

МИНИСТЕРСТВО ОБРАЗОВАНИЯ НИЖЕГОРОДСКОЙ ОБЛАСТИ  
Государственное бюджетное профессиональное образовательное учреждение  
НИЖЕГОРОДСКИЙ РАДИОТЕХНИЧЕСКИЙ КОЛЛЕДЖ

Специальность 09.02.07 Информационные системы и программирование

**КУРСОВАЯ РАБОТА**

МДК 01.01 Разработка программных модулей

Тема: «Разработка информационной системы для магазина мясной  
продукции»

Выполнил студент  
группы 4ИСИП-18-2  
Неугомонов В. О.

Проверил преподаватель

Попцов А. В.

Проект защищён с оценкой

Дата защиты \_\_\_\_\_

Подпись \_\_\_\_\_

## **ЗАДАНИЕ**

### **на курсовую работу**

Специальность Информационные системы и программирование

МДК 01.01 Разработка программных модулей

Студенту Неугомонову Валерию Олеговичу

Группа 4ИСИП-18-2

**Тема курсовой работы «Разработка информационной системы для магазина мясной продукции»**

Дата выдачи задания «24» сентября 2021 г.

Срок сдачи работы «8» декабря 2021 г.

#### **Перечень вопросов, подлежащих разработке:**

- Введение (название выбранной темы, обзор раскрываемых вопросов).
- Разработка технического задания.
- Проектирование информационной системы.
- Разработка информационной системы.
- Тестирование информационной системы.
- Руководство пользователя.
- Заключение.
- Список использованной литературы.

#### **Перечень диаграмм проектирования:**

1. Диаграмма прецедентов. Сценарий варианта использования.
2. Диаграмма последовательностей.
3. Диаграмма компонентов.
4. Информационно-логическая модель базы данных.

Задание выдал преподаватель \_\_\_\_\_

/А. В. Попцов/

## Оглавление

<b>Введение</b>	<b>5</b>
<b>1 Теоретические основы разрабатываемой темы</b>	<b>7</b>
1.1 Анализ проектируемой системы	7
1.2 Обоснование выбора средств разработки информационной системы	7
<b>2 Разработка технического задания</b>	<b>10</b>
2.1 Назначение и цели создания системы	10
2.1.1 Назначение системы	10
2.1.2 Цели создания системы	10
2.2 Характеристика объектов автоматизации	10
2.3 Требования к структуре системы	10
2.3.1 Требования к структуре и функционированию системы	10
2.3.2 Требования к способам и средствам связи для информационного обмена между компонентами системы	11
2.3.3 Требования к режимам функционирования системы	11
2.3.4 Требования по диагностированию системы	12
2.3.5 Требования к надежности	12
2.3.6 Требования к защите информации	12
2.4 Требования к функциям, выполняемым системой	13
2.5 Требования к видам обеспечения	13
2.5.1 Требования к программному обеспечению	13
2.5.2 Требования к техническому обеспечению	13
<b>3. Проектирование информационной системы</b>	<b>13</b>
3.1. Диаграмма прецедентов (вариантов использования)	14
3.1.1 Прецедент “Вход в систему”	16
3.1.2 Прецедент “Регистрация работника-подчинённого”	16
3.1.3 Прецедент “Обработка, доставка заказов”	16
3.1.4 Прецедент “Регистрация заказчика”	16
3.1.5 Прецедент “Составление своих заказов”	16
3.1.6 Прецедент “Просмотр своих заказов”	17
3.1.7 Прецедент “Просмотр меню”	17
3.1.8 Прецедент “Оплата своего заказа”	17

3.1.9 Прецедент “Изменение своего адреса”	17
3.1.10 Прецедент “Изменение своего пароля”	17
3.2 Диаграмма последовательности	18
3.3 Диаграмма компонентов	20
3.4 Информационно-логическая модель базы данных	21
3.4.1 Структура БД	21
<b>4 Разработка информационной системы</b>	<b>23</b>
4.1 Описание структуры системы	23
4.2 Разработка графического пользовательского интерфейса и переключения между формами и панелями	23
4.3 Разработка структуры системы графического пользовательского интерфейса, логики приложения	26
<b>5. Тестирование информационной системы</b>	<b>27</b>
<b>6 Руководство пользователя</b>	<b>30</b>
6.1 Страница приветствия и регистрации клиента	30
6.2 Страница входа клиента	31
6.3 Основная страница личного кабинета клиента	32
6.4 Страница оплаты заказа	33
6.5 Страница подтверждения успешного заказа	33
6.6 Страница авторизации сотрудника	34
6.7 Страница подтверждения доставки заказов	34
<b>Заключение</b>	<b>35</b>
<b>Список использованных источников</b>	<b>36</b>

## Введение

Информационная система - совокупность средств, методов и организационных ресурсов (человеческие, технические, финансовые и т. д.), используемых для хранения, обработки и выдачи информации в интересах достижения поставленной цели.

В настоящее время мир тесно связан с информационными технологиями и информационными системами. На каждом предприятии, в школе, компании есть свой сайт и личный кабинет.

Несмотря на то, что информационные системы имеют большую популярность, не все компании грузоперевозок перешли с бумажных носителей на электронный вид: заполнение заявки от руки, информация об автомобилях в тетрадях. Отсутствие личного кабинета имеет минусы:

- Менеджеры принимают ежедневно очень много звонков;
- Стопки бумаги с заявками занимают много места;
- Потеря документов без возможности восстановления.

