|  |  |
| --- | --- |
|  | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ Информатика и системы управления

КАФЕДРА Системы обработки информации и управления

**РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

***К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ***

***НА ТЕМУ:***

**Информационная система подачи объявлений для мелкооптовой торговли строительными материалами. Модуль фронтенд.**

Студент \_\_ИУ-5 84Б\_\_ **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_А.С.Братухин\_\_**

(Группа) (Подпись, дата) (И.О.Фамилия)

Руководитель ВКР **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_С.В.Черненький**

(Подпись, дата) (И.О.Фамилия)

Консультант **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_ М.В.Черненький**

(Подпись, дата) (И.О.Фамилия)

Нормоконтролер **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_Ю.Н.Кротов\_\_**

(Подпись, дата) (И.О.Фамилия)

2024 г.

АННОТАЦИЯ

Расчётно-пояснительная записка квалификационной работы бакалавра содержит 53 страниц. С приложениями объем составляет 76 страниц. Работа включает в себя 20 таблиц и 25 иллюстраций. В процессе выполнения было использовано 17 источников.

Объектом разработки является фронтенд модуль информационной системы подачи объявлений для мелкооптовой торговли строительными материалами.

Данная работа посвящена разработке автоматизированной информационной системы для работы с объявлениями, которые создаются пользователями в процессе мелкооптовой торговли строительными материалами. Пользователи просматривают объявления других пользователей сервиса, и при желании могут сделать заказ и приобрести интересующий товар. В системе реализованы такие возможности, как добавление, редактирование и снятие объявлений с публикации, загрузка и просмотр фотографий товара, просмотр информации о других пользователях и возможность связаться с ними. В системе реализовано разграничение доступа для неавторизованных гостей и авторизованных пользователей.

Цель работы заключается в программной реализации веб-интерфейса для системы подачи объявлений, постановке задач разработки и описании конструкторско-технологической части.

Пояснительная записка содержит 3 вида приложений. В приложении А приведены графические диаграммы, описывающие архитектуру системы, в приложении Б приведено техническое задание, в приложении В приведена программа и методика испытаний.

СОДЕРЖАНИЕ

[АННОТАЦИЯ 2](#_Toc168866957)

[СОДЕРЖАНИЕ 3](#_Toc168866958)

[СПИСОК ОБОЗНАЧЕНИЙ И СОКРАЩЕНИЙ 5](#_Toc168866959)

[ВВЕДЕНИЕ 6](#_Toc168866960)

[1 ПОСТАНОВКА ЗАДАЧ РАЗРАБОТКИ 7](#_Toc168866961)

[1.1 Цели и задачи проекта 7](#_Toc168866962)

[1.2 Описание предметной области 7](#_Toc168866963)

[1.3 Выбор критериев качества системы 9](#_Toc168866964)

[1.4 Анализ аналогов 10](#_Toc168866965)

[1.4.1 Понятия цифровая платформа и веб-сервис 10](#_Toc168866966)

[1.4.2 Сервис “Авито” 13](#_Toc168866967)

[1.4.3 Сервис “Озон” 15](#_Toc168866968)

[1.4.4 Сравнение разрабатываемой системы с аналогами 16](#_Toc168866969)

[1.5 Требования к разрабатываемому программному средству 20](#_Toc168866970)

[2 КОНСТРУКТОРСКО-ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ 22](#_Toc168866971)

[2.1 Описание технологий разработки программного средства 22](#_Toc168866972)

[2.1.1 Angular 22](#_Toc168866973)

[2.1.2 React 24](#_Toc168866974)

[2.1.3 Vue 27](#_Toc168866975)

[2.2 Выбор технологии разработки 28](#_Toc168866976)

[2.3 Архитектура системы 35](#_Toc168866977)

[2.4 Описание функционала интерфейса системы 39](#_Toc168866981)

[2.4.1 Функции гостя 39](#_Toc168866982)

[2.4.2 Функции авторизованного пользователя 43](#_Toc168866983)

[ЗАКЛЮЧЕНИЕ 53](#_Toc168866984)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 54](#_Toc168866985)

[ПРИЛОЖЕНИЕ А ГРАФИЧЕСКАЯ ЧАСТЬ 56](#_Toc168866986)

[ПРИЛОЖЕНИЕ Б ТЕХНИЧЕСКОЕ ЗАДАНИЕ 60](#_Toc168866987)

[ПРИЛОЖЕНИЕ В ПРОГРАММА И МЕТОДИКА ИСПЫТАНИЙ 66](#_Toc168866988)

СПИСОК ОБОЗНАЧЕНИЙ И СОКРАЩЕНИЙ

1. **Авторизация** – предоставление пользователю прав на выполнение определенных действий в системе.
2. **ИС –** информационная система.
3. **Интерфейс –** программа-посредник при передаче данных между пользователем и системой.
4. **Фронтенд –** пользовательский интерфейс системы.
5. **Веб-сервис –** программное средство в сети интернет, отвечающее на определенные требования пользователей.
6. **Фреймворк -** программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта.
7. **ИТ** –информационные технологии.

ВВЕДЕНИЕ

Многие задачи по решению бытовых вопросов и бизнес задач можно решить с помощью интернета: доставка еды, заказ модных вещей на дом, поиск работы, досуг можно сделать, обратившись к интернет сервисам. Такие привычные всем вещи, как магазин, рынок или доска объявлений, существуют как бы в двух мирах одновременно: в реальном и виртуальном.

Для того чтобы облегчить покупку или продажу различных вещей созданы специальные сервисы по подбору объявлений, которые работают по принципу «барахолки». Люди, желающие приобрести какой-либо товар, заходят на такой сайт, ищут интересующую их вещь, связываются с продавцом и договариваются о сделке. Благодаря таким сервисам человек экономит свое время.

В квалификационной работе рассматривается информационная система подачи объявлений в области продажи материалов для строительства. В отличие от уже созданных масштабных сервисов с объявлениями данная система предназначена под товары ограниченного типа. Благодаря этому система не будет перегружена лишними объявлениями о продаже товаров, не касающихся темы строительства. При отсутствии лишней информации пользователи смогут очень быстро найти и приобрести нужные товары, и при желании разместить свое объявление на продажу какого-либо строительного материала.

1 ПОСТАНОВКА ЗАДАЧ РАЗРАБОТКИ

1.1 Цели и задачи проекта

Целью работы является разработка фронтенд модуля информационной системы подачи объявлений для мелкооптовой торговли строительными материалами. Данный модуль является интерфейсом информационной системы и предназначен для пользователей, которые будут ее использовать.

Задачами данной работы являются:

* Анализ предметной области;
* Разработка требований к системе;
* Анализ критериев качества системы;
* Рассмотрение существующих аналогов;
* Анализ современных технологий разработки;
* Выбор оптимальной технологии для разработки;
* Разработка и описание архитектуры программного средства;
* Разработка программного кода;
* Тестирование и отладка разработанного программного средства.

1.2 Описание предметной области

В данной работе разрабатывается фронтенд-модуль информационной системы подачи объявлений для мелкооптовой торговли строительными материалами. Чтобы система была полезна людям, необходимо разобраться в том, какая целевая аудитория будет пользоваться разрабатываемой системой. Нужно подробно описать предметную область, под которую данная система разрабатывается.

Разрабатываемая система предназначена для индивидуальных предпринимателей и небольших магазинов, занимающихся мелкооптовой реализацией строительных материалов, а также их покупкой. Данная система является уникальной, поскольку ближайшие аналоги по типу сервисов Авито и Озон не заточены под определенные виды товаров, и при поиске нужного объявления клиенту сервиса приходится очень долго искать нужную категорию товара.

Распишем основные характеристики строительных материалов. Всего в рамках работы было выделено 8 значимых характеристик материалов, представленных в таблице 1.

Таблица 1 – Характеристики материалов

|  |
| --- |
| **Параметр объекта** |
| Стоимость |
| Износостойкость |
| Назначение материала |
| Легкость обработки |
| Прочность |
| Водостойкость |
| Устойчивость к огню |
| Вид материала |

Стоимость материала является важной характеристикой строительных материалов, так как данная характеристика помогает определить бюджет строительства, выбрать наиболее выгодный вариант, сравнить материалы по цене.

Износостойкость определяет то, насколько материал сопротивляется деформации и другим механическим воздействиям. Данный параметр определяет срок службы материала, его практичность в различных условиях эксплуатации.

Назначение материала необходимо для определения того, в какой сфере используется тот или иной материал. Данная характеристика позволяет подобрать подходящий вариант для конкретного проекта.

Характеристика «легкость обработки» определяет степень сложности обработки материала с помощью различных инструментов. Данная характеристика влияет на затраты при работе с материалом, а также на трудоемкость работы с ним.

Прочность - способность материала сопротивляться разрушению под действием нагрузки, т.е. определяет допустимые нагрузки на материал, его надежность. Данная характеристика необходима для обеспечения безопасности и долговечности конструкции.

Такие характеристики как, водостойкость и устойчивость к огню необходимы для того, чтобы понимать, насколько хорошо материал сопротивляется воздействию воды и огня. Обычно такие материалы используются в местах, с повышенными требованиями к пожарной безопасности, или местах, где материал контактирует с водой.

Вид материала – категория, к которой относится материал и данный критерий помогает ориентироваться в многообразии материалов, сужает круг поиска.[4]

1.3 Выбор критериев качества системы

Для разрабатываемой системы основными критериями качества являются:

* Простота и удобство использования интерфейса;
* Дешевизна разработки;
* Соответствие функционала рамкам предметной области;
* Быстрое нахождение нужной информации.

**Простота и удобство использования интерфейса.** Пользователю должен быть понятен интерфейс системы, для его освоения нужно минимальное количество времени.

**Дешевизна разработки.** Система разрабатывается минимальным количеством людей в сжатые сроки, стоимость разработки в идеальном случае бесплатна.

**Соответствие функционала рамкам предметной области.** Система должна удовлетворять требованиям заявленной предметной области.

**Быстрое нахождение нужной информации.** Пользователь должен иметь возможность за максимально короткое время найти необходимую для себя информацию, не отвлекаясь на посторонние вещи и не натыкаясь на лишний функционал.[2][5]

1.4 Анализ аналогов

1.4.1 Понятия цифровая платформа и веб-сервис

В данный момент реализовано огромное количество разнообразных сервисов по продаже товаров и услуг. Первая компьютерная электронная доска объявлений была создана [111].

Цифровые платформы — это сложные информационные системы, обеспечивающие функции взаимосвязи между участниками рынка, открытые для использования клиентами и партнерами, разработчиками приложений, поставщиками услуг и агентами. Используя пакеты цифровых технологий обработки данных и упрощая схемы разделения труда, цифровые платформы позволяют снижать транзакционные издержки и выстраивать взаимовыгодные отношения между хозяйствующими субъектами. [14]

Существует три основных вида цифровых платформ, приведем их общее описание.

* Инструментальные цифровые платформы, в основе которых находятся программные или программно-аппаратные комплексы (продукты), предназначенные для создания прикладных решений с использованием типовых функций и интерфейсов для обработки информации и с применением сквозных технологий работы с данными. Также цифровые платформы этого типа включают функционал для отладки прикладных программных или программно-аппаратных инструментов. Примеры таких платформ - Java, Android OS, Intel x86, Bitrix, Microsoft Azure и другие;
* Инфраструктурные цифровые платформы, связанные с экосистемами участников рынков информатизации, на которые выводятся ИТ-сервисы, использующие сквозные цифровые технологии работы с данными и доступ к источникам информации и применяемые в пределах экосистем. К платформам такого типа относятся General Electric Predix, ESRI ArcGIS, ЕСИА;
* Прикладные цифровые платформы - бизнес-модели, предоставляющие возможности алгоритмизированного обмена ценностями между значительным числом независимых участников рынка. Транзакции при этом проводятся в единой информационной среде. В данную группу входят, например, Uber, AirBnB, Aliexpress, Booking.com, Avito, Yandex Taxi, Ozon.

Также цифровые платформы различаются по группам участников и уровням обработки информации:

* инструментальные цифровые платформы объединяют разработчиков платформ и решений и проводят технологические операции по обработке информации;
* инфраструктурные цифровые платформы объединяют поставщиков информации, разработчиков и операторов платформ, разработчиков и потребителей ИТ-сервисов и применяются для выработки информации для принятия решений на уровне хозяйствующих субъектов;
* прикладные цифровые платформы используются для обработки информации о заключении и выполнении сделок между несколькими экономическими субъектами - поставщиками товаров, услуг, производственных ресурсов, их потребителями, а также операторами платформ и регуляторами.[15]

На рисунке 1 проиллюстрирована связь всех трех типов цифровых платформ между собой.

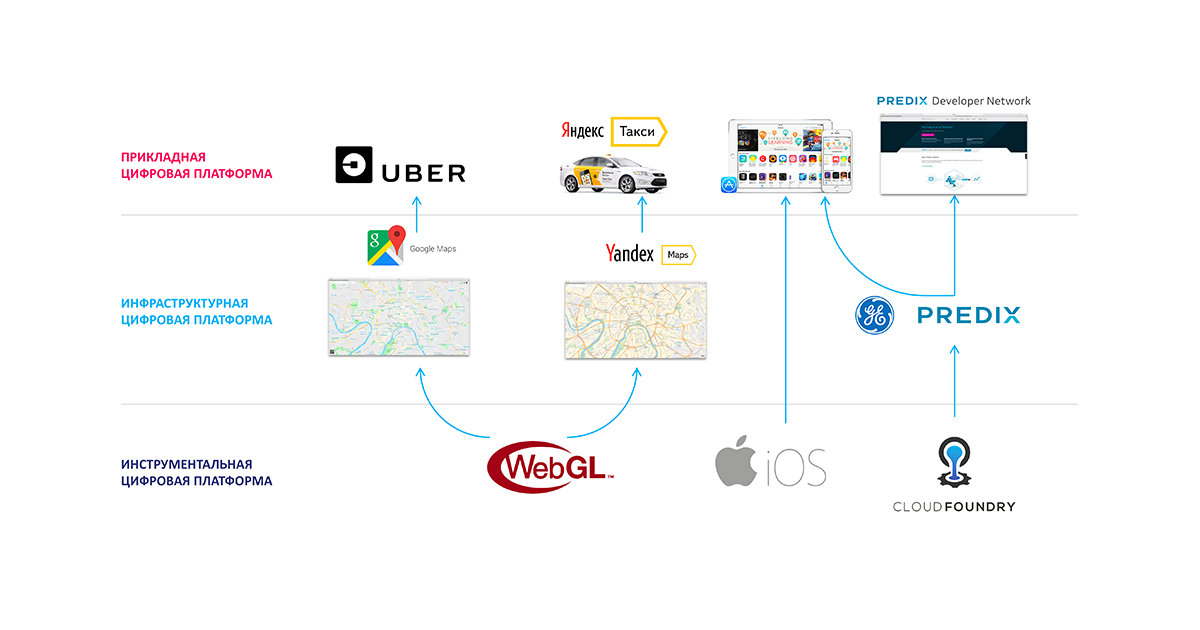


Рисунок 1 – Взаимодействие цифровых платформ

В рамках данной работы мы рассматриваем непосредственно прикладные цифровые платформы.

Веб-сервис – обладающая уникальным веб-адресом программная система, построенная на базе открытых стандартов и использующаяся для обмена данными между несколькими системами. Программы, которые на работают на разных платформах, могут использовать веб-сервисы для обмена данными по компьютерным сетям различного размера.

Веб-сервис обеспечивает связь между различными приложениями с использованием открытых стандартов, таких как:

* HTML — для обмена Web-страницами;
* XML — для описания данных;
* SOAP — для обмена структурированными сообщениями;
* JSON — для обмена текстовыми данными.

Основное преимущество веб-сервисов — универсальность. Они работают на базе стандартных технологий, не зависящих от поставщиков приложений, могут использоваться в любых операционных системах, серверах приложений, языках программирования и т.д.

Современные веб-сервисы получили широкое распространение. Компании используют их для интеграции информационных систем и доступа клиентов или партнеров к своим ресурсам. Например, банк разрабатывает веб-сервис для предоставления данных по курсам акций и валютам, условиям по вкладам и кредитам.[3]

Современные сервисы объявлений выдерживают огромный наплыв пользователей с десятками миллионов обращений в день. Важно поддерживать достаточно высокую скорость отклика. Рассмотрим две самые популярные цифровые платформы на сегодняшний день в России: Авито и Озон. Проанализируем их по нескольким параметрам: охват аудитории, средний возраст пользователя, количество посещений и так далее.

1.4.2 Сервис «Авито»

Авито – российский веб-сайт с [объявлениями](https://translated.turbopages.org/proxy_u/en-ru.ru.17f2aeb0-6587384e-c3f51825-74722d776562/https/en.wikipedia.org/wiki/Classified_advertising) с разделами, посвященными продаже товаров общего назначения, работе, [недвижимости](https://translated.turbopages.org/proxy_u/en-ru.ru.17f2aeb0-6587384e-c3f51825-74722d776562/https/en.wikipedia.org/wiki/Real_estate), [персоналиям](https://translated.turbopages.org/proxy_u/en-ru.ru.17f2aeb0-6587384e-c3f51825-74722d776562/https/en.wikipedia.org/wiki/Personal_advertisement), автомобилям на продажу и услугам. Avito - самый популярный сайт объявлений в России и крупнейший сайт объявлений в мире.

