

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное
образовательное учреждение высшего образования
«Южно-Уральский Государственный Университет
(национальный исследовательский университет)»

Высшая школа электроники и компьютерных наук
Кафедра «Электронные вычислительные машины»

ДОПУСТИТЬ К ЗАЩИТЕ
Заведующий кафедрой ЭВМ
_____ Д.В. Топольский
« ____ » _____ 2022 г.

ВЕБ-ПРИЛОЖЕНИЕ ДЛЯ ВЫБОРА РЕЦЕПТА ИЗ НАБОРА
ИМЕЮЩИХСЯ ПРОДУКТОВ

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ
ЮУРГУ-090301.2022.218 ПЗ ВКР

Руководитель работы,
к.п.н., доцент каф. ЭВМ
_____ М.А. Алтухова
« ____ » _____ 2022

Автор работы,
Студент группы КЭ-405
_____ Е.С. Малинин
« ____ » _____ 2022

Нормоконтролёр,
к.п.н., доцент каф. ЭВМ
_____ М.А. Алтухова
« ____ » _____ 2022

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего образования
«Южно-Уральский Государственный Университет
(национальный исследовательский университет)»
Высшая школа электроники и компьютерных наук
Кафедра «Электронные вычислительные машины»

УТВЕРЖДАЮ

Заведующий кафедрой ЭВМ

_____ Д.В. Топольский

«___» _____ 2022 г.

ЗАДАНИЕ

на выпускную квалификационную работу бакалавра
студенту группы КЭ-405
Малинину Егору Сергеевичу
обучающемуся по направлению
09.03.01 «Информатика и вычислительная техника»

1. **Тема работы:** «Веб-приложение для выбора рецепта из набора имеющихся продуктов» утверждена приказом по университету от "___" _____ 202__ г. №___

2. **Срок сдачи студентом законченной работы:** 1 июня 2022 г.

3. **Исходные данные к работе.**

Обеспечить основной функционал приложения:

1) подбор из базы данных рецептов блюд с наибольшим числом совпадений ингредиентов;

2) возможность выбора времени приёма пищи (завтрак, обед, ужин, перекус) и типа питания («не имеет значения», «для веганов», «для вегетарианцев», «спортивное питание», и т.д.);

3) возможность авторизоваться для пользователя и администратора;

4) возможность выбора ингредиентов из соответствующих категорий;

5) возможность использования «живого» поиска, который будет работать без перезагрузки страницы

б) возможность выбора ингредиентов из представленного списка на главной странице;

7) возможность предложить рецепт на рассмотрение администратору в отдельной форме.

Веб-приложение должно работать на ПК и иметь адаптивный интерфейс на мобильных устройствах.

Предусмотреть наличие ролей обычного пользователя, администратора.

Спроектировать структуру базы данных рецептов, чтобы впоследствии вручную заполнять её от имени администратора.

Обеспечить оптимизацию времени загрузки сайта – время загрузки главной страницы не должно превышать 5 секунд при нагрузке до 20 посетителей в минуту.

Среда и средства реализации – по выбору автора.

4. Перечень подлежащих разработке вопросов:

- рассмотрение существующих аналогов по выбранной тематике;
- анализ современных программных технологий для обеспечения указанного функционала в веб-браузере;
- формирование требований к веб-приложению;
- проектирование веб-приложения;
- тестирование работоспособности разработанного веб-приложения.

5. Дата выдачи задания: 1 декабря 2021 г.

Руководитель работы _____ /М.А. Алтухова/

Студент _____ /Е.С. Малинин /

КАЛЕНДАРНЫЙ ПЛАН

Этап	Срок сдачи	Подпись руководителя
Введение и обзор литературы	07.02.2022	
Разработка модели, проектирование	07.03.2022	
Реализация системы	04.04.2022	
Тестирование, отладка, эксперименты	10.05.2022	
Компоновка текста работы и сдача на нормоконтроль	16.05.2022	
Подготовка презентации и доклада	24.05.2022	

Руководитель работы _____ /М.А. Алтухова/

Студент _____ /Е.С. Малинин /

АННОТАЦИЯ

Е.С. Малинин. Веб-приложение для выбора рецепта из набора имеющихся продуктов. – Челябинск: ФГАОУ ВО «ЮУрГУ (НИУ)», ВШЭКН; 2021, 76 с., 11 табл., 43 ил., библиогр. список – 15 наим.

В рамках выпускной квалификационной работы производится разработка веб-приложения для выбора рецепта из набора имеющихся в наличии продуктов. Целью разработки данного веб-приложения является обеспечение рационального использования наибольшего числа продуктов из холодильника во избежание их порчи либо замораживания.

СОДЕРЖАНИЕ

ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ.....	8
ВВЕДЕНИЕ.....	9
1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ	11
1.1 Обзор аналогов	11
1.2 Анализ основных технологических решений	15
1.3 Вывод.....	18
2 ОПРЕДЕЛЕНИЕ ТРЕБОВАНИЙ	20
2.1 Функциональные требования.....	20
2.2 Нефункциональные требования.....	22
2.2.1 Требования к пользователям.....	22
2.2.2 Требования к системе безопасности	22
2.2.3 Требования к лингвистическому обеспечению.....	23
3 ПРОЕКТИРОВАНИЕ	24
3.1 Архитектура веб-приложения.....	24
3.2 Описание данных веб-приложения	25
3.2.1 Входные данные	25
3.2.2 Выходные данные	26
3.2.3 Описание базы данных веб-приложения	27
4 РЕАЛИЗАЦИЯ ВЕБ-ПРИЛОЖЕНИЯ	31
4.1 Реализация интерфейсов	31
4.2 Вид страниц веб-приложения	41
5 ТЕСТИРОВАНИЕ РАЗРАБОТАННОГО ВЕБ-ПРИЛОЖЕНИЯ	42

5.1 Методология тестирования	42
5.2 Проведение процедуры тестирования	42
5.2.1 Регистрация и авторизация пользователя	42
5.2.2 Функция «Предложить рецепт».....	45
5.2.3 Процесс получения рецепта	48
ЗАКЛЮЧЕНИЕ	49
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	51
ПРИЛОЖЕНИЕ А Снимки страниц веб-приложения.....	53
ПРИЛОЖЕНИЕ Б Процесс получения рецепта.....	57
ПРИЛОЖЕНИЕ В Исходный код javascript-файла «main.js»	61
ПРИЛОЖЕНИЕ Г Исходный код javascript-файла «forform.js»	64
ПРИЛОЖЕНИЕ Д Исходный код javascript-файла «cart.js»	65
ПРИЛОЖЕНИЕ Е Исходный код php-файла «index.php»	66
ПРИЛОЖЕНИЕ Ж Исходный код php-файла «header.php»	68
ПРИЛОЖЕНИЕ З Исходный код php-файла «suggest_recipe.php».....	70
ПРИЛОЖЕНИЕ И Исходный код php-файла «recipes.php»	71
ПРИЛОЖЕНИЕ К Исходный код php-файла «functions.php».....	74

ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

В настоящей пояснительной записке применяются следующие термины с соответствующими определениями.

Frontend – пользовательский интерфейс, видимая посетителями часть сайта или приложения

Backend – программно-аппаратная часть сайта или приложения, отвечающая за функционирование его внутренней (серверной) составляющей

CMS (от англ. content management system) – информационная система или компьютерная программа, используемая для обеспечения и организации совместного процесса создания, редактирования и управления содержимым сайта или приложения

Usability – качественная оценка простоты и комфорта работы с сайтом или приложением. Пользователь должен без труда находить нужную информацию, не теряясь в функционале и многочисленных страницах, и при этом получать эстетическое удовольствие от работы с ресурсом

ВВЕДЕНИЕ

Современные информационные технологии помогают людям как в упрощении или автоматизации рабочего процесса, так и в бытовых вопросах. К примеру, в такой обыденной и популярной сфере, как приготовление пищи. Уже несколько лет стремительно снижается тенденция хранения записей на листах в книге рецептов, так как инструкцию к приготовлению любого интересующего блюда можно быстро отыскать в интернете. На сегодняшний день существует множество функционирующих сайтов и приложений со своими базами данных с рецептами, но, при всём их многообразии, они не удовлетворяют всем запросам пользователя. Например, у человека часто возникает необходимость рационального использования продуктов, находящихся в холодильнике, во избежание их скорой порчи, при этом, он не должен докупать ничего дополнительного.

Исходя из таких потребностей, было решено написать веб-приложение [1] по подбору рецептов из набора имеющихся продуктов, которое позволит в несколько кликов выбрать необходимое число ингредиентов, время приёма пищи, тип питания, в зависимости от предпочтений пользователя, например, вегетарианское. После данных операций веб-приложение предложит список рецептов с подробными описаниями каждого шага и фотографиями. Сайт также будет адаптирован под различные размеры экрана, от десктопных до мобильных телефонов.

Целью представленной выпускной квалификационной работы является разработка адаптивного веб-приложения для выбора рецепта из набора имеющихся продуктов, ориентированное на рациональное использование наибольшего числа продуктов из холодильника во избежание их порчи либо замораживания.

Для достижения поставленной цели, необходимо решить следующие задачи:

рассмотреть существующие аналоги, предоставляющие услуги просмотра, публикации, редактирования статей по теме подбора рецептов блюд;

провести анализ современных технологий для разработки веб-приложения, просмотра, публикации, редактирования списков продуктов и рецептов;

разработать структуру базы данных для хранения информации о пользователях, ингредиентах, рецептах;

заполнить базу данных набором ингредиентов и рецептов вручную;

оценить работоспособность разработанного веб-приложения в различных условиях;

произвести выборку результатов и проанализировать работу веб-приложения.

1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

1.1 Обзор аналогов

В таблице 1.1 перечислены продукты, аналогичные создаваемому веб-приложению.

Таблица 1.1 – Аналогичные продукты

№	Название продукта	Тип	Поддерживаемая система	Разработчик
1	Подбери рецепт	Мобильное приложение	Android	Y&M
2	Webspoon.ru	Сайт	Любой веб-браузер	Неизвестен
3	Поварёнок.ру	Сайт	Любой веб-браузер	Евгения Грозова
4	Гастрономъ	Сайт	Любой веб-браузер	ООО «Гастроном Медиа», Дизайн — Coalla Agency
5	Рецепты по ингредиентам	Сайт	Любой веб-браузер	Неизвестен

Мобильное приложение «Подбери рецепт» [2] предоставляет широкий диапазон выбора ингредиентов и рецептов, также в нём присутствует таблица мер и весов, которая показывает фактическую массу продукта в посуде определенного объёма. Также в данной разработке есть возможность, при необходимости, исключать ингредиенты из подбора и выбирать категории продуктов и рецептов для подбора.

Однако, стоит заметить то, что, несмотря на большую базу данных, приложение не удовлетворяет всех потребностей пользователя. Например, в нём отсутствует выбор времени приёма пищи и типа питания, а также исключений для людей с определенными ограничениями в выборе продуктов. Также в приложении присутствует много плохочитаемого текста (рисунок 1.1), и в описании рецептов нет картинок, из-за чего человек должен искать дополнительную информацию по внешнему виду того или иного шага приготовления в сторонних источниках. Стоит отметить, что пользоваться приложением могут только владельцы мобильных устройств с операционной системой Android.

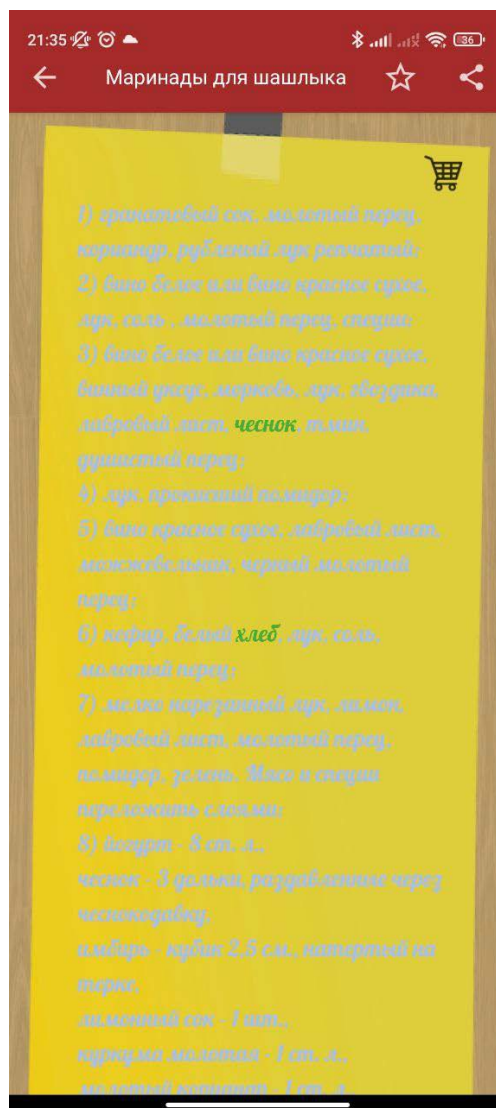


Рисунок 1.1 – Пример плохочитаемого текст в мобильном приложении «Подбери рецепт»

Кулинарный сайт «Webspoon.ru» [3] - сообщество, в котором люди могут делиться своими рецептами и коммуницировать друг с другом. На нём реализован поиск рецептов по ингредиентам, но стоит заметить несколько особенностей.

Например, для подбора можно выбрать только до четырёх ингредиентов, что не отвечает основному требованию этого функционала – задействовать наибольшее количество продуктов (рисунок 1.2). Также на сайте не реализованы возможности исключения ингредиентов, выбора времени приёма пищи и типа питания.

Подбор рецептов по ингредиентам

Для подбора можно выбрать до четырёх ингредиентов.

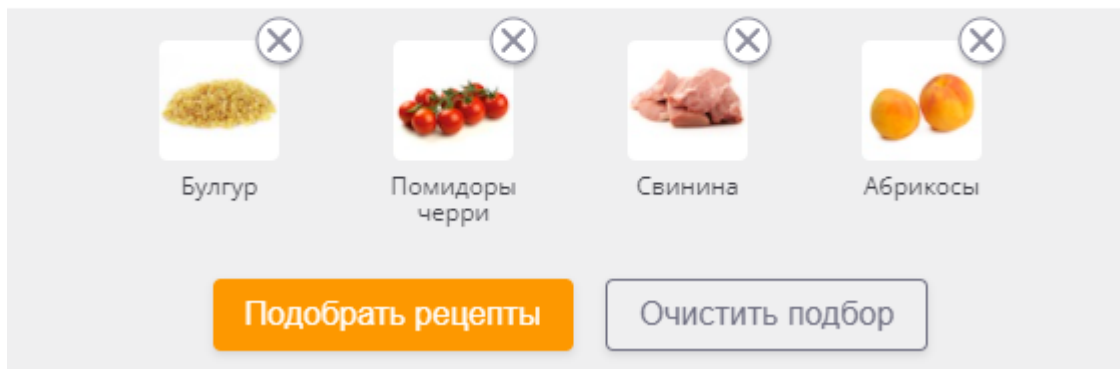


Рисунок 1.2 – Максимальное количество выбранных ингредиентов, равное четырём

«Поварёнок.ру» [4] является кулинарным сайтом, на котором пользователи могут самостоятельно загружать свои рецептуры, проходя модерацию. На компьютерном разрешении экрана наблюдается переизбыток информации, при котором посетитель сайта может просто потеряться и выйти. Например, в правой панели сайта присутствует раздел «Категории рецептов», содержимое которой отображается также в «шапке» сайта, а в левой части находится «Рубрикатор рецептов», который выполняет точно такую же функцию, но описан другими словами. Выделенные области представлены на рисунке 1.3. Что касается подбора рецептов по ингредиентам, то он выполняет второстепенную роль на сайте и реализован через ввод названия продуктов в строку через запятую, что, по мнению автора, может быть неудобным для пользователей со слабым зрением или отталкивающим с точки зрения «usability».

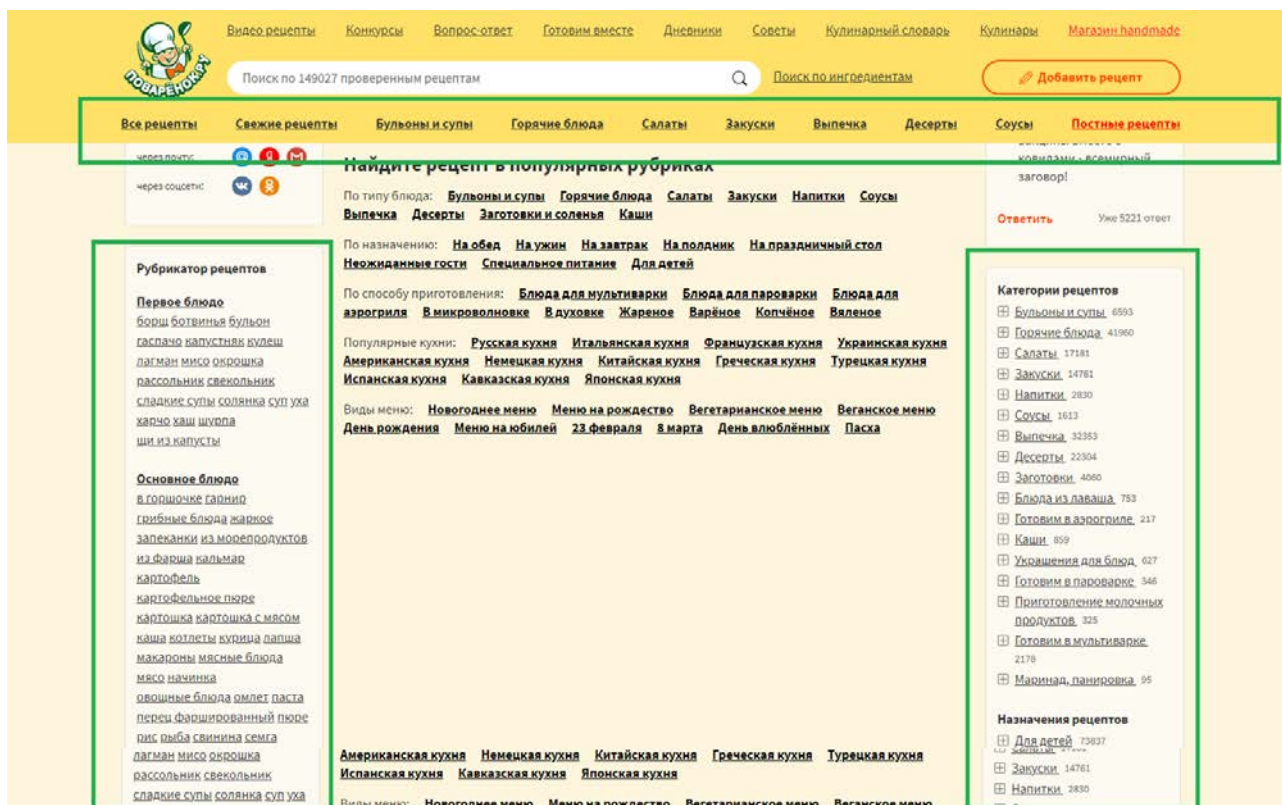


Рисунок 1.3 – Выделенные области с разделами с одинаковым функционалом.

«Гастрономъ» [5] является преимущественно новостным сайтом с кулинарной тематикой, но и на нём реализован поиск рецептов по ингредиентам. Также в данной разработке есть возможность, при необходимости, исключать ингредиенты из подбора и выбирать категории продуктов и рецептов для подбора. Аналогично с предыдущим сайтом, здесь необходимо вводить названия продуктов без ошибок и через запятую.

Кулинарный сайт «Рецепты по ингредиентам» [6] оснащён полным функционалом поиска инструкций по приготовлению, но каждая ссылка на них ведёт на сторонние сайты. Также на сайте невозможно исключить ингредиент из подбора, из-за чего людям с ограничениями в пище придется дольше подбирать подходящий рецепт.

1.2 Анализ основных технологических решений

Работа выполняется своими силами без стороннего финансирования, поэтому важную роль будет играть наличие бесплатных инструментов. При использовании новых для автора технологий важен низкий порог вхождения, наличие доступной документации, обеспечение высокой скорости разработки веб-приложений, средства для работы с базой данных.

Для создания backend части веб-приложения выбран язык программирования PHP [7], так как он ориентирован именно для решения задач веб-разработки. Главным фактором языка PHP является практичность. Практический характер PHP обусловлен четырьмя важными характеристиками:

- традиционностью;
- простотой;
- эффективностью;
- безопасностью.