Вышеперечисленные проблемы и недостатки бумажных носителей решают электронные носители:

- Занимают мало места;
- Не нужен особый способ хранения;
- Информацию легко скопировать и передать.

Но мало того, чтобы внедрить в работу компании электронные носители, нужно наладить информационную работу, которая будет взаимодействовать с данными.

В настоящее время компьютерные технологии нашли широкое применение в различных областях. Огромное значение они имеют и в сфере интернет-магазинов. Работа современной системы интернет-магазинов не может быть возможна без использования компьютеров. Сейчас они используются уже не только как вычислительные машины, но и как средства взаимодействия с магазинами.

Информационной системой (ИС) называется комплекс, включающий вычислительное и коммуникационное оборудование, программное обеспечение, лингвистические средства, информационные ресурсы, а также персонал, обеспечивающий поддержку динамической информационной модели предметной области для удовлетворения информационных потребностей пользователей.

В ИС часть функций управления и обработки данных выполняется компьютерами, а часть человеком.

Современный мир информационных технологий трудно представить себе без возможности обработки больших объёмов информации. Такие объёмы информации удобно обрабатывать с помощью баз данных. Практически все системы в той или иной степени связаны с долговременным хранением и обработкой информации. Фактически, информация становится фактором, определяющим эффективность любой сферы деятельности. Увеличились информационные потоки и повысились требования к скорости обработки данных. Большинство операций не может быть выполнено вручную. Любые административные решения требуют более чёткой и точной оценки текущей ситуации и возможных перспектив её изменения.

В данной курсовой работе необходимо разработать модуль информационной системы для личного кабинета пользователя и работника в приложении интернет-магазина «МЯСА», который позволит выполнять управление и производить контроль всех заказов клиента. Он должен обеспечивать просмотр, обработку и выборку заказов по идентификатору клиента.

Необходимо разработать настольное приложение для информационной системы, которая будет являться личным кабинетом.

Внедрение данного приложения позволит:

- Более легкое оформление заказа клиента.
- Отслеживание доставки заказа клиента
- Управление статусом доставки заказа клиента

# 1 Теоретические основы разрабатываемой темы

## 1.1 Анализ проектируемой системы

«МЯСА» — российский интернет-магазин и сеть ресторанов мясной продукции. «МЯСА» действует в 83 субъектах Российской Федерации, в Таджикистане, а также в частично признанных государствах Абхазии и Южной Осетии. По данным компании, количество пользователей компании на 01 октября 2015 года составляло 73,8 млн. человек.

Среди клиентов и партнеров «МЯСА» — частные лица, малые, средние и крупные предприятия, транснациональные корпорации.

ОАО «МЯСА Ритейл» - дочерняя Компания ОАО «МЯСА», развивает розничный бизнес Компании «МЯСА». В настоящее время розничная сеть «МЯСА» насчитывает около 1950 мясных магазинов-ресторанов в 500 городах России. В Компании работает более 13 тысяч человек.

В данной курсовой работе будет разработан модуль автоматизированной информационной системы для личного кабинета компании ОАО «МЯСА». Он позволит клиентам компании оформлять заказы на мясную продукцию, не обращаясь при этом в поддержку компании, сотрудникам - управлять статусами заказов клиентов в автоматизированной информационной базе.

## 1.2 Обоснование выбора средств разработки информационной системы

Java - межплатформенный язык

В языке Java используется технология объектно-ориентированного программирования, которая позволяет сократить общее время разработки и писать повторно используемый код. Java-приложения являются независимыми от платформы. Это достигается путем совмещения в языке свойств компилятора и интерпретатора. Платформенная независимость байт-кода обеспечивается наличием виртуальных java-машин для всех основных платформ. В комплект поставки Java входят стандартные классы, которые обладают достаточной функциональностью для быстрой разработки приложений, Развитые средства безопасности позволяют использовать Java

для разработки приложений, работающих в Интернете. Единственным недостатком Java является медленная скорость работы.

Java—это язык программирования общего назначения, который следует парадигме объектно-ориентированного программирования и подходу «Написать один раз и использовать везде» . Java используется для настольных, сетевых, мобильных и корпоративных приложений.

Java—это не только язык программирования, но и экосистема инструментов, охватывающая почти все, что может понадобиться при программировании на Java.

Java – это основной язык разработки для Android. Он используется в веб-приложениях, правительственных веб-сайтах и технологиях обработки больших данных. Java подходит и для научных проектов, особенно в области обработки естественного языка. Несмотря на растущую популярность Go и Python, Java остается на вершине списка уже более десятилетия.

## MySQL

MySQL — свободная реляционная система управления базами данных. Разработку и поддержку MySQL осуществляет корпорация Oracle, получившая права на торговую марку вместе с поглощённой Sun Microsystems, которая ранее приобрела шведскую компанию MySQL AB. Продукт распространяется как под GNU General Public License, так и под собственной коммерческой лицензией. Помимо этого, разработчики создают функциональность по заказу лицензионных пользователей. Именно благодаря такому заказу почти в самых ранних версиях появился механизм репликации.

MySQL является решением для малых и средних приложений. Входит в состав серверов WAMP, AppServ, LAMP и в портативные сборки серверов Денвер, XAMPP, VertrigoServ. Обычно MySQL используется в качестве сервера, к которому обращаются локальные или удалённые клиенты, однако в дистрибутив входит библиотека внутреннего сервера, позволяющая включать MySQL в автономные программы.