По количеству посещений сервис очень популярен – порядка 60 млн уникальных пользователей за месяц. Большинство пользователей – мужчины с долей 56.8%, доля женщин 43.2%. Самая многочисленная возрастная группа – люди от 25 до 34 лет. Среднее время на сайте – 11 минут. Самыми популярными товарами являются гаджеты (20%) и товары для дома (14%). С компьютеров сайт посещают 48.6% аудитории, с мобильных гаджетов – 51.4%. Использование сервиса людьми разного возраста показано на рисунке 2:

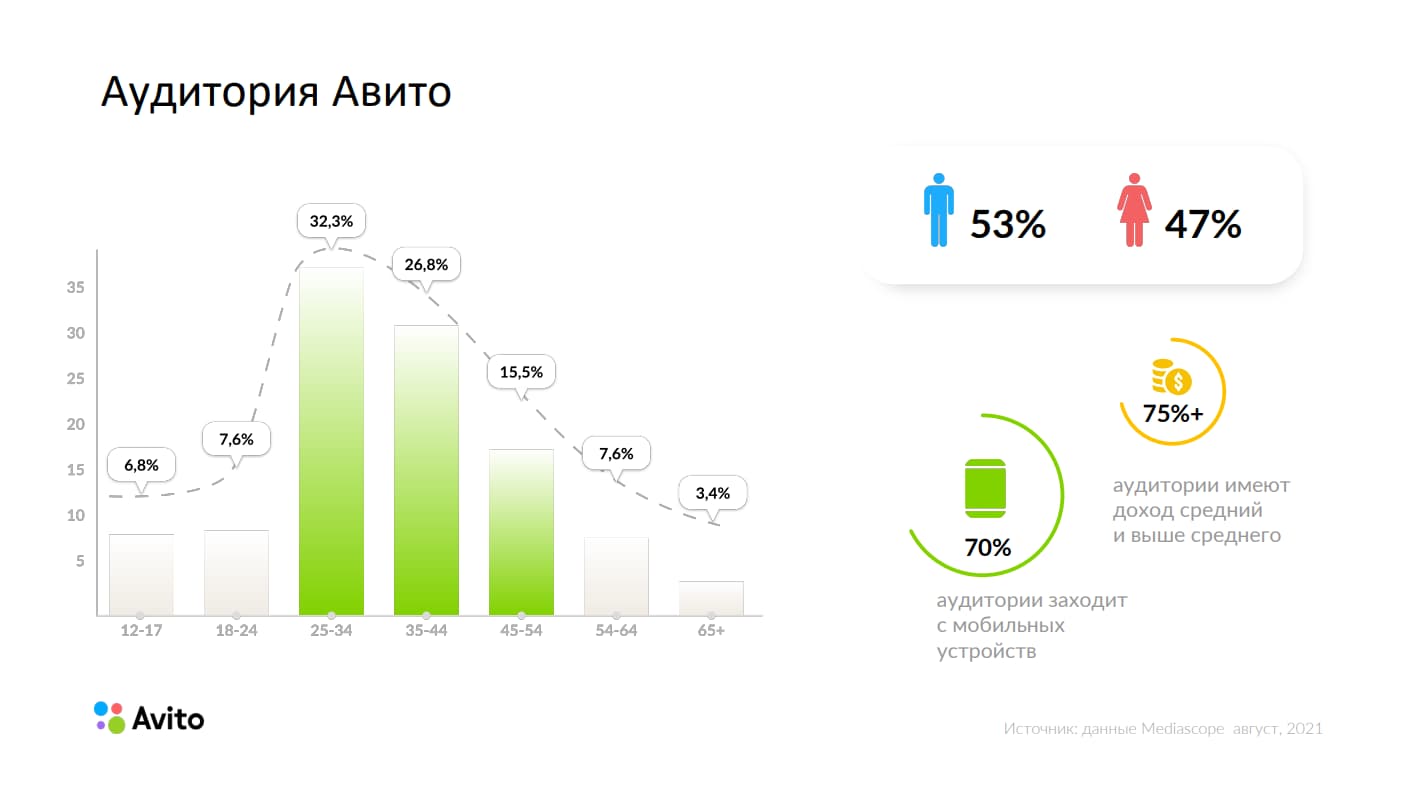


Рисунок 2 – Возрастные группы в “Авито”

При исследовании статистики сервиса Авито выявлена интересная закономерность. Пользователи, заходящие с десктопных устройств склонны больше времени проводить на сайте, тем самым просматривают больше страниц и более внимательно изучают информацию. Также замечен меньший процент отказов при оформлении заказов. Пользователи мобильных же устройств наоборот проводят меньше времени на сайте, делая упор на поверхностный просмотр. Данный аспект стоит учитывать при проектировании собственного сервиса (таблица 2).

Таблица 2 - Статистика мобильного трафика Авито

| **Метрика** | **Тип интерфейса** | |
| --- | --- | --- |
| Десктоп | Мобильная версия |
| Визиты в день, среднее | 6.15 млн | 6.5 млн |
| Уникальные визиты в день | 3.14 млн | 3.7 млн |
| Проведенное время | 12 мин 51 с | 9 мин 45 с |
| Страницы за визит | 13.7 | 8.3 |
| Просмотры страниц | 84.3 млн | 53.9 млн |

Интерфейс сайта очень информативен, на первой странице есть множество иконок и кнопок, интуитивно понятных пользователю. Рассмотрим на рисунке 3 цветовую гамму сайта – все функциональные элементы выполнены в спокойных тонах, присутствуют только три основных цвета. [11]

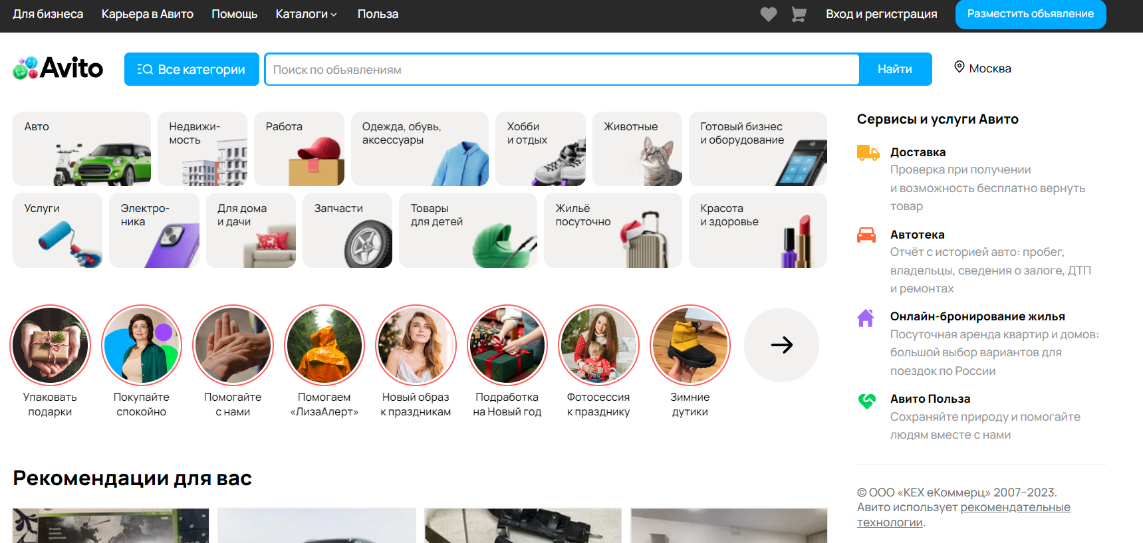


Рисунок 3 – Интерфейс сервиса “Авито”

1.4.3 Сервис «Озон»

Ozon — одна из лидирующих платформ электронной коммерции в России [12].

По количеству посещений сервис занимает лидирующие позиции в своей категории – 440 млн посещений за последний месяц. Большинство пользователей – мужчины с долей 62%, доля женщин 38%. Самая многочисленная возрастная группа – люди от 25 до 34 лет, что составляет четверть всех посетителей.

Среднее время на сайте – 8 минут 40 секунд. Самыми популярными категориями товаров являются: консоли и аксессуары для видеоигр. За один визит пользователь в среднем просматривает 12 страницы. Вышеизложенная статистика приведена для настольной (десктоп) версии сервиса.

Можно сказать, что интерфейс сайта использует более яркую палитру цветов, нежели Авито. Это объясняется тем, что Авито позиционируется как площадка подачи объявлений частными пользователями, а Озон – витрина товаров различных магазинов.[12]

На рисунке 4 можно увидеть интерфейс сервиса Озон.

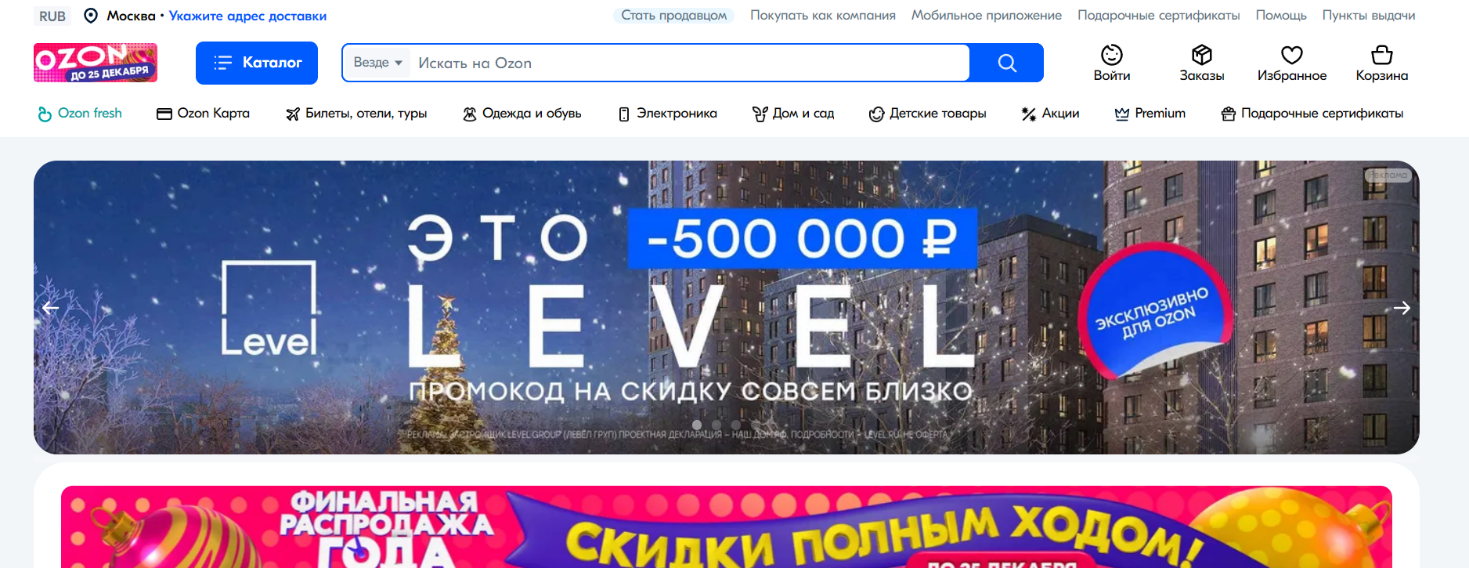


Рисунок 4 – Интерфейс сервиса “Озон”

1.4.4 Сравнение разрабатываемой системы с аналогами

Сервис подачи объявлений по мелкооптовой торговле строительными материалами в рамках своей предметной области не имеет прямых аналогов. Однако можно сказать, что ближайшими аналогами нашей системы являются сервисы Озон и Авито, описание которых приведено выше.

Для каждого критерия качества назначим весовые коэффициенты. Приведем их описание в таблице 3.

Каждому из критериев будет присуждаться определенная оценка в размере от 0 до 1. Введем вербально-числовую шкалу для проставления оценки по каждому из критериев качества.

Для критерия К1 простота и удобство использования интерфейса (таблица 4).

Таблица 3 – Весовые коэффициенты критериев качества

| **№** | **Критерий качества** | **Весовой коэффициент** |
| --- | --- | --- |
| К1 | Простота и удобство использования интерфейса | 0.3 |
| К2 | Дешевизна разработки | 0.1 |
| К3 | Соответствие функционала рамкам предметной области | 0.4 |
| К4 | Быстрое нахождение нужной информации | 0.2 |

Таблица 4 – Вербально-числовая шкала по критерию «простота и удобство использования интерфейса»

| **Качественная оценка** | **Численная оценка** |
| --- | --- |
| Интуитивно понятен | 1 |
| Вполне понятен | 0.8 |
| Понятен с небольшими затруднениями | 0.6 |
| Достаточно сложен | 0.4 |
| Очень сложен | 0.2 |

Для критерия К2 «дешевизна разработки» используется вербально-числовая шкала (таблица 5).

Таблица 5 – Вербально-числовая шкала по критерию «дешевизна разработки»

| **Качественная оценка** | **Численная оценка** |
| --- | --- |
| Бесплатно | 1 |
| Малый бюджет | 0.9 |
| Средний бюджет | 0.8 |
| Большой бюджет | 0.6 |

Для критерия К3 «соответствие функционала рамкам предметной области» (таблица 6):

Таблица 6 – Вербально-числовая шкала по критерию «соответствие функционала рамкам предметной области»

| **Качественная оценка** | **Численная оценка** |
| --- | --- |
| Полное соответствие | 1 |
| Соответствие с небольшими изъянами | 0.7 |
| Мало соответствует | 0.3 |

Таблица 7 – Вербально-числовая шкала по критерию «быстрое нахождение нужной информации»

| **Качественная оценка** | **Численная оценка** |
| --- | --- |
| Нет лишней информации | 1 |
| Есть незначительное количество ненужной информации | 0.7 |
| Достаточно много ненужной информации | 0.4 |
| Нужная информация отсутствует | 0 |

Оценим каждую систему по каждому из вышеуказанных критериев. Все результаты запишем в таблицу 8.

Таблица 8 – Качественная оценка систем по критериям качества

| **Критерий** | **Весовой коэффициент** | **Сравниваемая система** | | |
| --- | --- | --- | --- | --- |
| Авито | Озон | Разрабатываемая система |
| Простота и удобство использования интерфейса | 0.3 | Вполне понятен | Интуитивно понятен | Вполне понятен |
| Дешевизна разработки | 0.1 | Большой бюджет | Большой бюджет | Бесплатно |
| Соответствие функционала рамкам предметной области | 0.4 | Полное соответствие | Мало соответствует | Полное соответствие |
| Быстрое нахождение нужной информации | 0.2 | Достаточно много ненужной информации | Достаточно много ненужной информации | Нет лишней информации |

Воспользуемся вербально-числовыми шкалами, описанными выше, и расставим количественные оценки для каждой системы по всем критериям качества в таблицу 9.

Таблица 9 – Количественная оценка систем по критериям качества

| **Критерий** | **Весовой коэффициент** | **Сравниваемая система** | | |
| --- | --- | --- | --- | --- |
| Авито | Озон | Разрабатываемая система |
| Простота и удобство использования интерфейса | 0.3 | 0.8 | 1 | 0.8 |
| Дешевизна разработки | 0.1 | 0.6 | 0.6 | 1 |
| Соответствие функционала рамкам предметной области | 0.4 | 1 | 0.3 | 1 |
| Быстрое нахождение нужной информации | 0.2 | 0.4 | 0.4 | 1 |

Проведем анализ этих систем, используя метод близости сравниваемых вариантов к идеальному. Лучшим будет считаться вариант, набравший наименьшее значение[9] Для этого используем формулу 1:

(1)

где kij – оценка варианта в определенном критерии,

αi – вес критерия.

Вычислим Y для каждой системы по формуле 1:

Для Авито: Y1 =

Аналогично вычислим значения для остальных систем.

Для Озон: Y2 = 0,66

Для Сравнивамой системы: Y3 = 0,24

Запишем измерения в таблицу 10.

Таблица 10 – Метод близости сравниваемых вариантов к идеальному

| **Формула** | **Сравниваемая система** | | |
| --- | --- | --- | --- |
| Авито | Озон | Разрабатываемая система |
|  | 0.47 | 0.66 | 0.24 |

Как видно по результатам вычислений, наименьшее значение набрал вариант с разрабатываемой системой. Таким образом, по результату сравнения разрабатываемой системы с ближайшими аналогами по заданным критериям качества наша система обходит свои аналоги.

1.5 Требования к разрабатываемому программному средству

Разрабатываемая система должна удовлетворять нескольким требованиям:

* Понятный и удобный пользовательский интерфейс;
* Минимальные задержки при использовании функционала системы;
* Отсутствие ошибок и конфликтов при работе системы.

Разрабатываемая система должна исполнять следующие функции:

* По запросу пользователя показывать интересующие объявления;
* Поиск нужных объявлений по названию;
* Фильтрация объявлений по нескольким критериям;
* Регистрация новых пользователей;
* Вход пользователей по имеющимся в системе данным;
* Разграничение доступа к функционалу системы на авторизованного и неавторизованного пользователя;
* Добавление пользователем объявлений;
* Редактирование пользователем объявлений;
* Добавление пользователем интересующих объявлений в избранное;
* Создание заказов на интересующие объявления.

2 КОНСТРУКТОРСКО-ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ

2.1 Описание технологий разработки программного средства

Веб-интерфейс – это программное обеспечение, с помощью которого пользователь может взаимодействовать с веб-сервисами через браузер. Практически все современные веб-интерфейсы разрабатываются с использованием кода HTML, CSS и JavaScript. HTML используется для разметки гипертекста. CSS задаёт стили для текста, таблиц, блоков, заданных при помощи HTML. При помощи JavaScript создаётся красочная анимация, а также всевозможные взаимодействия с сервером.

В настоящее время очень популярен метод создания веб-интерфейсов под названием SPA (Single Page Application). SPA загружает одну HTML страницу и обновляет ее содержимое в случае, если оно изменилось, без полной перезагрузки страницы. В данном случае применяется технология асинхронных запросов к серверу (AJAX).

В 2024 году существует большое количество различных фреймворков для разработки интерфейса программных приложений. Для сравнения были выбраны три самых популярных и растиражированных: Angular, React, Vue. Данные фреймворки используют скриптовый язык программирования для создания веб-интерфейсов Java-Script. Также с этими технологиями можно использовать строго типизированную версию языка – Type-Script.

Под фреймворком будем понимать программное обеспечение, определяющее структуру программной системы. Оно облегчает разработку и объединение нескольких модулей одного большого программного проекта.

Фреймворк отличается от понятия [библиотеки](https://ru.wikipedia.org/wiki/%D0%91%D0%B8%D0%B1%D0%BB%D0%B8%D0%BE%D1%82%D0%B5%D0%BA%D0%B0_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) тем, что последняя может быть использована в программном продукте просто как набор подпрограмм похожей функциональности, не влияя на архитектуру программного продукта и не накладывая на неё никаких ограничений. В то время как фреймворк диктует правила построения архитектуры приложения, задавая на начальном этапе разработки поведение по умолчанию - «каркас», который нужно будет расширять и изменять согласно указанным требованиям.[10]

2.1.1 Angular

Angular представляет фреймворк от компании Google для создания клиентских приложений. Прежде всего он предназначен для разработки одностраничных приложений. В этом плане Angular является наследником другого фреймворка AngularJS. В то же время Angular это не новая версия AngularJS, а принципиально новый фреймворк.

Angular предоставляет такую функциональность, как двустороннее связывание, позволяющее динамически изменять данные в одном месте интерфейса при изменении данных модели в другом, шаблоны, маршрутизация и так далее.