Остальные преимущества языка PHP:

- является свободным программным обеспечением, распространяемым подсобой лицензией;
- несложен в освоении;
- имеет большое сообщество пользователей и разработчиков;
- имеет развитую поддержку баз данных;
- есть возможность использования в изолированной среде;
- предлагает нативные средства организации веб-сессий, программный интерфейс расширений;
- возможно развертывание почти на любом веб-сервере;
- поддерживается многими аппаратными платформами и операционными системами;
- предоставляет в распоряжение разработчиков и администраторов

гибкие и эффективные средства безопасности.

При выборе языков программирования для разработки Backend [8] части веб-приложения были рассмотрены и другие варианты: Python, Java, Ruby, C, C++, но оказались отвергнуты по следующим причинам:

Недостатки языка C++, C, Java:

- очень высокий порог вхождения;
- очень низкий уровень абстракции;
- очень низкая скорость разработки веб-приложений.

Недостатки языка Ruby:

- высокий порог вхождения;
- недостаток информационных ресурсов;
- медленно развивается.

Недостатки языка Python:

- плохая поддержка многопоточности;
- отсутствие коммерческой поддержки средств разработки;
- недостаток средств для работы с базами данных.

Для упрощения разработки было решено использовать систему управления содержимым WordPress.

Система управления содержимым сайта WordPress является самой популярной свободной системой (системой с открытым исходным кодом) управления сайтом по версии интернет-ресурса «Рейтинг Рунета» [9]. Количество проектов, учитываемые рейтингом и разработанные на данной системе, – 15 377.

Основные преимущества WordPress:

- удобная, простая и функциональная панель администратора (на освоение уйдет не более 3-4 часов);
- постоянные обновления и дополнения (бесплатные);
- развитое сообщество пользователей и соответствующая база знаний;
- возможность скачать дополнительные расширения.

CMS WordPress позволяет направлять контент из базы данных на сайт с помощью специальных PHP команд, что делает сайт очень гибким и динамическим.

Существует множество систем управления базами данных, которые можно использовать для веб-приложений, например MySQL, Microsoft SQL Server, PostgreSQL, MongoDB.

MySQL идеально подходит для проектов, которым требуется надежный инструмент управления базами данных, но бесплатный.

Преимущества MySQL:

- распространяется бесплатно;
- понятная документация;
- простая установка;
- поддерживает набор пользовательских интерфейсов;
- является распространенной на хостингах;
- хорошая поддержка.

Недостатки Microsoft SQL Server:

- является тяжеловесной;
- даже при тщательной настройке производительности СУБД может занять все доступные ресурсы;
- есть проблемы с использованием службы интеграции для импорта файлов.

Microsoft SQL Server идеально подходит для крупных организаций, которые уже используют ряд продуктов Microsoft.

Недостатки PostgreSQL:

- неразборчивая документация;
- сложная конфигурация;
- скорость работы может падать во время проведения пакетных операций или выполнения запросов чтения.

PostgreSQL идеально подходит для проектов с ограниченным

бюджетом, но квалифицированными специалистами.

Недостатки MongoDB:

- SQL не используется в качестве языка запросов;
- программа установки может занять много времени;

MongoDB подходит для проектов с разнородными данными, которые тяжело поддаются классификации. Для внедрения потребуются высококлассные специалисты

Таким образом, при разработке веб-приложения будет использоваться СУБД MySQL.

Для разработки клиентской части (frontend) используются следующие средства:

- язык разметки HTML;
- таблицы каскадных стилей CSS [10];
- язык программирования JavaScript с библиотекой jQuery;
- фреймворк Bootstrap 5, использующий современные наработки в области CSS и HTML.

jQuery [11] – это библиотека JavaScript, основанная на принципе «пиши меньше, делай больше». Это инструмент, который упрощает написание общих задач JavaScript [12]. Кроме того, jQuery обладает кроссбраузерной совместимостью, а значит, можно быть уверенным в том, что любой современный браузер корректно отобразит вывод программы.

Главным элементом Bootstrap является адаптивная сетка. В общем-то, без нее фреймворк утратил бы практически всю свою ценность, потому что именно благодаря сетке можно быстро создавать адаптивные шаблоны.

1.3 Вывод

Исследование рынка приложений и сайтов для подбора рецептов из набора имеющихся продуктов показало, что ни одно из них не является полнофункциональным. Следовательно, создание веб-приложения,

свободного от выявленных недостатков, является актуальной задачей. Разрабатываемое веб-приложение будет иметь следующий ряд преимуществ, по сравнению с аналогами:

- адаптивность, кроссбраузерность и кроссплатформенность;
- возможность выбора времени приёма пищи и типа питания;
- возможность предложить свой рецепт на рассмотрение администратору;
- полностью читаемая и приятная для пользователя информация на каждой странице;
- наличие в каждом рецепте пошаговой инструкции с картинками;
- наличие только самого необходимого функционала, который полностью соответствует назначению сайта, и отсутствие лишней информации, которая засоряет экран.

Для разработки веб-приложения выбраны следующие технологические решения: для backend – язык программирования PHP, CMS WordPress, СУБД MySQL, для frontend – язык разметки HTML, таблицы каскадных стилей CSS, язык программирования JavaScript с библиотекой jQuery, фреймворк Bootstrap 5.

2 ОПРЕДЕЛЕНИЕ ТРЕБОВАНИЙ

Для реализации веб-приложения используется следующий набор подсистем:

- HTTP-сервер Apache [13] с веб-интерфейсом в виде набора PHP-скриптов. С помощью веб-сайта обеспечивается общее управление системой, добавление ингредиентов, возможность предложить свой рецепт, получение результатов в виде рецептов, и регистрация новых пользователей;

- MySQL база данных, обеспечивающая хранение данных о пользователях, продуктах и рецептах;

- система управления содержимым сайта, обеспечивающая разработку веб-приложения с необходимым функционалом;

- веб-браузер, позволяющий тестировать разрабатываемый веб-сервис.

2.1 Функциональные требования

Основные требования к функционалу системы [14]:

- вся необходимая информация для выбора продуктов и подбора рецептов должна находиться в одном приложении;

- подбор рецептов из базы данных с наибольшим числом совпадения ингредиентов;

- выбор времени приёма пищи (завтрак, обед, ужин, перекус) и типа питания («не имеет значения», «для веганов», «для вегетарианцев», «спортивное питание», и т.д);

- возможность авторизоваться для пользователя и администратора;

- возможность предложить рецепт на рассмотрение администратору в отдельной форме;

– наличие системы администрирования веб-приложения.

Функционал выбора продуктов предоставляет пользователю:

- возможность выбора продуктов из предложенного списка;
- возможность добавления ингредиентов в «корзину» для дальнейшего поиска рецептов;
- возможность поиска ингредиентов в поле ввода слов;
- возможность выбора ингредиентов из определенных категорий.

Функционал подбора рецептов из базы данных предоставляет пользователю:

- возможность выбора времени приёма пищи;
- возможность выбора типа питания;
- возможность формирования списка подходящих рецептов, согласно выбранным ингредиентам;
- возможность сортировки полученных рецептов по убыванию числа совпадающих ингредиентов;
- возможность просмотра подробной инструкции по приготовлению блюд.

Функционал предложения рецепта предоставляет пользователю:

- возможность отправить предложенный рецепт на почту администратору;
- возможность заполнить формы «Название Вашего рецепта», «Описание Вашего рецепта», «Ссылка на источник рецепта (если рецепт взят из интернета)».

Функционал регистрации и авторизации пользователей предоставляет:

- возможность пользователю зарегистрироваться;
- возможность пользователю авторизоваться в системе;
- возможность пользователь сбросить пароль в случае его утраты.

2.2 Нефункциональные требования

2.2.1 Требования к пользователям

Пользователи должны обладать базовыми навыками владения компьютером и веб-браузерами [15].

Каждый пользователь, работающий в системе, имеет возможность:

- зарегистрироваться в системе и заполнить обязательные поля регистрационной формы (имя пользователя на латинице, адрес электронной почты, пароль для входа в систему, повторный пароль для подтверждения) и необязательные (имя, фамилия);

- войти в систему в случае выхода из аккаунта и заполнить следующие обязательные поля формы входа (имя пользователя на латинице или адрес электронной почты, пароль для входа в систему);

- позволить системе запомнить авторизацию в браузере;

- сбросить пароль, заполнив соответствующую форму.

Зарегистрированный пользователь должен иметь возможность воспользоваться любым функционалом системы, за исключением функций модерирования и администрирования.

Незарегистрированный пользователь должен иметь возможность воспользоваться любым функционалом системы, за исключением функций модерирования, администрирования и предложения своего рецепта на рассмотрение администратору.

Администратор должен иметь возможность воспользоваться функционалом администрирования.

2.2.2 Требования к системе безопасности

Система должна обладать следующими средствами обеспечения безопасности:

- пользователь при входе в систему обязан указывать логин и пароль;
- пользователь при регистрации в системе обязан указывать пароль два раза (для подтверждения);
- пароль должен храниться в зашифрованном виде.

2.2.3 Требования к лингвистическому обеспечению

Веб-приложение должно быть на русском языке, за исключением названия сайта, а также имени пользователя и адреса электронной почты, вводимых пользователями на латинице.

3 ПРОЕКТИРОВАНИЕ

3.1 Архитектура веб-приложения

Архитектура веб-приложения представлена на рисунке 3.1.

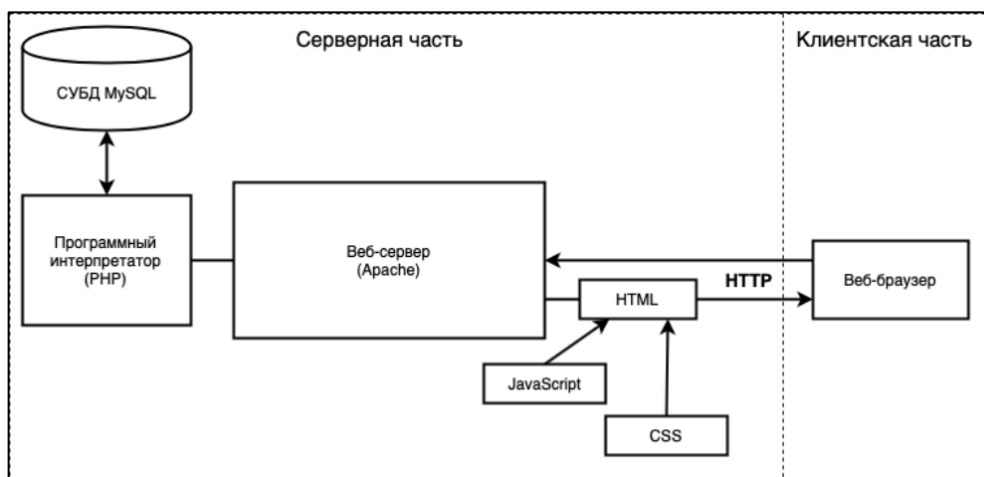


Рисунок 3.1 – Архитектура веб-приложения

Веб-приложение — это клиент-серверное приложение, в котором клиентом выступает браузер, а сервером — веб-сервер. Логика веб-приложения распределена между сервером и клиентом, хранение данных осуществляется преимущественно на сервере, обмен информацией происходит по сети. Одним из достоинств такого подхода является тот факт, что клиенты не зависят от конкретной операционной системы пользователя, поэтому веб-приложения являются кроссплатформенными сервисами.

Структурно серверная часть веб-приложения состоит из трех крупных компонентов:

- веб-сервера (HTTP-сервера);
- сервера базы данных (БД);
- программного интерпретатора.

Веб-сервер является связующим звеном между веб-приложением и клиентом. Он принимает запросы от клиента в виде набора данных по протоколу HTTP и возвращает в ответ документы на языке HTML и файлы, которые требуются дополнительно (изображения, листы стилей, другие

объекты).

Программный интерпретатор — это важная часть системы. С помощью программного интерпретатора выполняется код, содержащий логику работы системы (алгоритмы). Интерпретируемые программы медленнее выполняются, но процесс их разработки значительно проще и быстрее.

Сервер баз данных выполняет накопление структурированной информации, а также ее выдачу по запросам веб-приложения. От сервера баз данных зависят производительность приложения и функциональность с точки зрения данных. В качестве сервера баз данных в веб-приложении используется СУБД MySQL, имеющая серьезную репутацию в мире веб-разработок.

Центром клиентской части веб-приложения является браузер. Любой компьютер, подключенный к интернету, содержит программу-браузер, через которую можно начать работу с веб-приложением.

Со стороны клиентской части для придания интерактивности веб-страницам, помимо таблиц стилей CSS, к HTML-странице подключаются JavaScript-файлы

3.2 Описание данных веб-приложения

3.2.1 Входные данные

В качестве входных данных от пользователя в момент регистрации используются:

- имя пользователя;
- адрес электронной почты;
- имя;
- фамилия;

- пароль и его подтверждение.

В качестве входных данных от пользователя в момент авторизации или в случае утери пароля от учетной записи в системе используются:

- имя пользователя или адрес электронной почты;
- пароль для входа в систему.

В качестве входных данных от авторизованного пользователя при предложении рецепта на рассмотрение администратору используются:

- название рецепта;
- подробное описание приготовления блюда;
- ссылка на источник, если рецепт взят из интернета.

В качестве входных данных от пользователя при добавлении ингредиентов в «корзину» используются выбранные продукты самостоятельно или с помощью поиска в соответствующем поле ввода, а также посредством навигации по категориям ингредиентов.

3.2.2 Выходные данные

В качестве выходных данных в веб-приложении используются HTML, CSS, JavaScript, PHP файлы, формирующие внешний вид страниц:

- главная страница;
- страница регистрации;
- страница авторизации;
- страница восстановления пароля;
- страницы категорий ингредиентов;
- страница предложения своего рецепта;
- страница «корзины» с добавленными ингредиентами;
- страница со сформированными рецептами блюд.

3.2.3 Описание базы данных веб-приложения

Структура базы данных представлена на рисунке 3.2.

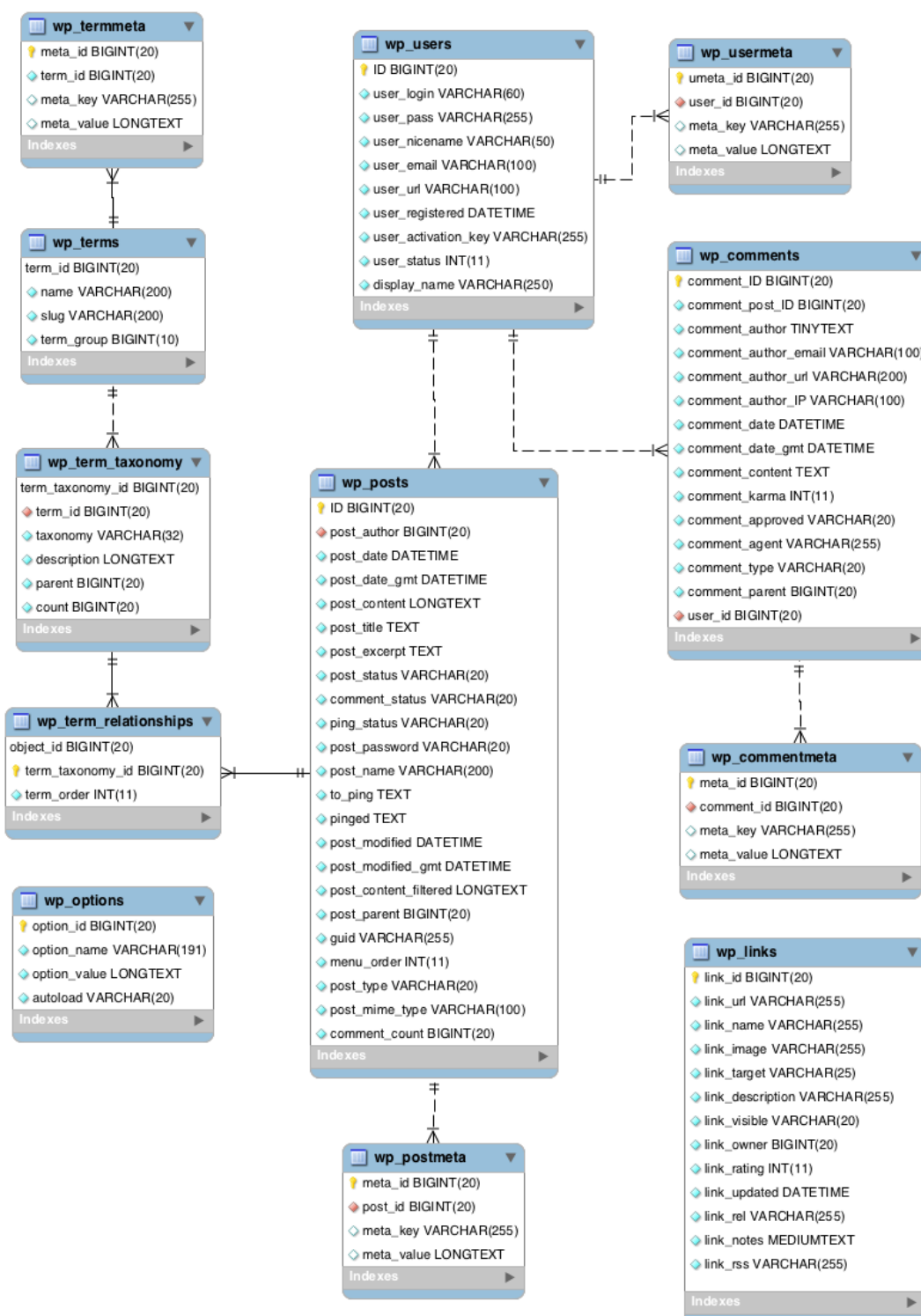


Рисунок 3.2 – Структура базы данных веб-приложения

Описание структуры дано в таблице 3.1.

Таблица 3.1 – Описание базы данных

Таблица	Данные
wp_termmeta	Метаданные терминов. Эта таблица дополняет таблицу «wp_terms».
wp_terms	Элементы таксономии (термины) и базовая информация о них.
wp_term_taxonomy	Дополнительные данные об элементе таксономии, в частности важные из них - к какой таксономии относится термин (поле «taxonomy») и с какой записью связан (поле «term_taxonomy_id»).
wp_term_relationships	Связи терминов с записями. Таблица показывает какая запись какому термину принадлежит. Только ID термина здесь связывается через поле «term_taxonomy_id».
wp_options	Основные опции и настройки сайта
wp_links	Ссылки сайта. На данный момент не используется
wp_users	Пользователи
wp_usermeta	Метаданные для каждого пользователя
wp_posts	Содержимое сайта
wp_postmeta	Данные о записях и страница сайта
wp_comments	Комментарии к записям сайта. На данный момент не используются.
wp_commentmeta	Метаданные комментариев к записям сайта. На данный момент не используются.

Описание полей каждой таблицы представлено в таблицах 3.2 – 3.10.

