

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
“ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ”

Факультет *компьютерных наук*
Кафедра *информационных систем*

Веб сервис информационной системы доставки еды.

Курсовой проект

по дисциплине

Технологии программирования

09.03.02 *Информационные системы в телекоммуникациях*

Обучающийся _____ *Н.А. Бузин, 3 курс, д/о*

Обучающийся _____ *Ф. Керимов, 3 курс, д/о*

Обучающийся _____ *М.А. Паронников, 3 курс, д/о*

Руководитель _____ *Тарасов В. С.*

Воронеж 2020

Оглавление

Введение	3
Используемые определения.....	4
1. Постановка задачи	5
2. Анализ предметной области	6
3. Обоснование выбора продуктовых воронок	13
1. IDEF0 диаграмма.....	15
2. Диаграмма прецедентов.....	16
3. Диаграмма классов.....	17
4. Диаграмма объектов	17
5. Диаграмма взаимодействия.....	19
6. Диаграмма состояний	20
7. Диаграмма активностей.....	20
8. Диаграмма развертывания.....	23
9. Обоснование архитектуры проекта	24

Введение

В наше время человек имеет все меньше свободного времени, следовательно, хочет всячески себе упростить те действия, которые требуют для него траты этого самого времени. Именно поэтому на данный момент стали так востребованы сервисы, которые выполняют какие-то элементы за него, и сервис доставки еды не исключение. Любой человек любит сытно и вкусно покушать, но не каждый может позволить сходить в ресторан из-за нехватки свободного времени или проблем с самоизоляцией, наш сервис как раз-таки помогает в этом.

Использование веб-сервиса, предоставляющего возможность онлайн заказа еды, дает следующие преимущества:

- Возможность заказать еду, не выходя из дома.
- Экономия личного времени.
- Отслеживание истории заказов.

Для достижения поставленной цели были определены следующие задачи:

- проектирование веб-сервиса средствами языка UML
- разработка базы данных для хранения информации о пользователях и их статьях
- разработка программной реализации веб-сервиса

Используемые определения

Веб-сервис	идентифицируемая уникальным веб-адресом (URL-адресом) программная система со стандартизированными интерфейсами, а также HTML-документ сайта, отображаемый браузером пользователя
Пользователь	Зашедший на портале человек, пользующийся функционалом веб-сервиса.
Курьер	Авторизованный пользователь, имеющий обязанности приема и доставки заказов.
Менеджер	Авторизованный пользователь, имеющий обязанности добавления, удаления и редактирования курьеров, списка еды, менеджеров

1. Постановка задачи

Необходимо разработать сервис для ресторана грузинской кухни, который позволит осуществлять заказ блюд из него.

Для достижения данных целей сервис должен отвечать следующим требованиям:

1. Сервис должен выполнять все предписанные ему функции, иметь простой и понятный дизайн.

2. Выполнение основных функциональных задач сервиса:

2.1. Возможность совершить авторизацию и регистрацию в системе

2.2. Возможность просмотра меню блюд, добавление их в корзину и оформление заказа.

2.3. Возможность добавления, удаления и редактирования курьеров, списка еды, менеджеров для авторизированных менеджеров.

2.4. Возможность просмотра и приема заказа для авторизованного курьера.

Для выполнения данных требований необходимо выполнить следующие задачи:

1. Проектирование веб-сервиса средствами языка UML

2. Разработка back-end части, включающая в себя:

2.1. Реализация ролей:

- Менеджер

- Курьер

- Пользователь(клиент)

2.2. Реализация функционала ролей

2.3. Подключение внешнего модуля для хранения данных

2.4. Разработка модели данных

2.4. Разработка функционала статических и динамических страниц

2.5. Реализация авторизации/регистрации пользователей

3. Разработка front-end части, включающей в себя:

3.1. Создание макета дизайна

3.2. Реализация макета дизайна

2. Анализ предметной области

2.1. Анализ существующих решений

Анализ существующих решений будем проводить на основе данных сервисов:

https://burgerbox777.ru/	Доставка бургеров Burger Box
https://chaihona.ru/	Доставка еды Чайхона №1
https://dominospizza.ru/	Доставка пиццы Domino's Pizza

1. Доставка бургеров Burger Box

Общее описание:

Данный сервис позволяет осуществлять онлайн заказ бургеров с доставкой на дом в городе Воронеж. На его главной странице, представленной на Рисунок 1.1, находится информация о проводимых акциях, распределение товаров по категориям, фильтр, несколько позиций блюд и информация о доставке



Рисунок 1.1 - Главная страница сайта Burger Box

При добавлении товара в корзину, мы можем оформить заказ, как продемонстрировано на Рисунок 1.2, на этой странице мы можем изменять количество товаров, очистить весь список и оформить заказ.

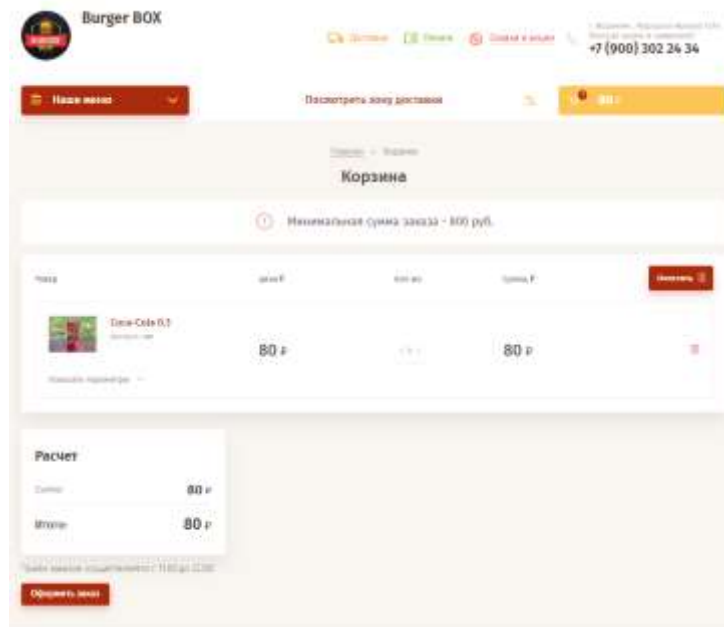


Рисунок 1.2 – Страница оформления заказа

Достоинства:

- Быстрая загрузка страниц
- Понятный интерфейс

Недостатки:

- На главной страницы слишком много информации

2. Сайт Чайхона №1

У ресторана Чайхона №1 есть свой сайт, на котором мы можем увидеть новости от этой организации, осуществить заказ еды, просмотреть вакансии, забронировать столик.