Одной из ключевых особенностей Angular является то, что он использует в качестве языка программирования TypeScript. При желании можно писать приложения на Angular с помощью таких языков как JavaScript. Но TypeScript тем не менее является основным языком для Angular.[13]

Фреймворк ценен не сам по себе, а за счет своих составляющих, упрощающих разработку. Что предлагает Angular:

* Компоненты. Структурные блоки, из которых и состоит приложение, написанное на Angular. Если вам нужно написать сайт – минимальными компонентами будут шапка, основной контент и футер. Физически Angular повторяет структуру Java – все структурные блоки хранятся в отдельных файлах, что упрощает разработку и поддержку сложного приложения. Важная особенность: компоненты полностью автономны, изначально зависимостей между компонентами нет. Это позволяет вам беспроблемно добавлять или удалять их;
* Модули. Если просто, то модули – это скрипты компонента. У каждого компонента есть как минимум один центральный модуль, описывающий основную линию поведения. Обычно к основному модулю присоединяют ряд дополнительных, расширяющих функционал оригинального – это позволяет построить систему с высокой инкапсуляцией (изолированностью частей друг от друга);
* Формы. Набор готовых форм, через которые пользователь отправляет данные на сервер. Форм – много, на любой вкус и для любой ситуации;
* Сервисы. Если модули – это скрипты, то сервисы – это классы. Они могут как хранить информацию, так и производить действия. Сервисами удобно соединять между собой разные части приложений, потому что для сервиса нет проблемы в том, чтобы обратиться к какому-либо компоненту, модулю или форме. Сервисы могут иметь (и чаще всего имеют) свою собственную архитектуру, которая строится по принципу «Один сервис = одно действие»;
* Директивы. Глобальные модули с поведением сервисов. Если с помощью сервисов управляется одна страница, то с помощью директив можно управлять всеми страницами сразу. Правильно составленные директивы во много раз упрощают разработку нового функционала уже для готового приложения.

Кратко: приложение состоит из компонентов и форм, которыми управляют модули. Если вам нужно прописать взаимодействие между компонентами – используете сервисы, если вам нужно настроить глобальное взаимодействие всех частей приложения – используете директивы.[6] Описание взаимодействия проиллюстрировано на рисунке 5:

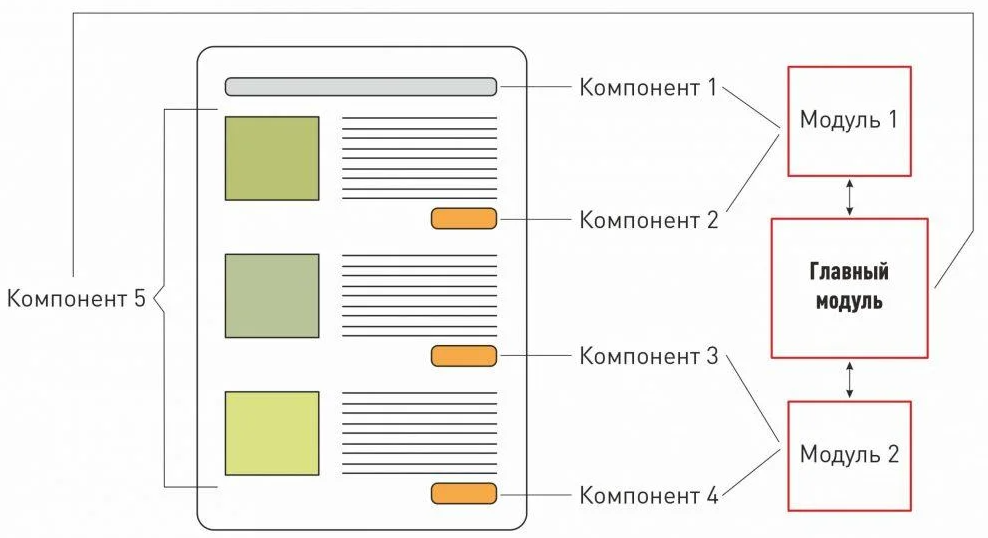


Рисунок 5 – Компоненты фреймворка “Angular”

2.1.2 React

React – JavaScript библиотека с открытым исходным кодом для работы с пользовательскими интерфейсами, разработанная компанией Facebook. Библиотека была выпущена в открытый доступ в 2013 году.

С помощью React разработчики создают веб-приложения, которые изменяют отображение без перезагрузки страницы. Благодаря этому сайты быстро реагируют на действия пользователя, например, заполнение форм, применение фильтров, добавление товаров в корзину и так далее.

React применяют для отрисовки компонентов пользовательского интерфейса. Также библиотека может полностью управлять фронтендом. В этом случае React используют с библиотеками для роутинга и управления состоянием, например, Redux и React Router.[8]

React использует декларативный подход в программировании. Если в императивном программировании необходимо задать каждое действие, то в декларативном достаточно описания конечного результата и реакции на действия.

Библиотека позволяет оптимизировать процесс разработки за счет того, что программисту не нужно описывать детали. React будет самостоятельно обновлять элементы. Задача программиста — грамотно описать эти самые элементы. Использование такого подхода позволяет сокращать время на создание приложений.[1]

Главными объектами в React являются компоненты. Компонент – это элемент, который можно использовать неограниченное количество раз. Каждый компонент инкапсулирован (ограничен от других элементов) и содержит все методы и данные, необходимые для работы с ним. Для того чтобы воспользоваться компонентом, его нужно вызвать в коде и далее использовать по своему усмотрению.

У библиотеки React есть специальный механизм для управления рендерингом всех компонентов на странице. То есть создания визуального представления элементов. У каждой страницы в интернете есть своя структура. Это так называемое DOM-дерево или объектная модель документа. Он состоит из тэгов HTML — языка разметки и каскадных таблиц стилей — CSS, а также подключенного к нему коду на JavaScript. Для этого React сохраняет в кэше копию DOM-дерева. Она изменяется быстрее оригинала, что позволяет приложению так же быстро обновлять страницу.

Например, если после какого-либо действия страница нуждается в перезагрузке, React будет работать не с реальной структурой, а с копией. Что позволит пользователю быстрее увидеть результат. За счет этого в приложениях, написанных на React, изменения вносятся практически мгновенно — ждать несколько секунд не нужно. В этом помогают так называемые “состояния”, хранящиеся в локальном хранилище React приложения.

Изначально в React использовали классы — тип компонентов, включающих в себя методы управления состоянием. Сейчас вместо этого используют хуки. Хуки - это функции, которые можно прикрепить к определенному состоянию или элементу. При изменении состояния либо элемента хук отслеживает это действие, и затем отрабатывает свой функционал.

В React реализовали собственное расширение для JavaScript. С его помощью можно описывать элементы HTML-кодом на этом языке программирования. В итоге разработчики получают возможность менять структуру DOM, используя привычный JavaScript.[7]

В React используется роутинг для управления навигацией между разными частями приложения. React отслеживает действия пользователя и в зависимости от того, какое действие пользователь произведет, подгружает необходимую информацию.

2.1.3 Vue

Vue.js — это современный JavaScript-фреймворк, предназначенный для разработки пользовательских интерфейсов и одностраничных веб-приложений. Он решает задачи уровня представления и облегчает взаимодействие с различными библиотеками. Одной из отличительных особенностей Vue.js является возможность постепенного внедрения, что выделяет его среди других фреймворков. Построенный на основе стандартов HTML, CSS и JavaScript, он предлагает декларативную и компонентную модель программирования, которая способствует эффективной разработке интерфейсов любой сложности.

Vue.js обычно используется для разработки:

* Быстрых веб-сайтов и приложений, блогов небольшого размера, лендингов;
* Одностраничных приложений;
* Адаптивных интерфейсов;
* Разделов личных кабинетов, пользовательских страниц;
* Интерфейсов авторизации, онлайн-чатов, форм заявки и так далее.

Основные концепции фреймворка:

* Конструктор;
* Компоненты;
* Директивы;
* Переходы.

Конструктор – функция, которая используется для создания экземпляра. При создании экземпляра, конструктор инициализирует свойства и настройки экземпляра и связывает их с определённым DOM-элементом. Таким образом, происходит управление компонентами и реактивностью в приложении.

Компоненты – элементы, помогающие расширить основные html-элементы и вписать код, который можно использовать повторно в разных частях приложения. На этапе проектирования приложение разбивается на независимые между собой части и получается древовидная структура компонентов.

Директивы — специальные атрибуты для добавления элементам html дополнительной функциональности. Несколько примеров таких директив:

* V-bind — динамически связывается с одним или несколькими атрибутами;
* v-if — условие для рендера элемента;
* V-else — обозначает “else блок” для v-if;
* V-for — циклично проходит массив объектов.

Все Vue-директивы имеют префикс “v-”. В директиву передается какое-то значение состояния, а в качестве аргументов могут быть атрибуты html или события.

Vue предоставляет различные способы применения анимационных эффектов, когда элементы отрисованы, обновлены или удалены из DOM. Они включают в себя инструменты для:

* Автоматического применения классов для CSS-переходов и анимаций;
* Интеграции сторонних библиотек для CSS-анимаций, таких как Animate.css;
* Использования JavaScript для манипуляции DOM-ом;
* Интеграции сторонних JavaScript библиотек для анимаций, таких как Velocity.js.[16]

2.2 Выбор технологии разработки

Выбраны 3 популярных технологии разработки веб-интерфейса: Angular, React и Vue. Зададим каждой технологии вариант с порядковым номером:

* В1 – Angular
* В2 – React
* В3 – Vue

Выполнять сравнение данных фреймворков будем по следующим критериям:

* К1 - Трудовременные затраты при разработке;
* К2 – Востребованность (среднее количество вакансий на досках объявлений по данному фреймворку);
* К3 - Поддержка (частота обновлений от разработчиков фреймворка);
* К4 - Сфера применения.

Для каждого критерия назначим весовые коэффициенты бальным анализом. Баллы выставляются по шкале от 1 до 5, где максимальный балл получит критерий, являющийся самым значимым, а меньшие баллы получат критерии, которые менее важны. В данном случае экспертом выступает разработчик проекта.

Определим весовой коэффициент критерия по следующей формуле 2:

, (2)

где: – весовой коэффицент критерия,

bi – балл критерия,

b – сумма всех баллов.

Критерию трудовременные затраты при разработке (К1) поставлено 5 баллов, т.к. в данной работе важна быстрота разработки программного средства из-за ограниченности времени.

Критерию сфера применения (К4) поставлено 4 балла, т.к. в нашей предметной области нужно разработать программное средство определенного масштаба (среднего), и технологии, используемые для разработки высоконагруженных приложений нам в целом необязательны.

Критерию востребованность (К2) поставлено 3 балла. Этот критерий уступает по значимости К1 и К4, однако в дальнейшем навык изучения выбранной технологии разработки поможет в построении будущей карьеры, по этому желательно выбрать ту технологию, которая в будущем будет актуальна на рынке труда.

Критерию поддержка (К3) поставлено 2 балла, т.к. разрабатываемое программное средство в дальнейшем не планируется масштабировать и обновлять, по этому данный критерий имеет наименьшую важность в рамках данной работы.

В итоге сумма баллов вышла равной 14. Рассчитаем по формуле 1 весовые коэффициенты каждого критерия и запишем получившиеся значения в таблицу 11.

Таблица 11 – Расчёт весовых коэффициентов критериев

| **№** | **Критерий** | **Балл критерия** | **Весовой коэффициент** |
| --- | --- | --- | --- |
| К1 | Трудовременные затраты при разработке | 5 | 0.36 |
| К2 | Востребованность | 3 | 0.21 |
| К3 | Поддержка | 2 | 0.14 |
| К4 | Сфера применения | 4 | 0.29 |

Для каждого рассматриваемого варианта по критериям введем бальную характеристику по шкале от 0 до 1, где 0 – очень плохо, 1 – очень хорошо. Так как выбранные критерии трудно измерить числовой характеристикой без предварительных объяснений, также введем вербально-числовые шкалы для перевода качественных параметров в количественные.

Введём вербально-числовую шкалу для критерия трудовременные затраты при разработке (таблица 12).

Таблица 12 – Вербально-числовая шкала для критерия К1

| **Качественная оценка** | **Количественная оценка** |
| --- | --- |
| Быстро | 1 |
| Относительно быстро | 0.8 |
| Нормально | 0.6 |
| Довольно долго | 0.4 |
| Очень долго | 0.2 |

Введем вербально-числовую шкалу для критерия востребованность (таблица 13):

Таблица 13 – Вербально-числовая шкала для критерия К2

| **Качественная оценка** | **Количественная оценка** |
| --- | --- |
| Очень популярен | 1 |
| Достаточно популярен | 0.7 |
| Теряет актуальность | 0.4 |
| Практически неактуален | 0.1 |

Введем вербально-числовую шкалу для критерия поддержка (таблица 14):

Таблица 14 – Вербально-числовая шкала для критерия К3

| **Качественная оценка** | **Количественная оценка** |
| --- | --- |
| Очень часто | 1 |
| Часто | 0.9 |
| С незначительными перерывами | 0.8 |
| С задержками | 0.6 |
| Редко | 0.4 |
| Не поддерживается | 0.1 |

Оценки критерия сфера применения (К4) ранжированы непосредственно для нашей предметной области. Так как разрабатываемая система имеет относительно широкий функционал, но без уклона в промышленную разработку, то самым лучшим фреймворков будет тот, который поддерживает разработку проектов средней величины (таблица 15).

Таблица 15 – Вербально-числовая шкала для критерия К4

| **Качественная оценка** | **Количественная оценка** |
| --- | --- |
| Проекты средней величины | 1 |
| Большие проекты | 0.9 |
| Маленькие проекты | 0.8 |

Расставим баллы каждой технологии по критериям (таблица 16):

Таблица 16 – Сравнение фреймворков по заданным критериям:

| **Критерий** | **Весовой коэффициент** | **Количественная оценка** | | |
| --- | --- | --- | --- | --- |
| В1 | В2 | В3 |
| К1 | 0.36 | 0.4 | 0.8 | 1 |
| К2 | 0.21 | 0.7 | 1 | 0.7 |
| К3 | 0.14 | 0.8 | 0.9 | 0.9 |
| К4 | 0.29 | 0.9 | 1 | 0.8 |

Произведем ранжирование вариантов технологий по набору показателей сравнения по методу Борда используя формулу (3):

, (3)

где – ранг (место), который имеет *j*-й вариант по *i*-му показателю.

Выигрывает вариант, набравший наименьшую сумму баллов.

Найдем значение интегрального критерия по таблице 17:

Таблица 17 – Метод ранжирования Борда

| **Критерий** | **Ранги** | | |
| --- | --- | --- | --- |
| **В1** | **В2** | **В3** |
| К1 | 3 | 2 | 1 |
| К2 | 2.5 | 1 | 2.5 |
| К3 | 3 | 1,5 | 1,5 |
| К4 | 2 | 1 | 3 |
|  | 10.5 | **5,5** | 8 |

По полученным результатам можно сделать вывод, что вариант В2 является предпочтительным по методу ранжирования Борда.

Далее воспользуемся методом взвешенного произведения локальных критериев. В данном методе наилучший вариант находят, используя формулу 4:

, (4)

где

kij – оценка варианта в определенном критерии,

αi – весовой коэффициент критерия.

Тогда рассчитаем Yl по формуле 3 для каждого из вариантов:

Для В1: Y1 = 0.40.36 \* 0.70.21 \* 0.80.14 \* 0.90.29 = 0.63

Для В2: Y2 = **0.91**

Для В3: Y3 = 0.86

Запишем все измерения в таблицу 18.

Таблица 18 – Метод взвешенного произведения локальных критериев

| **Критерий** | **Весовой коэффициент** | **Количественная оценка** | | |
| --- | --- | --- | --- | --- |
| В1 | В2 | В3 |
| К1 | 0.36 | 0.4 | 0.8 | 1 |
| К2 | 0.21 | 0.7 | 1 | 0.7 |
| К3 | 0.14 | 0.8 | 0.9 | 0.9 |
| К4 | 0.29 | 0.9 | 1 | 0.8 |
|  | | 0.63 | 0.91 | 0.86 |

Y2набрал наибольшее значение среди всех. Таким образом В2 является лучшим среди всех вариантов по методу взвешенного произведения локальных критериев.

Далее найдем лучший вариант технологии методом близости сравниваемых вариантов к идеальному.

Вычислим Y по формуле 4 для каждого из 3 вариантов:

Для В1: Y1 =

Для В2: Y2 = 0,29

Для В3: Y3 = 0,37

Запишем измерения в таблицу 19.

Таблица 19 – Метод близости сравниваемых вариантов к идеальному

| **Критерий** | **Весовой коэффициент** | **Количественная оценка** | | |
| --- | --- | --- | --- | --- |
| В1 | В2 | В3 |
| К1 | 0.36 | 0.4 | 0.8 | 1 |
| К2 | 0.21 | 0.7 | 1 | 0.7 |
| К3 | 0.14 | 0.8 | 0.9 | 0.9 |
| К4 | 0.29 | 0.9 | 1 | 0.8 |
|  | | 0.58 | 0.29 | 0.37 |

Y2 набрал наименьшее значение из всех. Таким образом по методу сравниваемых вариантов к идеальному В2 является лучшим из всех.

Далее приведена сравнительная таблица всех технологий по каждому из методов анализа (таблица 20):

Таблица 20 – Сравнение технологий по каждому методу

| **Метод анализа** | **Лучший вариант** |
| --- | --- |
| Метод ранжирования Борда | В2 |
| Взвешенное произведение локальных критериев | В2 |
| Метод близости сравниваемых вариантов к идеальному | В2 |

По итогам анализа технологий разработки программного средства выяснилось, что наилучшим по результату отбора по 3 методам является вариант В2. Таким образом, для разработки фронтенд модуля системы будет использоваться библиотека React.

2.3 Архитектура системы

Диаграмма прецедентов позволяет проиллюстрировать разные типы ролей в системе и взаимодействие данных ролей с системой. Роли и функции отражены на диаграмме прецедентов на рисунке 6:

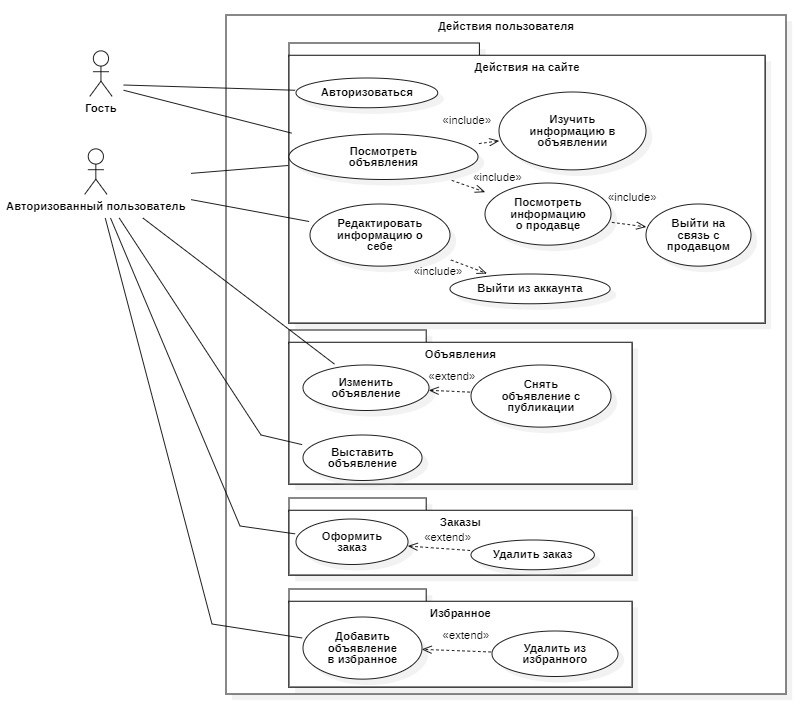


Рисунок 6 – Диаграмма прецедентов

Разрабатываемая система обеспечивает:

* Разделение пользователей по ролям: гость и авторизованный пользователь;
* Функционал гостя ограничен просмотром объявлений, регистрацией и авторизацией;
* Функционал авторизованного пользователя по сравнению с гостем расширен до следующих действий: редактирование информации в личном кабинете, добавление и изменение собственных объявлений, оформление и удаление заказов, добавление объявлений в избранное и удаление их оттуда.