Таблица 3.2 – Описание полей таблицы «wp_termmeta»

Поле	Тип	Назначение	Индекс
meta_id	bigint(20)	ID метаполя	Основной
term_id	bigint(20)	ID термина	term_id (wp_terms)
meta_key	varchar(255)	Ключ метаполя	meta_key
meta_value	longtext	Значение метаполя	Нет

Таблица 3.3 – Описание полей таблицы «wp_terms»

Поле	Тип	Назначение	Индекс
term_id	bigint(20)	ID термина	Основной
name	varchar(200)	Название термина	name
slug	varchar(200)	Ярлык термина	slug
term_group	bigint(10)	Группа термина	Нет

Таблица 3.4 – Описание полей таблицы «wp_term_taxonomy»

Поле	Тип	Назначение	Индекс
term_taxonomy_id	bigint(20)	ID строки в таблице	Основной
term_id	bigint(20)	ID термина	term_id (wp_terms)
taxonomy	varchar(32)	Название таксономии	taxonomy
description	longtext	Описание термина	Нет
parent	bigint(20)	ID родительского термина (для древоводных таксономий)	Нет
count	bigint(20)	Количество записей в термине	Нет

Таблица 3.5 – Описание полей таблицы «wp_term_relationships»

Поле	Тип	Назначение	Индекс
object_id	bigint(20)	ID записи	Основной
term_taxonomy_id	bigint(20)	ID строки в таблице	term_taxonomy_id (wp_term_taxonomy)
term_order	int(11)	Порядок термина при прикреплении к записи	Нет

Таблица 3.6 – Описание полей таблицы «wp_options»

Поле	Тип	Назначение	Индекс
option_id	bigint(20)	ID опции	Основной

Продолжение таблицы 3.6

Поле	Тип	Назначение	Индекс
option_name	varchar(191)	Название опции	option_name
option_value	longtext	Значение опции	Нет
autoload	varchar(20)	Автозагрузка	Нет

Таблица 3.7 – Описание полей таблицы «wp_users»

Поле	Тип	Назначение	Индекс
ID	bigint(20)	ID пользователя	Основной
user_login	varchar(60)	Имя пользователя	user_login
user_pass	varchar(255)	Пароль пользователя	Нет
user_nicename	varchar(50)	Никнейм	user_nicename
user_email	varchar(100)	Адрес электронной почты	user_email
user_url	varchar(100)	Веб-сайт	Нет
user_registered	datetime	Дата регистрации	Нет
user_activation_key	varchar(255)	Ключ активации	Нет
user_status	int(11)	Статус	Нет
display_name	varchar(250)	Публичное имя	Нет

Таблица 3.8 – Описание полей таблицы «wp_usermeta»

Поле	Тип	Назначение	Индекс
umeta_id	bigint(20)	ID метаполя	Основной
user_id	bigint(20)	ID пользователя	ID (wp_users)
meta_key	varchar(255)	Ключ метаполя	meta_key
meta_value	longtext	Значение метаполя	Нет

Таблица 3.9 – Описание полей таблицы «wp_posts»

Поле	Тип	Назначение	Индекс
ID	bigint(20)	ID записи	Основной
post_author	bigint(20)	ID пользователя	Основной
post_date	datetime	Дата создания	type_status_date
post_date_gmt	datetime	Дата создания	Нет
post_content	longtext	Содержимое	Нет
post_title	text	Название записи	Нет
post_excerpt	text	Анонс записи	Нет
post_status	varchar(20)	Статус записи (опубликовано, черновик и т.п.)	type_status_date
comment_status	varchar(20)	Разрешение комментариев	Нет
ping_status	varchar(20)	Статус разрешения отправки ping-запросов	Нет
post_password	varchar(20)	Пароль допуска	Нет
post_name	varchar(200)	Имя записи	Нет
to_ping	text	URL, на которые нужно отправлять ping при публикации	Нет
pinged	text	URL, куда ping был отправлен	Нет
post_modified	datetime	Дата последнего изменения	Нет
post_modified_gmt	datetime	Дата последнего изменения	Нет
post_content_filtered	longtext	Временное содержимое	Нет
post_parent	bigint(20)	ID родительской записи	post_parent
guid	varchar(255)	Уникальный идентификатор записи	Нет
menu_order	int(11)	Числовой порядок в меню	Нет
post_type	varchar(20)	Тип записи	type_status_date
post_mime_type	varchar(100)	MIME тип записи	Нет
comment-count	bigint(20)	Количество комментариев	Нет

Таблица 3.10 – Описание полей таблицы «wp_postmeta»

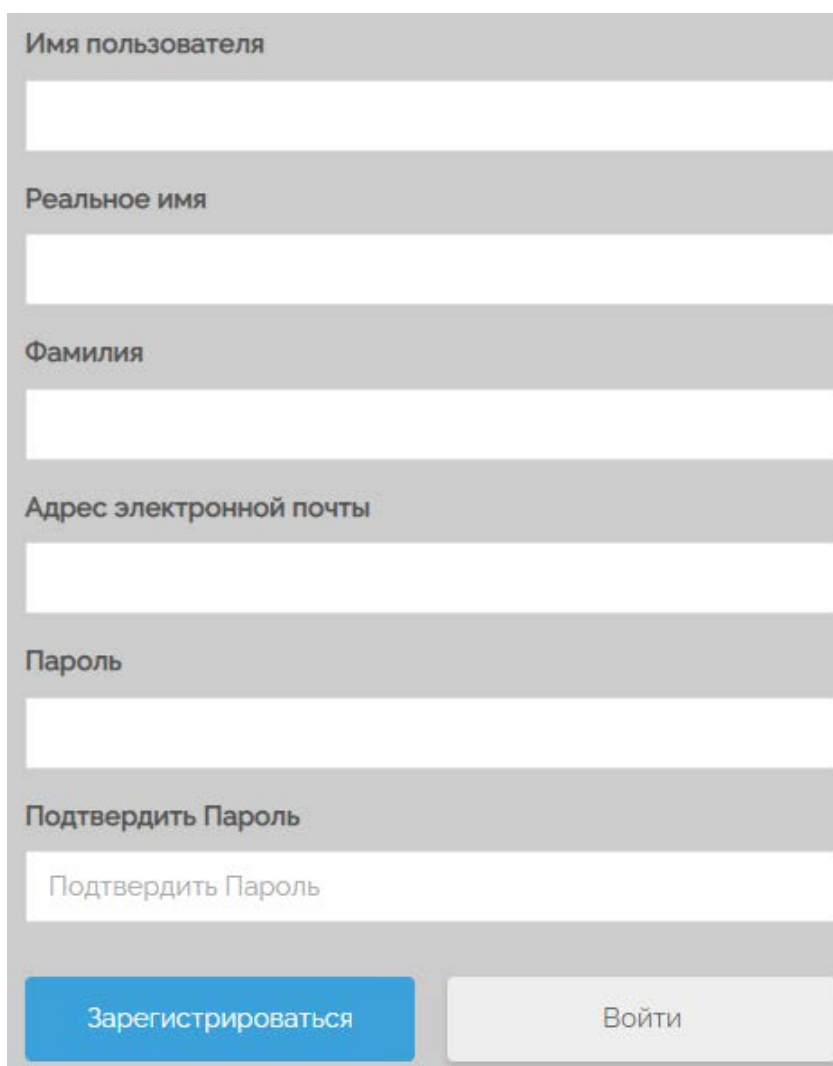
Поле	Тип	Назначение	Индекс
meta_id	bigint(20)	ID метаданных	Основной
post_id	bigint(20)	ID записи	post_id (wp_posts)
meta_key	varchar(255)	Ключ метаполя	meta_key
meta_value	longtext	Значение метаполя	Нет

4 РЕАЛИЗАЦИЯ ВЕБ-ПРИЛОЖЕНИЯ

4.1 Реализация интерфейсов

Для того, чтобы пользователь мог предлагать собственный рецепт на рассмотрение администратору, ему необходимо авторизоваться в системе. Если у пользователя нет учетной записи в системе, ему необходимо зарегистрироваться в системе. Экранная форма регистрации представлена на рисунке 4.1.

Исходный файл «main.js», где основной функционал всего веб-приложения приведен в листинге В.1 приложения В.



The image shows a registration form with the following fields and buttons:

- Имя пользователя (Username)
- Реальное имя (Real name)
- Фамилия (Surname)
- Адрес электронной почты (Email address)
- Пароль (Password)
- Подтвердить Пароль (Confirm Password)
- Зарегистрироваться (Register)
- Войти (Login)

Рисунок 4.1 – Экранная форма регистрации

Если все введенные данные корректны, система перенаправит пользователя на главную страницу, откуда пользователь сможет пользоваться полным функционалом сайта.

Если у пользователя уже есть аккаунт, то у него есть возможность сразу авторизоваться на сайте. Экранная форма авторизации представлена на рисунке 4.2.

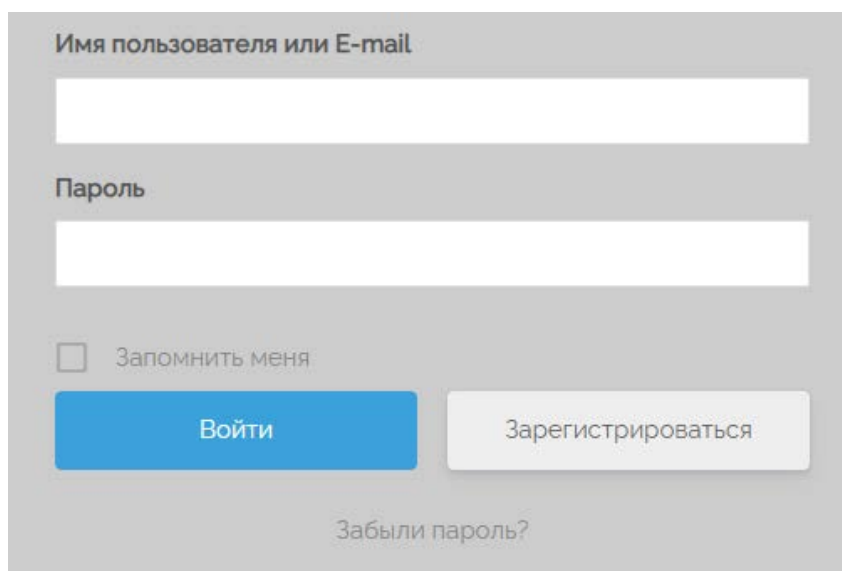


Рисунок 4.2 – Экранная форма авторизации

Если пользователь потерял/забыл пароль от своей учетной записи, он может указать адрес своей электронной почты, к которой привязана учетная запись, чтобы получить дальнейшие инструкции для сброса пароля. Экранная форма сброса пароля представлена на рисунке 4.3.

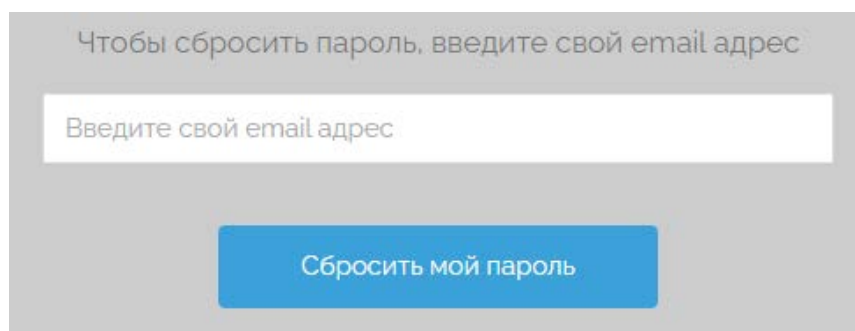


Рисунок 4.3 – Экранная форма сброса пароля

Для авторизованного пользователя в главном меню будет доступна форма для того, чтобы предложить собственный рецепт на рассмотрение администратору. Данная экранная форма представлена на рисунке 4.4.

Название Вашего рецепта

Описание Вашего рецепта

Ссылка на источник рецепта (если рецепт взят из интернета)

Отправить рецепт на почту Администратору для рассмотрения

Отправить

Рисунок 4.4 – Экранная форма предложения рецепта

Изначально пользователи попадают на главную страницу, интерфейс которой показан на рисунке 4.5. На ней доступны выбор ингредиентов из представленного списка, выбор категорий продуктов и реализован поиск.

Исходный файл «forForm.js», где показан функционал формы предложения рецепта приведен в листинге Г.1 приложения Г.

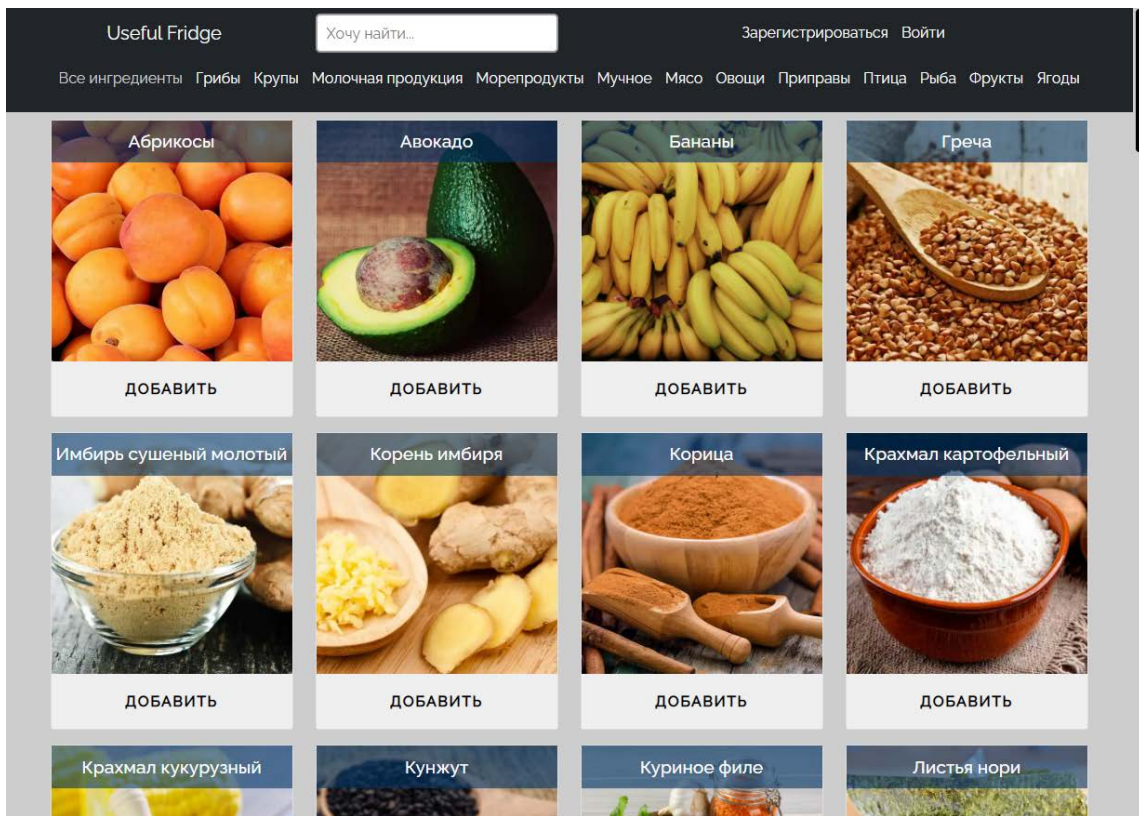


Рисунок 4.5 – Главная страница

На рисунке 4.6 демонстрируется одна из категорий, по которой пользователи будут способны сортировать продукты для более удобного поиска.

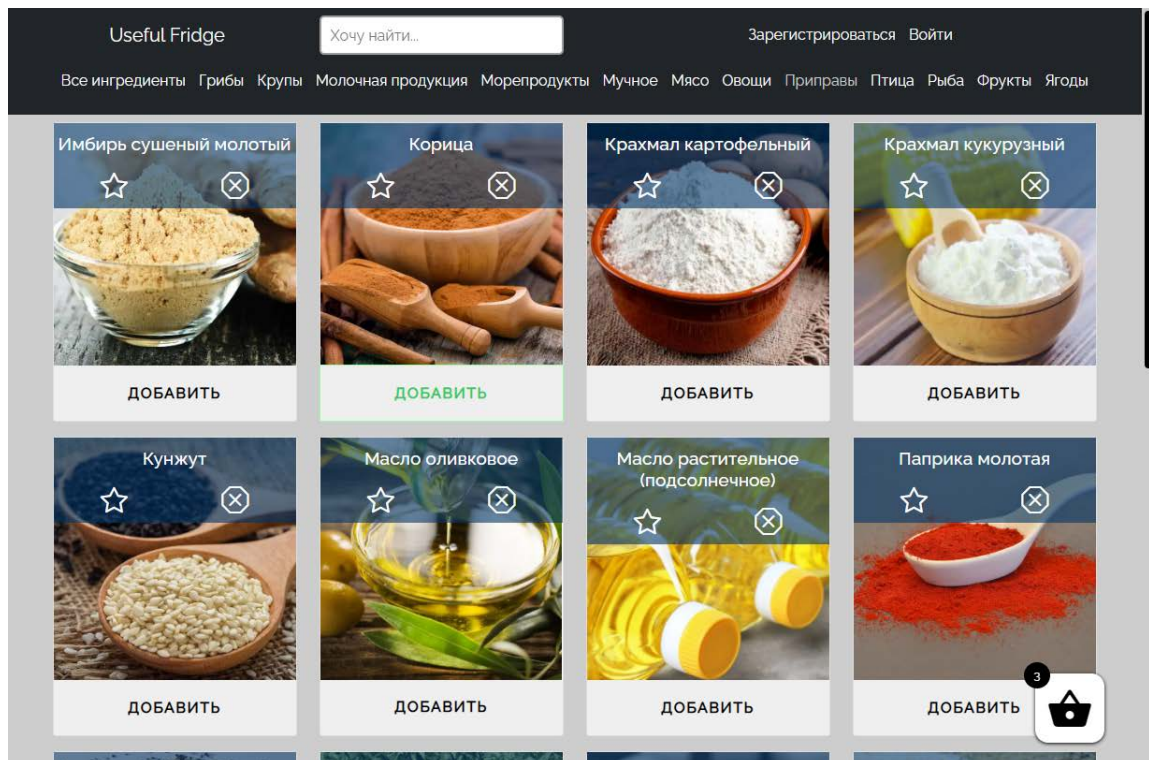


Рисунок 4.6 – Страница с продуктами в категории «Приправы»

На любой из страниц реализован «живой» поиск по первым введенным буквам, который без перезагрузки страницы обеспечит комфортный доступ к желаемым ингредиентам или рецептам. Демонстрация поиска приведена на рисунках 4.7 – 4.8.