Страница объявлений Рисунок 2.1 – Главная страница сайта Чайхона №1

выглядит простовато, но она хранит всю необходимую информацию, не перегружена лишними данными

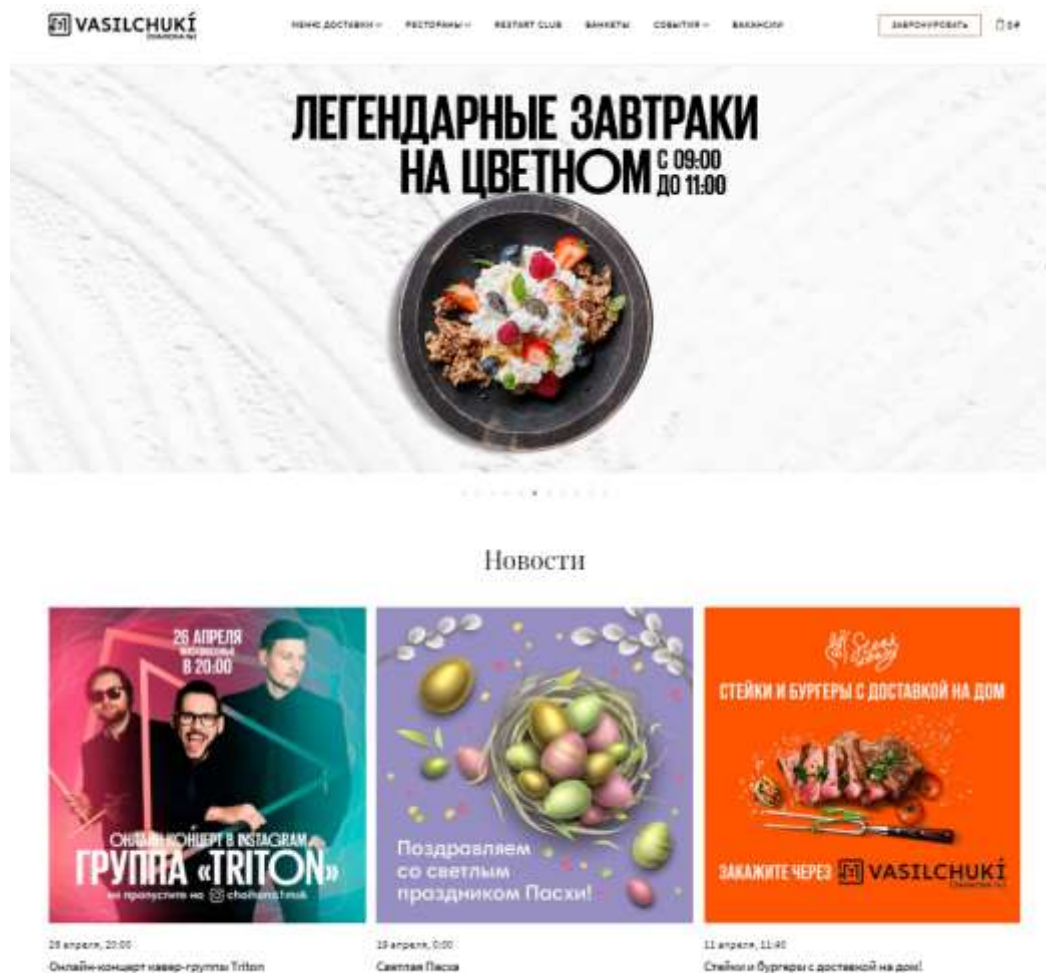


Рисунок 2.1 – Главная страница сайта Чайхона №1

На странице с меню, Рисунок 2.2 - 2 мы можем просмотреть текущий набор блюд, как-либо отфильтровать их и добавить в корзину.

Помощь | Меню доставки | 8328

Pizza

22 позиции

КАТЕГОРИИ

Пицца

Добавки к пицце

Выпечка

Супы и Салаты

Рис

Восточные и Шашлык

Салаты и Закуски

Чупчи

Плов

Паста

Напитки и Травы

Для приготовления

Гармоны

Чипсы

Десерты

Смузи

Салаты

Акции

Доставка за 15 мин

Специальные цены. Печенье

Подарки

Завтрак в Казу



Пицца Квартро Формаджи

330 P

ЗАКАЗАТЬ



Пицца Маргарита

260 P

ЗАКАЗАТЬ



Пицца миссия

620 P

ЗАКАЗАТЬ



Пицца с курицей и грибами

430 P

ЗАКАЗАТЬ



Пицца с пепперони

340 P

ЗАКАЗАТЬ



Котлетный Ветер (добавки к пицце)

110 P

ЗАКАЗАТЬ

Рисунок 2.2 - Страница меню

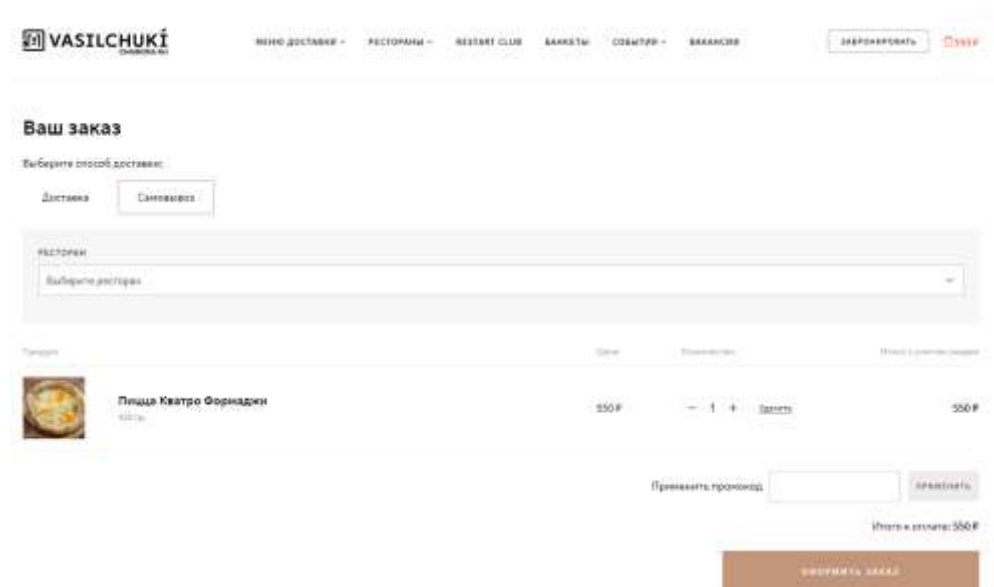


Рисунок 2.3 – Страница оформления заказа

На Рисунок 2.3 – Страница оформления заказа возможно также изменять количество товаров, оформить заказ или удалять позиции.

Достоинства:

- Контрастная и легкая для чтения цветовая гамма
- Быстрая работа сайта
- Возможность применить промокод

Недостатки:

- Пункт с доставкой еды не выделяется

3. Сайт Domino's Pizza

Сервис от международной компании Domino's Pizza, который позволяет осуществлять онлайн заказ пиццы и других блюд.

