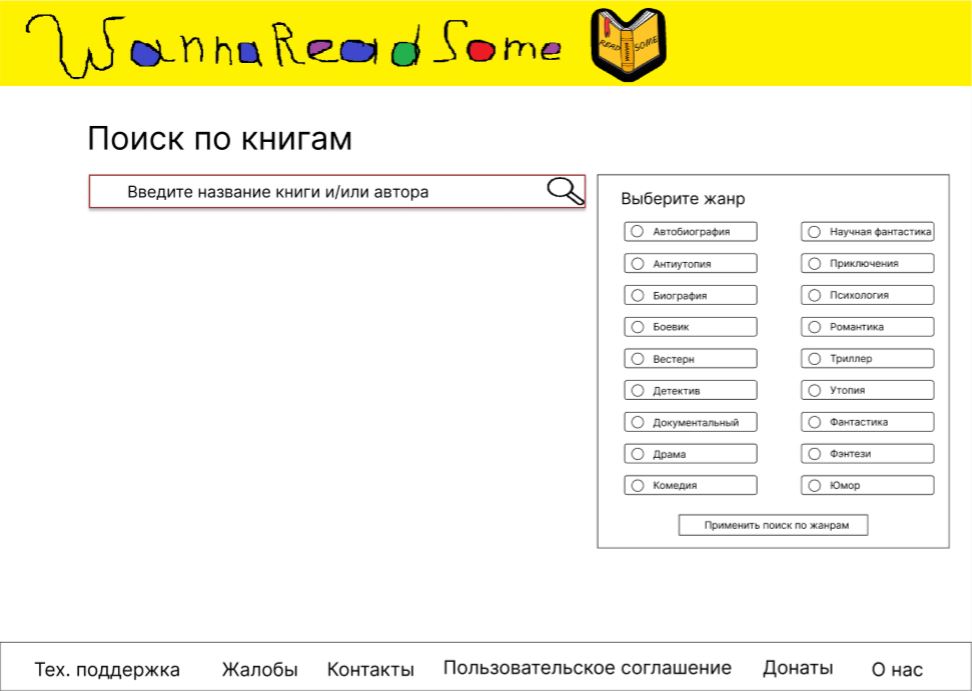
Ну вот теперь, когда мы написали о каждом из фреймворков и библиотек, пора выбрать фреймворк, что мы будем использовать. Мы решили остановиться на Angular, потому что

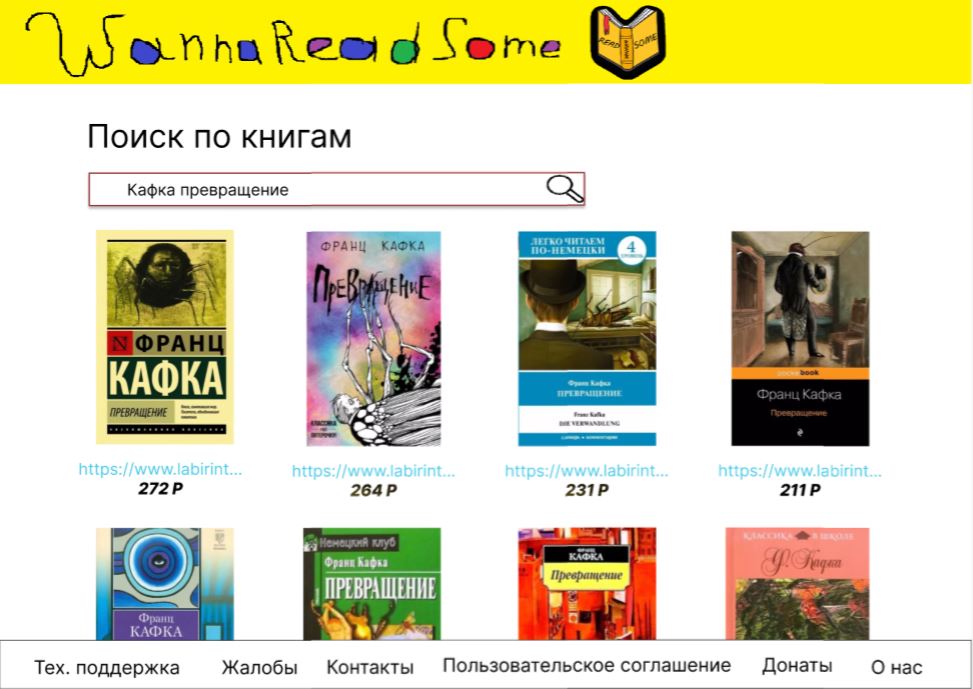
* Это гибкая среда разработки, обладающая большим количеством библиотек
* Также это стабильный фреймворк, при использовании которого обычно можно не беспокоиться за возникновение ошибок.
* Angular имеет компонентную структуру

**Эскизы сайта**

Мы планируем сделать сайт, который способен находить книгу в сети интернет, при этом мы собираемся сделать сортировку, которая поможет пользователям отобрать лучший для них вариант книги. Суть нашего сайта заключается в помощи пользователям находить книги, не лазая по десяткам сайтов для поиска книги по нужной категории, наш сайт будет делать это за них и поможет сохранить им время и нервы, которые они могут затратить на это.