Гибкость СУБД MySQL обеспечивается поддержкой большого количества типов таблиц: пользователи могут выбрать как таблицы типа MyISAM,



поддерживающие полнотекстовый поиск, так и таблицы InnoDB, поддерживающие транзакции на уровне отдельных записей. Более того, СУБД MySQL поставляется со специальным типом таблиц EXAMPLE, демонстрирующим принципы создания новых типов таблиц. Благодаря открытой архитектуре и GPL-лицензированию, в СУБД MySQL постоянно появляются новые типы таблиц.

## NetBeans

NetBeans является интегрированной средой разработки (IDE) для Java . NetBeans позволяет приложениям быть разработаны из набора модульных программных компонентов , называемых модулями. NetBeans работает на ОС Windows , MacOS , Linux и Solaris. Помимо разработки Java, он имеет расширения для других языков, таких как PHP, C, C ++, HTML5 и JavaScript. Приложения, основанные на NetBeans, включая IDE NetBeans, может быть расширены сторонними разработчиками.

## 2 Разработка технического задания

### 2.1 Назначение и цели создания системы

#### 2.1.1 Назначение системы

Личный кабинет клиента создается с целью обеспечения возможности самостоятельного заказа товаров клиентом.

Личный кабинет работника создается с целью обеспечения возможности подтверждения доставки заказов работником.

Модуль информационной системы разработан для активных клиентов компании «МЯСА» и предназначен для полного контроля за заказами.

#### 2.1.2 Цели создания системы

- Заказ продуктов со стороны клиента
- Обработка заказов клиентов со стороны работника

### 2.2 Характеристика объектов автоматизации

Автоматизируется процесс записи и хранения данных о клиентах компании «МЯСА» и полной детализации их заказов. Информация о клиентах, их заказах, адресах хранится в соответствующих таблицах БД и заносится в них путем ввода контактных данных клиента, введенных им при регистрации.

### 2.3 Требования к структуре системы

#### 2.3.1 Требования к структуре и функционированию системы

Система магазина мясной продукции должна быть централизованной, т.е. все данные должны располагаться в центральном хранилище. В системе предлагается выделить следующие функциональные модули:

- Модуль «Авторизация». При входе клиент вводит почту, пароль, чтобы зайти в личный кабинет.

- Модуль «Регистрация». При входе клиент вводит запрашиваемые данные.
- Модуль «Меню», который предназначен для хранения информации о товарах компании.
- Модуль «Корзина», который предназначен для хранения и обработки информации о оформляющемся заказе.
- Модуль «Заказы», который предназначен для хранения информации о прошлых заказах.
- Модуль «Настройки», который предназначен для изменения данных клиента.
- Модуль «Доставка заказов», который предназначен для хранения и обработки информации по конкретным заказам клиентов, а также для отображения информации о номере телефона и адресе клиента, сделавшего заказ.
- Модуль «Панель администратора», который предназначен для регистрации новых работников работником-администратором.

### 2.3.2 Требования к способам и средствам связи для информационного обмена между компонентами системы

Взаимодействие компонентов системы осуществляется стандартными средствами платформы, на которой разработана система.

### 2.3.3 Требования к режимам функционирования системы

Система должна поддерживать следующие режимы функционирования:

- Основной режим, в котором подсистемы ММП выполняют все свои основные функции.
- Профилактический режим, в котором одна или все подсистемы ММП не выполняют своих функций.

В основном режиме функционирования Система ММП должна обеспечивать:

- работу пользователей режиме – 24 часов в день, 7 дней в неделю (24x7);
- выполнение своих функций – сбор, обработка и загрузка данных; хранение данных, предоставление отчетности.

В профилактическом режиме Система ММП должна обеспечивать возможность проведения следующих работ:

- техническое обслуживание;
- модернизацию аппаратно-программного комплекса;
- устранение аварийных ситуаций.

#### 2.3.4 Требования по диагностированию системы

Для обеспечения высокой надежности функционирования Системы как системы в целом, так и её отдельных компонентов должно обеспечиваться выполнение требований по диагностированию ее состояния. Обязательно ведение журналов инцидентов в электронной форме, а также графиков и журналов проведения ППР. Для всех технических компонентов необходимо обеспечить регулярный и постоянный контроль состояния и техническое обслуживание.

#### 2.3.5 Требования к надежности

Уровень надежности должен достигаться согласованным применением организационных, организационно-технических мероприятий и программно-аппаратных средств. При работе системы возможны следующие аварийные ситуации, которые влияют на надежность работы системы:

- сбой в электроснабжении сервера;
- сбой в электроснабжении рабочей станции пользователей системы;
- сбой в электроснабжении обеспечения локальной сети (поломка сети);
- сбой программного обеспечения сервера.

Проверка выполнения требований по надежности должна производиться на этапе проектирования расчетным путем, а на этапах испытаний и эксплуатации - по методике Разработчика, согласованной с Заказчиком.

#### 2.3.6 Требования к защите информации

Обеспечение информационное безопасности Системы ММП должно удовлетворять следующим требованиям:

- Защита Системы должна обеспечиваться комплексом программно-технических средств и поддерживающих их организационных мер.
- Защита Системы должна обеспечиваться на всех технологических этапах обработки информации и во всех режимах функционирования, в том числе при проведении ремонтных и регламентных работ.
- Программно-технические средства защиты не должны существенно ухудшать основные функциональные характеристики Системы (надежность, быстродействие, возможность изменения конфигурации).

## 2.4 Требования к функциям, выполняемым системой

- Хранение и обработка информации о клиентах;
- Хранение и обработка информации о заказах;
- Ведение БД

## 2.5 Требования к видам обеспечения

### 2.5.1 Требования к программному обеспечению

Доступные аппаратные средства:

- ПК с доступом в БД.