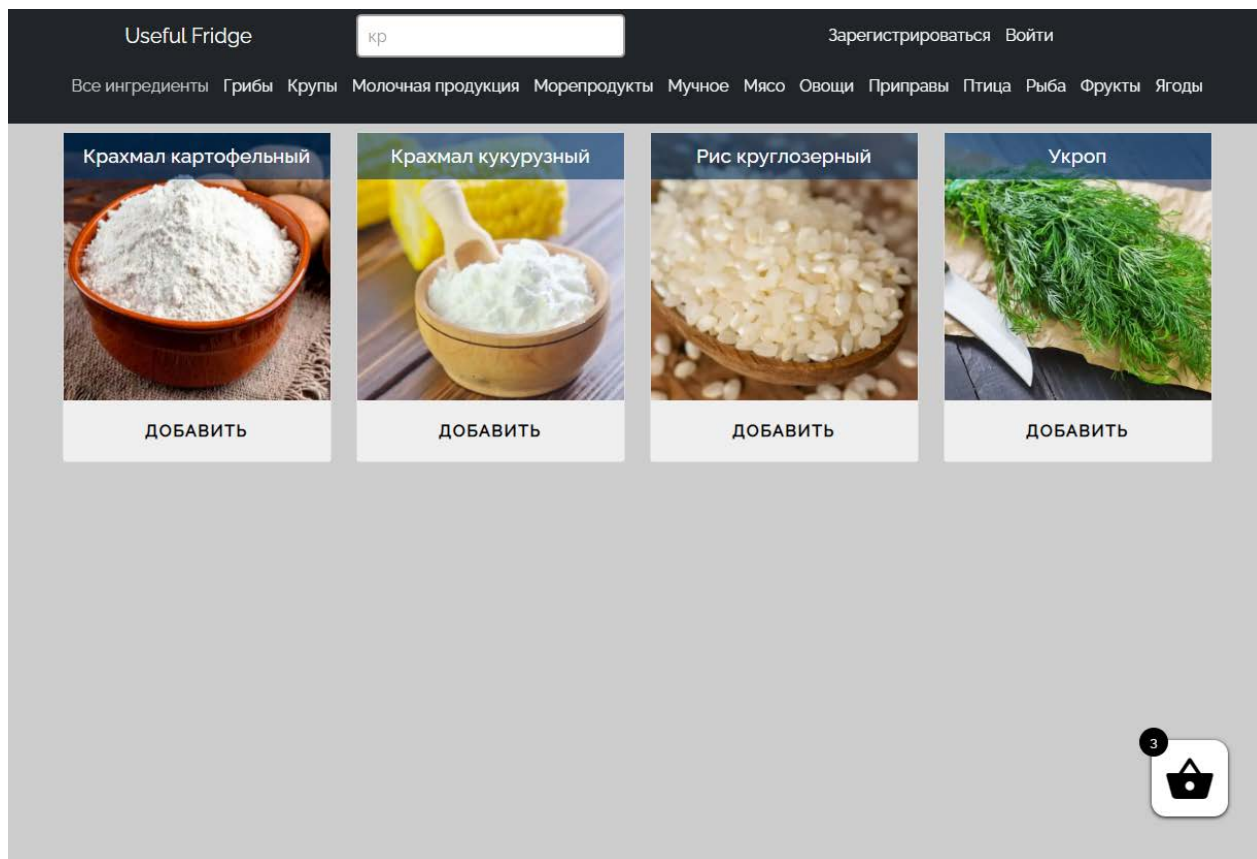


Рисунок 4.7 – Результат поиска с введенным содержимым «кр»

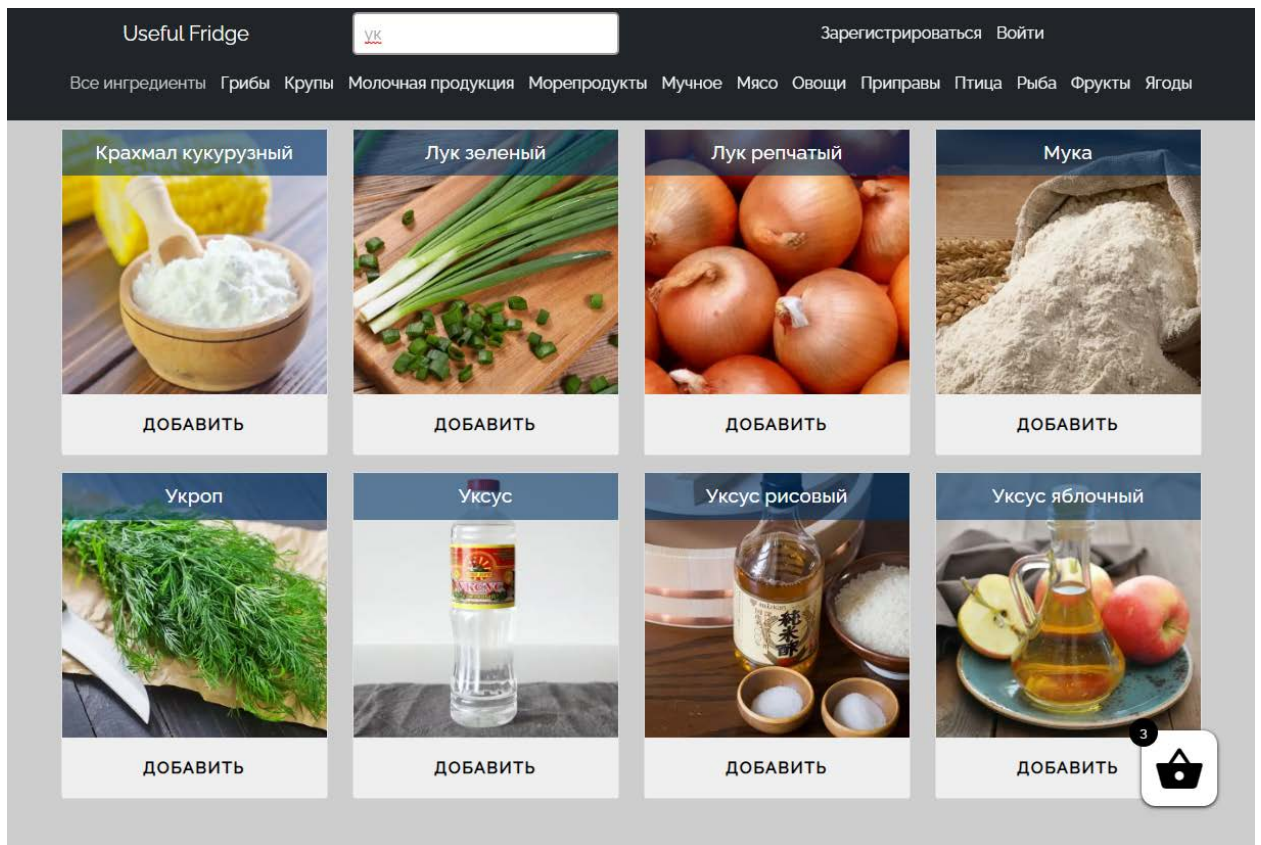


Рисунок 4.8 – Результат поиска с введенным содержимым «ук»

Для пользователей также доступна «Корзина», в которой размещаются выбранные ингредиенты перед подбором рецептов. Интерфейс данного окна представлен на рисунке 4.9.

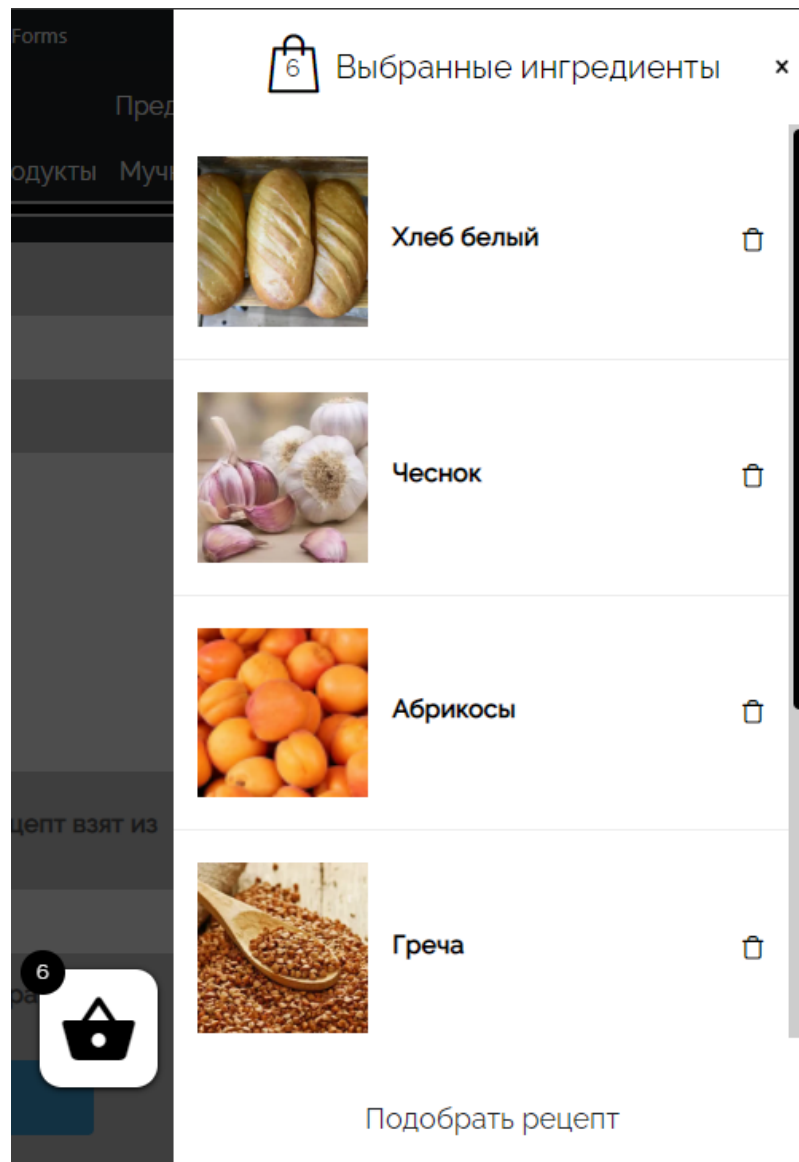


Рисунок 4.9 – «Корзина» с расположенными в ней ингредиентами

Когда нужные продукты помещены в «корзину», пользователь может перейти по ссылке «Подобрать рецепт», чтобы отобразилось модальное окно с выбором времени приёма пищи и типа питания. Демонстрация данного функционала приведена на рисунках 4.10 – 4.12.

Исходный файл «cart.js», где показан функционал отображения корзины приведен в листинге Д.1 приложения Д.

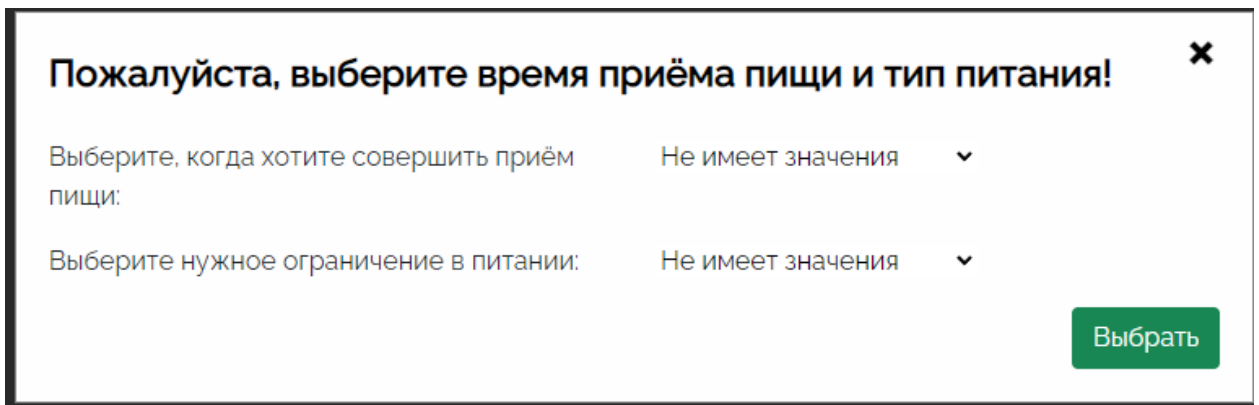


Рисунок 4.10 – Модальное окно с выбором времени приёма пищи и типа питания

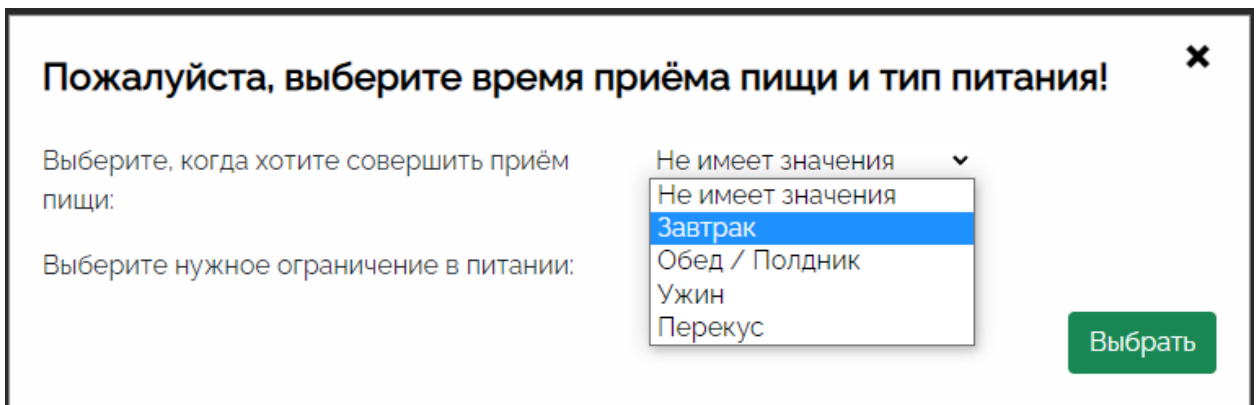


Рисунок 4.11 – Варианты выбора времени приёма пищи

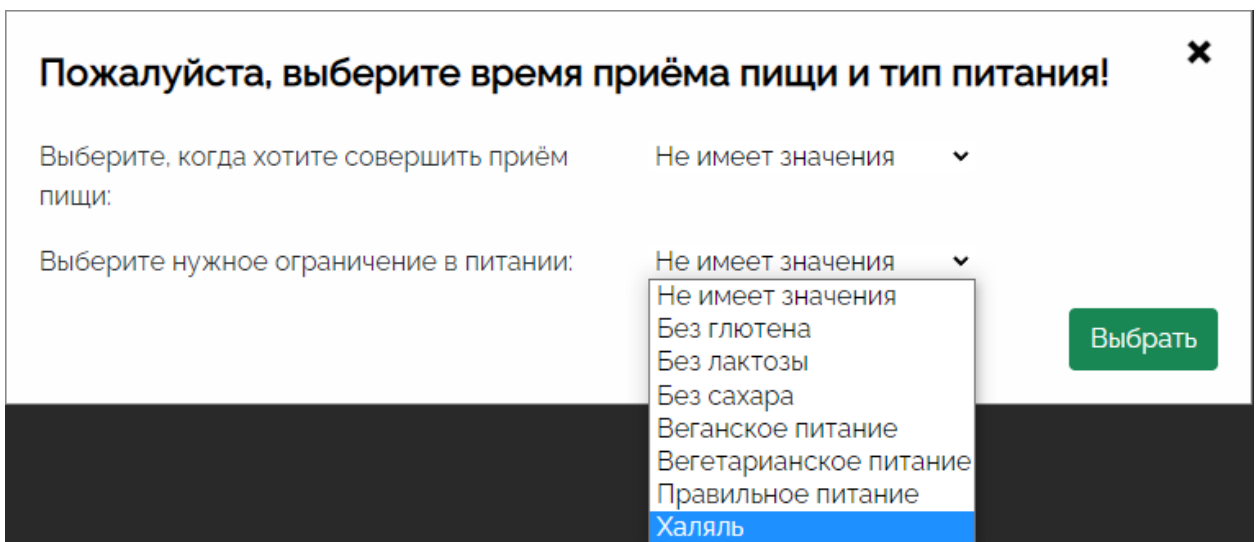


Рисунок 4.12 – Варианты выбора типа питания

После того, как выбраны необходимые параметры откроется страница с подобранными рецептами, которые будут отсортированы по предыдущим параметрам и выстроены по убыванию соответствия ингредиентов. Демонстрация страницы «Рецепты» представлена на рисунке 4.13.

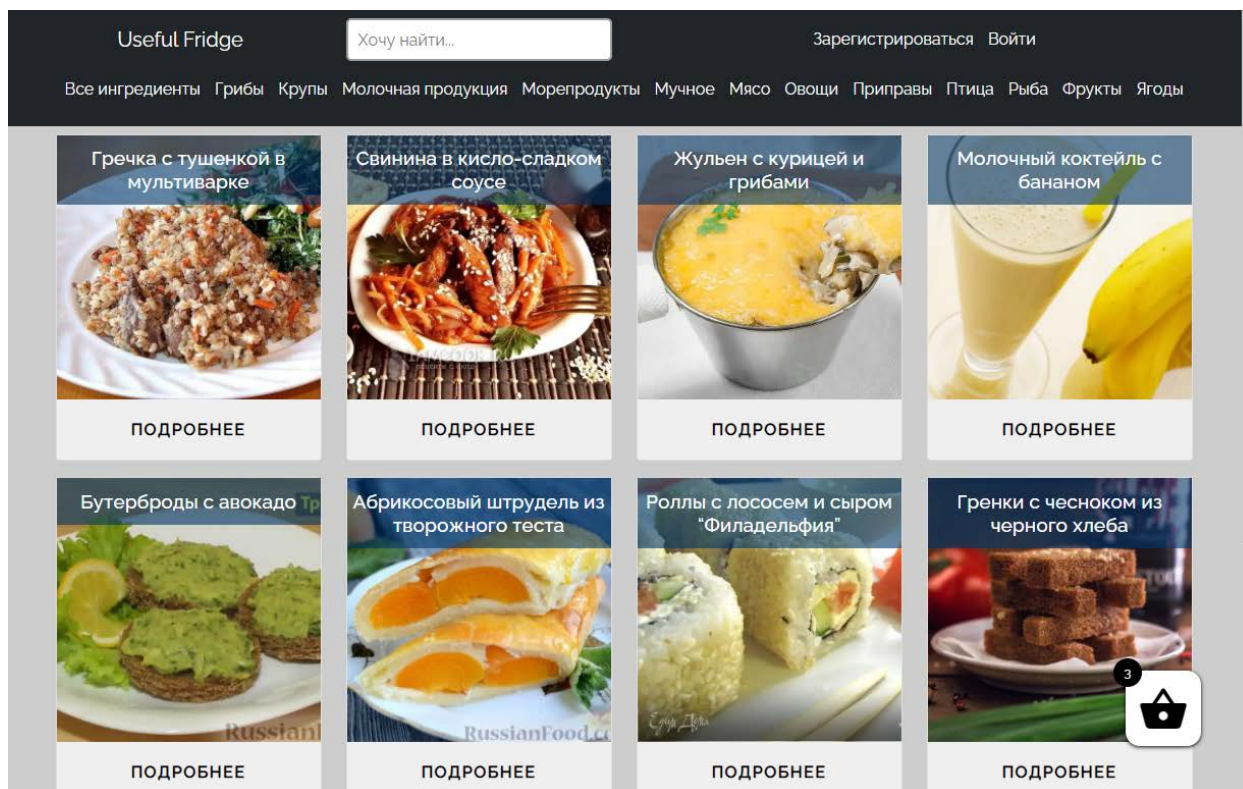


Рисунок 4.13 – Страница «Рецепты»

Если нажать на кнопку «Подробнее», то можно перейти в описание самого рецепта, которое содержит список необходимых ингредиентов и подробную инструкцию по приготовлению с фотографиями. Демонстрация данного функционала приведена на рисунках 4.14 – 4.15. Также во избежание проблем с авторскими правами снизу каждого описания размещена ссылка на источник рецепта, если он взят из интернета.

Гречка с тушенкой в мультиварке



Ингредиенты: Гречка, Тушенка из говядины, Лук репчатый, Морковь, Масло оливковое, Чеснок, Масло сливочное

Шаг 1 – Мелко нарезаем лук и морковь. Затем в мультиварку добавляем оливкового масла, выставляем режим "выпечка" на 20 минут. Затем закидываем овощи и закрываем крышкой. Иногда помешиваем.



Рисунок 4.14 – Подробная инструкция по приготовлению одного из блюд (Начало)



Шаг 5 – Затем подаем к столу. Приятного аппетита!



Источник https://obvar.ru/receptas/grechka_s_tushenkoi_v_multivarke-to306.html

Рисунок 4.15 – Подробная инструкция по приготовлению одного из блюд (Продолжение)

4.2 Вид страниц веб-приложения

Снимки страниц веб-приложения изображены на рисунках А.1 – А.8 приложения А.

Исходный файл «index.php», где показано отображение основного содержимого главной страницы представлен в листинге Е.1 приложения Е. Исходный файл «header.php», где показано отображение основного содержимого «шапки» сайта представлен в листинге Ж.1 приложения Ж. Исходный файл «suggest_recipe.php», где показано отображение основного содержимого страницы «Предложить свой рецепт» представлен в листинге З.1 приложения З. Исходный файл «recipes.php», где показано отображение основного содержимого страницы «Рецепты» представлен в листинге И.1 приложения И. Исходный файл «functions.php», где показаны основные серверные настройки сайта представлен в листинге К.1 приложения К.

5 ТЕСТИРОВАНИЕ РАЗРАБОТАННОГО ВЕБ-ПРИЛОЖЕНИЯ

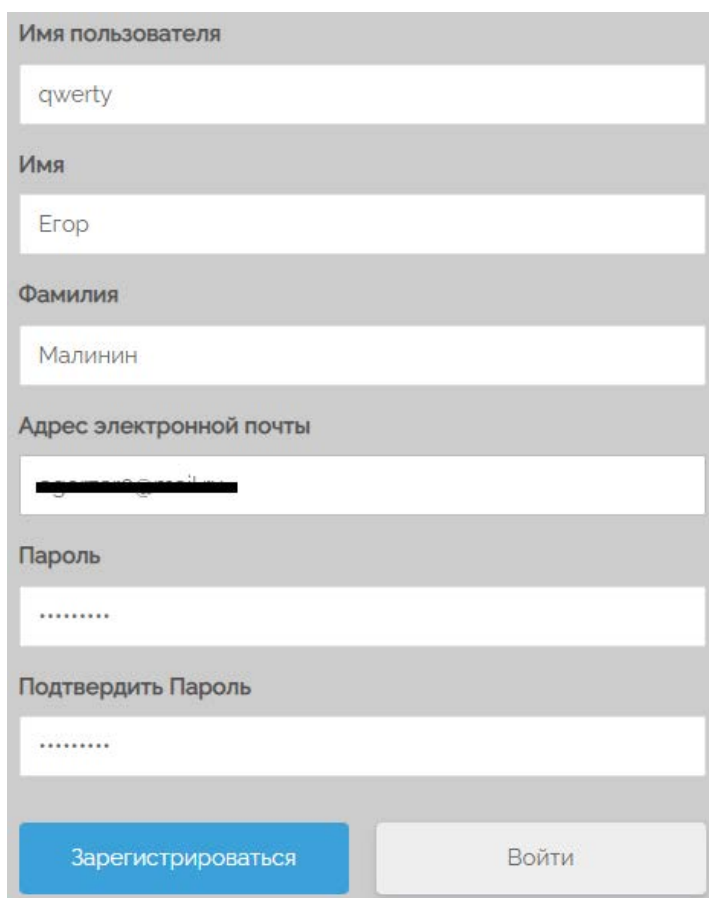
5.1 Методология тестирования

На текущий момент разработки веб-приложение прошло «альфа»-тестирование, то есть протестировано разработчиком, так как веб-приложение разрабатывается и находится на локальном компьютере. Таким образом, производится системное тестирование веб-приложения.