На рисунке 3.1 представлена главная страница сайта, на которой располагаются блоки с информацией о акциях, текущих доступных пиццах, дополнительная информация.

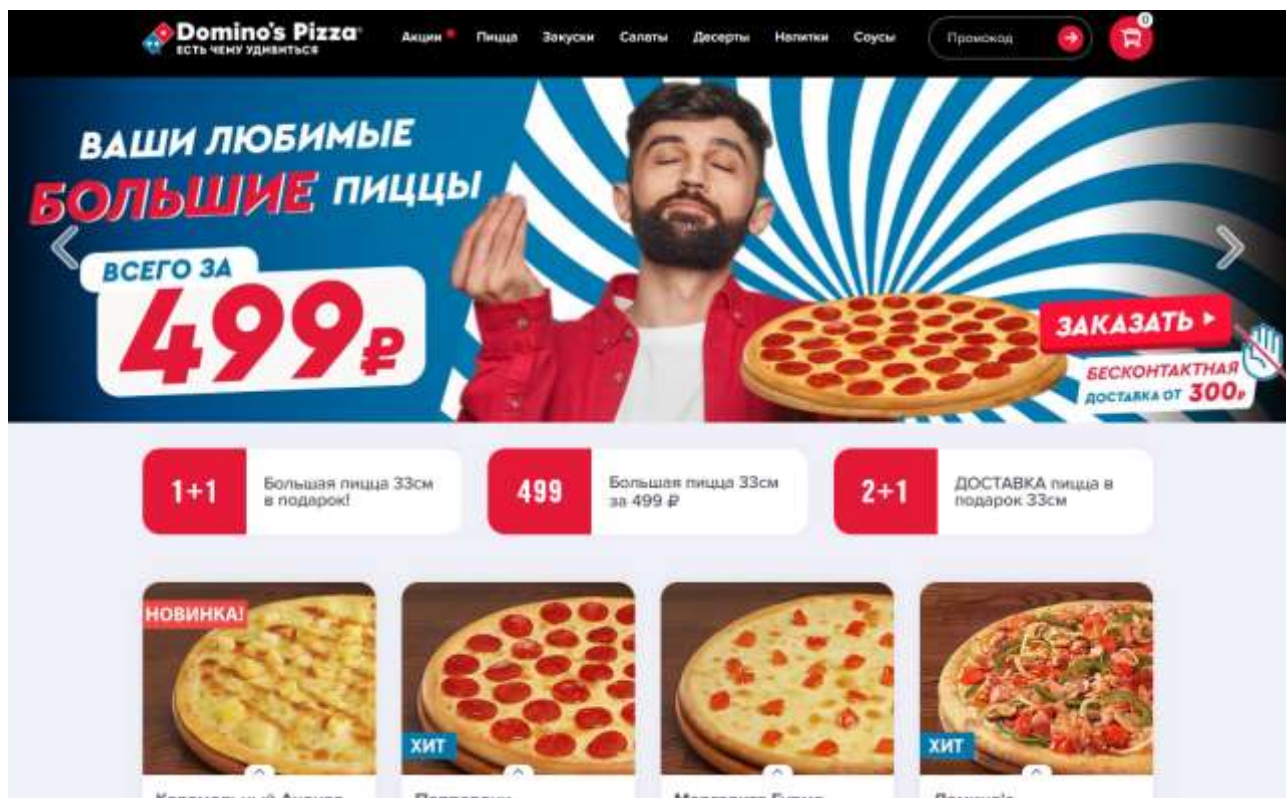


Рисунок 3.1 - Личный кабинет пользователя

На рисунке 3.2 представлена корзина с товарами, где можно оформить заказ, изменить количество товаров, удалить их, активировать промокод.

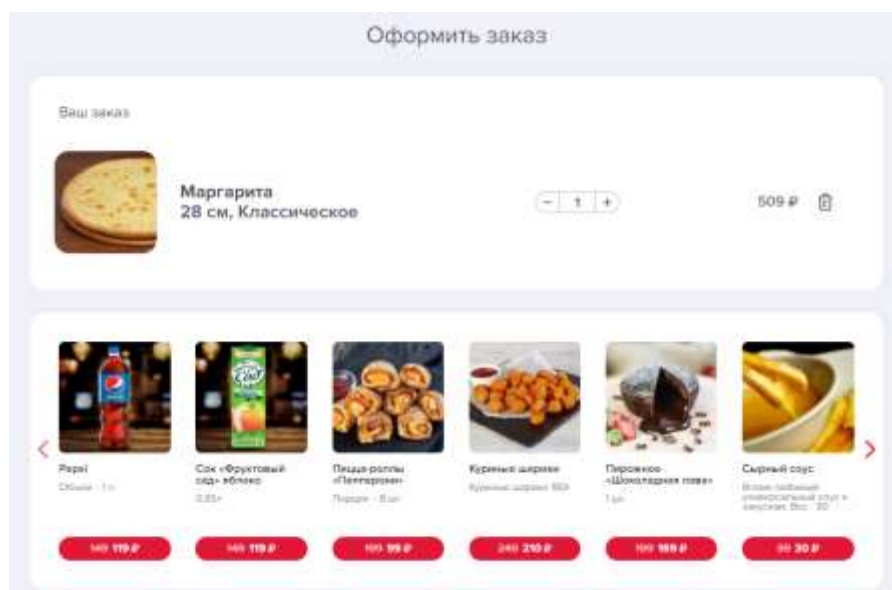


Рисунок 3.2 – Страница корзины товара

Достоинства:

- Хорошая читаемость текста
- Быстрая работа сайта
- Возможность активировать промокод

Недостатки:

- Мало заметные пункты с согласием о рассылки на Email, SMS.

Что должно быть в нашей системе:

- Высокая контрастность текста и фона
- Быстрая работа сайта
- Не перегружать страницу информацией
- Возможность изменения количества товара в корзине

2.2. Анализ потребности

На данный момент существует множество доставок еды, но у этого конкретного ресторана ее нет. Также учитывая на данный момент самоизоляцию, многие люди сидят дома и могут захотеть поесть из нового места.

3. Обоснование выбора продуктовых воронок

С помощью Яндекс.Метрики будет собираться информация о трех сценариях:

1. Пользователь выбрал товар и оформил заказ.
2. Пользователь выбрал заказ и отменил заказ.
3. Пользователь пользуется фильтром.



Рисунок 4 – 1 воронка



Рисунок 5 – 1 воронка

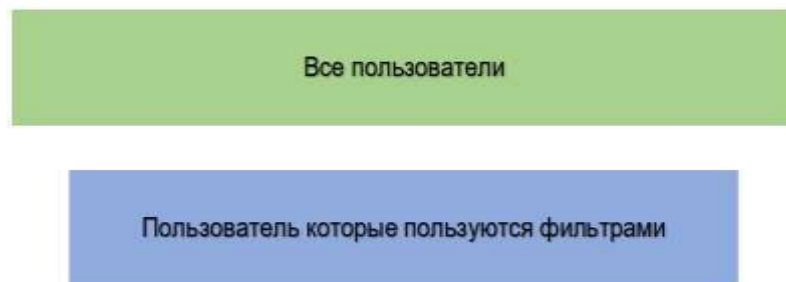


Рисунок 6 – 1 воронка

1. IDEF0 диаграмма

Основной процесс рассмотрим на примере IDEF0 диаграммы, на которой представлен общее видение работы Сервиса.