Доступные программные средства:

- Сервер БД MySQL;
- Любая операционная система с предустановленным программным обеспечением Java.

### 2.5.2 Требования к техническому обеспечению

- Локальный сервер

## 3. Проектирование информационной системы

Для проектирования использовался язык UML объектно-ориентированного подхода. Язык UML — это графический язык моделирования общего

назначения, предназначенный для спецификации, визуализации, проектирования и документирования всех артефактов, создаваемых при разработке программных систем.

### 3.1. Диаграмма прецедентов (вариантов использования)

Используется для представления функциональной модели информационной системы. Прецеденты (варианты использования — Use Cases) — это подробные процедурные описания вариантов использования системы всеми заинтересованными лицами, а также внешними системами, т. е. всеми, кто (или что) может рассматриваться как актёры (actors) — действующие лица. По сути, это своего рода алгоритмы работы с системой с точки зрения внешнего мира. Прецеденты являются основой функциональных требований к системе, позволяют описывать границы проектируемой системы, ее интерфейс, а затем выступают как основа для тестирования системы заказчиком с помощью приемочных тестов

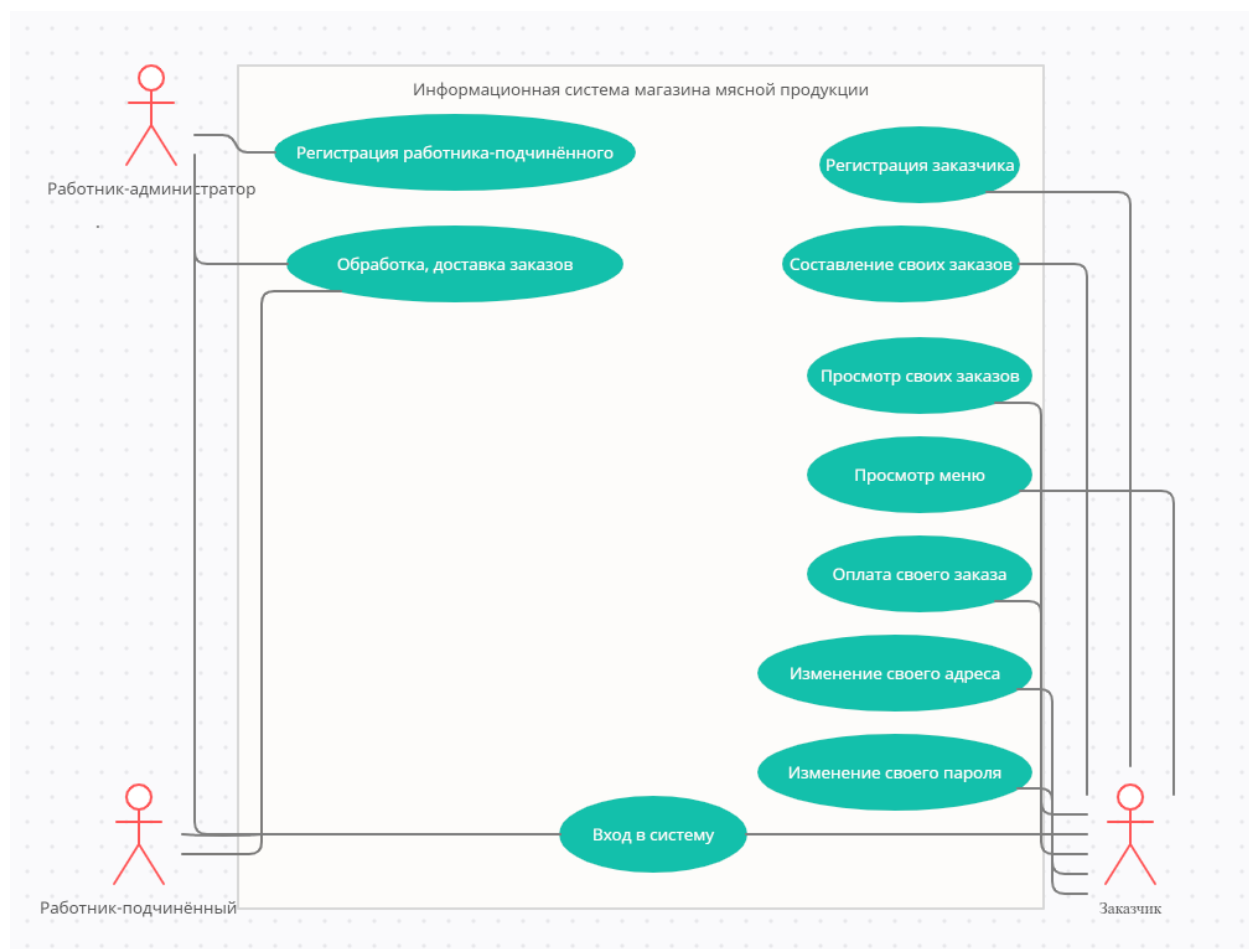


Рис 1. Диаграмма прецедентов

В разрабатываемой системе был выявлен следующий базовый список прецедентов:

- Вход в систему
- Регистрация работника-подчинённого
- Обработка, доставка заказов
- Регистрация заказчика
- Составление своих заказов
- Просмотр своих заказов
- Просмотр меню
- Оплата своего заказа
- Изменение своего адреса
- Изменение своего пароля

### 3.1.1 Прецедент “Вход в систему”

Основной исполнитель:

Работник-администратор/Работник-подчинённый/Заказчик.