5.2 Проведение процедуры тестирования

5.2.1 Регистрация и авторизация пользователя

Заполнение экранной формы регистрации нового пользователя приведена на рисунке 5.1.

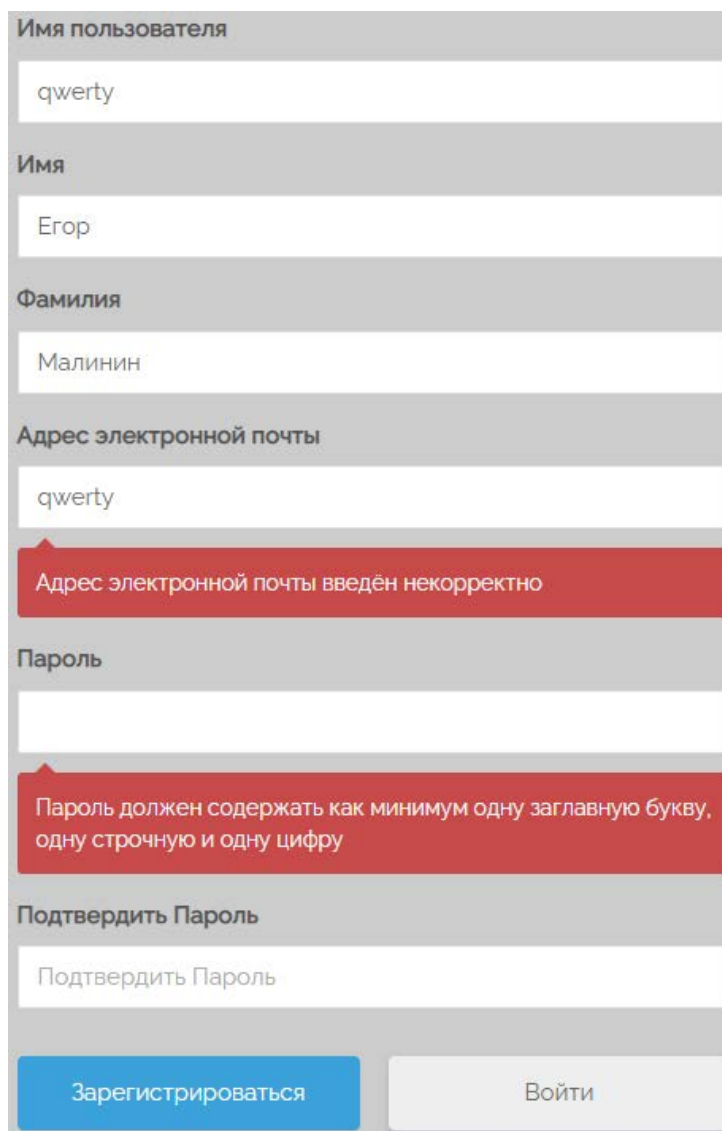


The image shows a registration form with the following fields and buttons:

- Имя пользователя: qwerty
- Имя: Егор
- Фамилия: Малинин
- Адрес электронной почты: [redacted]
- Пароль: [redacted]
- Подтвердить Пароль: [redacted]
- Buttons: Зарегистрироваться (blue), Войти (grey)

Рисунок 5.1 – Заполненная форма регистрации нового пользователя

В форме присутствует проверка на правильность введенных значений и ограничение на слишком простые пароли. Пароль должен содержать как минимум одну заглавную букву, одну строчную и одну цифру. Если пользователь заполнит поля неправильно, то появятся соответствующие ошибки в форме, которые представлены на рисунке 5.2.



The image shows a registration form with the following fields and error messages:

- Имя пользователя:** Input field containing "qwerty".
- Имя:** Input field containing "Егор".
- Фамилия:** Input field containing "Малинин".
- Адрес электронной почты:** Input field containing "qwerty". Below it is a red error message: "Адрес электронной почты введён некорректно".
- Пароль:** Empty input field. Below it is a red error message: "Пароль должен содержать как минимум одну заглавную букву, одну строчную и одну цифру".
- Подтвердить Пароль:** Input field containing "Подтвердить Пароль".

At the bottom of the form are two buttons: "Зарегистрироваться" (blue) and "Войти" (grey).

Рисунок 5.2 – Ошибки, возникшие в результате ввода некорректных данных

Посетитель сайта не может зарегистрироваться с уже использующимися электронной почтой или логином. Пример заполнения экранной формы регистрации с уже использующейся электронной почтой и именем пользователя представлен на рисунке 5.3.

Имя пользователя

Имя пользователя уже используется на данном сайте

Имя

Фамилия

Адрес электронной почты

Адрес электронной почты уже используется на данном сайте

Пароль

Подтвердить Пароль

Зарегистрироваться Войти

Рисунок 5.3 – Ошибки, возникшие в результате ввода существующих логина и адреса электронной почты

При попытке авторизоваться на сайте с неверным паролем возникают ошибки из-за несоответствия данных, что демонстрируется на рисунке 5.4.

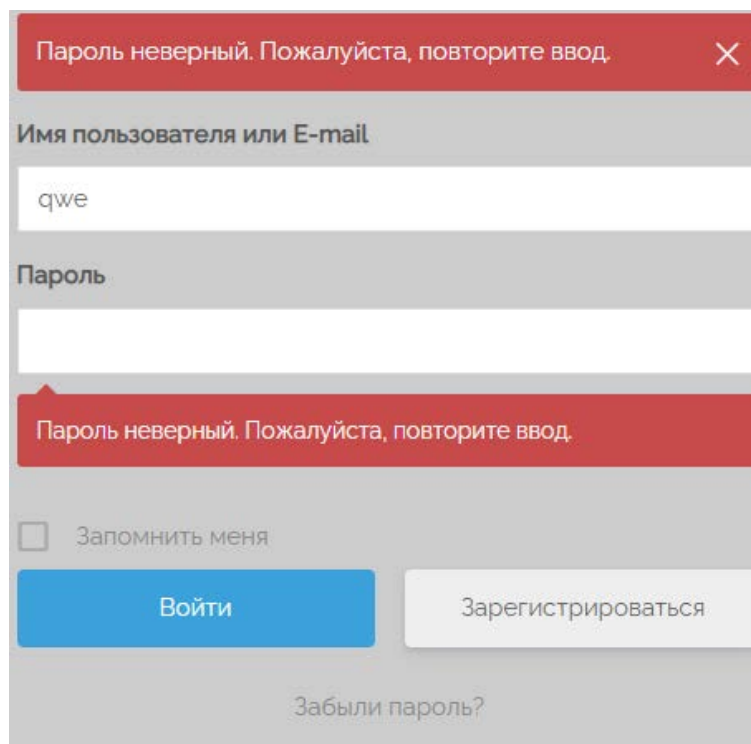


Рисунок 5.4 – Ошибки из-за неверно введенного пароля

После успешной регистрации или авторизации пользователь перенаправляется на главную страницу и у него становятся доступными функции «Предложить свой рецепт» и «Выйти из аккаунта» в основном меню, что представлено на рисунке 5.5.

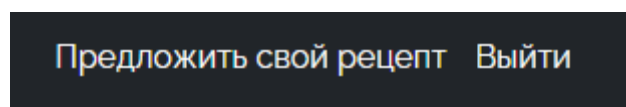


Рисунок 5.5 – Функции, доступные только авторизованному пользователю

5.2.2 Функция «Предложить рецепт»

Пользователь, успешно вошедший в свой аккаунт, способен предложить свой рецепт на рассмотрение администратору, для этого необходимо заполнить соответствующую форму, которая представлена на рисунке 5.6.

Название Вашего рецепта

Блины с творогом

Описание Вашего рецепта

Шаг 1
Приготовьте начинку для блинов. Изюм залейте кипящей водой и оставьте на 15–20 минут. Затем откиньте на дуршлаг, выложите на бумажное полотенце и дайте обсохнуть.

Шаг 2
Творог выложите в глубокую миску. Добавьте ванильный сахар, сметану и взбейте погружным блендером.

Ссылка на источник рецепта (если рецепт взят из интернета)

<https://www.gastronom.ru/recipe/33786/bliny-s-tvorogom>

Отправить рецепт на почту Администратору для рассмотрения

Отправить

Рисунок 5.6 – Заполненная форма «Предложить свой рецепт»

Так как данная форма не обязательна к заполнению, проверки на валидность текста отсутствуют – пользователю позволено писать что угодно.

После отправки рецепта будет выведено соответствующее сообщение, которое представлено на рисунке 5.7.

Название Вашего рецепта

Блины с творогом

Описание Вашего рецепта

Шаг 1
Приготовьте начинку для блинов. Изюм залейте кипящей водой и оставьте на 15–20 минут. Затем откиньте на дуршлаг, выложите на бумажное полотенце и дайте обсохнуть.

Шаг 2
Творог выложите в глубокую миску. Добавьте ванильный сахар, сметану и взбейте погружным

Ссылка на источник рецепта (если рецепт взят из интернета)

<https://www.gastronom.ru/recipe/33786/bliny-s-tvorogom>

Отправить рецепт на почту Администратору для рассмотрения

Отправить

Вы успешно предложили рецепт!

Рисунок 5.7 – Сообщение об успешной отправке формы

Данные из формы отправляются на почту Администратору, который принимает решение о добавлении рецепта в веб-приложение. Пример письма, которое пришло после заполнения предыдущей формы представлен на рисунке 5.8.

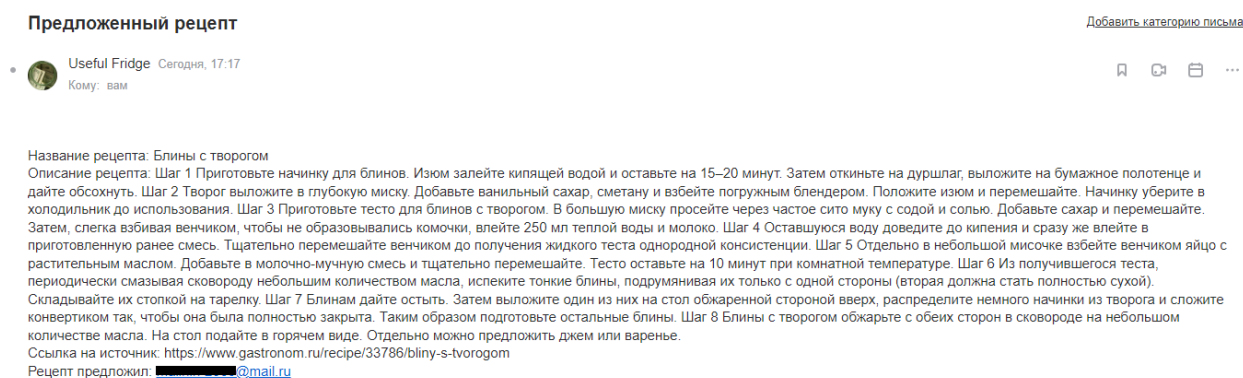


Рисунок 5.8 – Сообщение на почте администратора

5.2.3 Процесс получения рецепта

Полный процесс от добавления ингредиентов в «Корзину» до получения рецепта с выбранными временем приёма пищи и типа питания показан на рисунках Б.1 – Б.7 приложения Б.

ЗАКЛЮЧЕНИЕ

В ходе дипломного проектирования было выполнено следующее:

- проведен анализ предметной области и родственных разработок;
- проанализированы и выбраны средства для разработки веб-приложения;
- определены функциональные и нефункциональные требования к приложению;
- спроектирована архитектура веб-приложения;
- разработана база данных;
- разработано и протестировано программное обеспечение, решающее поставленную задачу.

Было создано веб-приложение, позволяющее:

- пользователям и посетителям, интересующимся кулинарией, подбирать рецепты блюд в желаемом времени приёма пищи и с выбранным типом питания;
- авторизованным пользователям предлагать свой рецепт на рассмотрение Администратору;
- искать продукты по категориям и в поле для ввода текста без перезагрузки страницы по первым напечатанным символам.

На стадии разработки, внедрения и эксплуатации использовалась бесплатная система управления содержимым сайта «WordPress». Озвучены причины, по которым производилась разработка веб-приложения на базе «WordPress». Так же в ходе анализа «WordPress» выяснилось, что она обладает рядом неприятных особенностей. В частности, ее отрицательное влияние на скорость работы системы ввиду большого количества исходного кода. Помимо этого, данная система управления содержимым сайта генерирует дубликаты страниц, а также дубликаты изображений в разных размерах. На основании этого сайт необходимо размещать на качественном

хостинге.

По итогам работы можно сделать следующий вывод:

– создано веб-приложение, позволяющее подбирать рецепты из набора выбранным ингредиентам;

– создано адаптивное веб-приложение, позволяющее пользователям выбирать время приёма пищи и тип питания;

– создано веб-приложение, позволяющее пользователям регистрироваться на сайте, входить в свой аккаунт и восстановить пароль;

– создано веб-приложение, позволяющее авторизованным пользователям предлагать свой рецепт на рассмотрение администратору;

– создано веб-приложение, позволяющее пользователям сортировать продукты по категориям;

– создано веб-приложение, позволяющее пользователям находить продукты и рецепты по первым введённым символам в поле для поиска;

Созданное, адаптированное под различные размеры экрана, от десктопных до мобильных телефонов, веб-приложение удовлетворяет потребностям пользователя в подборе рецептов из набора имеющихся продуктов. Сайт ориентирован на рациональное использование наибольшего числа ингредиентов во избежание их порчи либо замораживания.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Петриченко, И. Чем веб-приложения отличаются от веб-сайтов / И. Петриченко. – Изображение (подвижное ; двухмерное) : электронное // Glo Academy : [канал YouTube]. – 14 октября 2019. – URL: <https://www.youtube.com/watch?v=3ibxeW4itTg> (дата обращения: 22.05.2022).
2. Трофимов, С. Подбери рецепт – лучший помощник в приготовлении еды / С. Трофимов. – Текст : электронный // Optimakomp.ru : [интернет-портал]. – URL: <https://optimakomp.ru/podberi-recept-luchshijj-pomoshhnik-v-prigotovlenii-edy/> (дата обращения: 22.05.2022).
3. Webspoon.ru : [официальный сайт]. – URL: <https://webspoon.ru/> (дата обращения: 22.05.2022). – Текст: электронный.
4. Поварёнок.ру : [официальный сайт]. – URL: <https://www.povarenok.ru/> (дата обращения: 22.05.2022). – Текст: электронный.
5. Гастрономъ : [официальный сайт]. – URL: <https://www.gastronom.ru/> (дата обращения: 22.05.2022). – Текст: электронный.
6. Рецепты по ингредиентам : [официальный сайт]. – URL: <http://recepty-po-ingredientam.ru/> (дата обращения: 22.05.2022). – Текст: электронный.
7. Зандстра, М. PHP. Объекты, шаблоны и методики программирования / М. Зандстра ; [пер. с английского С. Г. Тригуба]. – 3-е изд. – Москва : Диалектика-Вильямс, 2019. – 736 с. : ил. – ISBN 978-5-907144-54-5.
8. Олифер, В. Г. Компьютерные сети. Принципы, технологии, протоколы / В. Г. Олифер, Н. А. Олифер. – 2-е изд., стер. – Санкт-Петербург : Питер, 2019. – 992 с. : ил. – ISBN 978-5-4461-1426-9.
9. Денисов, А. Рейтинг CMS, популярные движки для сайтов / А. Денисов. – Текст : электронный // Рейтинг Рунета : [интернет-портал]. – URL: <http://www.ratingruneta.ru/cms> (дата обращения: 22.05.2022).
10. Полянин, М. Красивый цветной текст в CSS: как это сделать / М. Полянин. – Текст : электронный // Журнал Яндекс Практикума : [интернет-

портал]. – 30 августа 2021. – URL: <https://thecode.media/fill-color/> (дата обращения: 22.05.2022).

11. Дакетт, Д. Javascript и jQuery. Интерактивная веб-разработка / Д. Дакетт ; [пер. с английского М. А. Райтмана]. – Москва : Издательство «Э», 2014. – 319 с. : ил. – ISBN 978-5-699-80285-2.

12. Сухов, К. К. Node.js. Путеводитель по технологии / К. К. Сухов. – Москва : ДМК Пресс, 2015. – 416 с. : ил. – ISBN 978-5-97060-164-8.

13. Полянин, М. Что такое Apache и как он работает / М. Полянин. – Текст : электронный // Журнал Яндекс Практикума : [интернет-портал]. – 7 сентября 2021. – URL: <https://thecode.media/apache/> (дата обращения: 22.05.2022).

14. Книга веб-программиста. Секреты профессиональной разработки веб-сайтов / Б. Хоган, К. Уоррен, М. Уэбер, К. Джонсон, А. Годин ; [пер. с английского А. Силкина]. – Санкт-Петербург : Питер, 2013. – 288 с. : ил. – ISBN 978-5-459-01510-2.

15. Антонов, И. В. Выбор подхода к разработке веб-приложений бакалаврами направления подготовки "Информационные системы и технологии" / И. В. Антонов, Ю. В. Бруттан // Вестник Псковского государственного университета. – 2015. – Вып. 2. – С. 107-113.