На вход у нас поступает Пользователь, БД без изменений.

Работу нашего сервиса будут регулировать Пользовательское соглашение и Разграничение прав, которое определяет возможности Пользователя в системе и регулируется Менеджером.

Работа сервиса обеспечивается с помощью Менеджера, Курьера.

Таким образом, на вход в систему поступает БД без изменений и Пользователь, который после авторизации может стать Менеджером, Курьером, или остаться Пользователем и БД без изменений. Если был осуществлен заказ, то следует изменение в БД и заказ закрывается.



Рисунок 7 - Контекстная диаграмма

2. Диаграмма прецедентов

На диаграмме прецедентов описаны роли, которые есть в нашем сервисе, и

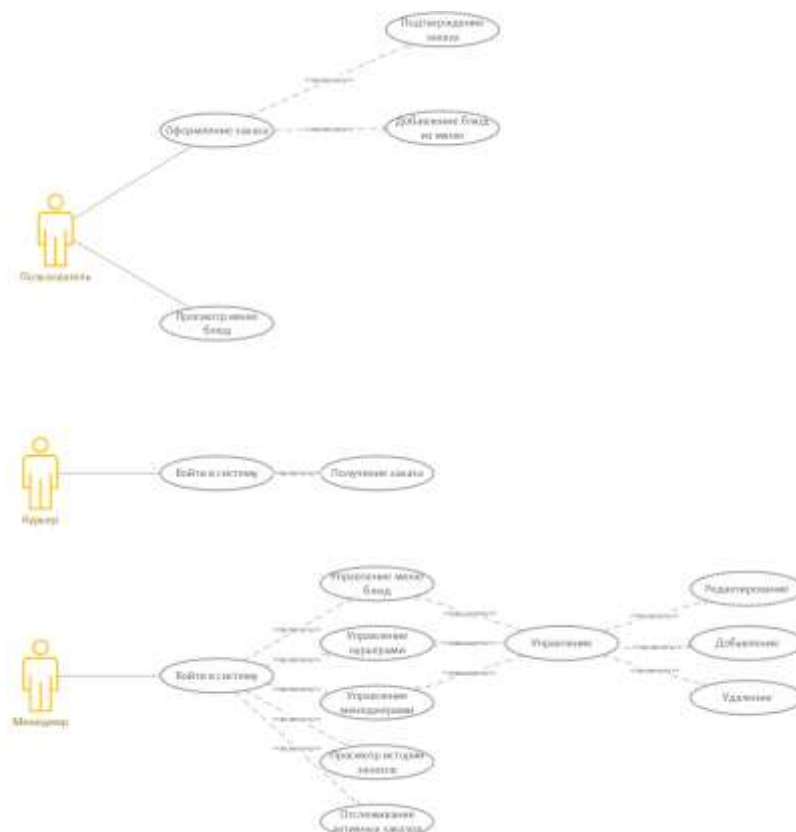


Рисунок 9 – Диаграмма прецедентов

возможности этих ролей.

Так пользователь может, просматривать блюда, оформлять заказ, добавлять блюда из меню

Курьер может принять заказ.

Менеджер имеет право на изменение всей информации на сайте, такой как информация о менеджерах, курьерах, блюдах, изменения их количества.

3. Диаграмма классов

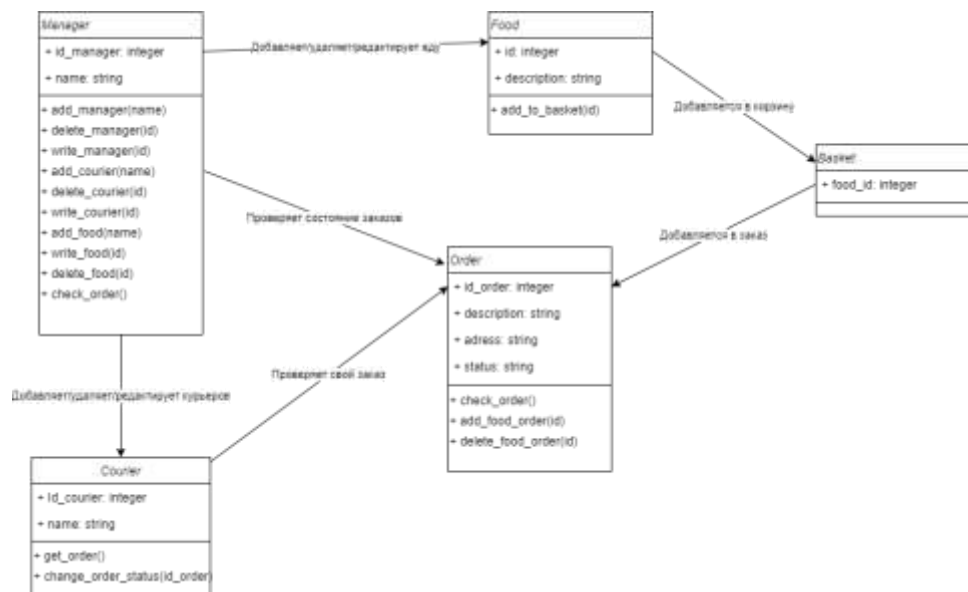


Рисунок 12 - Диаграмма классов

4. Диаграмма объектов

На рисунке №10 можно увидеть примерный вид приложения в определенный момент.