Диаграмма активностей показывает динамические аспекты поведения системы. В диаграмме показывается, как поток управления переходит от одной деятельности к другой.

В данной диаграмме на рисунке 7 отражены 2 деятельности (гость и авторизованный пользователь) и действия, выполняемые в каждой из них.

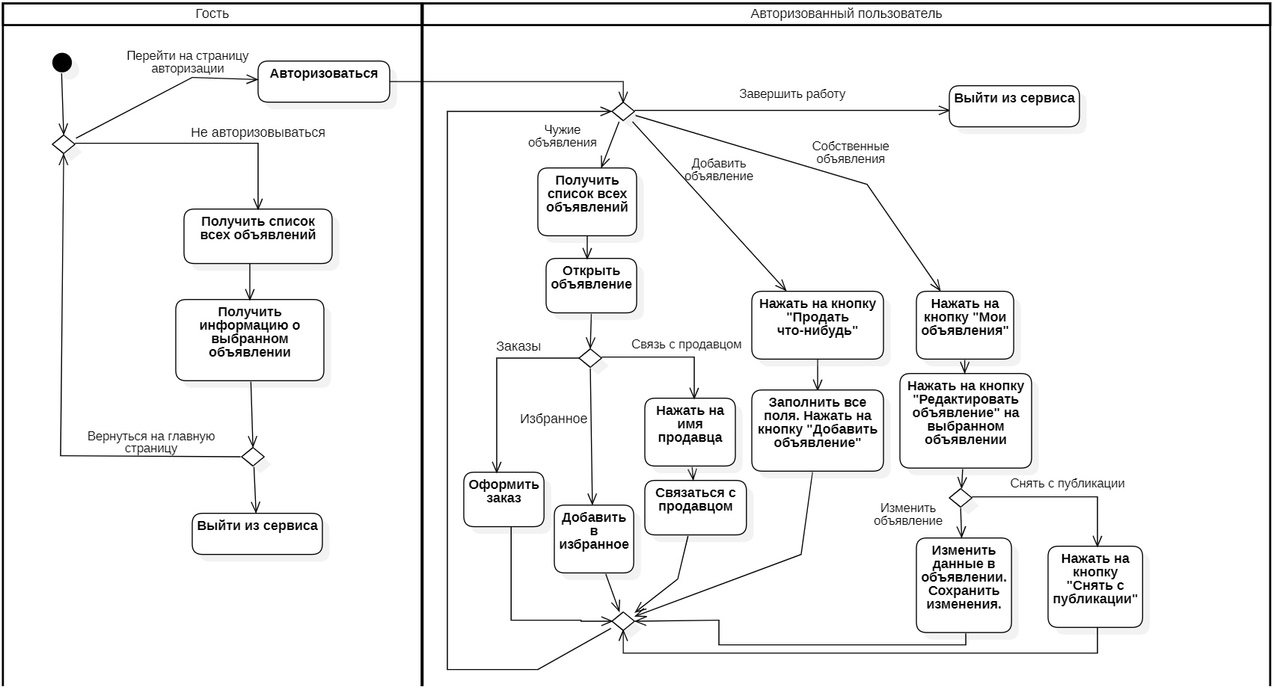


Рисунок 7 – Диаграмма активностей

Диаграмма состояний показывает поведение объекта, его переход из одного состояния в другое. На рисунке 8 изображён процесс приобретения товара.

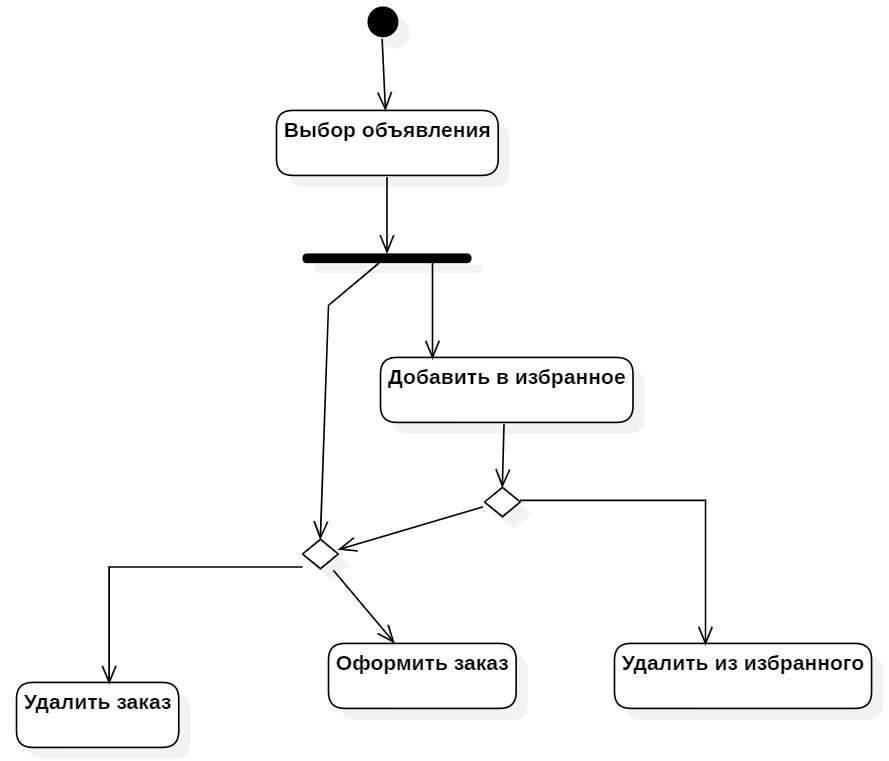


Рисунок 8 – Диаграмма состояний бизнес-процесса покупки

2.4 Описание функционала интерфейса системы

2.4.1 Функции гостя

При входе в систему пользователь попадает на главную страницу. Здесь у него есть несколько возможностей: вход, регистрация, поиск по объявлениям. На рисунке 9 изображена главная страница сервиса.

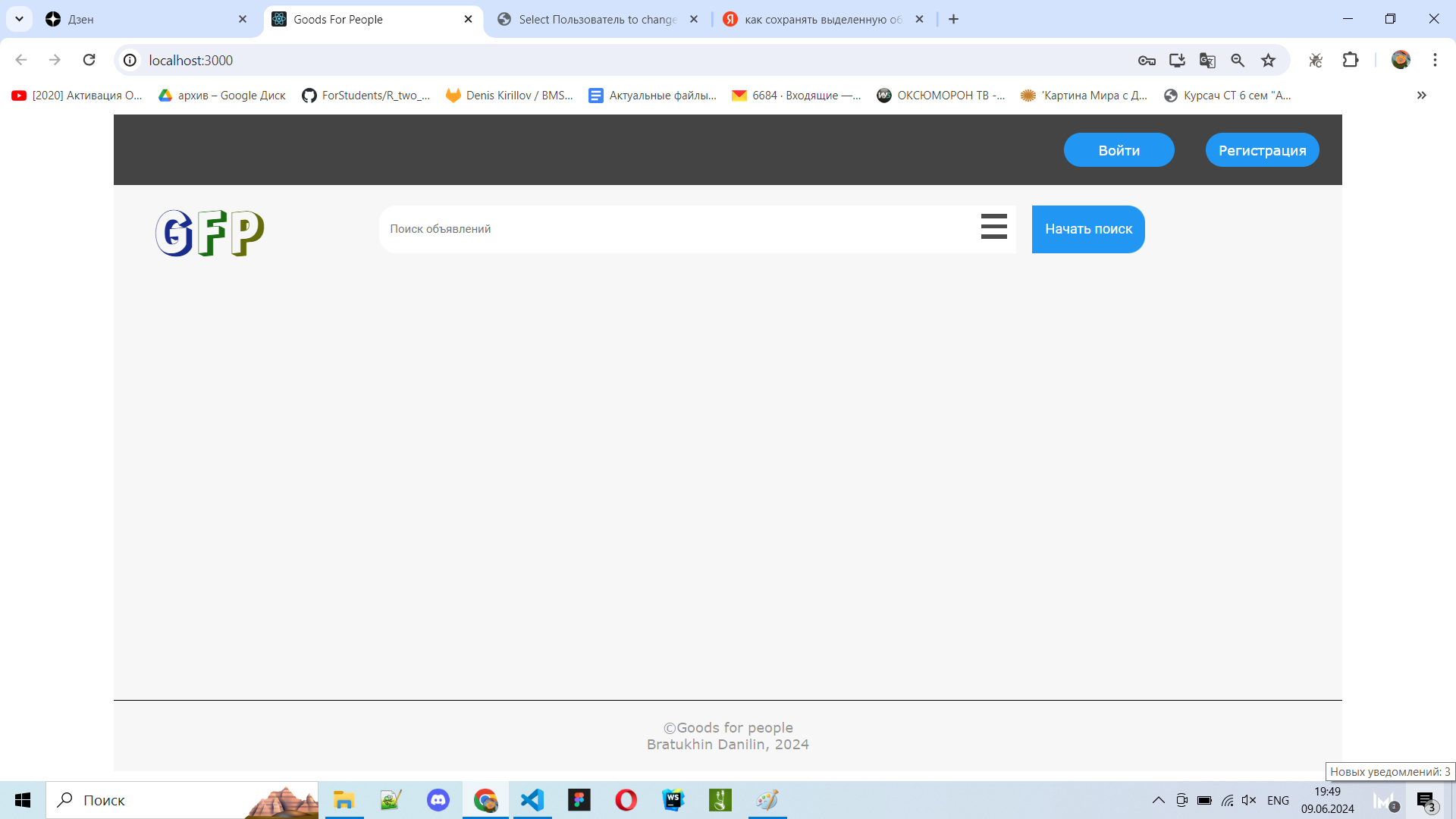


Рисунок 9 – Главная страница

Пользователь может осуществить поиск объявлений. Для этого он вводит в текстовое поле фразу, по которой происходит поиск нужного товара и жмет кнопку Начать поиск. На рисунке 10 изображен результат:

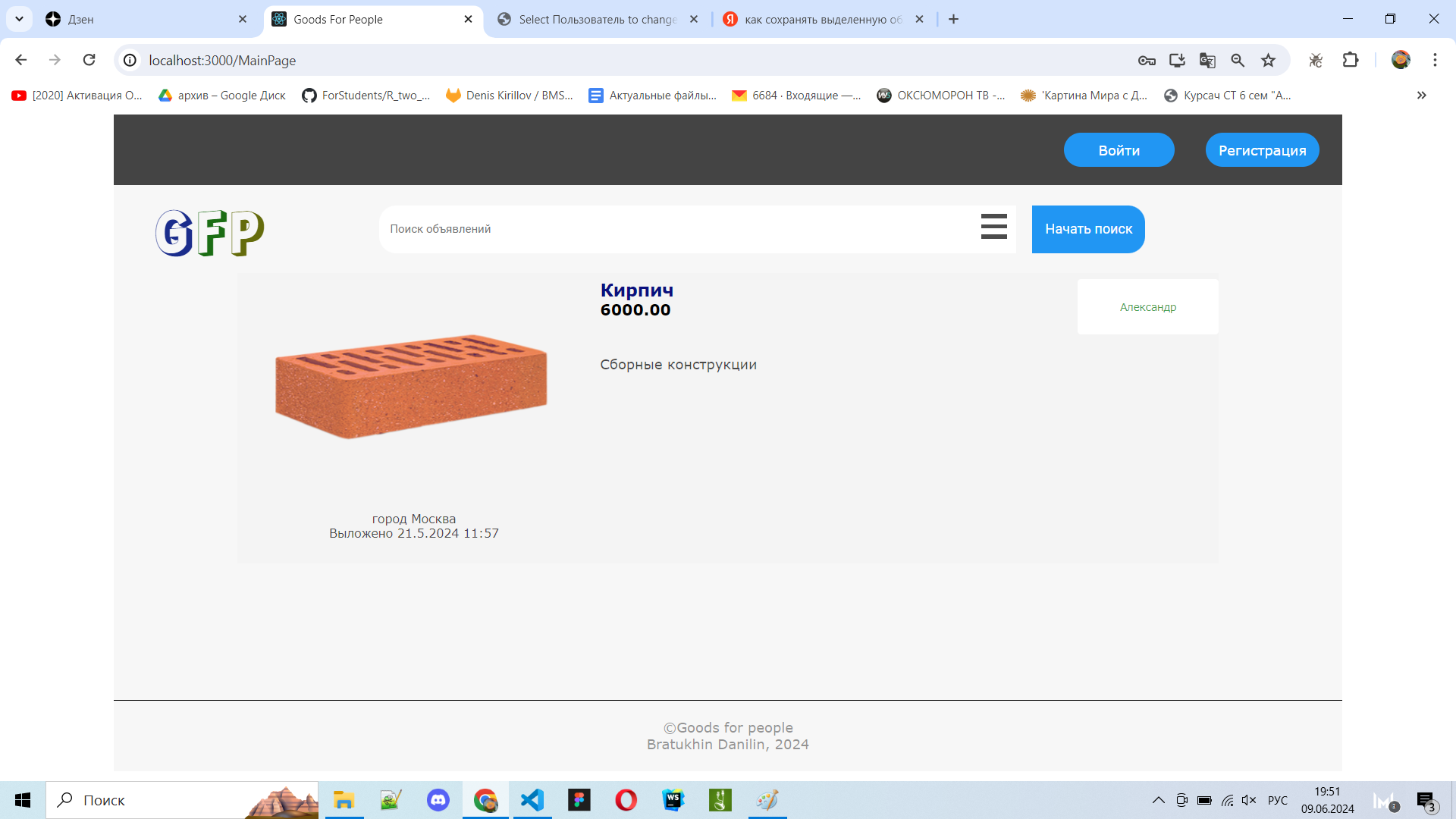


Рисунок 10 – Результат поиска гостя

Также пользователь может осуществить фильтрацию объявлений по нескольким парметрам: категории, цене и городу. Эта функция проиллюстрирована на рисунке 11:

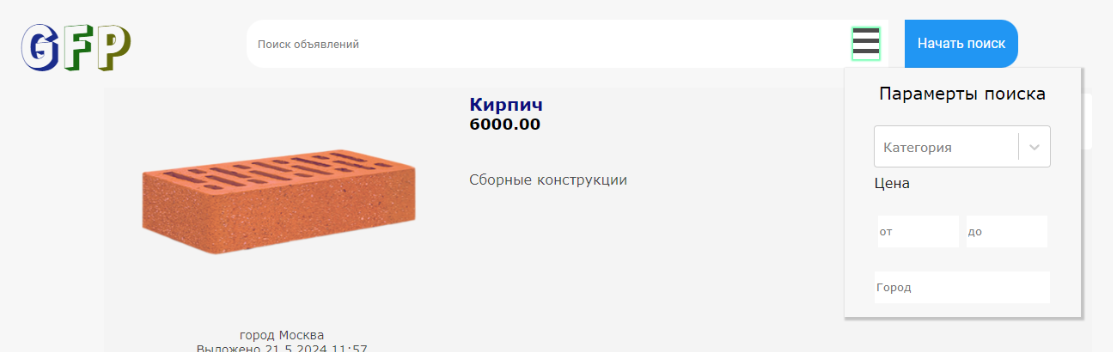


Рисунок 11 – Функция фильтрации объявлений

У гостя есть возможность зарегистрироваться в системе. Для этого нужно нажать на кнопку Регистрация в верхней части экрана. Пользователь попадает в форму регистрации, изображенную на рисунке 12:

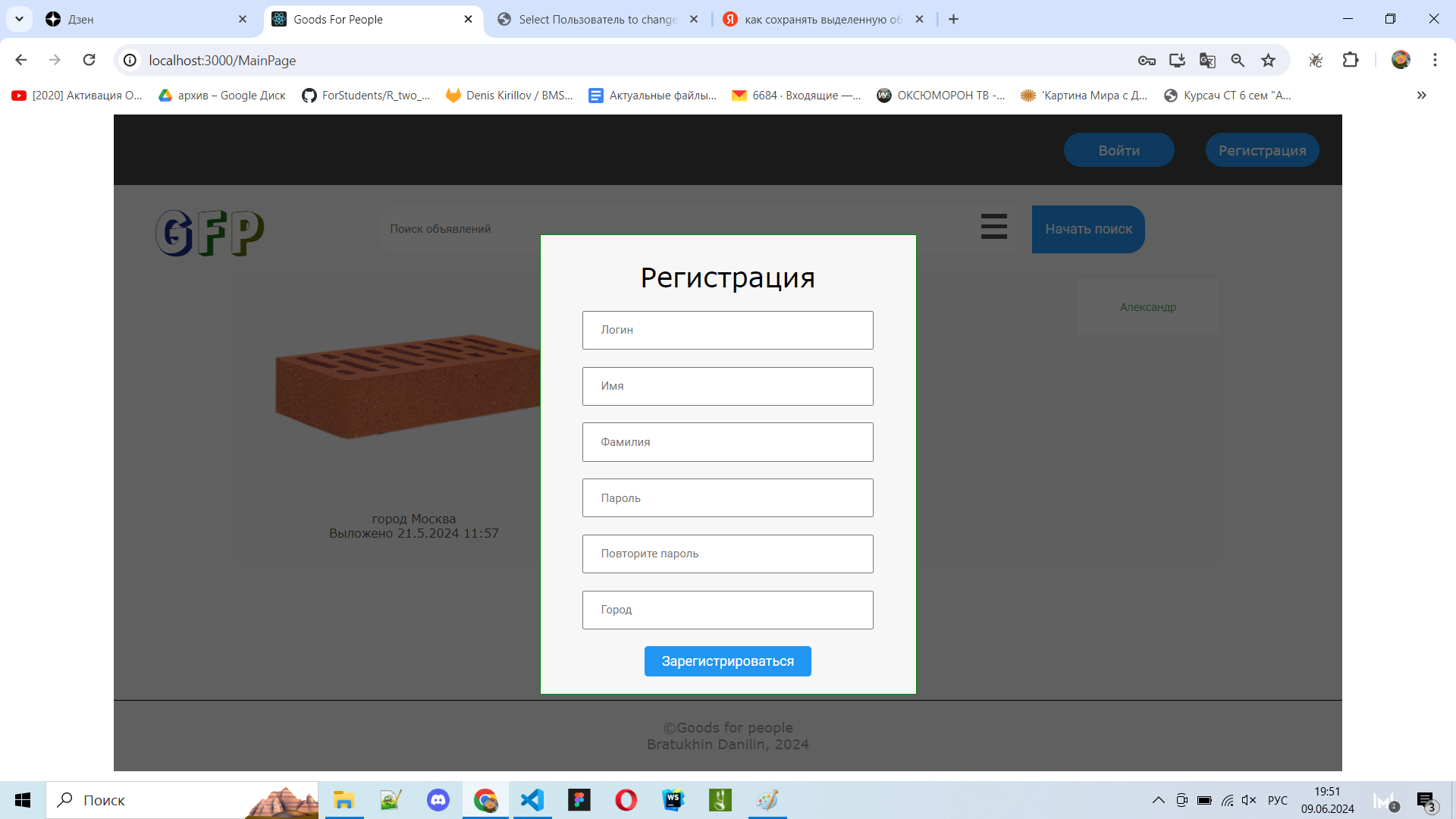


Рисунок 12 – Форма регистрации

Порядок регистрации:

* Нажать кнопку «Регистрация» в правом верхнем углу;
* Заполнить поля «Логин», «Имя», «Фамилия», «Пароль», «Повторите пароль», «Город»;
* Нажать кнопку «Зарегистрироваться»;

После успешного завершения регистрации пользователю придется пройти авторизацию в системе по имеющимся данным.