Основной успешный сценарий:

- Пользователь входит в систему.

### 3.1.2 Прецедент “Регистрация работника-подчинённого”

Основной исполнитель: Работник-администратор.

Основной успешный сценарий:

- Пользователь регистрирует подчинённого в системе.

### 3.1.3 Прецедент “Обработка, доставка заказов”

Основной исполнитель: Работник-администратор/Работник-подчинённый.

Основной успешный сценарий:

- Пользователь смотрит размещённые недоставленные заказы
- Пользователь подтверждает доставку заказов.

### 3.1.4 Прецедент “Регистрация заказчика”

Основной исполнитель: Заказчик.

Основной успешный сценарий:

- Пользователь регистрируется в системе.
- Пользователь может авторизоваться и пользоваться системой.

### 3.1.5 Прецедент “Составление своих заказов”

Основной исполнитель: Заказчик.

Основной успешный сценарий:

- Пользователь составляет и размещает свой заказ в системе.



- Работники могут размещённый подтвердить заказ.

### 3.1.6 Прецедент “Просмотр своих заказов”

Основной исполнитель: Заказчик.

Основной успешный сценарий:

- Пользователь смотрит свои прошлые заказы, статус.

### 3.1.7 Прецедент “Просмотр меню”

Основной исполнитель: Заказчик.

Основной успешный сценарий:

- Пользователь смотрит меню - все товары компании.
- Пользователь может добавить товары в корзину.

### 3.1.8 Прецедент “Оплата своего заказа”

Основной исполнитель: Заказчик.

Основной успешный сценарий:

- Пользователь оплачивает свой заказ - выбирает оплату онлайн и вводит данные своей банковской карты.

### 3.1.9 Прецедент “Изменение своего адреса”

Основной исполнитель: Заказчик.

Основной успешный сценарий:

- Пользователь заходит во вкладку “Настройки” в меню, изменяет свой адрес доставки.

### 3.1.10 Прецедент “Изменение своего пароля”

Основной исполнитель: Заказчик.

Основной успешный сценарий:

- Пользователь заходит во вкладку “Настройки” в меню, изменяет свой пароль.

### 3.2 Диаграмма последовательности

Диаграмма последовательности — диаграмма, на которой показано взаимодействие объектов (обмен между ними сигналами и сообщениями), упорядоченное по времени, с отражением продолжительности обработки и последовательности их проявления.

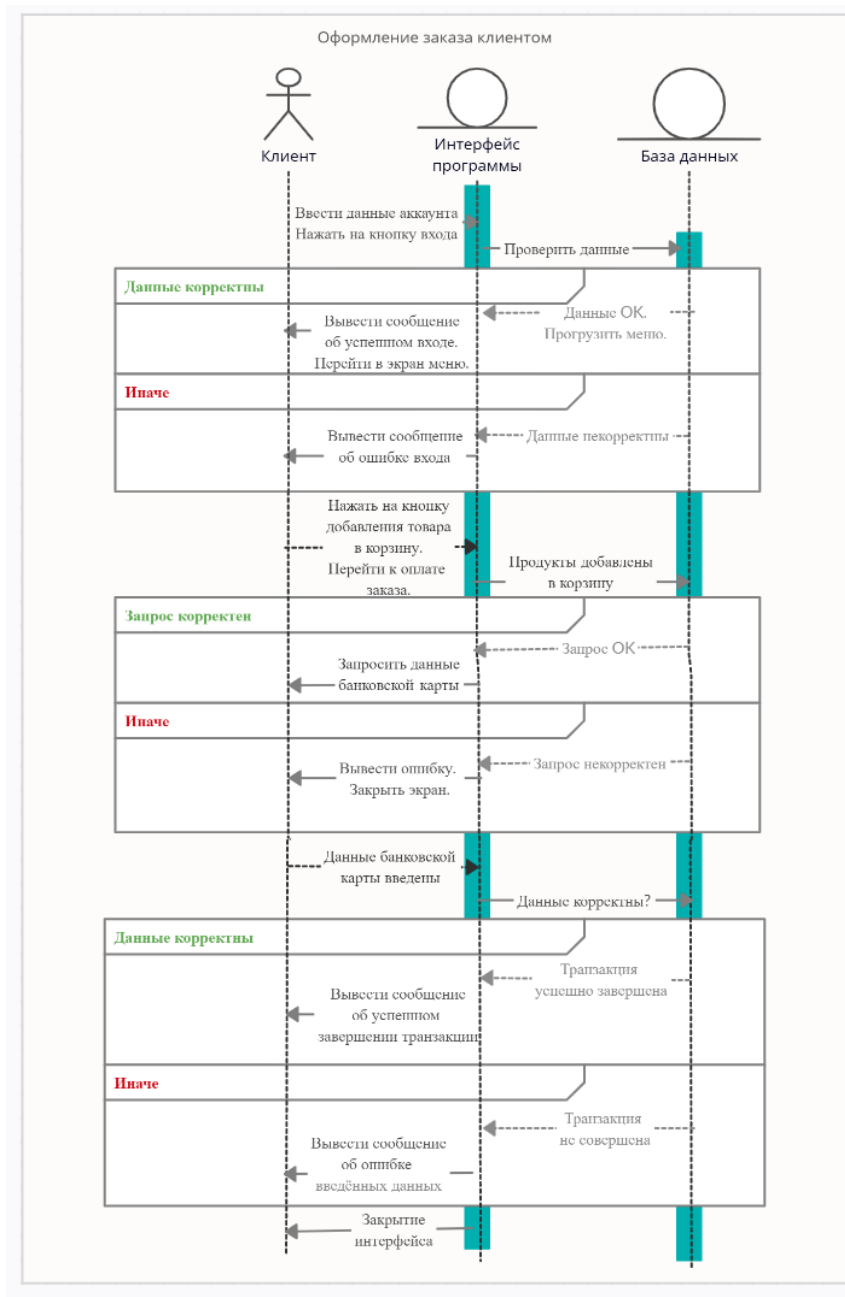


Рис 2. Диаграмма последовательности. Оформление заказа клиентом.