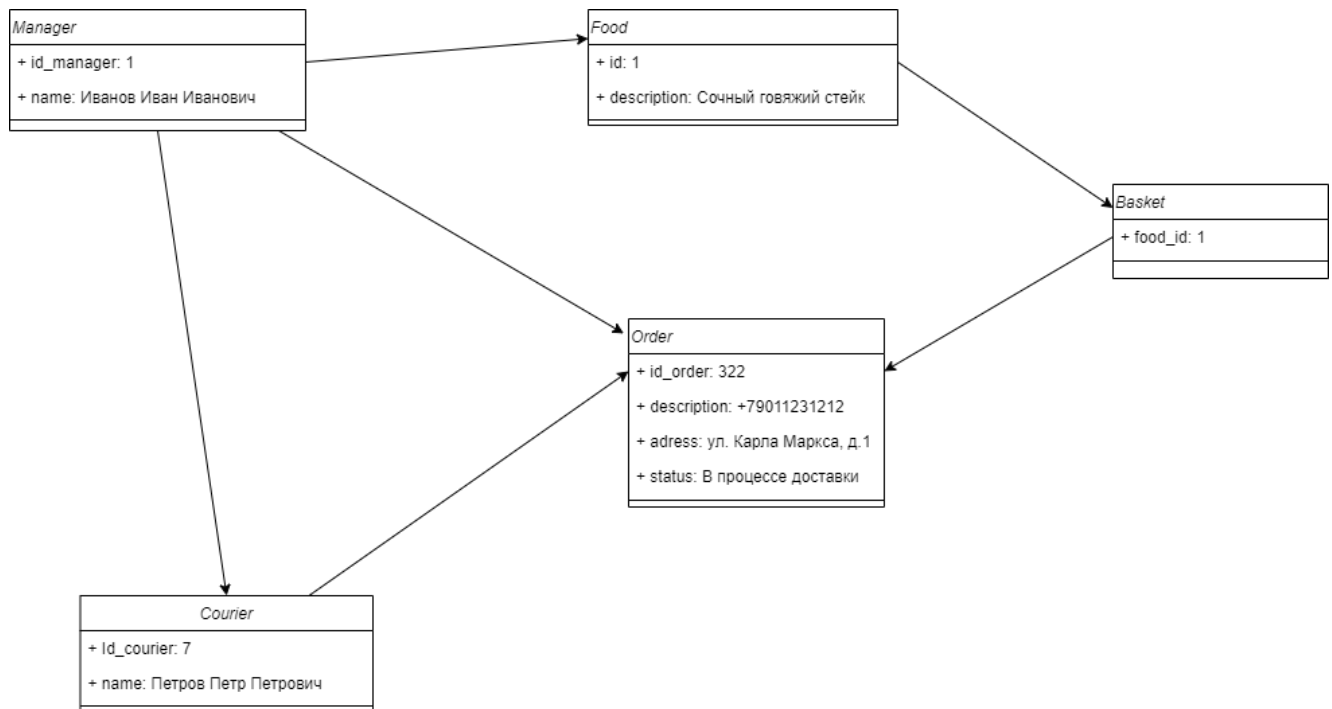


Рисунок 15 - Диаграмма объектов

5. Диаграмма взаимодействия

Необходимо теперь определить, какие способы взаимодействия с системой есть в распоряжении пользователей. Она позволяет видеть все взаимодействия запросов в системе и дополняет Рисунок 18 - Диаграмма взаимодействия.

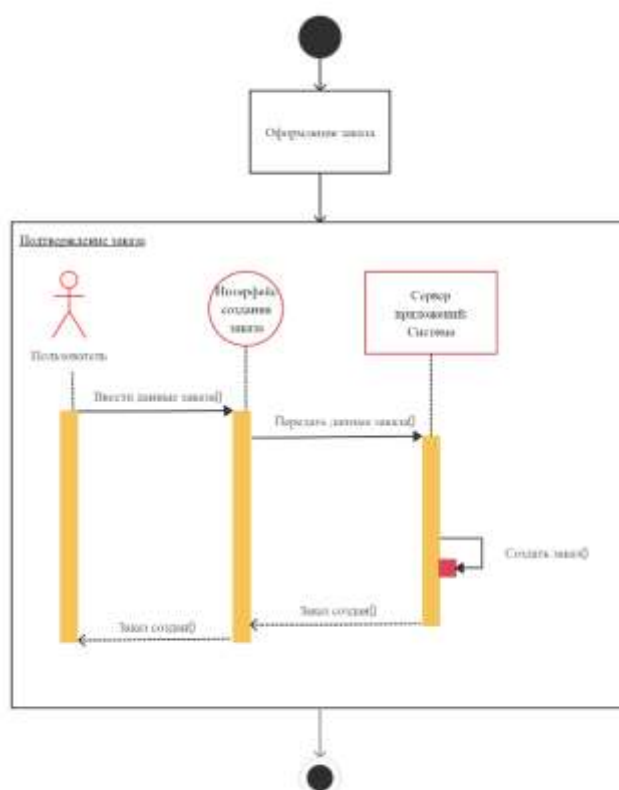


Рисунок 18 - Диаграмма взаимодействия

6. Диаграмма состояний

Диаграмма состояний показывает, как объект переходит из одного состояния в другое. На **Ошибка! Источник ссылки не найден.** показано изменение состояний заказа. в то время как на Рисунок 19 - Диаграмма состояний.



Рисунок 19 - Диаграмма состояний.

Рисунок 20 - Диаграмма состояний. ФормаРисунок 21 - Диаграмма состояний. Статьяизменение состояний Статьи.