Для того чтобы авторизоваться в системе, требуется нажать кнопку Войти вверху экрана. Откроется форма авторизации, изображенная на рисунке 13:

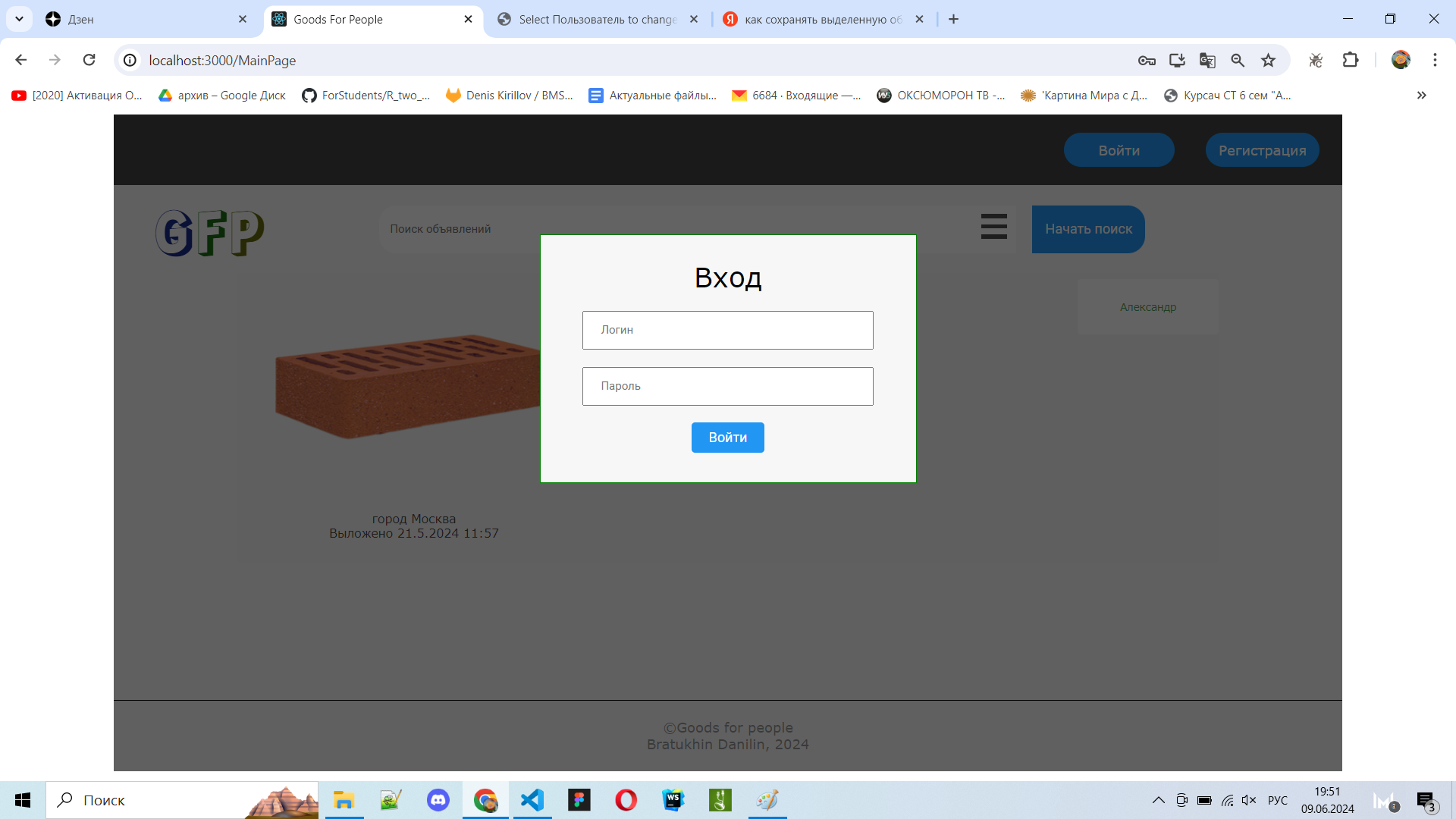


Рисунок 13 – Форма авторизации

Порядок авторизации:

* Нажать кнопку «Войти» в правом верхнем углу;
* Заполнить поля «Логин», «Пароль»;
* Нажать кнопку «Войти».

Логином может являться номер телефона, электронная почта или непосредственно логин пользователя (user name).

Если логин или пароль пользователя были введены неверно, то будет выведена ошибка авторизации, и Пользователю будет необходимо пройти авторизацию еще раз.

Если логин и пароль пользователя были введены правильно, то Пользователь будет перемещен на главную страницу, ему откроются функции авторизированного пользователя.

Неавторизованный в системе гость может просматривать все объявления других пользователей, но у него нет возможности добавить объявление в избранное либо оформить заказ. Просмотр объявления гостем изображен на рисунке 14:

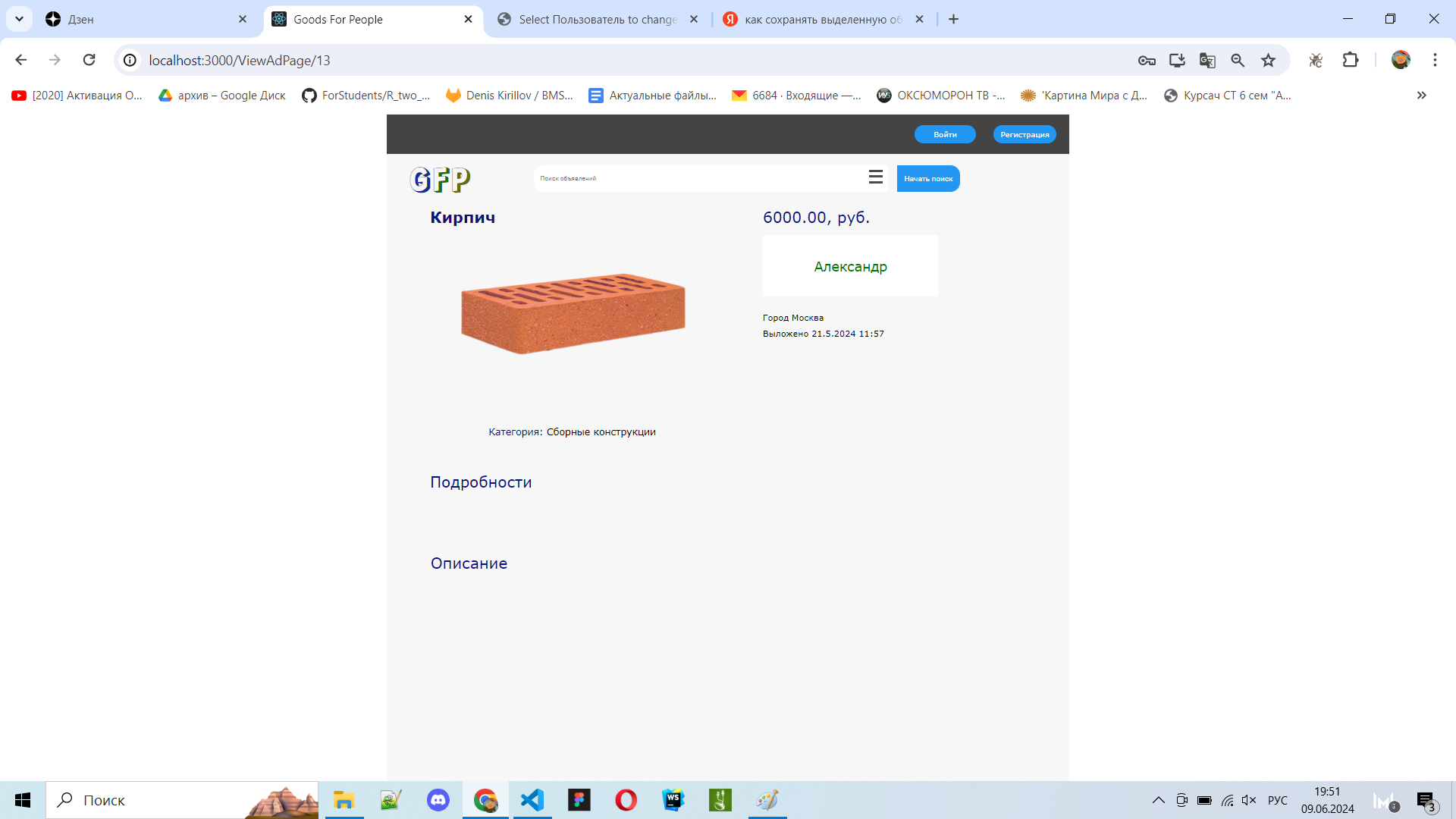


Рисунок 14 – Просмотр объявления гостем

2.4.2 Функции авторизованного пользователя

Авторизованный пользователь имеет те же функции, прописанные в пункте 2.6.1, что и гость. Далее приведено описание тех действий, которые пользователь может совершить, находясь на любой странице сервиса.

Для того, чтобы просмотреть и редактировать информацию о себе, требуется совершить несколько действий:

Для изменения информации о себе (без пароля):

* На любой странице нажать сверху слева на кнопку с собственными фамилией и именем.
* На странице редактирования информации о себе ввести данные, которые нужно изменить.
* Нажать на кнопку «Обновить данные».

Страница с личными данными изображена на рисунке 15:

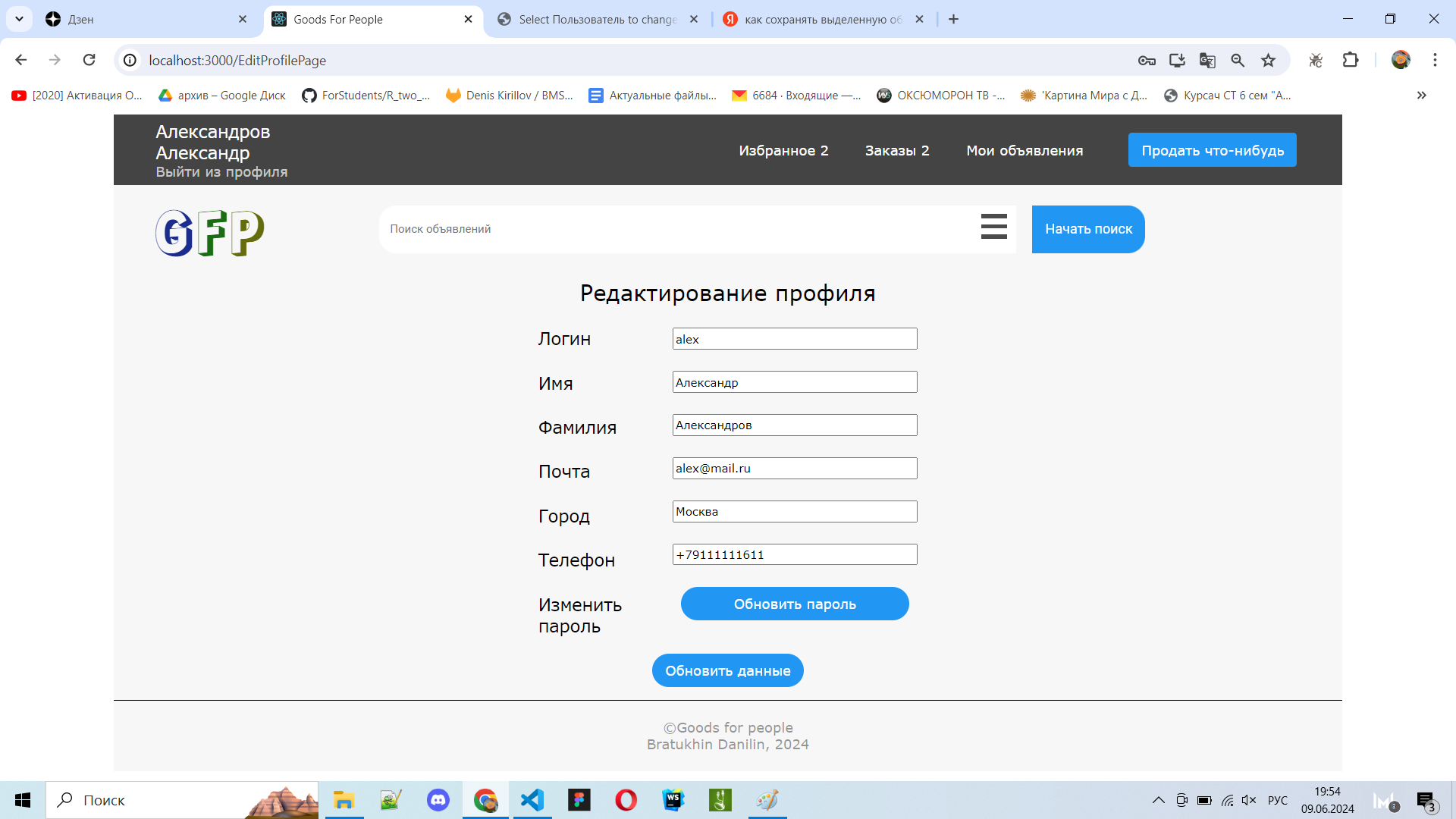


Рисунок 15 – Страничка личного профиля

Для изменения пароля:

* На странице редактирования информации о себе нажать на кнопку «Обновить пароль».
* На странице редактирования пароля ввести свой старый пароль, новый пароль, и повторить ввод нового пароля.
* Нажать на кнопку «Обновить пароль».

При правильно введенных данных информация о профиле пользователя обновится и система перекинет пользователя на главную страницу, иначе система потребует повторно ввести данные и оставит пользователя на этой же странице. Страница изменения пароля представлена на рисунке 16:

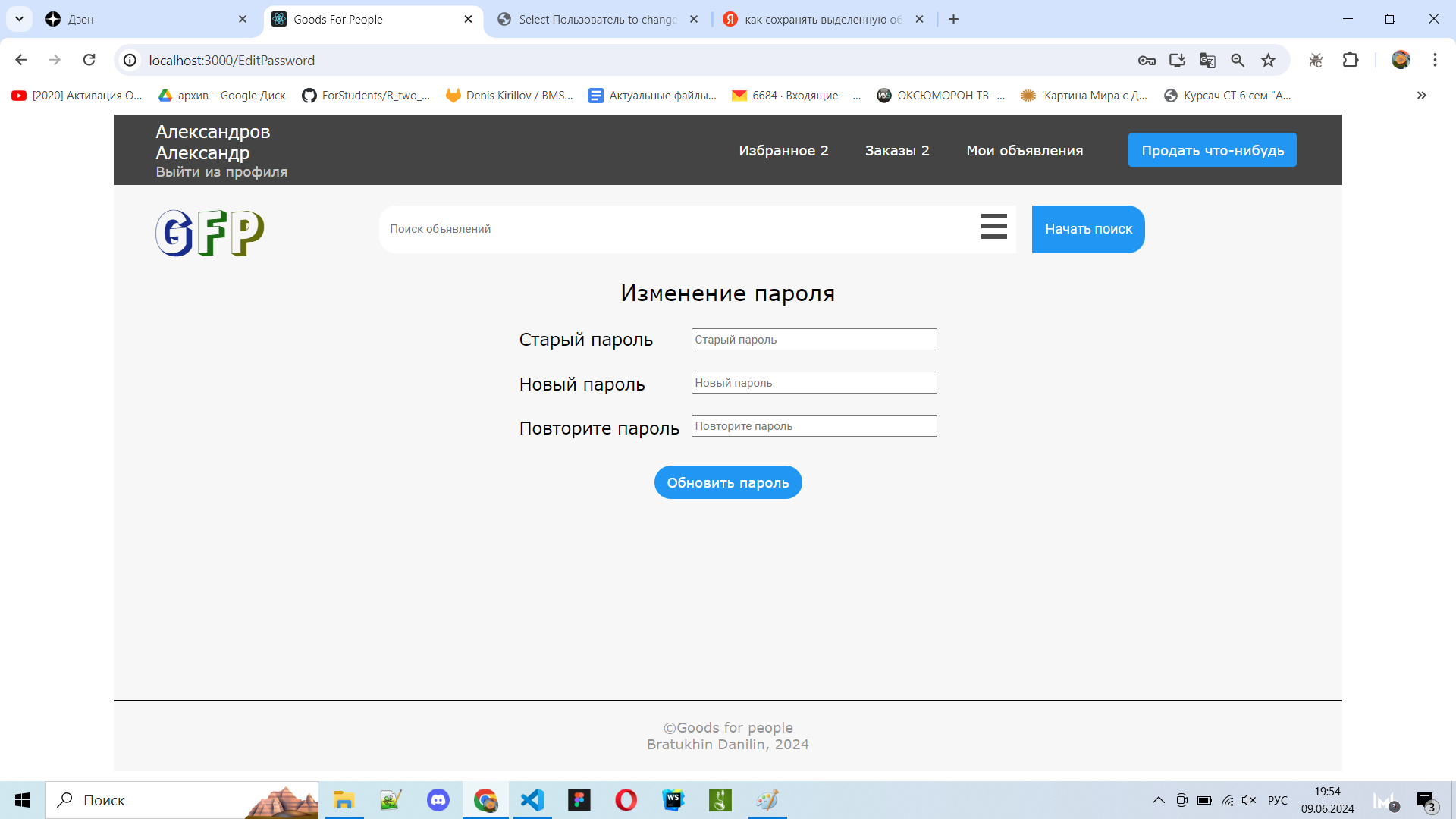


Рисунок 16 – Страница изменения пароля

Помимо просмотра объявлений пользователь может просматривать информацию о другом пользователь, для связи с ним. Для того чтобы это сделать, нужно провести следующие действия:

* Выбрать и нажать на название понравившегося объявления другого пользователя. Произойдет переход на страницу с показом информации об объявлении.
* На странице показа объявления нажать на имя пользователя.
* Произойдет переход на страницу просмотра информации другого пользователя.