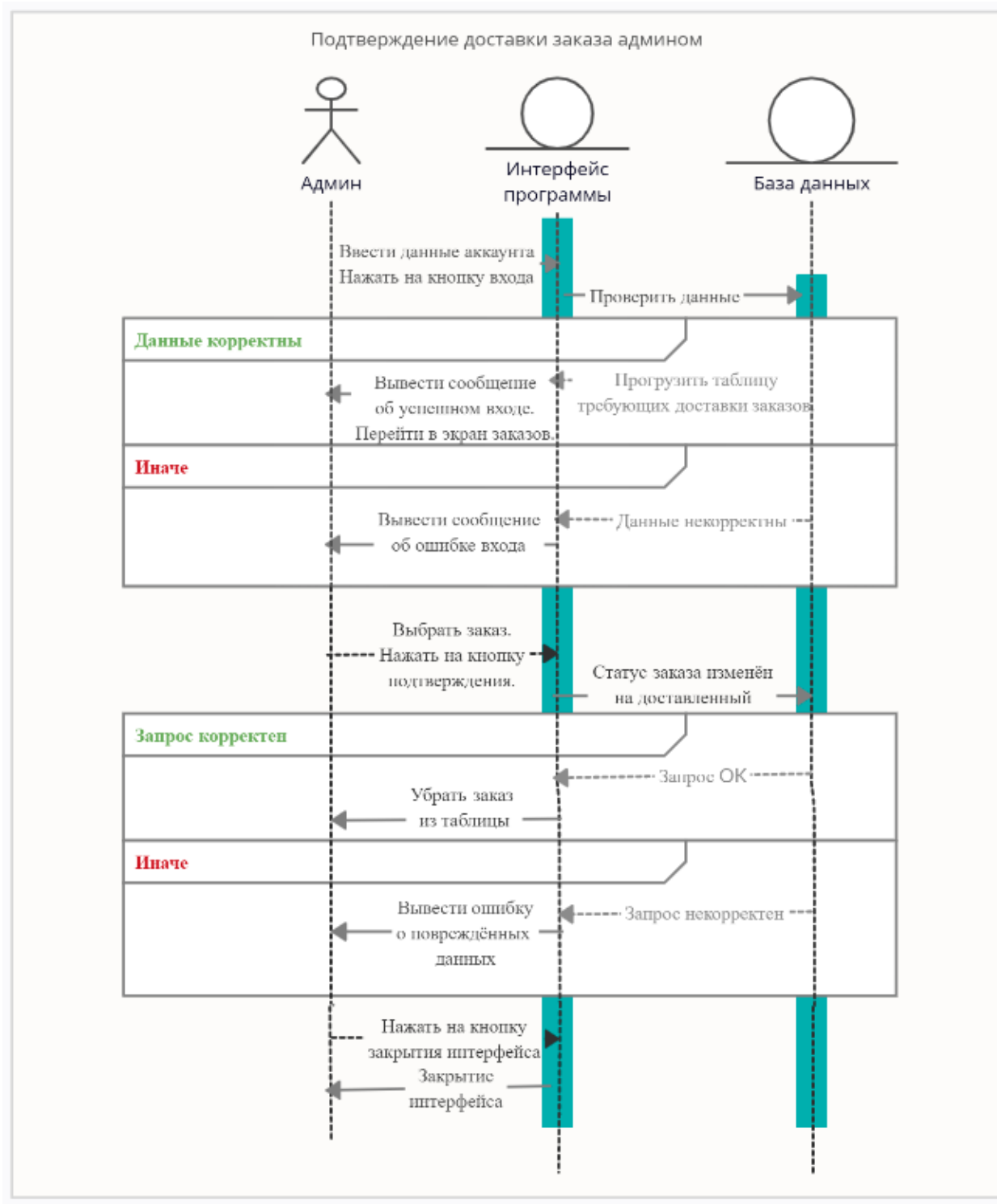


Рис 3. Диаграмма последовательности. Подтверждение доставки заказа админом.

### 3.3 Диаграмма компонентов

Диаграммы компонентов - это один из двух видов диаграмм, применяемых при моделировании физических аспектов объектно-ориентированной системы. Они показывают организацию наборов компонентов и зависимости между ними.

Диаграммы компонентов применяются для моделирования статического вида системы с точки зрения реализации. Сюда относится моделирование физических сущностей, развернутых в узле, например исполняемых программ, библиотек, таблиц, файлов и документов. По существу диаграммы компонентов - это не что иное, как диаграммы классов, сфокусированные на системных компонентах.