ПРИЛОЖЕНИЕ А

Снимки страниц веб-приложения

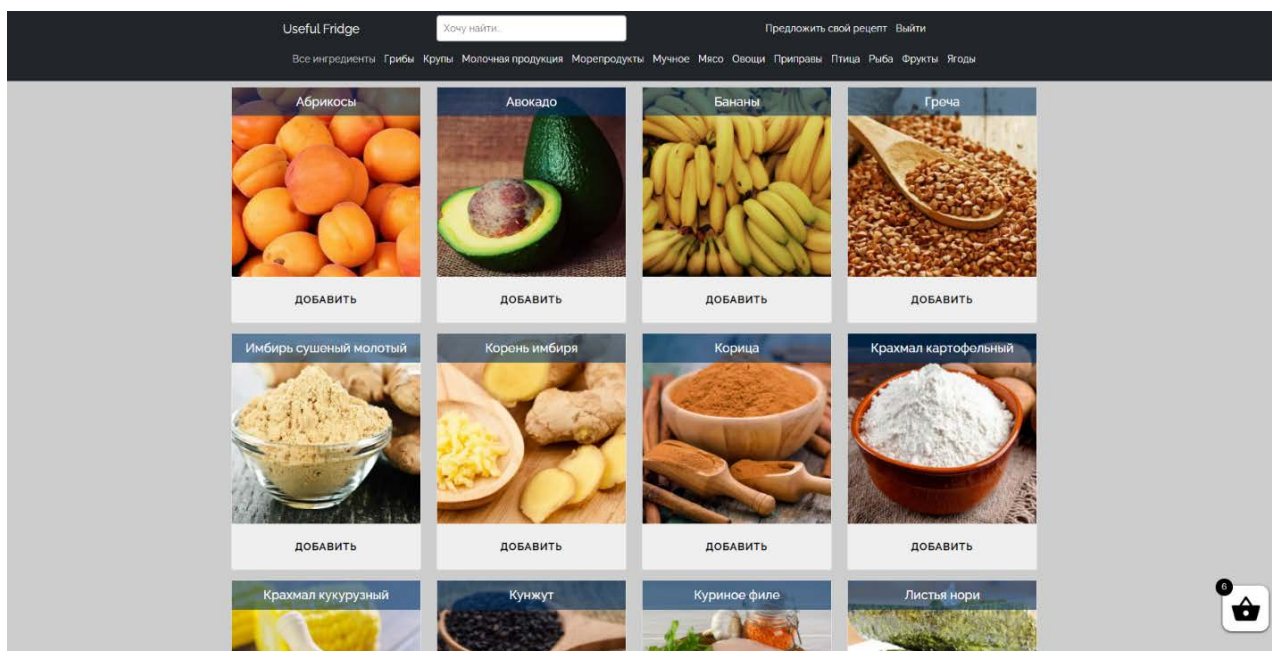


Рисунок А.1 – Главная страница веб-приложения

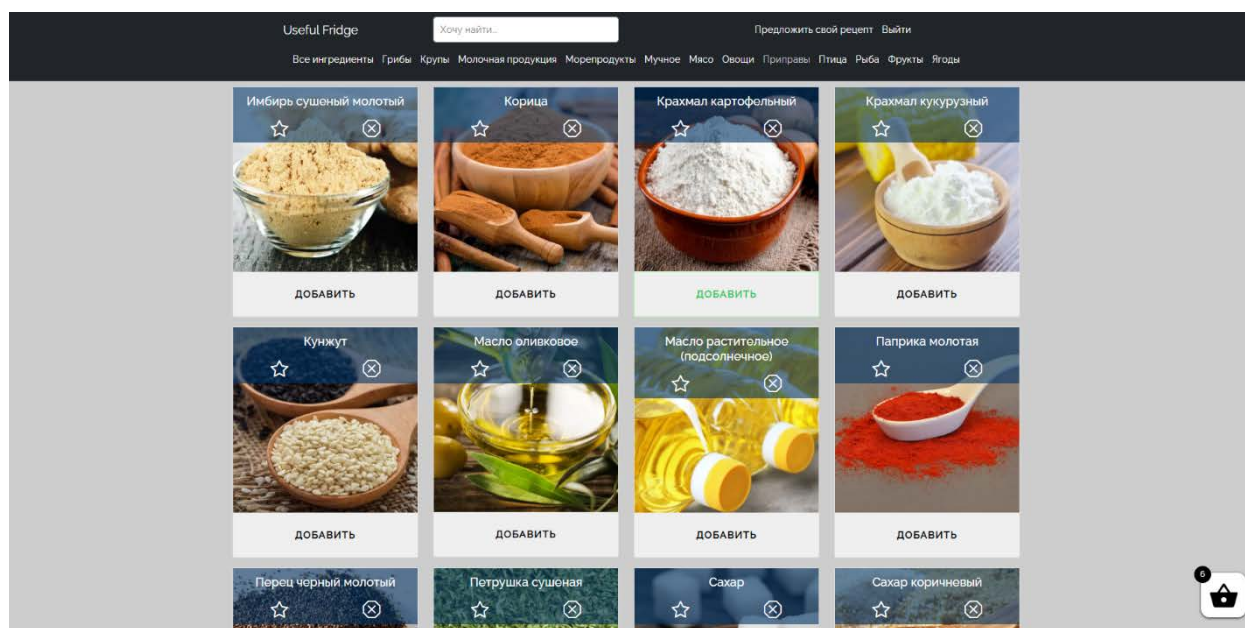
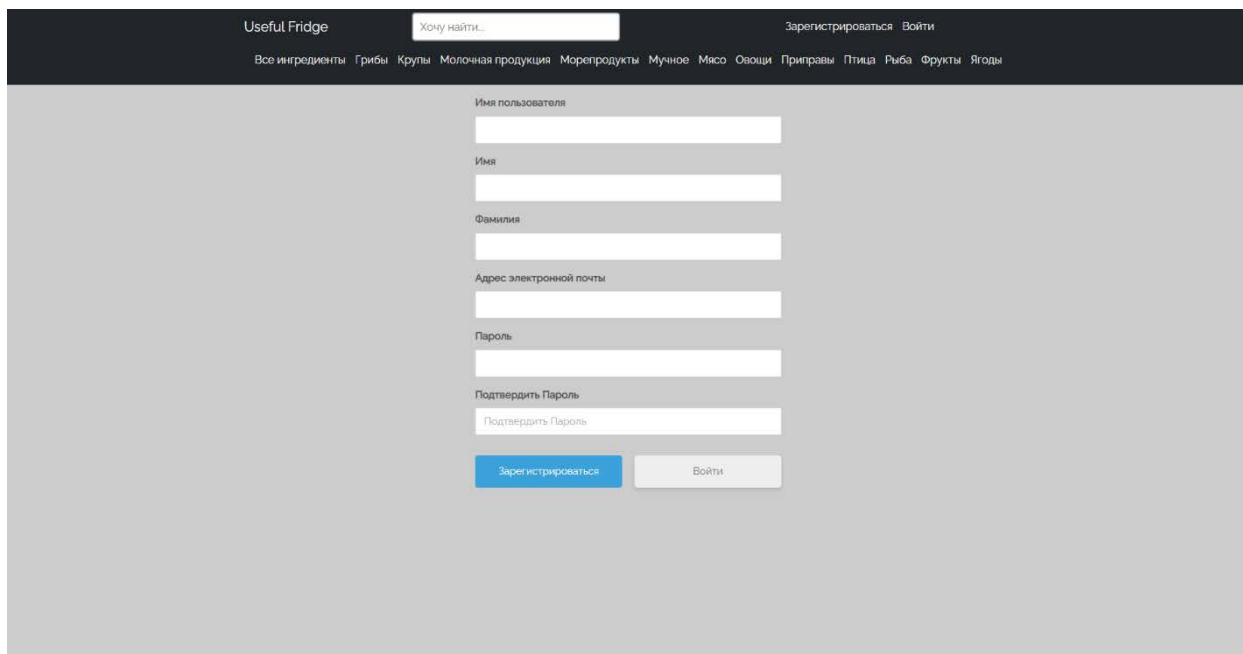


Рисунок А.2 – Одна из категорий ингредиентов. Страница «Приправы»



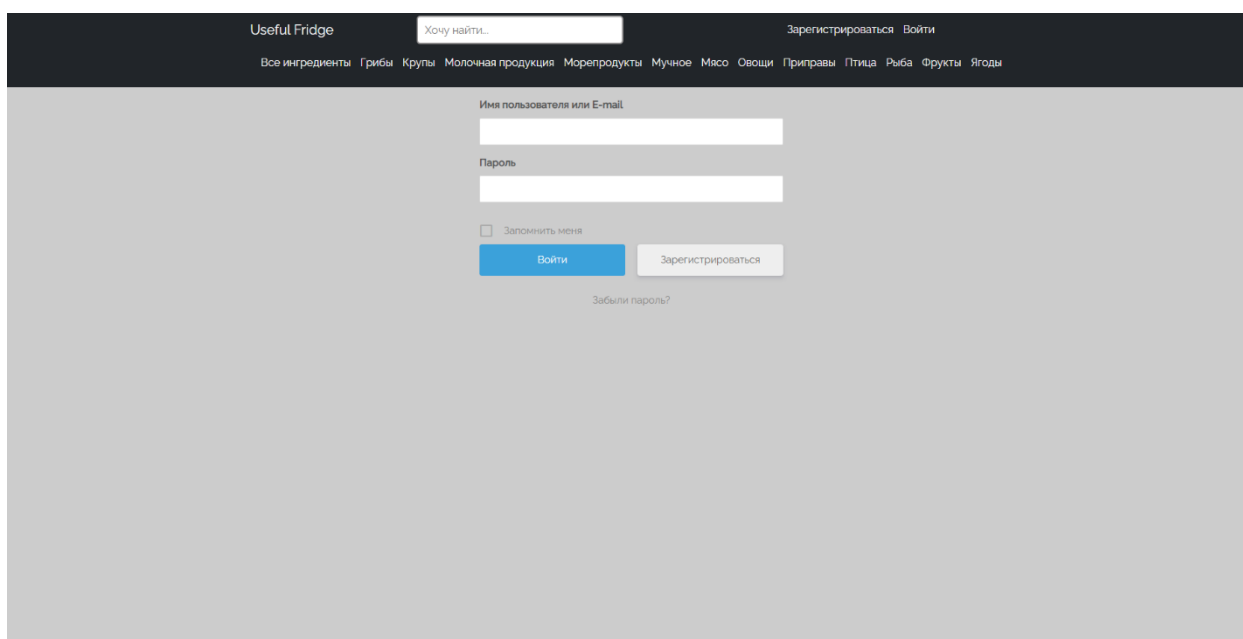
Useful Fridge Хочу найти... Зарегистрироваться Войти

Все ингредиенты Грибы Крупы Молочная продукция Морепродукты Мучное Мясо Овощи Приправы Птица Рыба Фрукты Ягоды

Имя пользователя
Имя
Фамилия
Адрес электронной почты
Пароль
Подтвердить Пароль

Зарегистрироваться Войти

Рисунок А.3 – Страница регистрации



Useful Fridge Хочу найти... Зарегистрироваться Войти

Все ингредиенты Грибы Крупы Молочная продукция Морепродукты Мучное Мясо Овощи Приправы Птица Рыба Фрукты Ягоды

Имя пользователя или E-mail
Пароль

Запомнить меня

Войти Зарегистрироваться

Забыли пароль?

Рисунок А.4 – Страница авторизации

The screenshot shows the password recovery page of the Useful Fridge website. At the top, there is a dark navigation bar with the site name 'Useful Fridge' on the left, a search bar with the placeholder text 'Хочу найти...', and links for 'Зарегистрироваться' and 'Войти' on the right. Below the navigation bar, a horizontal menu lists various food categories: 'Все ингредиенты', 'Грибы', 'Крупы', 'Молочная продукция', 'Морепродукты', 'Мучное', 'Мясо', 'Овощи', 'Приправы', 'Птица', 'Рыба', 'Фрукты', and 'Ягоды'. The main content area has a light gray background and contains the text 'Чтобы сбросить пароль, введите свой email адрес' above a white input field with the placeholder 'Введите свой email адрес'. Below the input field is a blue button labeled 'Сбросить мой пароль'.

Рисунок А.5 – Страница восстановления пароля

The screenshot shows the 'Propose your recipe' page of the Useful Fridge website. The navigation bar and menu are identical to the previous screenshot. The main content area has a light gray background and contains the following elements: a label 'Название Вашего рецепта' above a white input field; a label 'Описание Вашего рецепта' above a larger white text area; a label 'Ссылка на источник рецепта (если рецепт взят из интернета)' above a white input field; and a blue button labeled 'Отправить' at the bottom. Below the button, there is a small note: 'Отправить рецепт на почту Администратору для рассмотрения'.

Рисунок А.6 – Страница «Предложить свой рецепт»

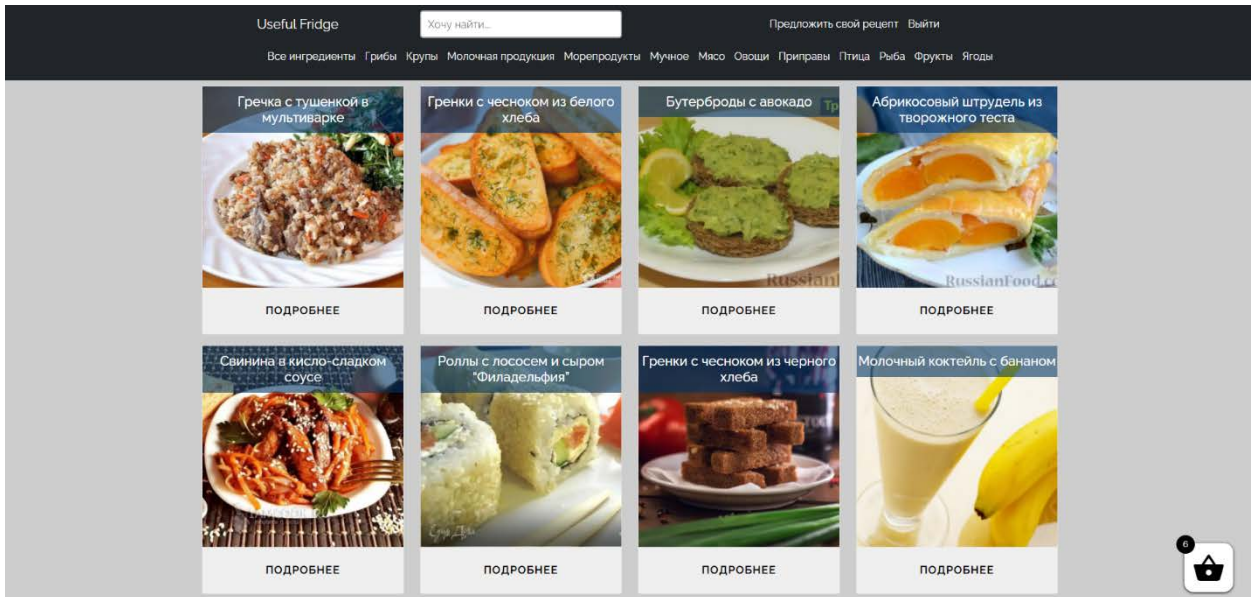


Рисунок А.7 – Страница «Рецепты»

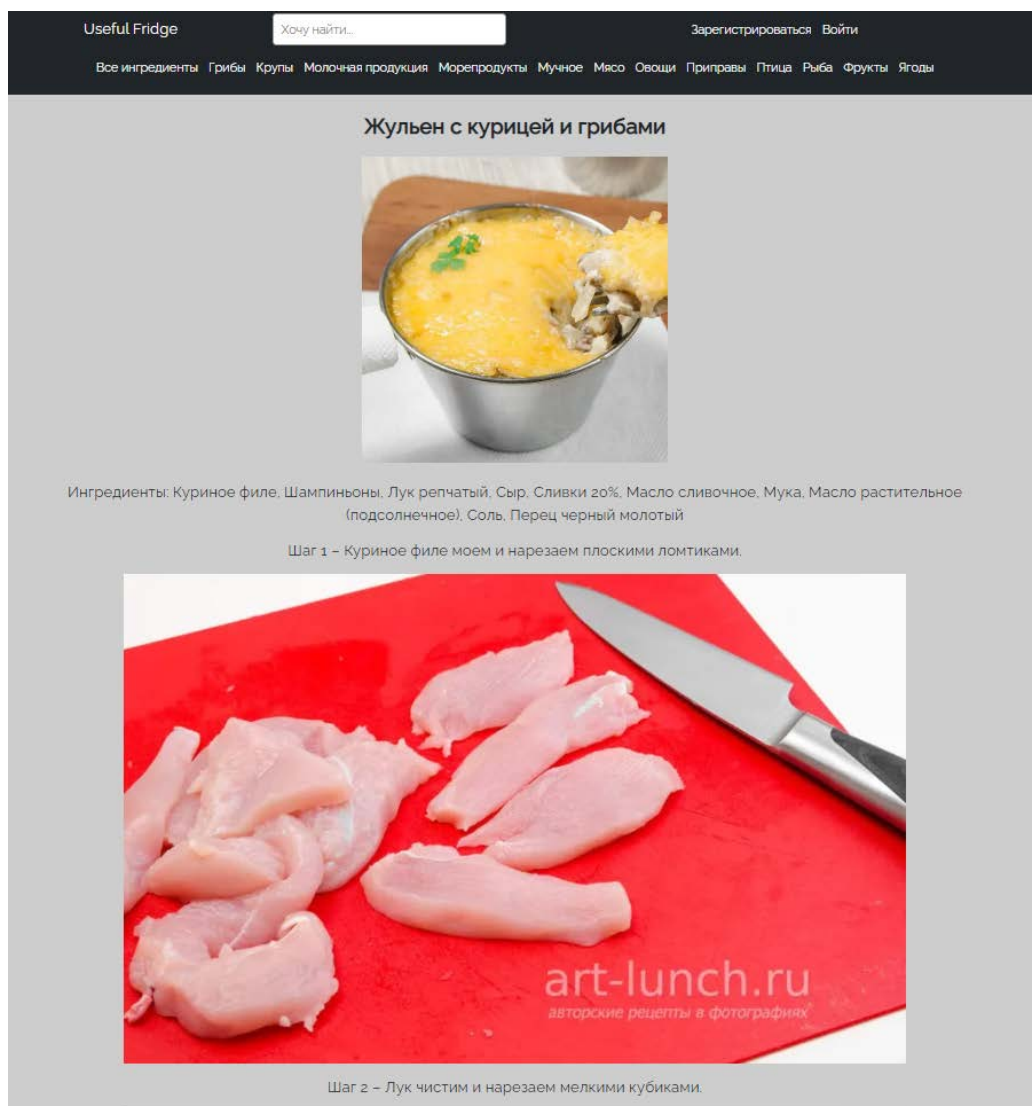


Рисунок А.8 – Страница с инструкцией про приготовлению

ПРИЛОЖЕНИЕ Б

Процесс получения рецепта

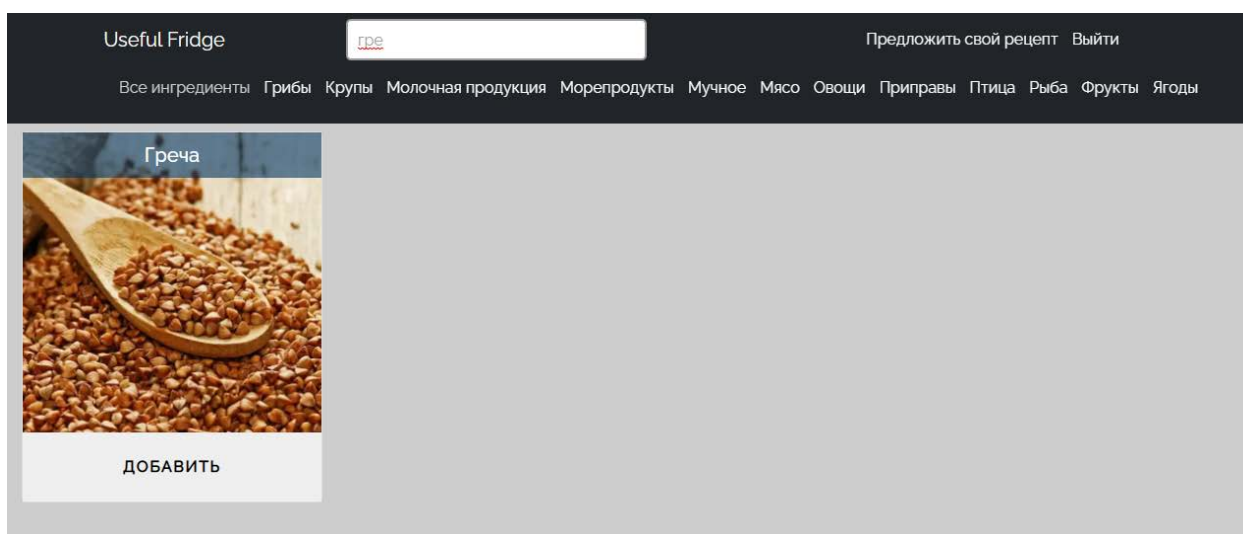


Рисунок Б.1 – Поиск продукта в поле ввода

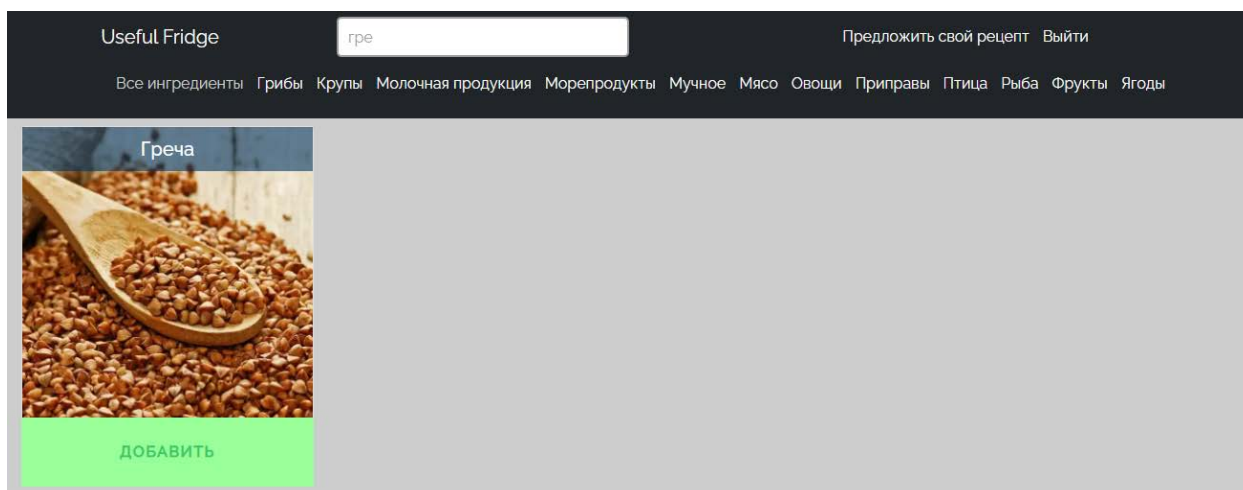


Рисунок Б.2 – Добавление ингредиента в «Корзину»