7. Диаграмма активностей

На данной диаграмме показывается процесс оформления заказ

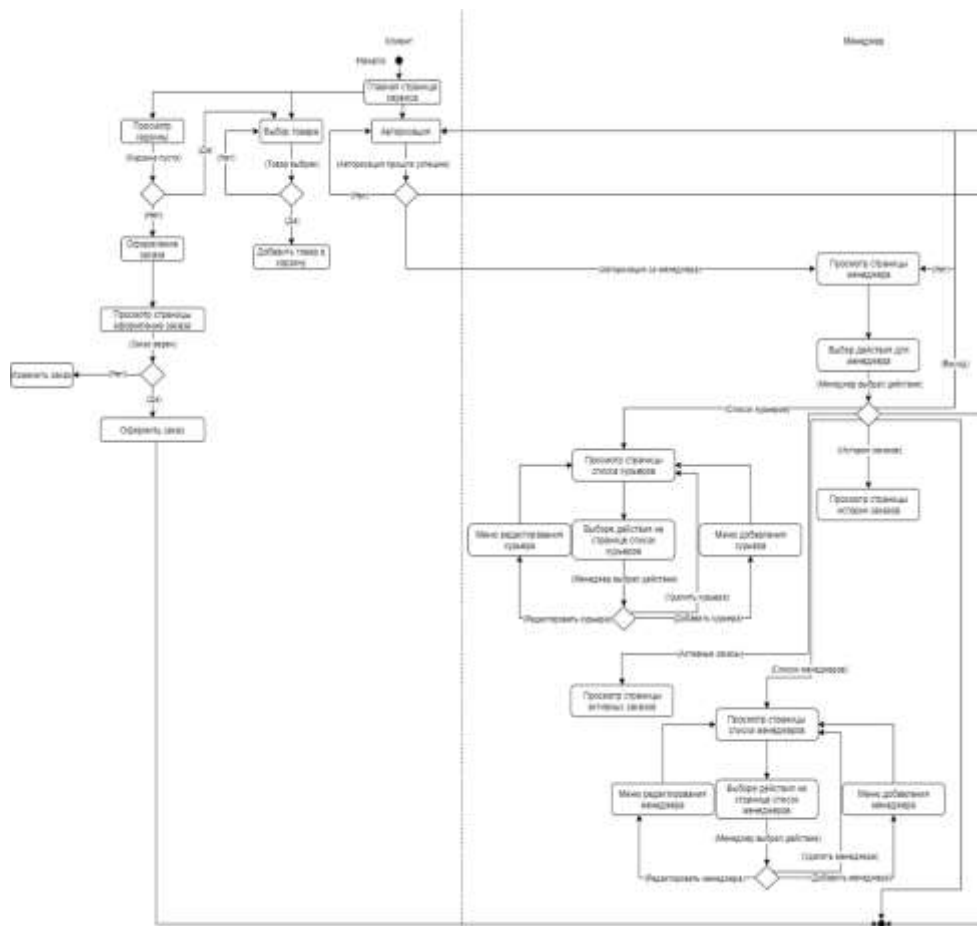


Рисунок 22 - Диаграмма активностей. Действия клиента и менеджера

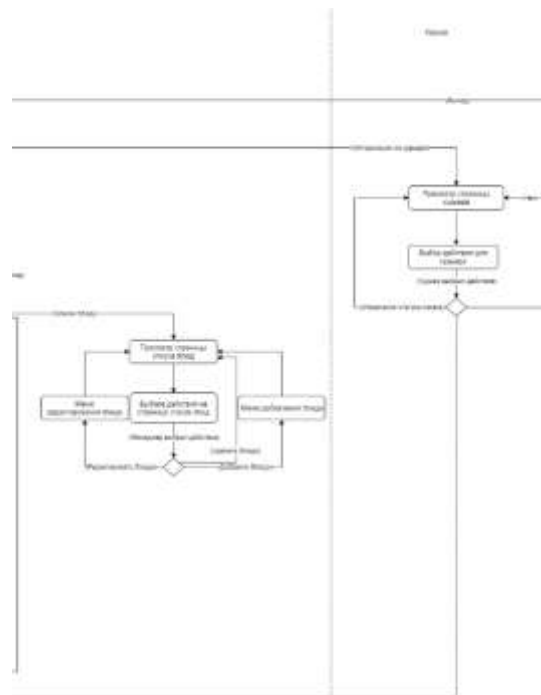


Рисунок 23 - Диаграмма активности. Действия менеджера и курьера

8. Диаграмма развертывания

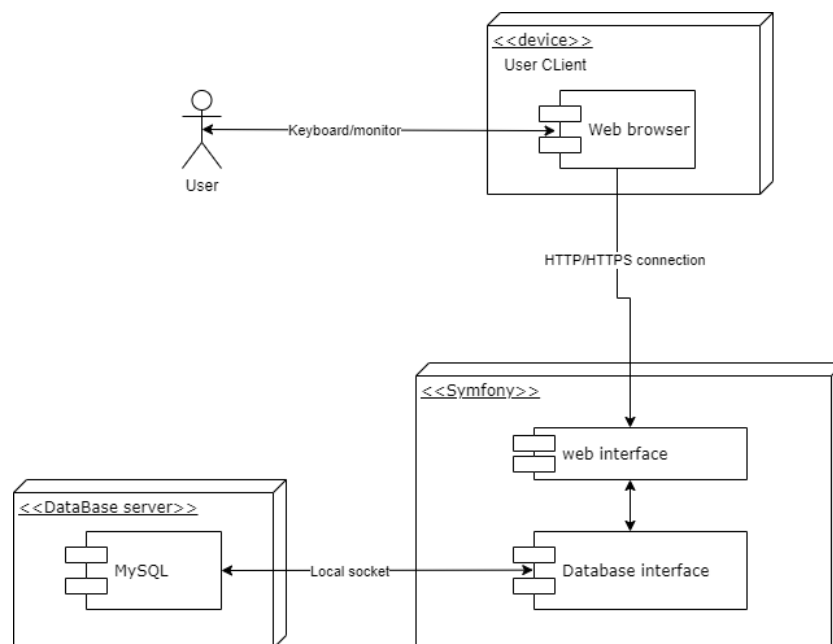


Рисунок 24 - Диаграмма развертывания

9. Обоснование архитектуры проекта

Архитектура проекта: при проектировании приложения был использован паттерн проектирования MVC

Паттерн MVC позволяет разделять приложение на части, такие как Model, View, Controller. Это позволяет модифицировать отдельные компоненты приложения не затрагивая другие части.

Схематично паттерн можно определить следующим образом:

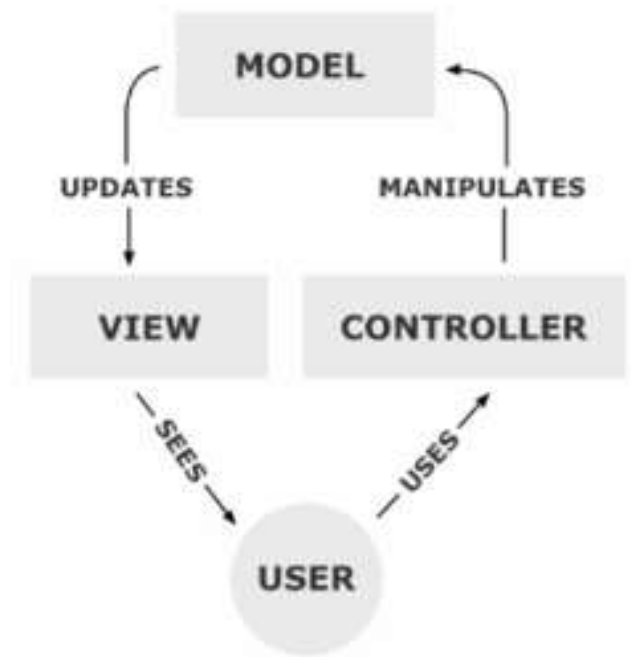


Рисунок 25 - Схема паттерна MVC

Основные элементы паттерна:

- Модель предоставляет данные и реагирует на команды контроллера, изменяя своё состояние.
- Представление отвечает за отображение данных модели пользователю, реагируя на изменения модели.
- Контроллер интерпретирует действия пользователя, оповещая модель о необходимости изменений.

10. Реализация приложения

10.1. Серверная часть

В серверной части приложения можно выделить следующие элементы. Начнем с репозиториев.

На рисунке 26 изображен пример части репозитория, ответственного за работу с таблицей Courier в базе данных.

```
/**
 * @method Courier|null find($id, $lockMode = null, $lockVersion = null)
 * @method Courier|null findOneBy(array $criteria, array $orderBy = null)
 * @method Courier[]    findAll()
 * @method Courier[]    findBy(array $criteria, array $orderBy = null, $limit = null, $offset = null)
 */
class CourierRepository extends ServiceEntityRepository implements PasswordUpgraderInterface
{
    public function __construct(ManagerRegistry $registry)
    {
        parent::__construct($registry, \emptyClass::class);
    }
}
```

Рисунок 26 – CourierRepository

С помощью репозитория мы можем выбрать из таблицы все элементы, отдельно по id, по критерию или несколько по критерию.