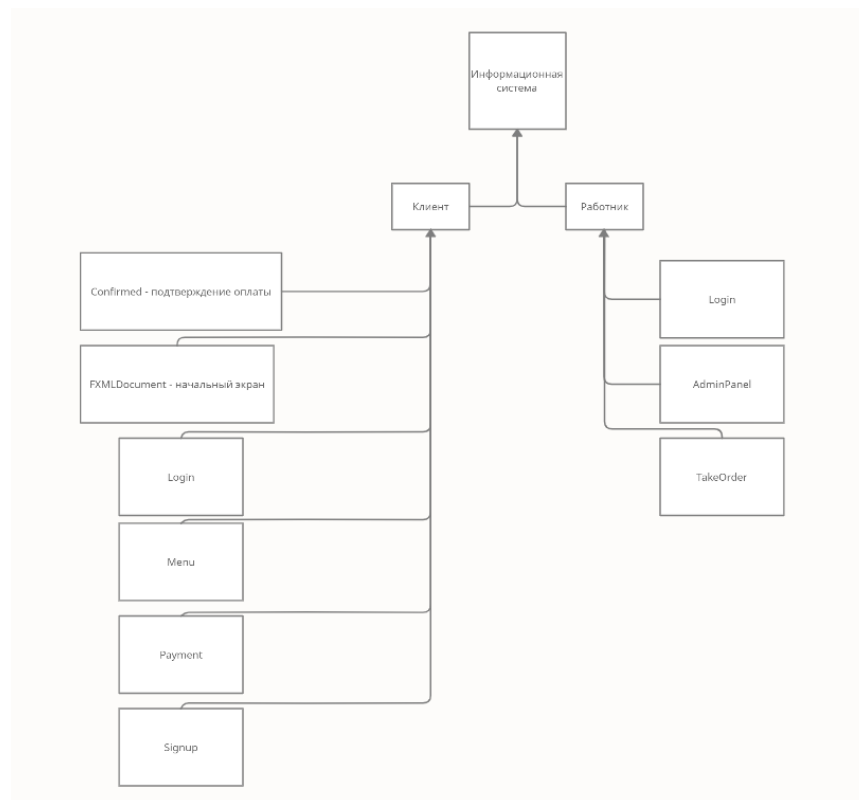


Рис 4. Диаграмма компонентов.

### 3.4 Информационно-логическая модель базы данных

Информационно-логическая модель отображает данные предметной области в виде совокупности информационных объектов и связей между ними. Эта модель представляет данные, подлежащие хранению в базе данных.

#### 3.4.1 Структура БД

В нижеследующем рисунке - поля всех таблиц БД, используемых разработанной информационной системой.