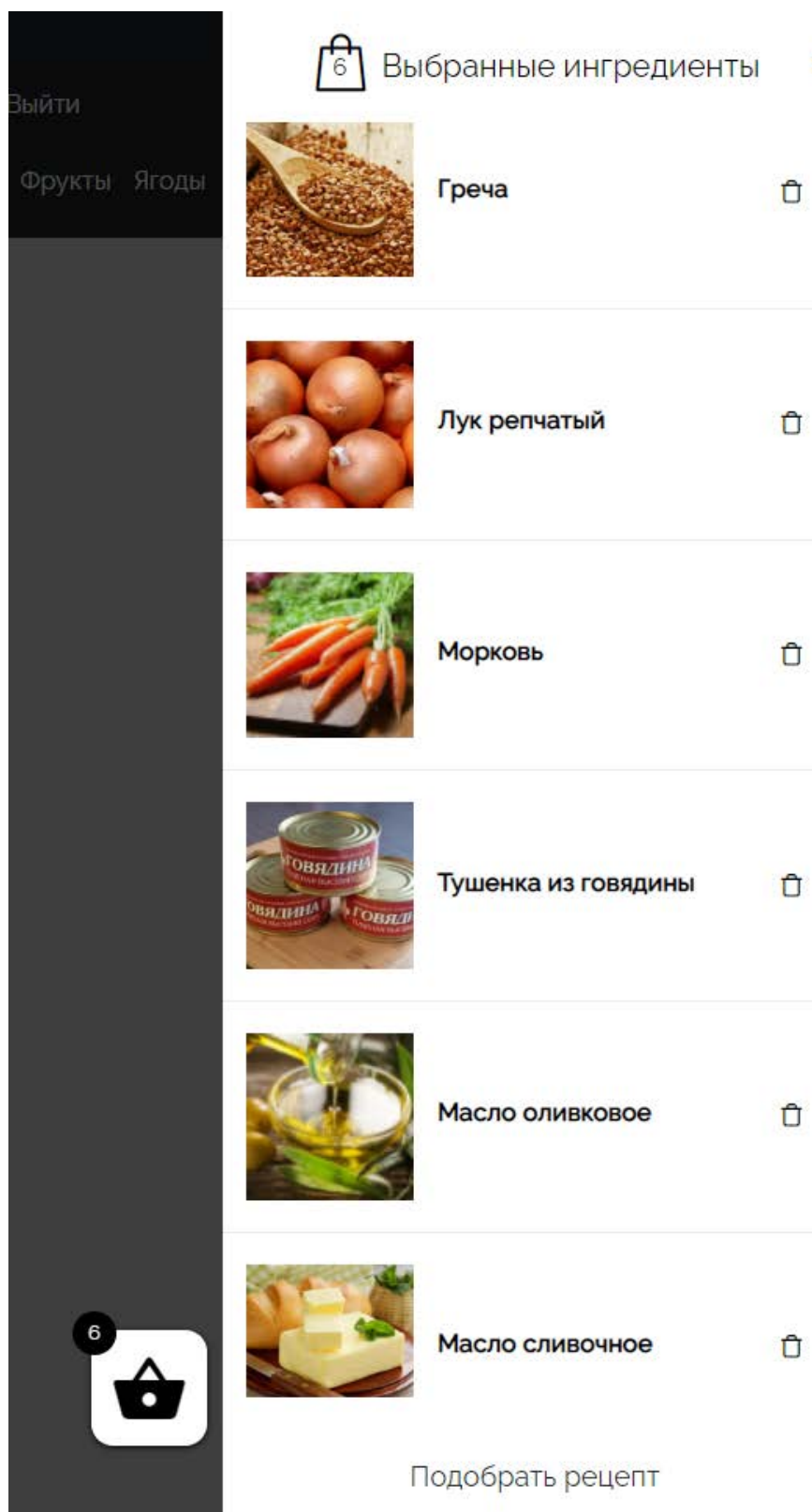


Рисунок Б.3 – Содержимое «Корзины» после добавления всех ингредиентов

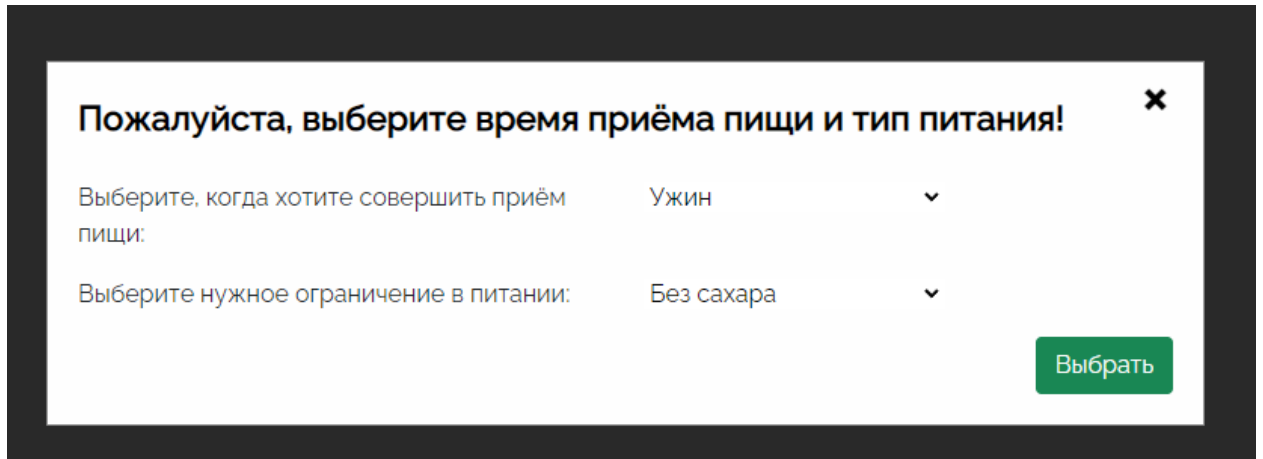


Рисунок Б.4 – Выбранные время приёма пищи и тип питания

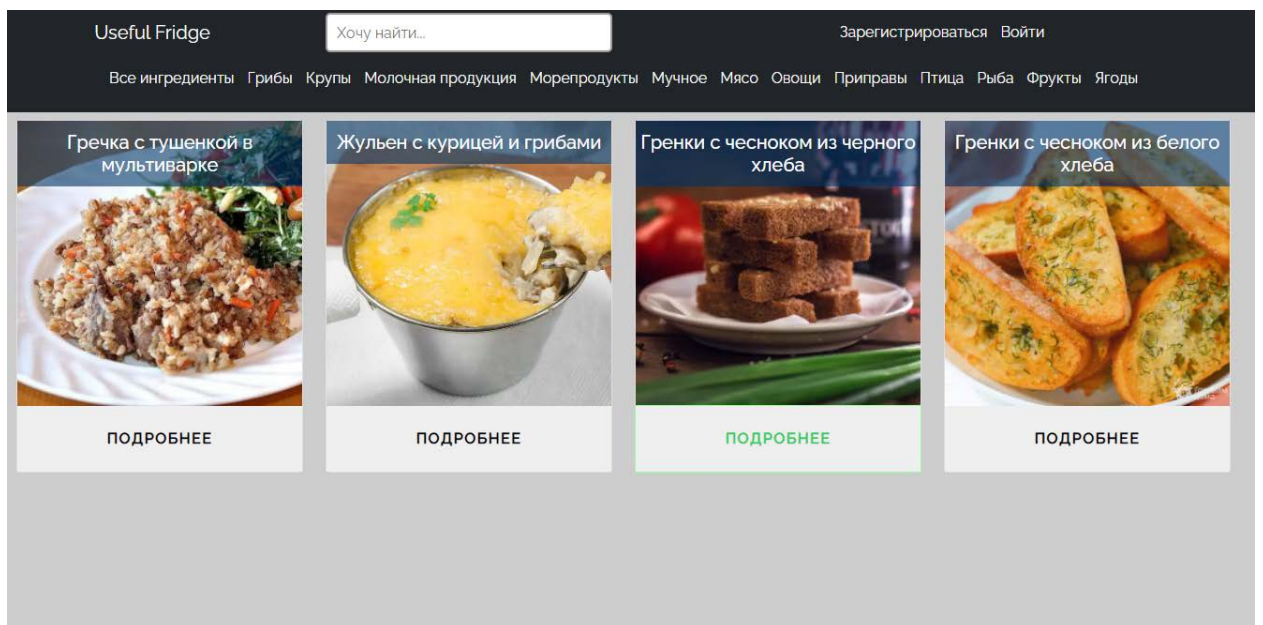


Рисунок Б.5 – Вывод рецептов, отсортированных по совпадению ингредиентов, с учётом предыдущих настроек

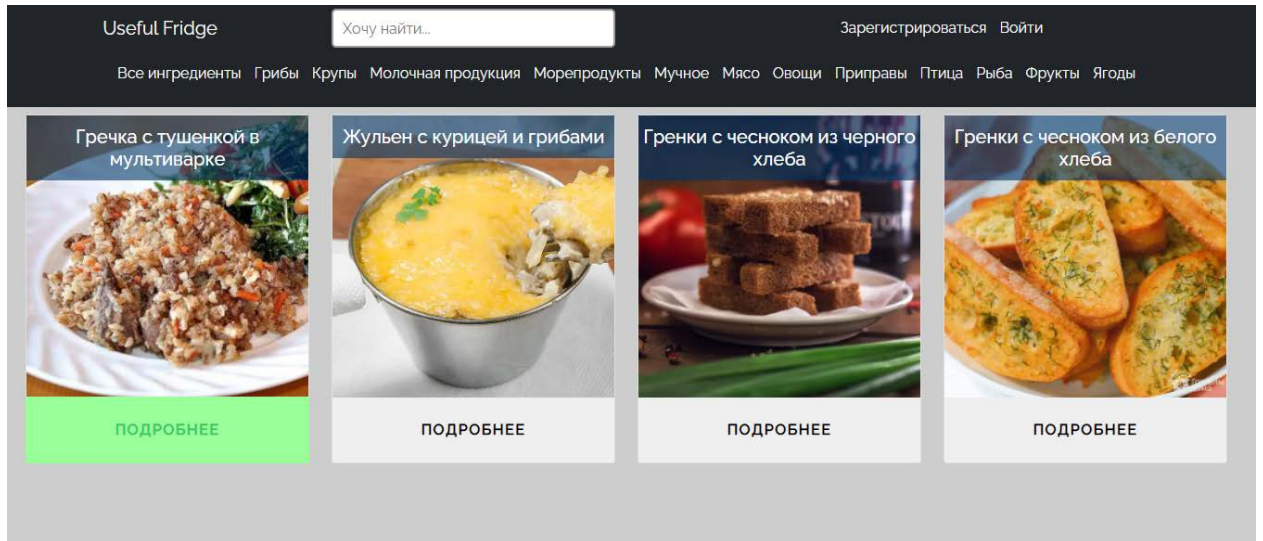


Рисунок Б.6 – Выбор наилучшего результата

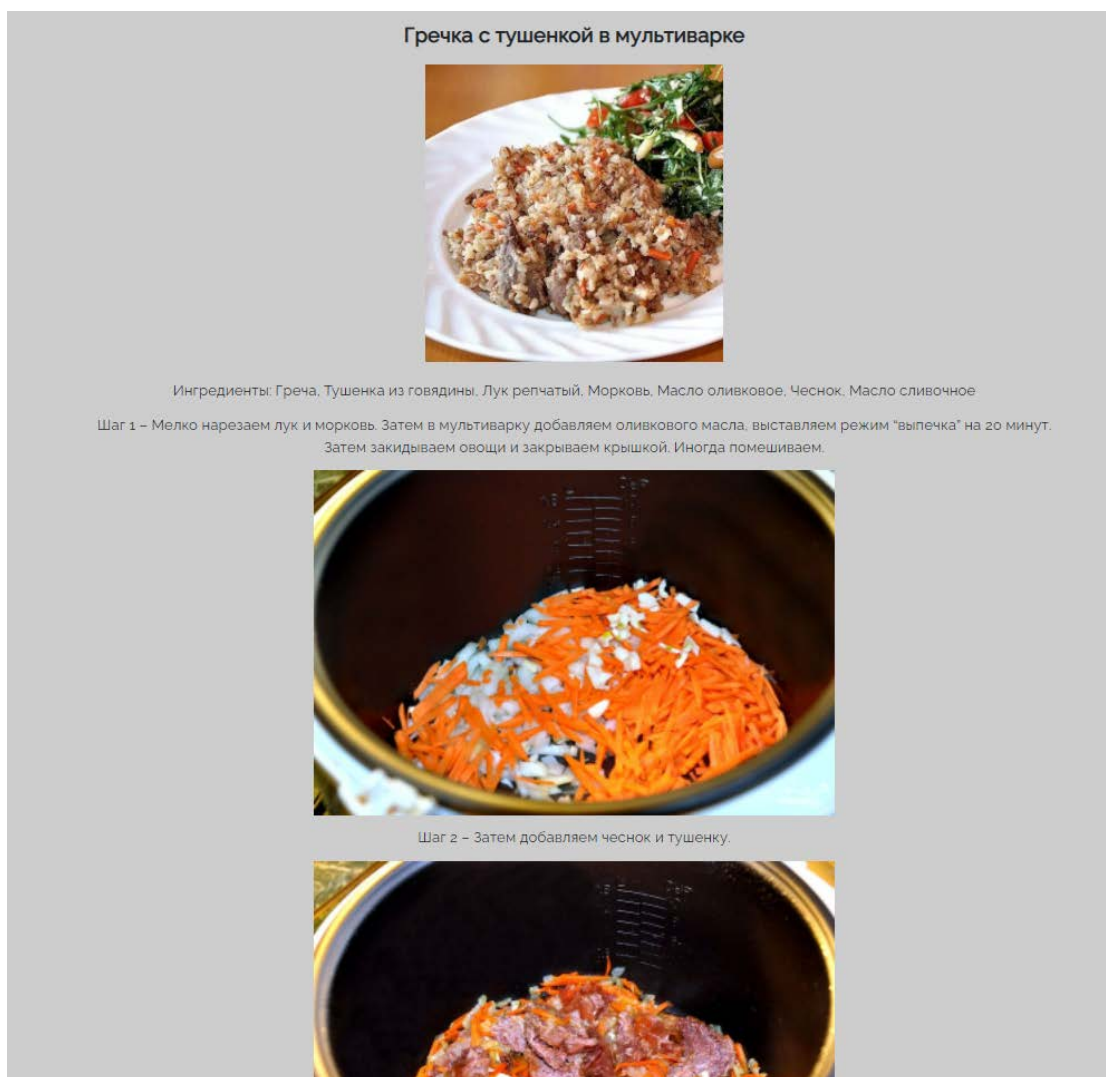


Рисунок Б.7 – Просмотр инструкции по приготовлению

ПРИЛОЖЕНИЕ В

Исходный код javascript-файла «main.js»

Листинг В.1 – Исходный код файла «main.js»

```
$(document).ready(function($) { //необходимая проверка на то, что все элементы
                                на странице подгрузились, чтобы только после
                                этого выполнять javascript код

var modal = $("#myModal"); //переменная, хранящая ссылку на модальное окно
                                для выбора времени приема пищи и типа питания
var modalBtn = $("#myModalBtn"); //переменная, хранящая ссылку на кнопку
                                «Выбрать» в этом модальном окне
var modalSpan = $("#modalClose"); //переменная, хранящая ссылку на иконку
                                для закрытия «X» в этом модальном окне

modalSpan.click(() => { //обработка события по клику на «X» для закрытия
                        модального окна
    modal.css({display: "none"});
});

modal.click((event) => { //обработка события по клику, чтобы модальное
                        окно не закрывалось при клике на её содержимое
    if(event.target == $(".myModal-content")[0]
        || event.target == $(".myModal-content p")[0]
        || event.target == $(".myModal-content div")[0]
        || event.target == $(".myModal-content select")[0]
        || event.target == $(".myModal-content p")[1]
        || event.target == $(".myModal-content div")[1]
        || event.target == $(".myModal-content select")[1]
        || event.target == $(".myModal-content p")[2]
        || event.target == $(".myModal-content div")[2]
        || event.target == $(".myModal-content div")[3]
        || event.target == $(".myModal-content div")[4]
        || event.target == $(".myModal-content div")[5]
        || event.target == $(".myModal-content div")[6]
        || event.target == $(".myModal-content div")[7]
        || event.target == $(".myModal-content div")[8]
        || event.target == $(".myModal-content div")[9]
        || event.target == $(".myModal-content div")[10]) {
        modal.css({display: "block"});
    } else modal.css({display: "none"});
});

$("#submit-button").click(() => { //обработка события по клику для
                                сохранения выбранных времени приема
                                пищи и типа питания в хранилище сессии
                                - хранилище данных в браузере,
                                которые обновляются только при
                                закрытии вкладки или самого браузера

    let lsItem1 = sessionStorage.getItem('lsChoice1');
    let choice1 = $(".select")[0].value; //выбор времени приема пищи
    choice1 = lsItem1;
    choice1 = $(".select")[0].value;
    sessionStorage.setItem('lsChoice1', choice1);
    let lsItem2 = sessionStorage.getItem('lsChoice2');
    let choice2 = $(".select")[1].value; //выбор типа питания
    choice2 = lsItem2;
    choice2 = $(".select")[1].value;
```

Продолжение приложения В

```
    sessionStorage.setItem('lsChoice2', choice2);
  });

$("#searchInput").on('input', function(e) { //обработчик события «ввод
    информации в поле input» для
    функционала поиска
    e.preventDefault(); //отключение стандартного поведения поля ввода
    (отключение обновления страницы при нажатии
    enter)
    var value = this.value.toLowerCase().trim(); //переменная, хранящая
    информацию, которую ввел пользователь, она
    преобразует её в нижний регистр и удаляет
    пробельные символы в начале и в конце строки
    var posts = $(".content .card .card-title"); //переменная, хранящая
    заголовки ингредиентов, в будущем нужно
    будет сравнивать то, что ввёл пользователь
    с этими заголовками (названиями)

    if (value != '') { //проверка на отсутствие пустого введённого
        значения
        posts.each(function(elem) { //цикл для перебора всех заголовков
            if(this.innerText.toLowerCase().search(value) == -1) { //если
                совпадений нет,
                this.classList.add('hide'); //то добавить класс «hide»,
                $(this).parents(".eachPost").css({display: "none"});
                //затем скрыть все блоки с этим
                классом в заголовке
            }
        });
    } else { //иначе
        posts.each(function(elem) {
            this.classList.remove('hide'); //удалить класс «hide»
            $(this).parents(".eachPost").css({display: "block"});
            //показать только те блоки, где
            есть совпадения с поиском
        });
    }
});

let splitStrArr = []; //Массив для хранения ингредиентов в рецепте
let cartArr = []; //Массив для хранения ингредиентов в «корзине»
for (let i = 0; i < $(".xoo-wsc-pname").length; i++) {
    cartArr.push($(".xoo-wsc-pname")[i].innerText); //Перебрав всё
    содержимое «козины», добавляем в ранее
    созданный массив названия ингредиентов
}
const intersect = function (nums1, nums2) { //функция для нахождения
    «пересечений» между двумя
    массивами, описанными ранее
    let result = []; //переменная, в которой будет результат «пересечения»
    let map = nums1.reduce((acc, i) => { //объект, который будет заполняться
        содержимым «корзины»
        acc[i] = acc[i] ? acc[i] + 1 : 1; //acc - аккумулятор (встроенный
        параметр, который принимает метод
        reduce для подсчёта количества полей
        в объекте
```

```

    return acc;
  }, {});

  for (let i = 0; i < nums2.length; i++) { //перебор массива с
                                          ингредиентами рецепта

    const current = nums2[i];
    let count = map[current];

    if(count && count > 0) { //сравнение текущего элемента в массиве с
                            ингредиентами «корзины» и текущего элемента
                            в массиве с ингредиентами рецепта
      result.push(current); //заполнение объекта с результатом
                            «пересечений»

      map[current] -= 1;
    }
  }

  return result.length; //возвращение количества ингредиентов, которые
                        «пересеклись»

}

$(".eachPost").each(function(i){
  splitStrArr = $(".card__ingredients")[i].innerText.split(', ');
  let newPer = intersect(splitStrArr, cartArr);
  $(this).css({order: "-" + newPer}); //css стиль order выводит элементы
  на странице в определенной последовательности. Им можно
  задавать приоритет. Чем меньше число, тем левее (перее)
  будет располагаться элемент. Например, order: -10 будет
  стоять перее, чем order: -5, а order: 0 будет стоять
  перее, чем order: 7.
  if ($(this).css('order') == '0') {
    $(this).css({display: "none"}); //скрыть элементы без «пересечений»
  }
});

});