Находясь на странице просмотра информации другого пользователя можно посмотреть имя, фамилию, город, телефон и почту пользователя. Также будут показаны активные и снятые с публикации объявления. Показанные в виде небольших квадратов объявления кликабельны, по нажатию на них произойдет переход на страницу просмотра объявления. Страница информации о другом пользователе представлена на рисунке 17:

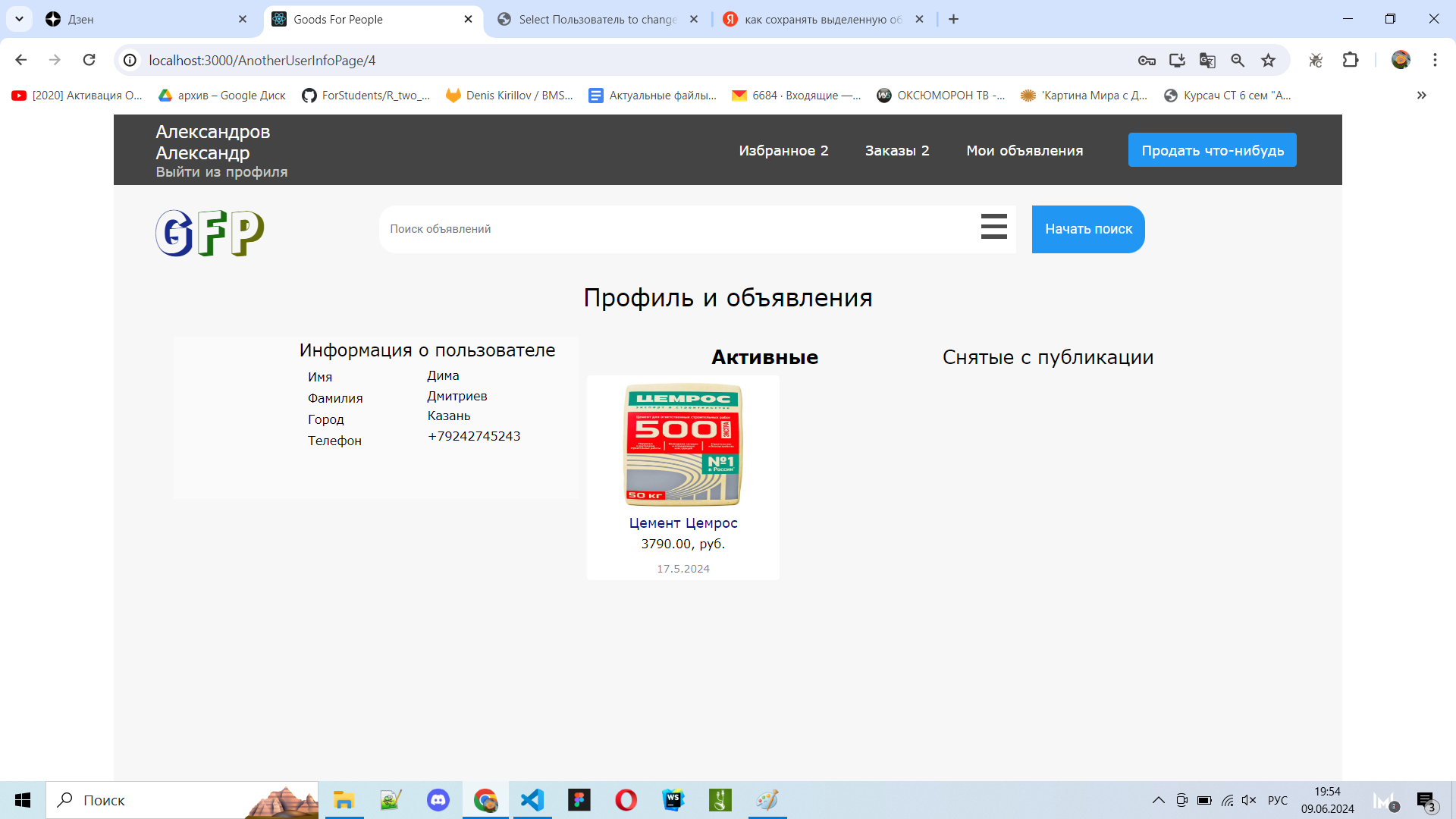


Рисунок 17 – Информация о другом пользователе

Авторизованный пользователь может добавлять понравившиеся объявления в избранное.

* На главной странице при полученном списке объявлений выбрать понравившееся объявление(рисунок 18).

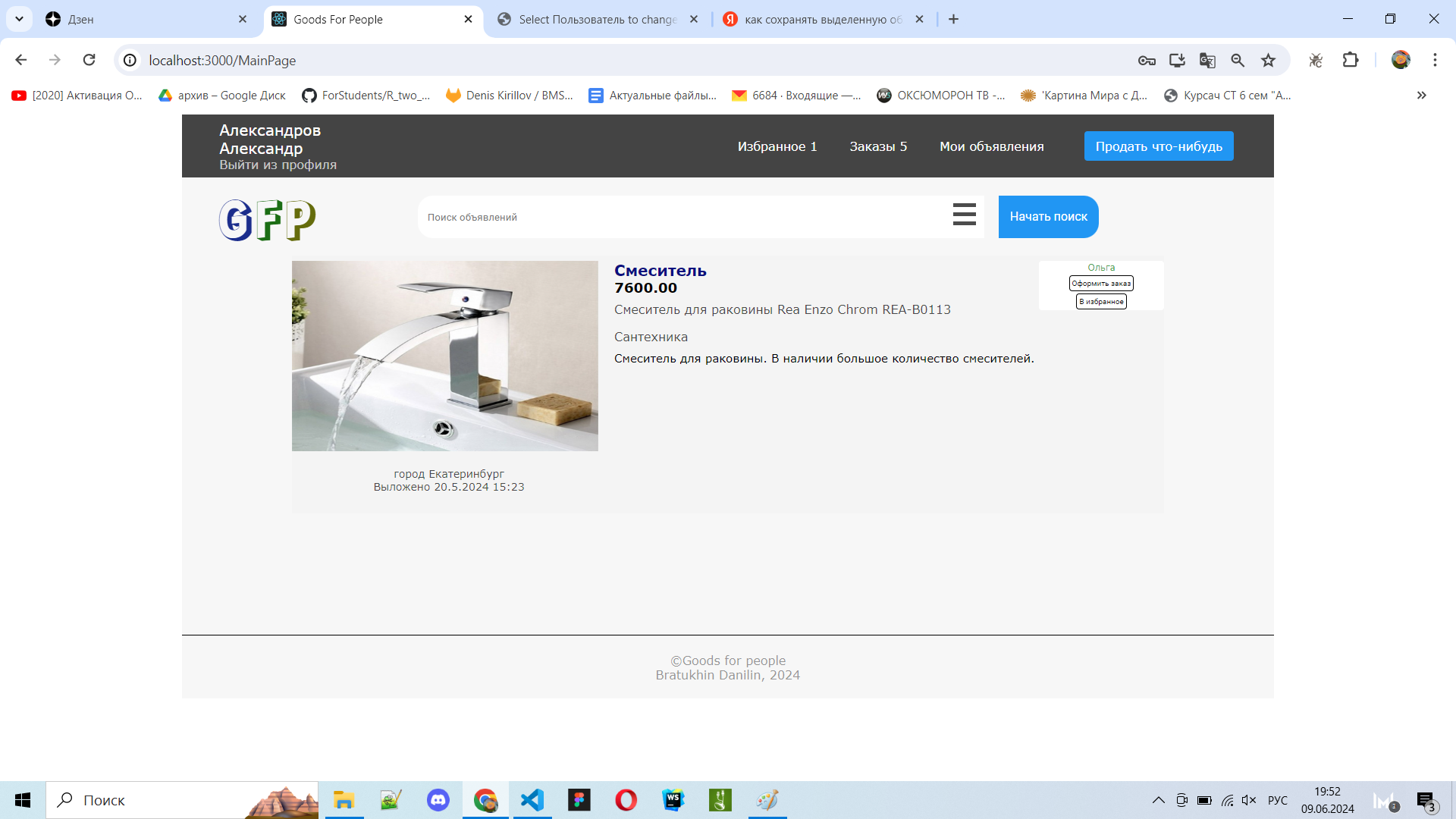


Рисунок 18 – Добавление объявления в избранное

* Справа от объявления нажать на кнопку «В избранное»(рисунок 19).

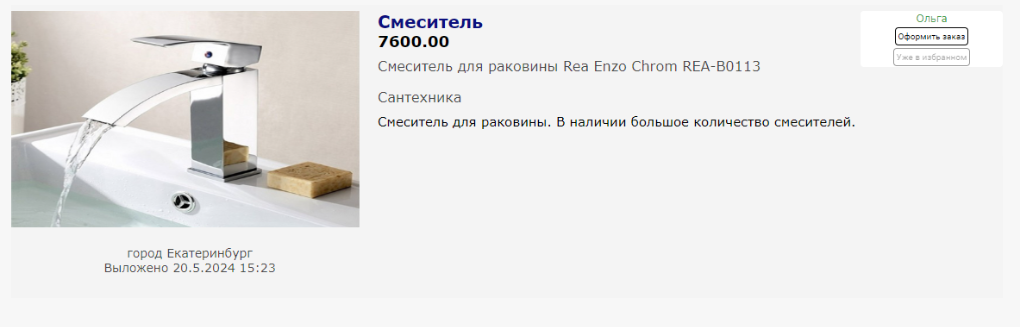


Рисунок 19 – Добавленное в избранное объявление

* Объявление добавится в избранные.

Одно объявление можно добавить в избранное один раз. Для удаления объявления из избранного нужно проделать следующие действия:

* На любой странице нажать сверху экрана на кнопку «Избранное».
* Произойдет переход на страницу с избранными объявлениями(рисунок 20).

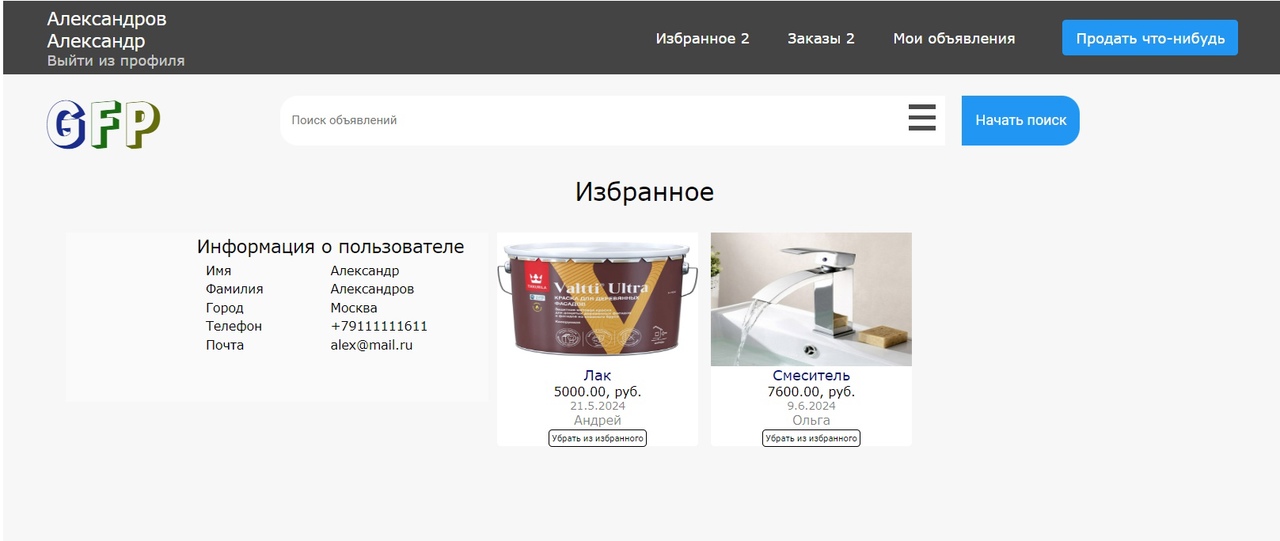


Рисунок 20 – Избранные объявления

* Выбрать нужное объявление и в нем нажать на кнопку «Убрать из избранного».

Авторизованный пользователь имеет возможность создавать заказы. Можно сделать несколько заказов на одно объявление. Для этого:

* На главной странице при полученном списке объявлений выбрать понравившееся объявление.
* Справа от объявления нажать на кнопку «Оформить заказ». Заказ создастся и его можно будет просмотреть как у пользователя, оформившего заказ, так и пользователя, чье объявление добавили в заказ(рисунок 21).

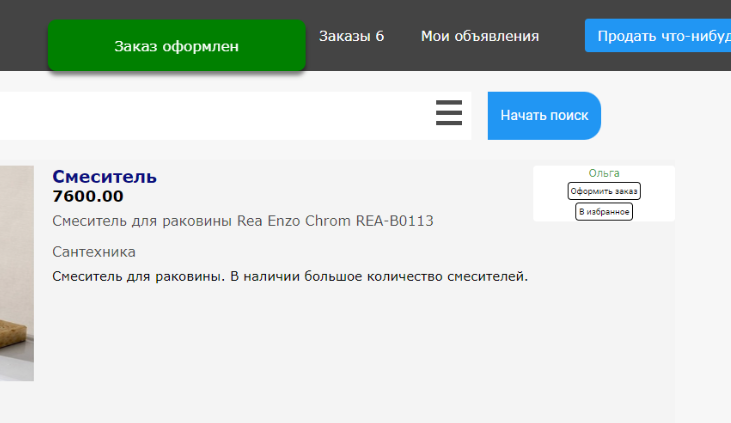


Рисунок 21 – Создание заказа

Для просмотра всех своих заказов провести следующие действия:

* На любой странице нажать сверху экрана на кнопку «Заказы».
* На странице заказов есть две вкладки: Мои заказы и Заказаны у меня. Автоматически выбрано Мои заказы(рисунок 22).



Рисунок 22 – Страница собственных заказов

* Выбрать нужное заказанное объявление и в нем нажать на кнопку «Удалить заказ».

На рисунке 23 показана страница заказов, оформленных другими пользователями у меня.

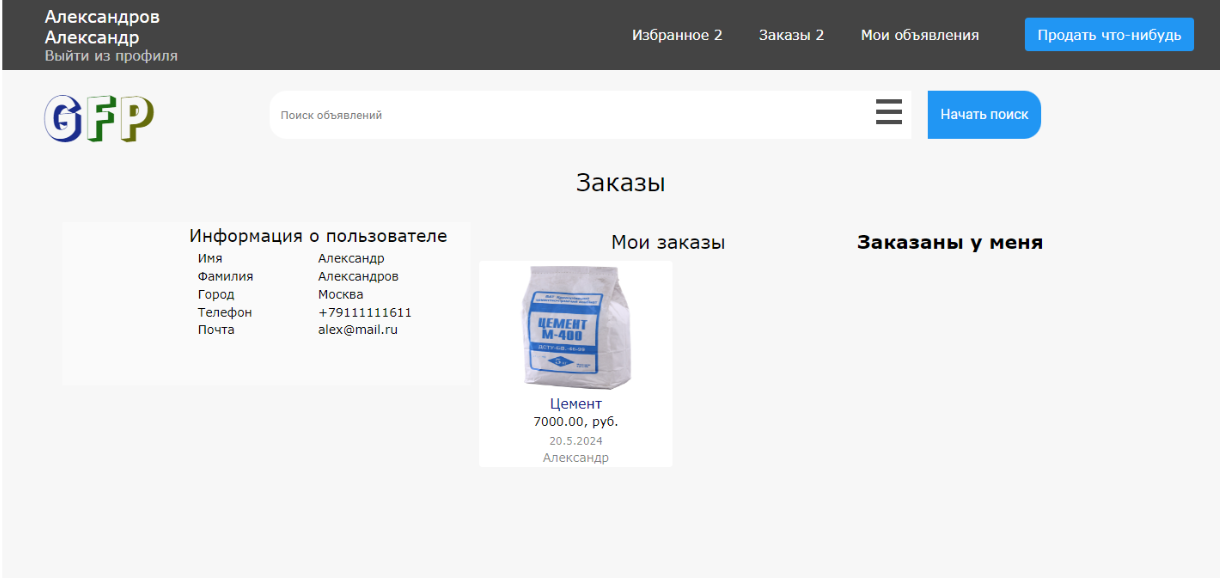


Рисунок 23 – Страница заказанных у меня товаров

На странице создания объявлений для создания своего объявления нужно выбрать категорию из выпадающего списка, ввести название, установить цену. Далее при необходимости можно: загрузить фотографию товара, ввести краткое и подробное описание.

* На любой странице нажать сверху экрана на кнопку «Продать что-нибудь».
* На странице создания объявления провести следующие действия: выбрать категорию из выпадающего списка, ввести название, установить цену, загрузить фотографию товара, ввести краткое и подробное описание.
* По окончании заполнения данных внизу страницы нажать на кнопку «Добавить объявление»(рисунок 24).

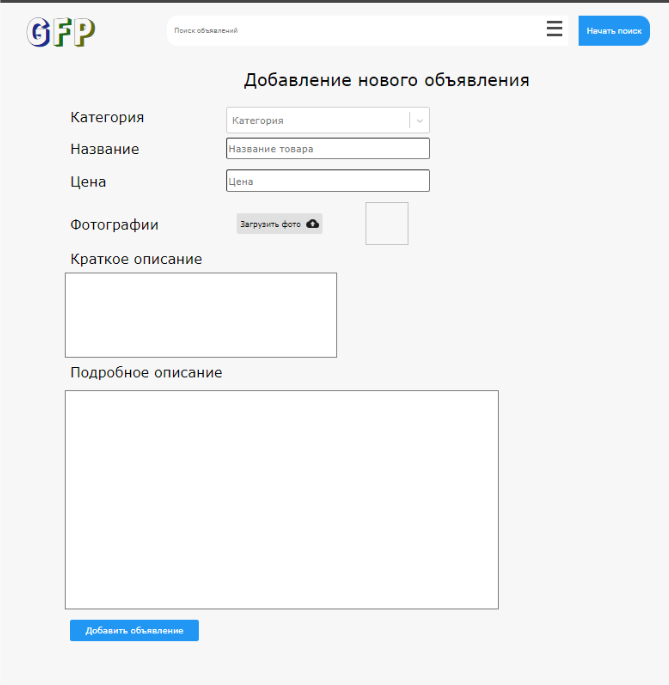


Рисунок 24 – Добавление нового объявления

При корректном вводе информации в обязательные поля (категория, название, цена) объявление добавится в систему и будет доступно для просмотра всем пользователям.