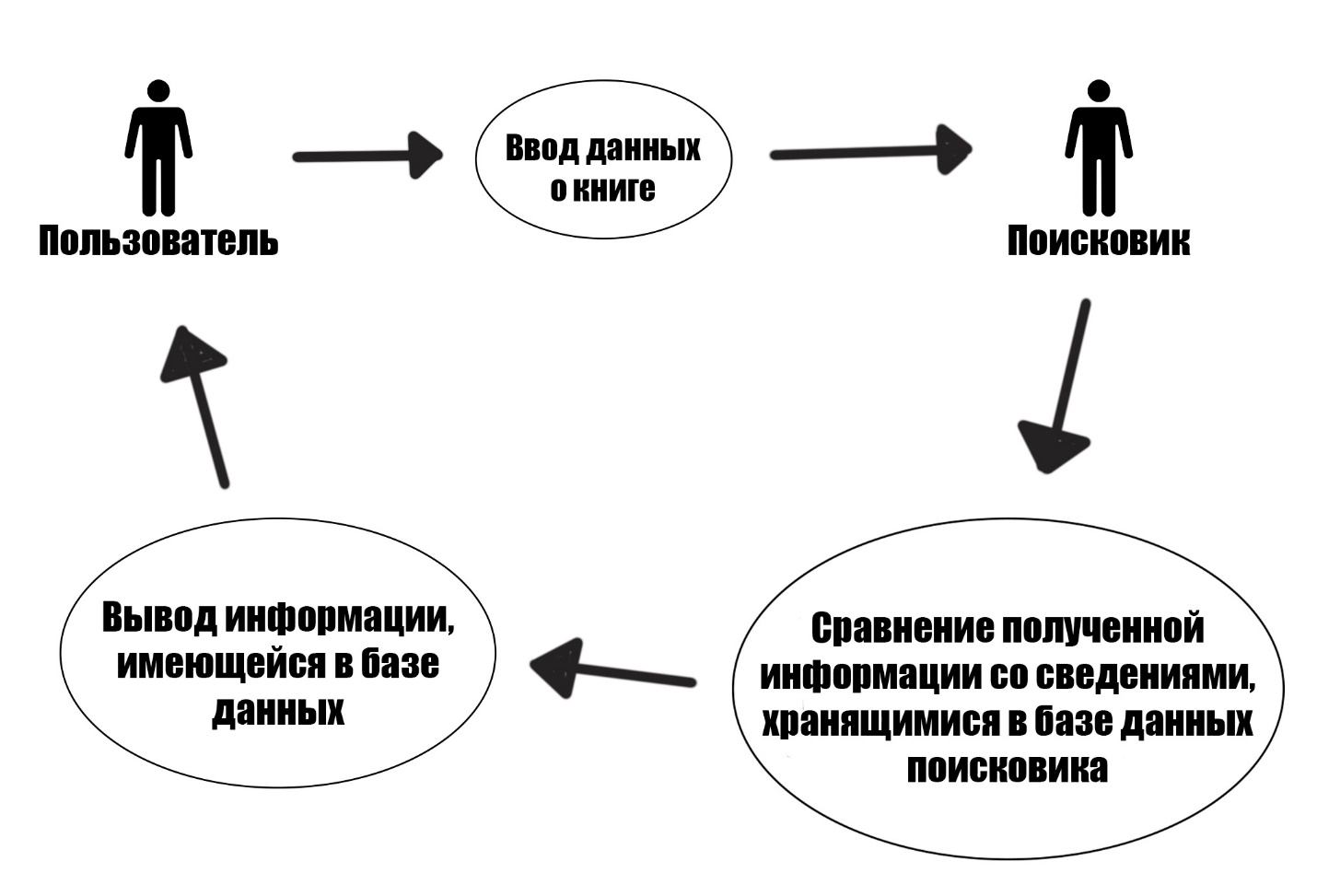
Наш ИИ ищет книги на многих сайтах и закидывает в нашу базу данных обложку, название книги, ссылку на страницу сайта, где она продается и ее ценник. Пользователь должен ввести название книги и автора и выбрать сортировку по категории которая ему нужна, затем из нашей базы данных на основную страницу сайта переходят все данные, которые хранятся в нашей базе данных, по нужной пользователю книге, чтобы пользователь мог их увидеть и найти нужный ему вариант сравнении эти данные.

****

****

**Диаграмма вариантов использования**

Пользователь вводит данные о книге в поисковик, далее поисковик обращается к базе данных с целью найти соответствие со сведениями, введёнными пользователем, после чего выводит соответствующую информацию из базы данных на экран.



**Вывод**

За текущий семестр мы выполнили основную часть поставленных задач, в основном направленных на анализ и теоритическое знакомство с инструментами и ресурсами, которые мы собираемся использовать для создания полноценного сайта, но помимо этого, определив целевую аудиторию, цели, задачи и требования к проекту, а также составив начальный эскиз сайта и диаграмму вариантов его использования, мы смогли понять то, какой именно веб-сайт мы хотим создать.

В течение следующего семестра мы планируем реализовать следующие задачи составить UML-диаграммы: классов (если они есть), компонентов, последовательности, деятельности, сверстать веб-страницы, реализовать и протестировать front-end функции работы веб-сайта, окончательно определить и доработать дизайн страниц сайта.

**Источники:**

<https://www.youtube.com/watch?v=MoD8KxpCYp4>

<https://tproger.ru/translations/web-frameworks-how-to-get-started/>

<https://drozd.red/glossary/framework/>

<https://vc.ru/dev/147263-populyarnye-freymvorki-javascript>

<https://secretmag.ru/enciklopediya/chto-takoe-freimvork-obyasnyaem-prostymi-slovami.htm>

<https://ru.wikipedia.org/wiki/Фреймворк>

<https://librebook.me/search/advanced>

<https://www.findbook.ru>

<https://www.livelib.ru>

<https://ru.m.wikipedia.org/wiki/Диаграмма_прецедентов>

<https://topseo.su/blog/development-of-article-sites/yazyki-dlya-sozdaniya-saytov/>