Для создания базы данных использовалась Doctrine. С его помощью создавались сущности для репозитория, задавались для них поля, типы и размерность. Пример такой сущности изображен на рисунке 27.

```
/**
 * @ORM\Entity(repositoryClass=CourierRepository::class)
 */
class Courier implements UserInterface
{
    /**
     * @ORM\Id()
     * @ORM\GeneratedValue()
     * @ORM\Column(type="integer")
     */
    private $id;

    /**
     * @ORM\Column(type="string", length=180, unique=true)
     */
    private $login;
```

Рисунок 27 – Entity Courier

Автоматически были созданы методы для взятия информации getLogin(), и для ее задачи setLogin(). Подобные методы были созданы для каждого поля в каждой сущности, за исключением id, данное поле является автоинкрементным и его можно только взять, но не задать.

Контроллер это php-функция, созданная нами, которая в нашем случае возвращает редиректы на другие url или html.twig страницы. В нашем проекте примером такого контроллера является FoodListController, изображенный на рисунке 28.

```
use Symfony\Component\BrowserKit\Request;
use Symfony\Component\HttpFoundation\Response;
use Symfony\Component\Routing\Annotation\Route;
use Symfony\Bundle\FrameworkBundle\Controller\AbstractController;

class FoodListController extends AbstractController
{
    /**
     * Show food list
     * @Route("/foodList")
     */
    public function number()
    {
        $foods = $this->getDoctrine()->getRepository('App:Food')->findAll();

        return $this->render('view: foodlist.html.twig', [
            'foods' => $foods,
        ]);
    }
}
```

Рисунок 28 – FoodListController

Для работы контроллера необходимо подключить Route в верху файла. Для создания route используются аннотации, также там прописывается информация для swagger.

10.2. Интерфейс

На рисунке 29 изображена главная страница сайта, на ней представлены блюда, можно отсортировать нужные блюда, перейти на страницу входа, добавить блюдо в корзину и перейти в корзину.

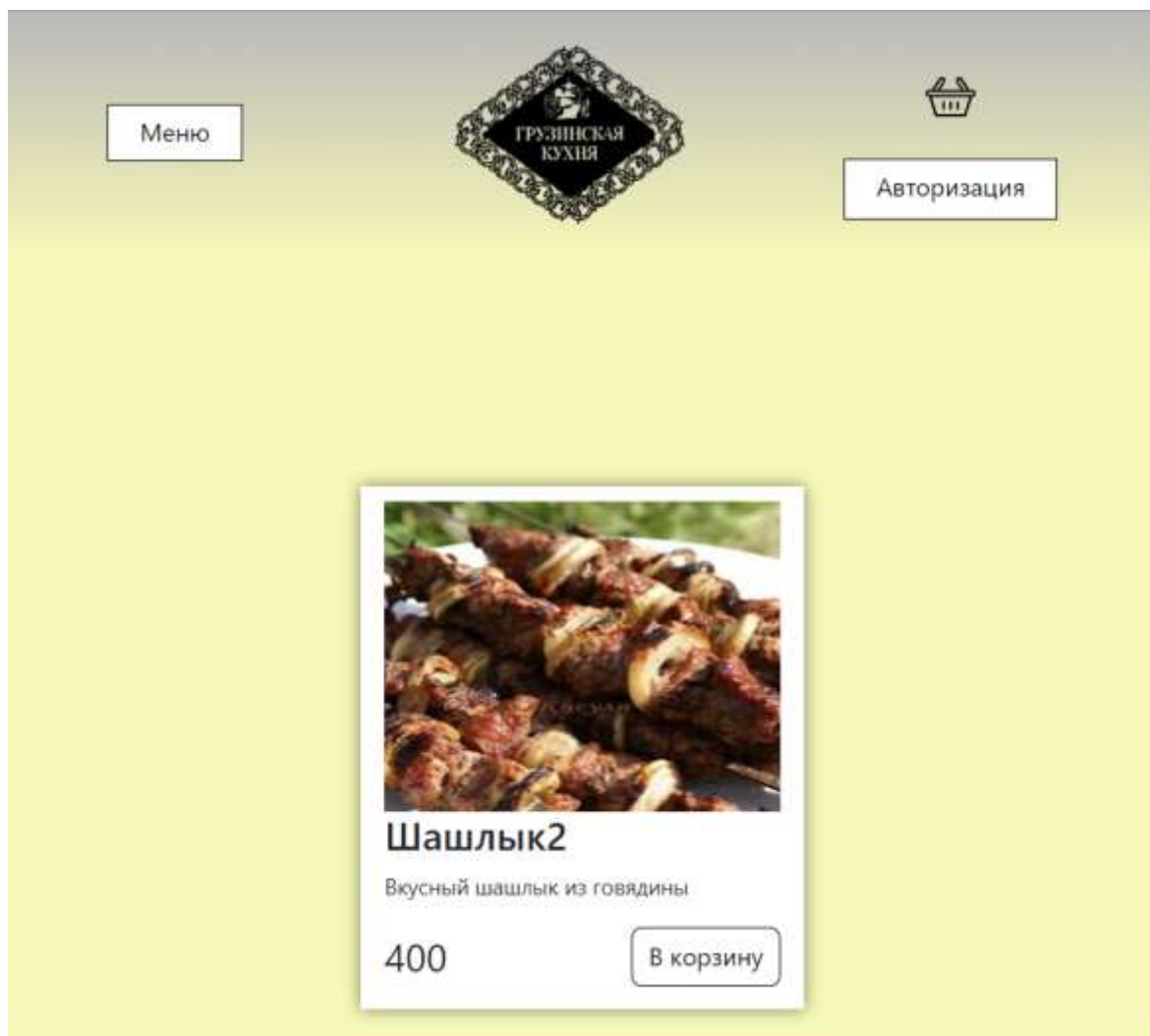


Рисунок 29 – Главная страница

При успешном входе с логином и паролем от аккаунта менеджера, открывается страница с личным кабинетом менеджера, изображенная на рисунке 30. На данной странице можно перейти на страницу со всеми заказами, только с активными заказами, просмотреть курьеров, менеджеров и список блюд. При нажатии на кнопку выход пользователь выйдет на страницу входа, изображенную на рисунке 31.