Для редактирования собственных объявлений требуется выполнить следующие действия:

* На любой странице нажать сверху экрана на кнопку «Мои объявления».
* На странице собственных объявлений в графе активные либо снятые с публикации выбрать интересующее объявление и нажать на кнопку «Редактировать».
* На странице редактирования объявления изменить необходимые данные.
* Если требуется только обновить информацию в объявлении, нажать на кнопку «Обновить объявление». Иначе нажать на кнопку «Снять с публикации», чтобы сделать объявление невидимым другим пользователям. Чтобы объявление было опять видно всем, нажать на кнопку «Опубликовать».

На рисунке 25 изображено окно редактирования объявления.

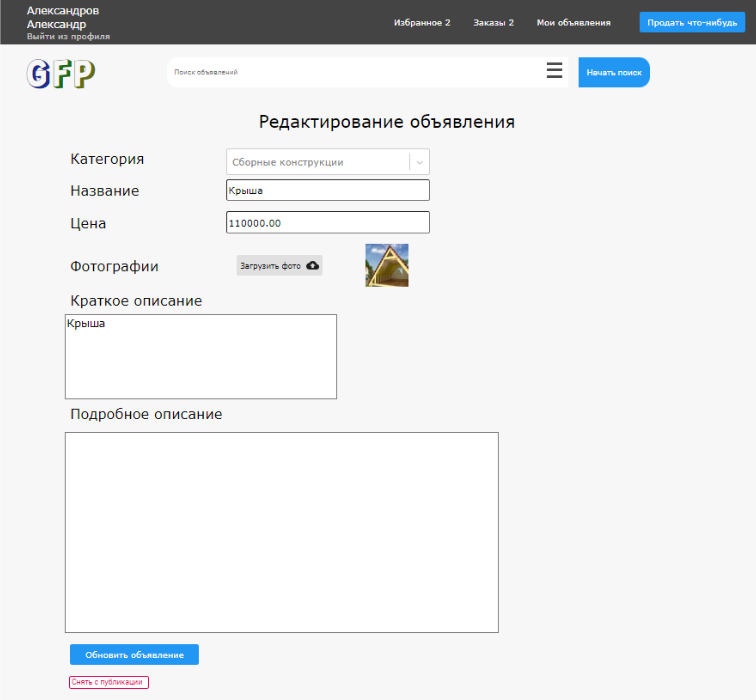


Рисунок 25 – Страница редактирования объявления

После проведенных действий при корректно введенных данных информация в объявлении обновится.

ЗАКЛЮЧЕНИЕ

В рамках квалификационной работы был проведен комплекс мероприятий по проектированию архитектуры, написанию документации, составлению диаграмм и схем, разработке и тестированию кода программного средства для интерфейса информационной системы подачи объявлений по теме торговли строительными материалами.

Была рассмотрена и проанализирована предметная область исследуемой работы.

Выбраны и описаны критерии качества системы, а также проведено сравнение ближайших аналогов разработанной системы по выявленным критериям.

Был проведен анализ существующих технологий разработки современных веб-интерфейсов и выбрана оптимальная технология для создания собственной системы.

Таким образом, с общими понятиями, принципами и требованиями, применяемыми при проектировании и разработке веб-сервиса публикации объявлений.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. React.js. Быстрый старт./Стефанов С. — СПб.: Питер, 2017. — 304 с.
2. Автоматизированные системы управления и связь: Учебник/ В.И. Зыков, А.В. Командиров, А.Б. Мосягин, И.М. Тетерин; под общей ред. В.И. Зыкова. – 2-е изд., перераб. и доп. – М.: Академия ГПС МЧС России, 2006. – 655 с.
3. Веб-сервис [Электронный ресурс] - <https://wiki.loginom.ru/articles/web-service.html> (дата обращения 2024-05-29)
4. Виды строительных материалов [Электронный ресурс] <https://werton.ru/newsline/articles/stroitelnye-materialy-kakovy-ix-vidy-i-razdelenie/> (дата обращения 2024-06-02)
5. ГОСТ 7.32-2017 Отчет о научно-исследовательской работе. Структура и правила оформления.
6. Документация Angular [Электронный ресурс] - <https://docs.angularjs.org/guide> (дата обращения 2024-05-22)
7. Документация React [Электронный ресурс] - <https://ru.legacy.reactjs.org/tutorial/tutorial.html> (дата обращения 2024-05-22)
8. Компоненты в React [Электронный ресурс] - <https://metanit.com/web/react/2.2.php> (дата обращения 2024-05-28)
9. Основы эксплуатации АСОИиУ Tом 1. Tехническое обслуживание АСОИиУ. 2- е издание переработанное и дополненное, учеб. пособие / В. М. Постников. — М.: Изд-во МГTУ им. Н. Э. Баумана, 2015. — 214 [2] с.: ил.
10. Пять строк кода. Роберт Мартин рекомендует. — СПб.: Питер, 2023. — 368 с.: ил
11. Статистика авито [Электронный ресурс] - <https://www.similarweb.com/ru/website/avito.ru/#ranking> (дата обращения 2024-05-17)
12. Статистика озон [Электронный ресурс] - <https://seller-edu.ozon.ru/mobile-app/statistics> (дата обращения 2024-05-17)
13. Фримен А. Angular для профессионалов. — СПб.: Питер, 2018. — 800 с.: ил
14. Цифровые платформы [Электронный ресурс] - <https://hsbi.hse.ru/articles/tsifrovye-platformy/> (дата обращения 2024-05-19)
15. ЦИФРОВЫЕ ПЛАТФОРМЫ В ЭКОНОМИКЕ: СУЩНОСТЬ, МОДЕЛИ, ТЕНДЕНЦИИ РАЗВИТИЯ - Научно-технические ведомости СПбГПУ. Экономические науки. Том 11, № 6, 2018 Авторы: И.З. Гелисханов , Т.Н. Юдина, А.В. Бабкин
16. Фреймворк Vue [Электронный ресурс] -<https://habr.com/ru/articles/329452/> (дата обращения 2024-05-18)

**ПРИЛОЖЕНИЕ А  
ГРАФИЧЕСКАЯ ЧАСТЬ**

А1. Диаграмма прецедентов.

А2. Диаграмма активностей.

А3. Диаграмма состояний.

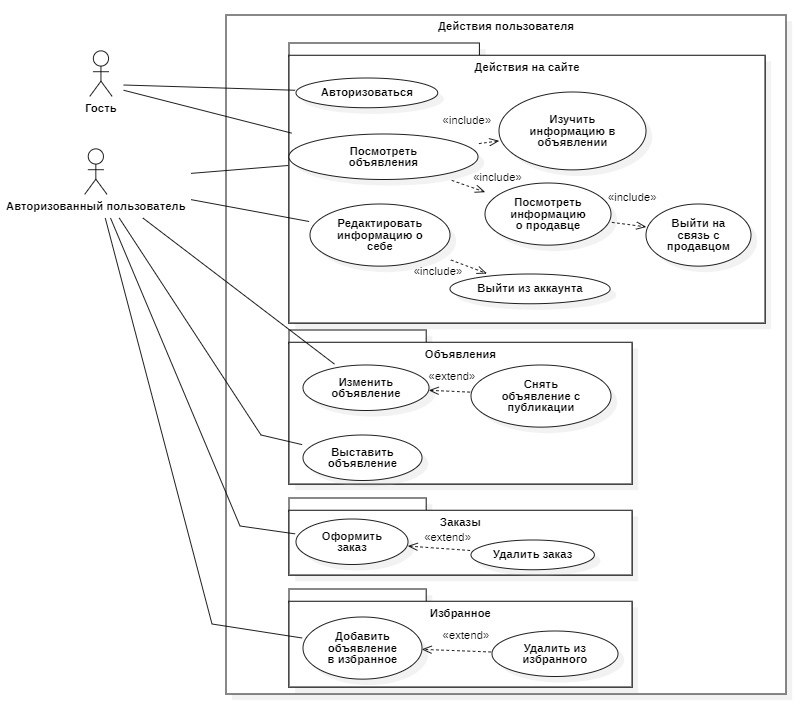
А4. Схема Redux.