```

ПРИЛОЖЕНИЕ Г

Исходный код javascript-файла «forform.js»

Листинг Г.1 – Исходный код файла «forForm.js»

```
$(document).ready(function($){ //обработчик формы «Предложить рецепт»

    let btn = $(".suggest-recipe__form__submit");
    form = $("#suggest-recipe__form");
    mess = $(".mess");
    inputTitle = $("#suggest-recipe__form__title");
    inputDescr = $("#suggest-recipe__form__descr");
    inputLink = $("#suggest-recipe__form__link");

    btn.click((ev) => { //обработчик события по клику для кнопки «Отправить»
        ev.preventDefault();

        let title = inputTitle.val();
            descr = inputDescr.val();
            link = inputLink.val();

        btn.addClass('waiting-send'); //добавление кнопки класс, который
            отвечает за серый цвет кнопки и
            ограничение повторного нажатия до
            отправки формы

        $.ajax({ //ajax-запрос для отправки данных на сервер и
            выполнения операций после успешного или
            провального ответа от сервера
            url: myPlugin.ajaxurl+'?action=get_in_touch', //путь до сервера
            type: "POST", //тип отправляемого POST-запроса
            processData: false, //отмена автоматического преобразования
            отправляемых данных в тип «строка»
            contentType: false,
            data: form.serialize(), //привести форму в массив
            success: function(){ //при успешном ответе от сервера
                btn.removeClass('waiting-send'); //удалить класс с кнопки
                mess.html('<div class="alert alert-success">Вы успешно
предложили рецепт!</div>'); //вывести сообщение об успешной отправки
            },
            error: function(){ //при провальном ответе от сервера
                mess.html('<div class="alert alert-danger">Ошибка
отправки!</div>'); //вывести соответствующее сообщение
                btn.removeClass('waiting-send'); //удалить класс с кнопки
            }
        });
    });
});
```


ПРИЛОЖЕНИЕ Д

Исходный код javascript-файла «cart.js»

Листинг Д.1 – Исходный код файла «cart.js»

```
var cart = {}; //объект для хранения данных в «корзине»

const checkCart = () => { //проверка на наличие элементов в «корзине»
  if( localStorage.getItem('cart') != null) { //если значение элемента
    в корзине не недопустимое значение null
    cart = JSON.parse(localStorage.getItem('cart')); //то создать
    запись в локальном хранилище (хранилище в
    памяти браузера, которое обновляется только
    при очистке кэша, не сбрасывается даже при
    закрытии браузера), и заполнять его этими
    элементами
  }
}

const renderCart = () => { //функция для отображения «корзины»
  var out = '';
  for (var key in cart){
    out += $(".card-title").text() + cart[key] + '<br>';
  }
  $('#.cartContent').html(out);
};

checkCart();
renderCart();
console.log(cart);
```

ПРИЛОЖЕНИЕ Е

Исходный код php-файла «index.php»

Листинг Е.1 – Исходный код файла «index.php»

```
<?php get_header(); ?> //вывести на экран содержимое файла header.php

<div class="content"> //основное содержимое страницы
  <div class="container"> //содержимое помещается в «контейнер» для
    отображения его по центру экрана
    <div class="row"> //горизонтальное выравнивание элементов

      <?php

        $filter = array( //параметры для вывода элементов по
          определенным критериям из БД
          'product_cat' => get_query_var('product_cat'),
//выводить элементы из текущей (у каждого ингредиента своя) категории
          'posts_per_page' => -1,
//выводить неограниченное количество ингредиентов
          'order' => 'ASC',
//сортировка по алфавиту
          'orderby' => 'title',
//сортировать по алфавиту будет заголовки
          'post_type' => 'product',
//тип постов с ингредиентами
        );
        $query = new WP_Query( $filter );

        if ( $query->have_posts() ){
          while ( $query->have_posts() ){
            $query->the_post();
            //цикл для вывода постов из БД
          }
        }

        <div class="eachPost col-md-4 col-lg-3 col-sm-6 col-6 my-2">
          //классы для корректного отображения
          элементов на разных экранах
          <div class="card text-white">
            <?php
              the_post_thumbnail();
            //вывод картинок ингредиентов,
            получаемых из БД
            ?>

            <div class="card-img-overlay">
              <h5 class="card-title"><?php the_title(); ?></h5>
              //Вывод названий ингредиентов,
              получаемых из БД

            </div>

            <a href="?add-to-cart=?php
echo $post->ID; ?>" rel="nofollow" data-product_id="<?php echo $post->ID; ?>"
data-quantity="1" class="add_to_cart button product_type_simple
add_to_cart_button ajax_add_to_cart">
            //кнопка для отправки ингредиента
            в «корзину»
            <button class="btn_add">Добавить</button>
```

Окончание приложения Е

```

        </a>
      </div>
    </div>
    <?php
    }
  }
  wp_reset_postdata(); //очистка памяти после окончания
                        работы цикла для вывода постов
  ?>
</div>
</div>
<?php get_footer(); ?> //Вывести на экран содержимое
                        файла footer.php
</body>
</html>
```

ПРИЛОЖЕНИЕ Ж

Исходный код php-файла «header.php»

Листинг Ж.1 – Исходный код файла «header.php»

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-
icons@1.8.0/font/bootstrap-icons.css">
  <title><?php echo bloginfo('name'); echo " | "; bloginfo('description');
?></title>

  <?php
    wp_head(); //подключение файл с основными php функциями
              functions.php
  ?>
</head>
<body>
  <header class="header">
    <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
      <div class="container">
        <div class="row">
          <div class="col-lg-3 col-md-3 col-12">
            <a href="/" class="navbar-brand"><?php echo
bloginfo('name'); ?></a> //логотип веб-приложения Useful Fridge
          </div>
          <div class="col-lg-3 col-md-3 col-12">
            <form method="get" action="<?php print
site_url(); ?>" class="searchForm">
              <input type="text" id="searchInput"
class="searchInput" placeholder="Хочу найти..." /> //поле поиска
            </form>
          </div>
          <div id="navbar-menu" class="navbar-nav col-lg-6
col-md-6 col-12">
            <?php wp_nav_menu(); ?>
            //Вывод основного меню сайта
          </div>
        </div>
        <div class="row">
          <div class="col navbar__scroll-menu">
            <ul class="navbar-nav">
              <?php
                $args_cat = array(
                  'taxonomy' => 'product_cat',
                  'orderby' => 'name',
                  'order' => 'ASC',
                  'exclude' => '1',
                  'hide_empty' => 1,
                  'title_li' => '',
                  'current_category' => '',
                );
              ?>
            <?php $url =
$_SERVER["REQUEST_URI"];?>
```

Окончание приложения Ж

```
<li <?php if ($url == "/") {echo 'class="current-cat"';}?><a href="/">Все
ингредиенты</a></li>
<li <?php if ($url == "/mushrooms/") {echo 'class="current-cat"';}?><a
href="/mushrooms">Грибы</a></li>
<li <?php if ($url == "/cereals/") {echo 'class="current-cat"';}?><a
href="/cereals">Крупы</a></li>
<li <?php if ($url == "/milk_products/") {echo 'class="current-cat"';}?><a
href="/milk_products">Молочная продукция</a></li>
<li <?php if ($url == "/seafood/") {echo 'class="current-cat"';}?><a
href="/seafood">Морепродукты</a></li>
<li <?php if ($url == "/flour_products/") {echo 'class="current-cat"';}?><a
href="/flour_products">Мучное</a></li>
<li <?php if ($url == "/meat/") {echo 'class="current-cat"';}?><a
href="/meat">Мясо</a></li>
<li <?php if ($url == "/vegetables/") {echo 'class="current-cat"';}?><a
href="/vegetables">Овощи</a></li>
<li <?php if ($url == "/condiments/") {echo 'class="current-cat"';}?><a
href="/condiments">Приправы</a></li>
<li <?php if ($url == "/bird/") {echo 'class="current-cat"';}?><a
href="/bird">Птица</a></li>
<li <?php if ($url == "/fish/") {echo 'class="current-cat"';}?><a
href="/fish">Рыба</a></li>
<li <?php if ($url == "/fruits/") {echo 'class="current-cat"';}?><a
href="/fruits">Фрукты</a></li>
<li <?php if ($url == "/berries/") {echo 'class="current-cat"';}?><a
href="/berries">Ягоды</a></li>
//вывод категорий ингредиентов
</ul>
</div>
</div>
</div>
</nav>
</header>
```

ПРИЛОЖЕНИЕ 3

Исходный код php-файла «suggest_recipe.php»

Листинг 3.1 – Исходный код файла «suggest_recipe.php»

```
<?php //Вёрстка и вывод формы «Предложить свой рецепт»

/**
 * Template Name: Предложить рецепт
 */

?>

<?php get_header(); ?>

<div class="content">
  <div class="container">
    <div class="suggest-recipe__row row">
      <div class="col suggest-recipe__col">
        <div>
          <form method="post" action="" class="suggest-
recipe__form" id="suggest-recipe__form">
            <div class="suggest-recipe__form__title">
              <label>Название Вашего рецепта</label>
              <input id="suggest-recipe__form__title"
type="text" name="suggest-recipe__form__title" method="post">
            </div>
            <div class="suggest-recipe__form__descr">
              <label>Описание Вашего рецепта</label>
              <textarea id="suggest-recipe__form__descr"
name="suggest-recipe__form__descr"></textarea>
            </div>
            <div class="suggest-recipe__form__link">
              <label>Ссылка на источник рецепта (если
рецепт взят из интернета)</label>
              <input id="suggest-recipe__form__link"
type="text" name="suggest-recipe__form__link" method="post">
            </div>
            <div class="suggest-recipe__form__button">
              <label>Отправить рецепт на почту
Администратору для рассмотрения</label>
              <button class="suggest-
recipe__form__sumbit" type="submit" name="suggest-recipe__form__submit"
method="post">Отправить</button>
            </div>
            <div class="mess"></div>
          </form>
        </div>
      </div>
    </div>
  </div>
</div>

<?php get_footer(); ?>
</body>
</html>
```

ПРИЛОЖЕНИЕ И

Исходный код php-файла «recipes.php»

Листинг И.1 – Исходный код файла «recipes.php»

```
<?php //Вывод и верстка страница «Рецепты»

/**
 * Template Name: Рецепты
 */

?>

<?php get_header(); ?>
<script>
$(window).on('load',function(){ //Проверка, если до этого не были
    //выбраны время приема пищи и тип
    //питания, то показать модальное окно
    if (sessionStorage.getItem('lsChoice1') == null &&
sessionStorage.getItem('lsChoice2') == null)
        $('#myModal').css({display: "block"});
});
</script>
<div class="content">
    <div class="container"> //Вёрстка модального окна с выбором
        //времени приема пищи и типа питания

        <!-- <button class="btn" id="myBtn">
            Открыть модальное окно для выборов (для теста)
        </button> -->

        <div id="myModal" class="mymodal">
            <div class="mymodal-content">
                <div class="row">
                    <div class="col-11" id="myModal__title">
                        <p>Пожалуйста, выберите время приёма
пищи и тип питания!</p>
                    </div>
                    <div class="col-1">
                        <span id="myModalclose">
                            &times;
                        </span>
                    </div>
                </div>
                <div class="row">
                    <div class="col-6">
                        <p>Выберите, когда хотите совершить
приём пищи:</p>
                    </div>
                    <div class="col-6">
                        <select class="select">
                            <option value="53">Не имеет
значения</option>
                            //Каждому выбору присваиваю value, которое
                            //равно ID категории в WordPress
                            <!-- breakfast --><option value="48">Завтрак</option>
                            <!-- dinner --><option value="49">Обед / Полдник</option>
                            <!-- supper --><option value="50">Ужин</option>
                            <!-- snack --><option value="51">Перекус</option>
```

Продолжение приложения И

```

        </select>
    </div>
</div>
<div class="row">
    <div class="col-6">
        <p>Выберите нужное ограничение в
питании:</p>
    </div>
    <div class="col-6 ">
        <select class="select">
            <option value="53">Не имеет
значения</option>
            <!-- without_gluten --><option value="44">Без глютена</option>
            <!-- without_lactose --><option value="46">Без лактозы</option>
            <!-- without_sugar --><option value="45">Без сахара</option>
            <!-- vegan --><option value="6">Веганское питание</option>
            <!-- vegetarian --><option value="5">Вегетарианское питание</option>
            <!-- healthy-diet --><option value="4">Правильное питание</option>
            <!-- halal --><option value="47">Халяль</option>
        </select>
    </div>
</div>
<div class="row">
    <div class="col-12" id="submit-button">
        <a href="recipes"><button
type="submit" class="btn btn-success">Выбрать</button></a>
    </div>
</div>
</div>
</div>
<script type="text/javascript">
var jsLsChoice1 = sessionStorage.getItem(`lsChoice1`); //присваиваю полученные
value в хранилище сессии
var jsLsChoice2 = sessionStorage.getItem(`lsChoice2`);
</script>
<?php
if (isset($_GET['Choice1'])) //необходимо написать GET-запрос на сервер,
который будет переадресовывать ссылку на
аналогичную, но с дополнительными
параметрами
{
    $php1 = $_GET['Choice1']; //назначаю полученные ранее переменные в
новые, но уже для языка php
    $php2 = $_GET['Choice2'];
}
else
{
    echo '<script>'; //Формирую GET-запрос
    echo 'document.location.href="http://usefulfridge/recipes?Choice1=' +
jsLsChoice1 . '+' '&Choice2=' + jsLsChoice2';
    // echo 'document.location.href="' . $_SERVER['REQUEST_URI'] .
'?phpLsChoice1=' + jsLsChoice1 . '+' '&phpLsChoice2=' + jsLsChoice2';
    echo '</script>';
    exit();
}
?>
```



```

<div class="row">

    <?php
    $filter = array(
        'category__and' => array($php1, $php2),
                                //выводим рецепты, подходящие
                                по 2-м параметрам - времени
                                приема пищи и типу питания
        'posts_per_page' => -1,
    );
    $query = new WP_Query( $filter );

    if ( $query->have_posts()){

        //вывод вёрстки постов с рецептами

        while ( $query->have_posts() ){
            $query->the_post();
            ?>
            <div class="eachPost col-md-4 col-lg-3 col-
sm-6 col-6 my-2">

                <div class="card text-white">
                    <?php
                    the_post_thumbnail();
                    ?>

                    <div class="card-img-
overlay">

                        <h5 class="card-
title"><?php the_title(); ?></h5>

                        </div>
                        <a href="<?php echo
get_permalink(); ?>"><button class="btn_add">
                            Подробнее
                        </button></a>
                    </div>
                    <p class="card__ingredients">
                        <?php
the_field('recipes_ingredients'); ?>
                    </p>
                </div>
            <?php
            }
        }
        wp_reset_postdata();
    ?>
    </div>
</div>
</div>
<?php get_footer(); ?>
</body>
</html>

```

ПРИЛОЖЕНИЕ К

Исходный код php-файла «functions.php»

Листинг К.1 – Исходный код файла «functions.php»

```
<?php

//данный файл отвечает за подключение сторонних файлов и
скриптов, за подключение некоторых плагинов WordPress, а
также за работу с хуками. Хуки - это функции, которые
позволяют манипулировать процедурой без изменения файла в
ядре WordPress. Хук может быть применён как к событию (action
hook), так и к фильтру (filter hook).

add_action('wp_enqueue_scripts', 'fridge_styles');

        //хук, который позволяет во время подгрузки основных
        файлов WordPress подключать свои файлы со стилями

add_action('wp_enqueue_scripts', 'fridge_scripts');

        //хук, который позволяет во время подгрузки основных
        файлов WordPress подключать свои файлы со скриптами

function fridge_styles() {
    wp_enqueue_style( 'fridge-style', get_stylesheet_uri() );

        //Подключение основного файла стилей style.css

    wp_enqueue_style( 'bootstrap-style', get_template_directory_uri() .
'/assets/css/bootstrap.min.css');
};

        //Подключение файла со стилями фреймворка
        Bootstrap 5

function fridge_scripts() {
    wp_deregister_script( 'jquery' );
    wp_register_script('jquery', get_template_directory_uri() .
'/assets/js/jquery.js');
    wp_enqueue_script( 'jquery' );

        //Подключение библиотеки jQuery

    wp_enqueue_script( 'fridge-script', get_template_directory_uri() .
'/assets/js/main.js', array('jquery'), null, true);

        //Подключение основного файла
        скриптов main.js

};

add_theme_support('custom-logo');
add_theme_support('menus'); //Подключение дополнительных пунктов в меню
        администратора WordPress

function fridge_add_woocommerce_support() {
    add_theme_support( 'woocommerce' ); //Подключение плагина Woocommerce
}
add_action('after_setup_theme', 'fridge_add_woocommerce_support');
```

Продолжение приложения К

```
add_filter("gettext_ultimate-
member","um_021522_change_reset_password_labels", 10, 3);
function um_021522_change_reset_password_labels( $translation, $text, $domain
){

    if( "To reset your password, please enter your email address or username
below." == $text ){
        $translation = "Чтобы сбросить пароль, введите свой email адрес";
    }else if( "Enter your username or email" == $text ){
        $translation = "Введите свой email адрес";
    }else if( "Please provide your username or email" == $text ){
        $translation = "Пожалуйста, укажите свой адрес электронной почты";
    }else if( "If an account matching the provided details exists, we will
send a password reset link. Please check your inbox." == $text ){
        $translation = "Если учетная запись, соответствующая предоставленным
данным, существует, мы отправим ссылку для сброса пароля. Пожалуйста, проверьте
свой почтовый ящик.";
    }
    return $translation;
}

//Изменение некоторых английских надписей,
которые не были переведены в плагине

add_action('wp_ajax_get_in_touch', 'get_in_touch_mail_function');
add_action('wp_ajax_nopriv_get_in_touch', 'get_in_touch_mail_function');

//Подключение функционала для работы с ajax-
запросами

function get_in_touch_mail_function(){

    //Здесь формируется письмо, которое будет
отправляться на почту Администратору после
отправки формы «Предложить рецепт»

    $title = $_POST['suggest-recipe__form__title'];

    //Принимает данные из полей формы

    $descr = $_POST['suggest-recipe__form__descr'];
    $link = $_POST['suggest-recipe__form__link'];
    $current_user = wp_get_current_user();
    //Получаем email зарегистрированного
пользователя

    $to = "malinin-2000@mail.ru"; //Адрес для отправки письма
    $date = date ("d.m.Y"); //Дата отправки письма
    $time = date ("h:i"); //Часы отправки письма
    $from = $current_user->user_email; //От кого пришло письмо
    $subject = "Предложенный рецепт"; //Тема письма

    $msg = " //Составление самого письма
Название рецепта: $title //Передаются соответствующие
переменные, полученные по ajax-
запросу

Описание рецепта: $descr
Ссылка на источник: $link
Рецепт предложил: $from";
wp_mail($to, $subject, $msg, "From: $from "); //функция отправки письма

wp_die(); //функция конца работы ajax-запроса и очистка
```

```
}  
  
add_action('wp_enqueue_scripts', 'form__func');  
  
function form__func() {  
    //Отсюда получаю данные для ajax-запроса  
    wp_enqueue_script('form-script', get_template_directory_uri() .  
    '/assets/js/forForm.js', array('jquery'), null, true);  
  
    //Указываю путь на сервер, куда будет  
    //отправляться ajax-запрос  
    wp_localize_script('form-script', 'myPlugin', array(  
        'ajaxurl' => admin_url('admin-ajax.php'),  
    ));  
  
}  
  
?>
```