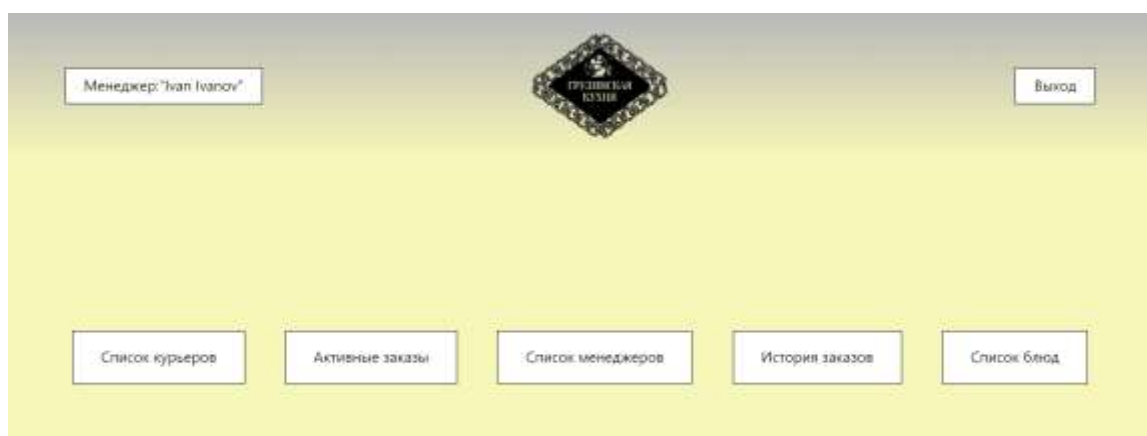


Рисунок 30 – Личный кабинет менеджера



Рисунок 31 – Страница входа

При переходе со страницы менеджера в список курьеров, представленный на рисунке 32, можно увидеть зарегистрированных курьеров в системе, редактировать информацию о них, добавлять новых или удалять старых.



Рисунок 32 – Список курьеров

При переходе со страницы менеджера на страницу с активными заказами, изображенной на рисунке 33, перед вами будет список активных заказов и возможность назначить курьеров на эти заказы.



Выход

№	Доступные заказы	Статус	Курьер
4	Блюда:Суп Харчо Сумма: 700 ФИО:Иван Иванов Номер: 89201231222	ул. Большая д.2	New Георгий Гирьев Иванович Назначить на заказ
5	Блюда:Суп Харчо Сумма: 700 ФИО:Иван Иванов Номер: 89201231222	ул. Большая д.1	New Георгий Гирьев Иванович Назначить на заказ
6	Блюда:Суп Харчо Сумма: 700 ФИО:Иван Иванов Номер: 89201231222	ул. Большая д.1	New Георгий Гирьев Иванович Назначить на заказ
7	Блюда:Суп Харчо Сумма: 350 ФИО:Иван Иванов Номер: 89201231222	ул. Большая д.1	New Георгий Гирьев Иванович Назначить на заказ

Активация Windows

Рисунок 33 – Страница с активными заказами

При переходе со страницы менеджера в список менеджеров, представленный на рисунке 34, можно увидеть зарегистрированных менеджеров в системе, редактировать информацию о них, добавлять новых или удалять старых.



Выход

№	Логин	Пароль	ФИО	
1	example@mail.com	qwerty	Ivan Ivanov	Редактировать менеджера Удалить

Добавить менеджера

Рисунок 34 – Список менеджеров

При переходе со страницы менеджера в историю заказов, представленный на рисунке 35, можно увидеть все заказы, сделанные в этой системе.

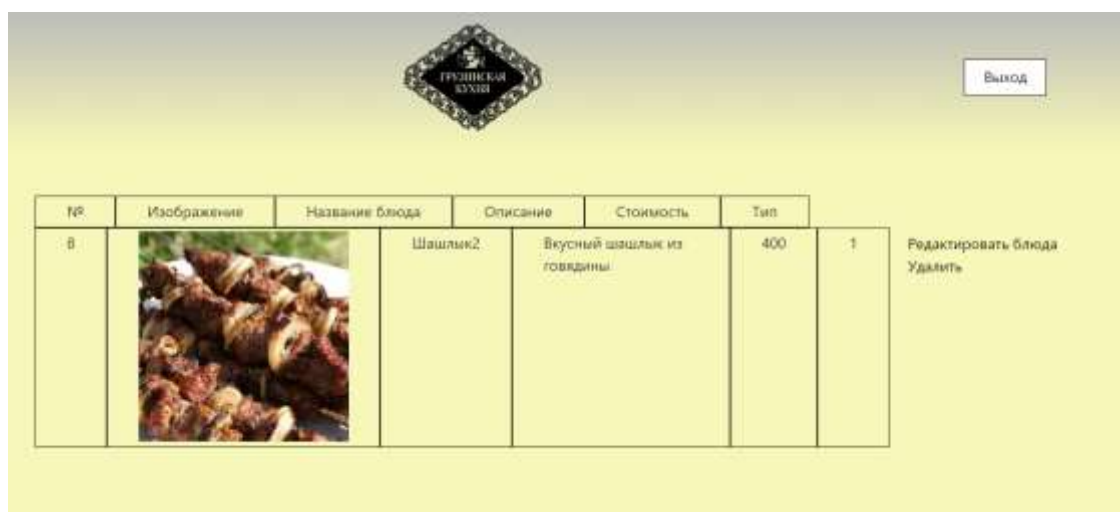



Выход

№	Описание	Адрес	Статус
3	Блюда:Шашлык Сумма: 800 ФИО:Иван Иванов Номер: 89201231222	ул. Большая д.1	Finish

Рисунок 35 – История заказов

При переходе со страницы менеджера на страницу со списком блюд, изображенной на рисунке 36, можно увидеть все блюда в системе, редактировать информацию о них, добавлять новые или удалять старые.



№	Изображение	Название блюда	Описание	Стоимость	Тип
8		Шашлык2	Вкусный шашлык из говядины	400	1

Редактировать блюда
Удалить

Рисунок 36 – Список блюд

При успешном входе с логином и паролем от аккаунта курьера, открывается страница с личным кабинетом курьера, изображенная на рисунке 37. На данной странице можно увидеть заказы, которые обязан доставить курьер, и возможность отметить заказ как выполненный



№	Описание	Адрес	Статус
3	Блюда: Шашлык Сумма: 800 ФИО: Иван Иванов Номер: 89201231222	ул. Большая д.1	Finish

Готово

Рисунок 37 – Личный кабинет курьера

При переходе в корзину, изображенной на рисунке 39, можно увидеть блюда, добавленные в корзину, их количество. Для оформления заказа необходимо ввести ФИО, номер телефона и адрес доставки.



Ваш заказ:

№	Название	Цена	Удалить
8.	Шашлык2	400	Удалить

ФИО
 Номер телефона
 Адрес доставки

☐ Дано согласие на обработку персональных данных

[Подтвердить заказ](#)
[Отменить заказ](#)

Рисунок 39 – Корзина

10.3. Яндекс.Метрика

Посещаемость сайта на рисунке 40



Рисунок 40 – Посещаемость

11. Заключение

По итогу выполненной работы можно сказать, что мы достигли выполнения задач, указанных в техническом задании, овладели новыми навыками. Наш сервис является простым для новых пользователей, он позволяет заказывать еду из ресторана, не выходя на улицу. В нашем продукте упрощен процесс заказа блюд, обработки заказа со стороны менеджера и курьера.