А5. Схема Dispatcher

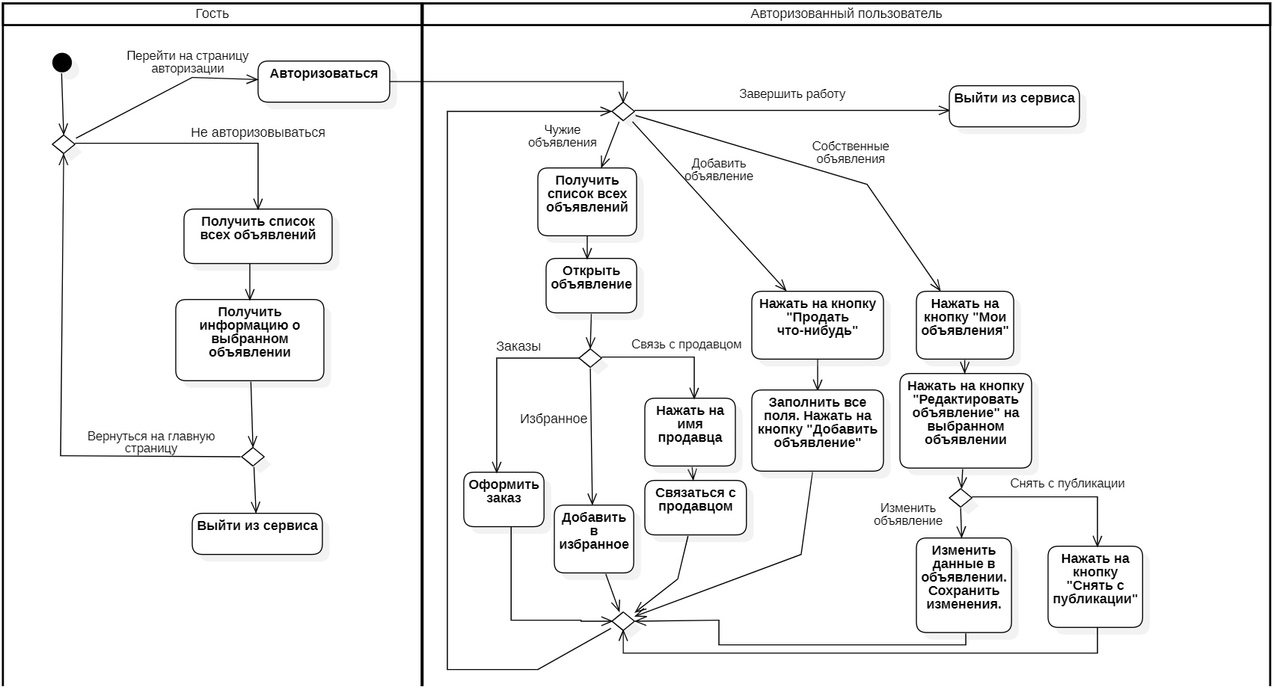
А6. Жизненный цикл React

А7. Технология AJAX

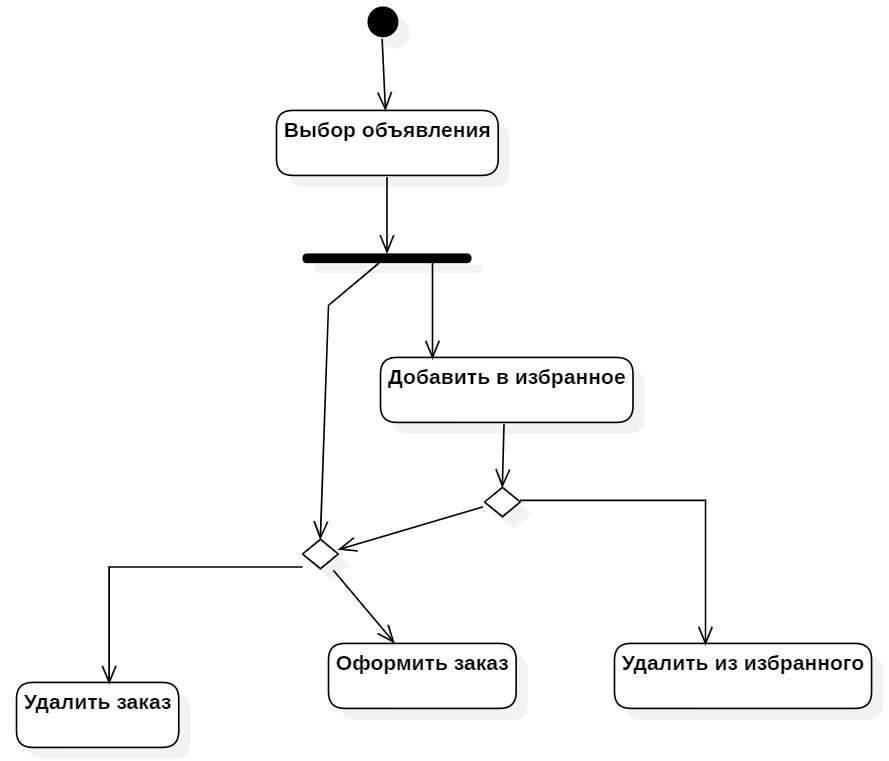
А1 Диаграмма прецедентов



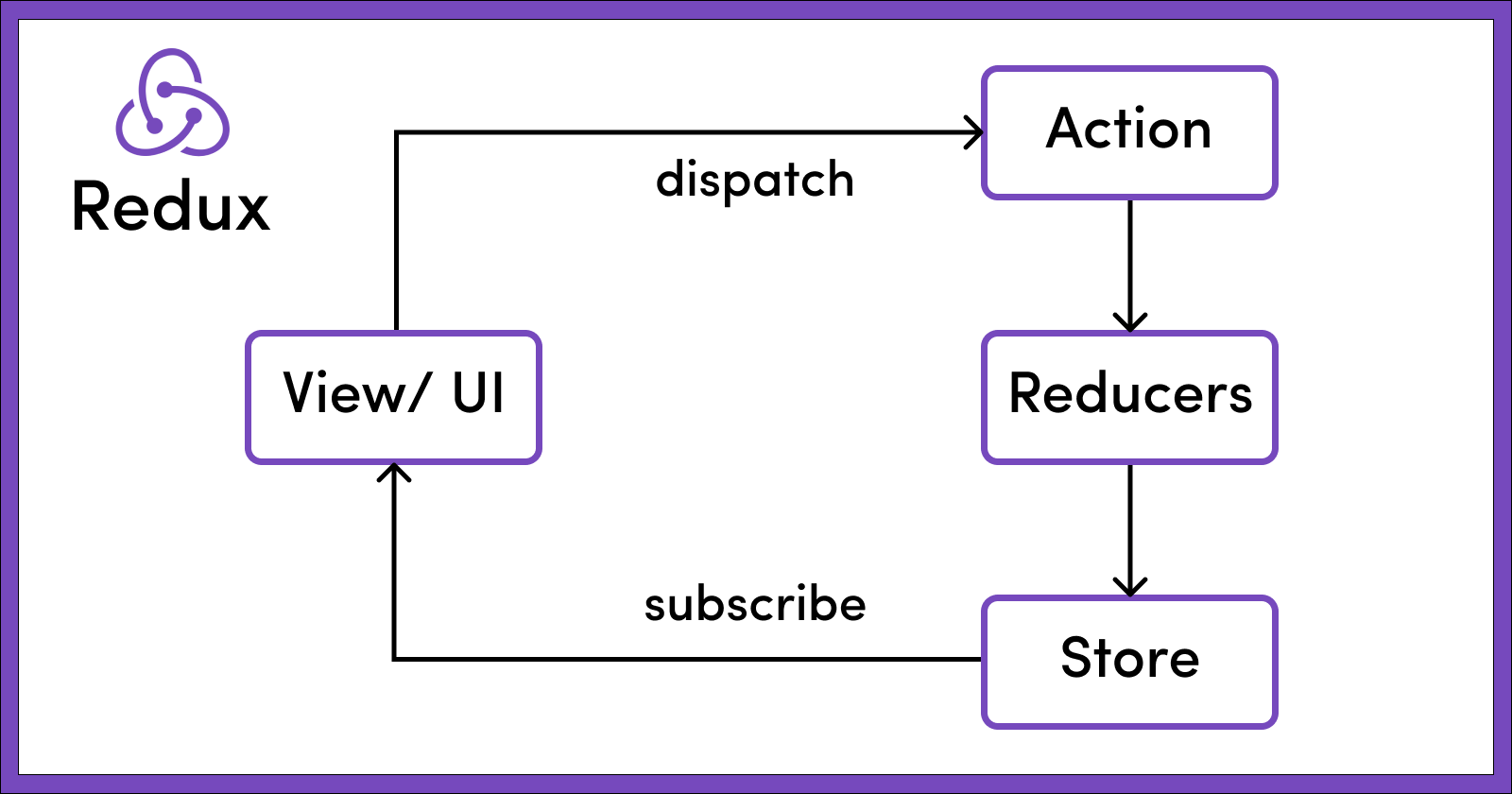
А2. Диаграмма активностей



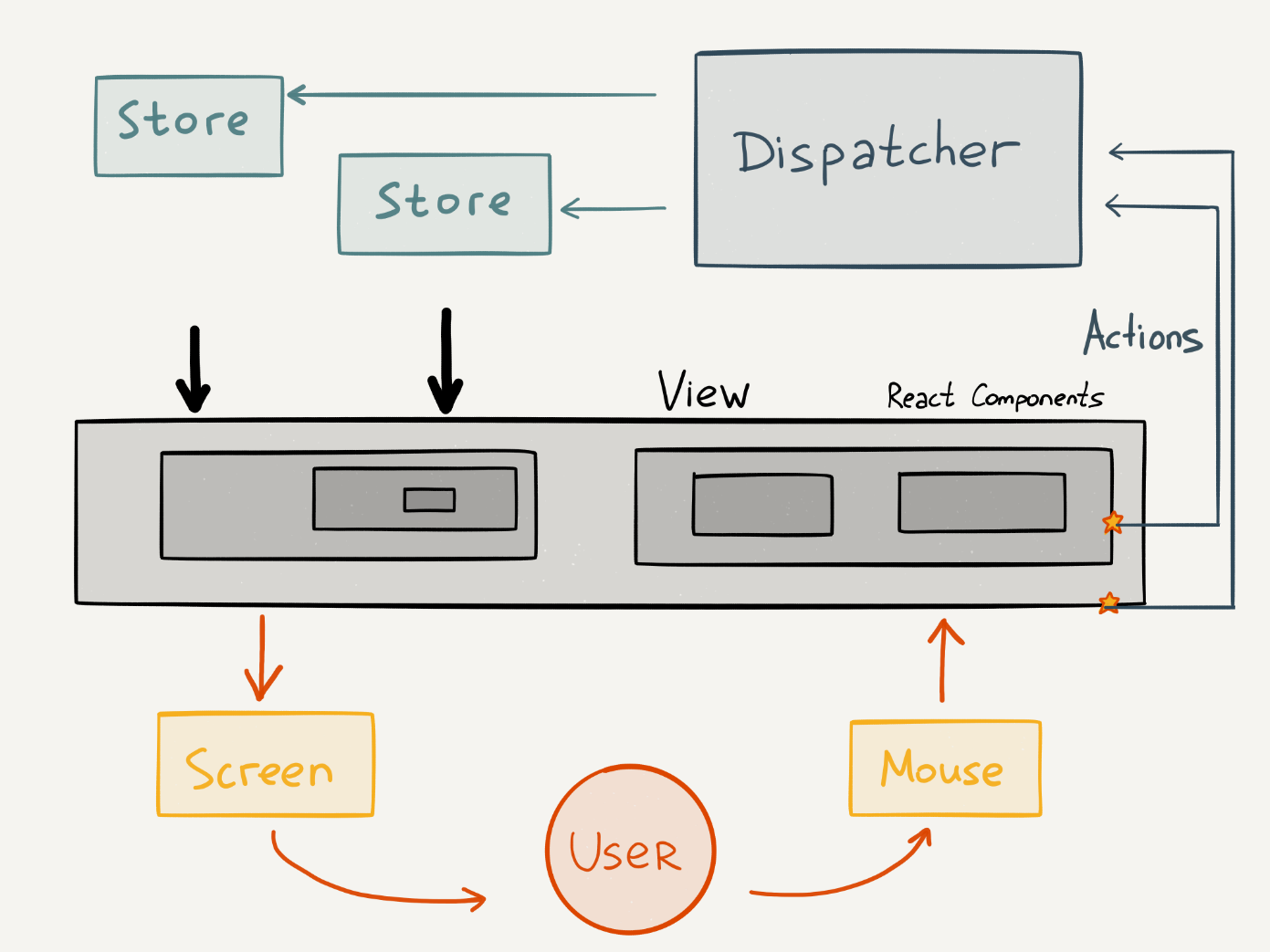
А3. Диаграмма состояний



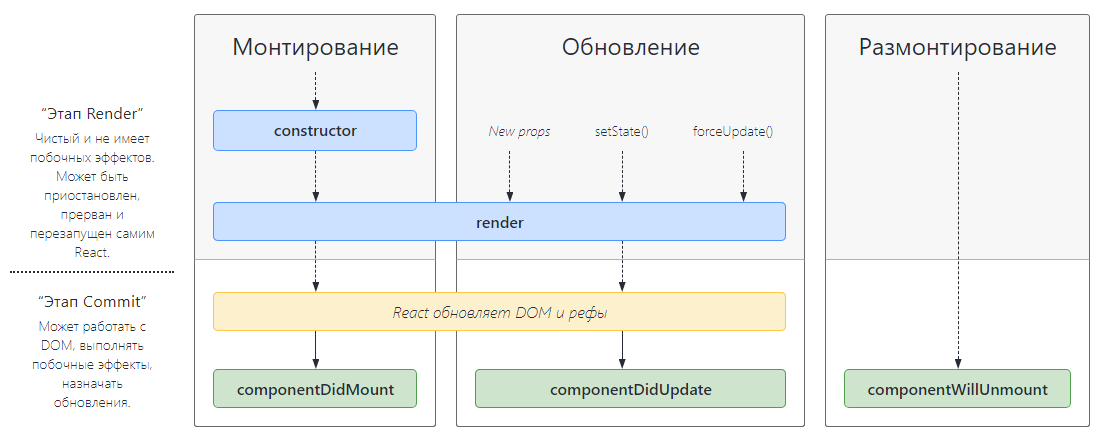
А4. Схема Redux



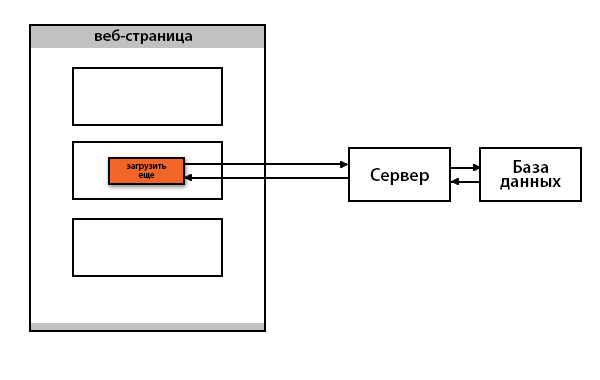
А5. Схема Dispatcher



А6. Жизненный цикл React



А7. Технология AJAX



**ПРИЛОЖЕНИЕ Б  
ТЕХНИЧЕСКОЕ ЗАДАНИЕ**

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

им. Н.Э. Баумана

Кафедра «Системы обработки информации и управления»

|  |  |  |  |
| --- | --- | --- | --- |
|  | Утверждаю  Заведующий кафедрой ИУ-5 |  | Согласовано  научный руководитель |
|  | \_\_\_\_\_\_\_\_ В.И.Терехов  "\_\_"\_\_\_\_\_\_\_\_\_\_2024 г. |  | \_\_\_\_\_\_\_С.В. Черненький  "\_\_"\_\_\_\_\_\_\_\_\_\_2024 г. |

**Информационная система подачи объявлений для мелкооптовой торговли строительными материалами. Модуль фронтенд.**

Техническое задание

(вид документа)

писчая бумага

(вид носителя)

5

(количество листов)

ИСПОЛНИТЕЛЬ: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Братухин Александр Сергеевич

"\_\_"\_\_\_\_\_\_\_\_\_\_\_2024 г.

Москва – 2024

**1. Введение**

Целью работы является разработка фронтенд модуля информационной системы подачи объявлений для мелкооптовой торговли строительными материалами. Данный модуль является интерфейсом информационной системы и предназначен для клиентов, которые будут использовать функционал системы.

**2. Основания для разработки**

Основанием для разработки является задание на выпускную квалификационную работу, подписанное руководителем выпускной работы и утверждённое заведующим кафедрой ИУ5 МГТУ им. Н.Э. Баумана 15 декабря 2023 года.

**3. Назначение разработки**

Назначением разработки является создание фронтенд модуля информационной системы, предназначенного для использования данного сервиса в рамках предметной области.

**4. Требования к программе или программному изделию**

4.1 Требования к функциональным характеристикам

Программное средство должно выполнять нижеперечисленные функции:

* + 1. Демонстрация объявлений в рамках предметной области.
    2. Поиск объявлений по названию.
    3. Поиск объявлений по выставленным параметрам поиска.
    4. Возможность получения подробной информации о выбранном объявлении.
    5. Возможность регистрации новых пользователей.
    6. Возможность авторизации пользователей.
    7. Возможность создания объявлений пользователями в рамках предметной области.
    8. Возможность редактирования добавленных пользователем объявлений в рамках предметной области.
    9. Возможность создания заказов.
    10. Возможность просмотра и удаления заказов, добавленных пользователем.
    11. Возможность просмотра заказов, добавленных другими пользователями на объявления авторизованного пользователя.
    12. Возможность добавления объявлений в избранное.
    13. Возможность просмотра всех объявлений, добавленных в избранное.
    14. Возможность просмотра информации о других пользователях.
    15. Возможность просмотра и редактирования информации о себе.
    16. Возможность смены пароля.
    17. Возможность выхода из системы.

4.2 Требования к надежности:

4.2.1 Программное средство должно обеспечивать корректный вывод информации.

4.2.2 Программное средство должно обеспечивать отклик с задержкой не более 5 секунд при наличии стабильного интернет соединения.

4.2.3 Программное средство не должно выдавать ошибки и создавать конфликты при работе.

4.3 Требования к составу и параметрам технических средств:

Необходимые технические средства для корректной работы клиентской части:

4.3.1 Наличие оперативной памяти компьютера в размере 2ГБ.

4.3.2 Процессор с тактовой частотой 1.1 ГГц и более.

4.3.3 Внутренняя память компьютера в размере 20ГБ и более.

4.3.4 Наличие периферийных средств: монитор, мышь, клавиатура.

4.3.5 Стабильное интернет соединение.

4.4 Требования к информационной и программной совместимости:

4.4.1 Наличие современной операционной системы Windows(не ниже 7 версии), Linux или MacOS.

4.4.2 Установленный веб-браузер версии 2020 года и позднее.

**5. Требования к программной документации**

Для представления заказчику разрабатываются следующие документы:

5.1 Техническое задание.

5.2 Программа и методика испытаний.

5.3 Руководство пользователя.

**6. Технико-экономические показатели**

Требования по данному разделу не предъявляются.

**7. Стадии и этапы разработки**

График выполнения отдельных этапов работ приведен в соответствии с приказом об организации учебного процесса в 2023/2024 учебном году.

Таблица 1: Этапы разработки

| №  п/п | Наименование этапа и содержание работ | Сроки исполнения |
| --- | --- | --- |
| 1 | Исследование предметной области, постановка задач | сентябрь 2023 г. |
| 2 | Разработка архитектуры программного обеспечения | октябрь 2023 г. |
| 3 | Написание кода | ноябрь-январь 2023 г. |
| 4 | Тестирование и отладка | февраль-март 2024 г. |
| 5 | Оформление документации | апрель 2023 г. |
| 6 | Показ макета | май 2023 г. |

**8. Порядок контроля и приёмки**

Приём программного изделия в виде приёмочных испытаний осуществляется в ходе «Защиты макетов программ – предварительной защиты ВКРБ» в период с 15 по 24 мая 2024 года в соответствие с разработанной программой и методикой испытаний.

**ПРИЛОЖЕНИЕ В  
ПРОГРАММА И МЕТОДИКА ИСПЫТАНИЙ**

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

им. Н.Э. Баумана

Кафедра «Системы обработки информации и управления»

|  |  |  |  |
| --- | --- | --- | --- |
|  | Утверждаю  Заведующий кафедрой ИУ-5 |  | Согласовано  научный руководитель |
|  | \_\_\_\_\_\_\_\_\_\_\_В.И.Терехов  "\_\_"\_\_\_\_\_\_\_\_\_\_2024 г. |  | \_\_\_\_\_\_\_С.В. Черненький  "\_\_"\_\_\_\_\_\_\_\_\_\_2024 г. |

**Информационная система подачи объявлений для мелкооптовой торговли строительными материалами. Модуль фронтенд.**

Программа и методика испытаний

(вид документа)

писчая бумага

(вид носителя)

8

(количество листов)

ИСПОЛНИТЕЛЬ: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Братухин Александр Сергеевич

"\_\_"\_\_\_\_\_\_\_\_\_\_\_2024 г.

Москва - 2024

**Аннотация**

В данном документе описываются последовательность и методы проведения испытаний при тестировании программного изделия, состав и структура технических и программных средств, необходимых для проведения испытаний, а также приводятся требования к предъявляемой документации, характеристикам программы применительно к условиям эксплуатации и требования к информационной и программной совместимости. Описывается тестовый пример и реакция системы на него.

**Объект испытаний**

Фронтенд модуль информационной системы подачи объявлений для мелкооптовой торговли строительными материалами.

**Цель испытаний**

Цель испытания – проверка функционирования всех указанных в техническом задании функций программы. Сокращенно – веб-сервис подачи объявлений.

**Состав предъявляемой документации**

На испытания программного продукта предъявляются следующие документы:

1) Техническое задание.

2) Программа и методика испытаний.

3) Руководство пользователя.

**Технические требования**

**4.1 Требования к программной документации**

Комплектность программной документации должна удовлетворять разделу данного документа "Состав предъявляемой документации".

**4.2 Требования к техническим характеристикам**

**4.2.1 Требования к составу аппаратного обеспечения**

Веб-сервис подачи объявлений должен выполняться на компьютере, имеющем следующие характеристики:

* Наличие оперативной памяти компьютера в размере 2ГБ.
* Процессор с тактовой частотой 1.1 ГГц и более.
* Внутренняя память компьютера в размере 20ГБ и более.
* Наличие периферийных средств: монитор, мышь, клавиатура.
* Стабильное интернет соединение.

**4.2.2 Требование к составу программного обеспечения**

Веб-сервис подачи объявлений должен выполняться на компьютере, имеющем следующие характеристики:

* Наличие современной операционной системы Windows(не ниже 7 версии), Linux или MacOS.
* Установленный веб-браузер версии 2020 года и позднее.

**Порядок проведения испытаний**

Испытания данного программного продукта будут проводиться в следующем порядке:

1) Запуск веб-сервиса подачи объявлений.

2) Тестирование функционала веб-сервиса подачи объявлений.

**5.1. Требования к составу аппаратного обеспечения**

Требования к составу аппаратного обеспечения учитываются согласно пункту 4.2.1.

**5.2. Требования к составу программного обеспечения**

Требования к составу программного обеспечения учитываются согласно пункту 4.2.2.

**6. Методы испытаний**

Методы испытаний и их последовательность для веб-сервиса подачи объявлений приведены в таблице 1. Есть разграничение на авторизованного и неавторизованного пользователя.

Таблица 1 – Методы испытаний и их последовательность

| **№** | **Пункт**  **ТЗ** | **Выполняемое действие** | **Результат** |
| --- | --- | --- | --- |
| 1 |  | Демонстрация объявлений в рамках предметной области:   * Нажать на логотип “GFP”, находящийся слева сверху; * Просмотреть показанные объявления; | Показ всех объявлений. |
| 2 |  | Поиск объявлений по названию:   * В строку поиска “Поиск объявлений” ввести название интересующего объявления; * Нажать кнопку “Начать поиск”; * Просмотреть показанные объявления; | Показ объявлений после ввода информации в поиск. |
| 3 |  | Поиск объявлений по выставленным параметрам поиска:   * Слева от кнопки “Начать поиск” нажать на иконку с тремя горизонтальными полосками; * Ввести необходимые параметры поиска: выбрать категорию, выставить минимальную и максимальную цену, город * Нажать кнопку “Начать поиск” * Просмотреть показанные объявления | Показ объявлений по выставленным параметрам поиска. |
| 4 |  | Возможность получения подробной информации о выбранном объявлении:   * Получить список всех объявлений (пункты ТЗ 4.1.1 – 4.1.3) * Нажать на название интересующего объявления * Просмотреть информацию о выбранном объявлении | Показ подробной информации об объявлении. |
| 5 |  | Возможность регистрации новых пользователей:   * Сверху справа на сайте нажать на кнопку “Регистрация”; * В открывшейся форме в тесктовые поля ввести необходимую информацию: логин, имя, фамилия, пароль, повторите пароль, город * Внизу формы нажать на кнопку “Зарегистрироваться”; | Регистрация нового пользователя. |
| 6 |  | Возможность авторизации пользователей:   * Сверху справа на сайте нажать на кнопку “Войти”; * В открывшейся форме в тесктовые поля ввести необходимую информацию: логин (можно ввести телефон, email, логин), пароль; * Внизу формы нажать на кнопку “Войти”; | Авторизация пользователя. |
| 7 |  | Далее функционал для авторизованного пользователя.  Возможность создания объявлений пользователями в рамках предметной области:   * Сверху справа на сайте нажать на кнопку “Продать что-нибудь”; * На открывшейся странице выбрать категорию товара, загрузить фото товара, заполнить текстовые поля: название товара, цена, краткое описание, подробное описание; * Снизу на странице нажать на кнопку “Добавить объявление”; | Добавится новое объявление. |
| 8 |  | Возможность редактирования добавленных пользователем объявлений в рамках предметной области:   * На странице сайта сверху нажать на кнопку “Мои объявления”; * На открывшейся странице выбрать интересующее объявление и нажать кнопку “Редактировать”; * На открывшейся странице изменить какую-то информацию (название, категория, цена и так далее); * Нажать на кнопку “Обновить объявление”; | Изменится информация в редактированном объявлении. |
|  |  | Возможность создания заказов:   * На главной странице со всеми объявлениями справа от объявлением нажать на кнопку “Оформить заказ”; | Создастся заказ на объявление. |
|  |  | Возможность просмотра удаления заказов, добавленных пользователем:   * Сверху на сайте нажать на кнопку “Заказы”; * Под нужным заказом нажать кнопку “Удалить заказ”; | Показ информации о своих заказах. Удаление выбранного заказа. |
|  |  | Возможность просмотра заказов, добавленных другими пользователями на объявления авторизованного пользователя:   * Сверху на сайте нажать на кнопку “Заказы”; * На открывшейся странице нажать на вкладку “Заказаны у меня”. | Показ информации о заказах, которые оформили другие пользователи на наши объявления. |
|  |  | Возможность добавления объявлений в избранное:   * На главной странице со всеми объявлениями справа от объявлением нажать на кнопку “В избранное”; | Объявление другого пользователя добавится в избранное. |
|  |  | Возможность просмотра всех объявлений, добавленных в избранное:   * Сверху на сайте нажать на кнопку “Избранное”; | Показ информации о всех объявлениях, добавленных в избранное. |
|  |  | Возможность просмотра информации о других пользователях:   * На главной странице со всеми объявлениями выбрать интересующее объявление и нажать на его название; * На открывшейся странице нажать на имя продавца; | Показ информации о выбранном пользователе. |
|  |  | Возможность просмотра и редактирования информации о себе:   * Слева сверху на сайте нажать на иконку со своими фамилией и именем; * На открывшейся странице в текстовых полях написана информация о себе. Нужные поля можно отредактировать (логин, имя, фамилия, почта, город, телефон); * Нажать на кнопку “Обновить данные”; | Показ информации о себе. Изменение данных о себе. |
|  |  | Возможность смены пароля:   * Слева сверху на сайте нажать на иконку со своими фамилией и именем; * На открывшейся странице нажать на кнопку “Обновить пароль”; * На открывшейся странице ввести свой старый пароль, новый пароль и повторить ввод нового пароля; | Смена пароля. |
|  |  | Возможность выхода из системы:   * Слева сверху на сайте нажать на кнопку “Выйти из профиля”; | Произойдет выход из системы, пользователь окажется на главной странице без прав авторизированного пользователя. |