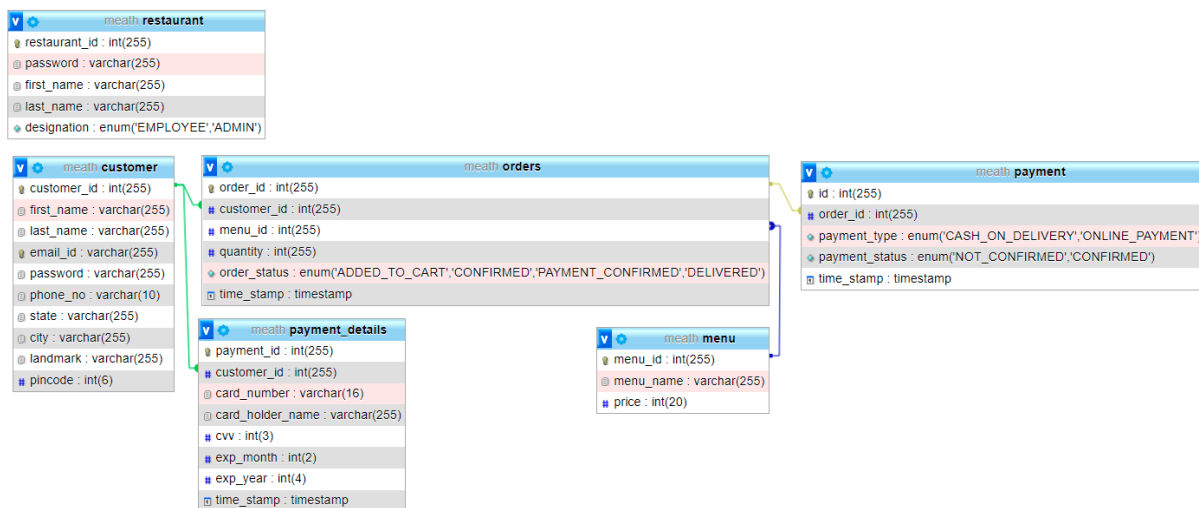


Рис 5. Информационно-логическая модель базы данных

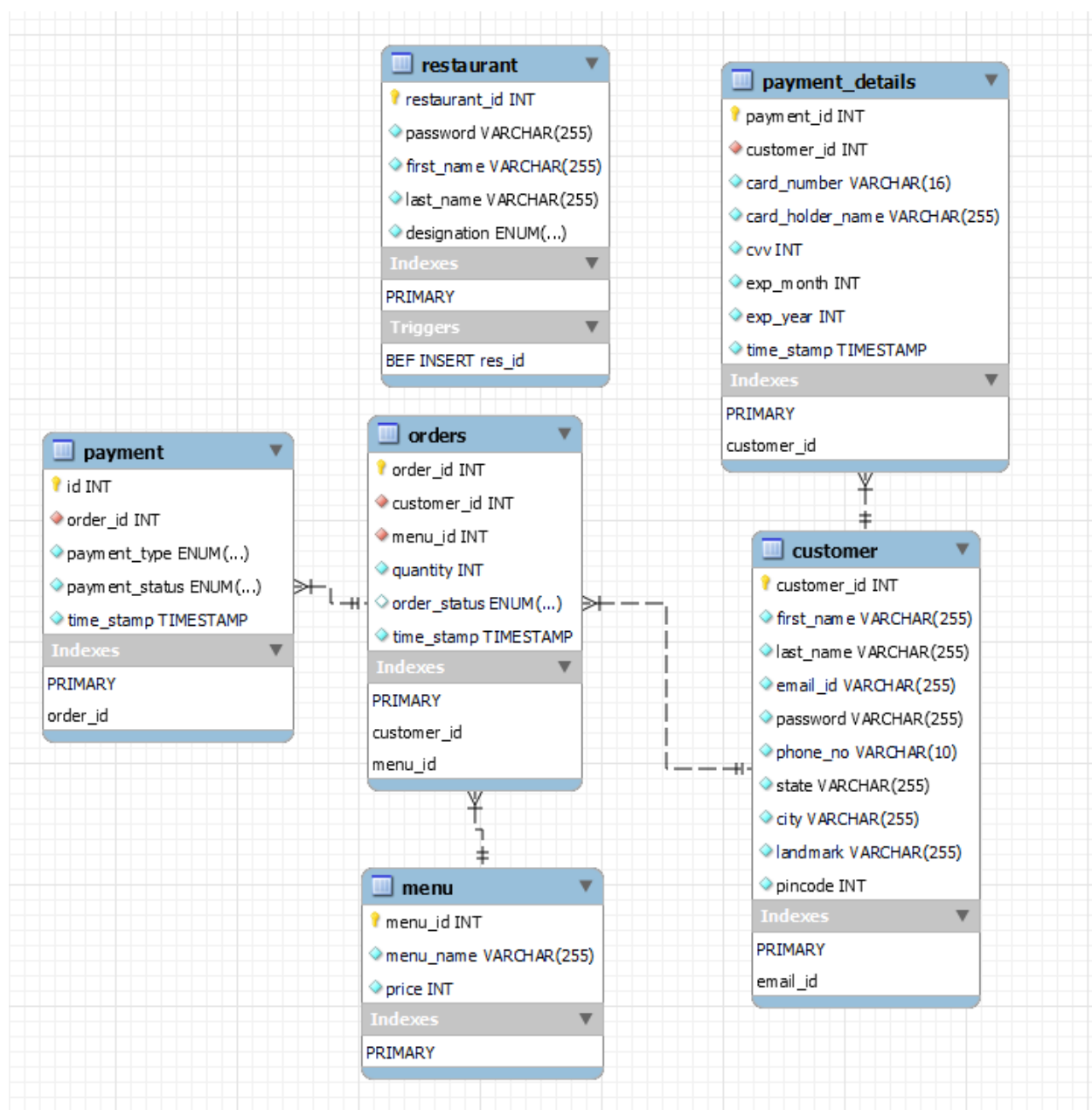


Рис 6. Информационно-логическая модель базы данных

## 4 Разработка информационной системы

### 4.1 Описание структуры системы

При запуске клиентского приложения пользователя встречает экран приветствия, который предлагает зарегистрироваться или авторизоваться.

Форма регистрации содержит такие поля: имя, фамилия, почта, телефон, пароль, область, город, адрес, почтовый индекс, и кнопку “Зарегистрироваться”. Форма авторизации содержит в себе два текстовых поля: почта, пароль, и кнопку «Войти».

Авторизация выполнена успешно – клиент попадает в меню. В нём располагаются продукты компании. Вкладка левее - корзина. Следующая вкладка - заказы. Последняя вкладка - настройки - позволяет изменить заполненные при регистрации данные адреса и изменить пароль. При нажатии на одну из вкладок открывается выбранная, остальные скрываются.

При запуске администраторского приложения пользователя встречает экран авторизации. Форма авторизации содержит в себе два текстовых поля: идентификатор, пароль, и кнопку «Войти».

Авторизация выполнена успешно - работник попадает в экран подтверждения доставки заказов.

### 4.2 Разработка графического пользовательского интерфейса и переключения между формами и панелями

Ниже приведены примеры внешнего вида взаимодействия с пользователем.

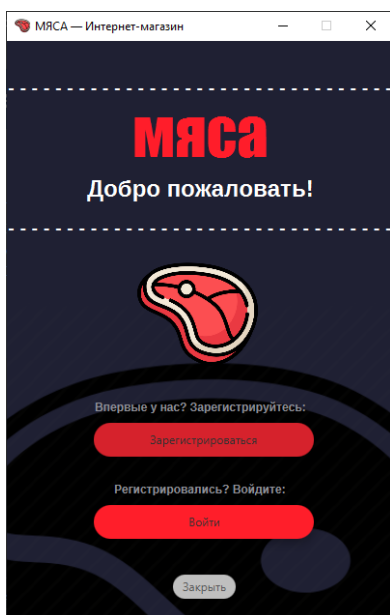


Рис 7. Экран приветствия

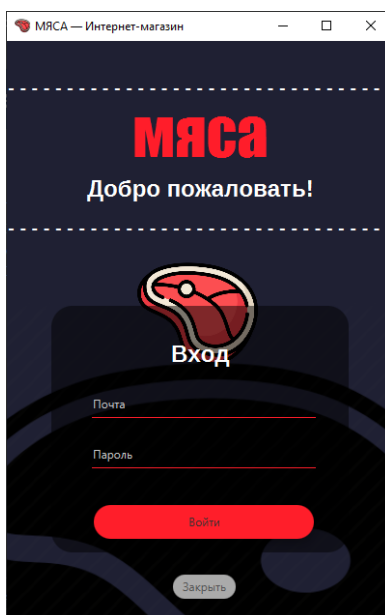


Рис 8. Экран авторизации

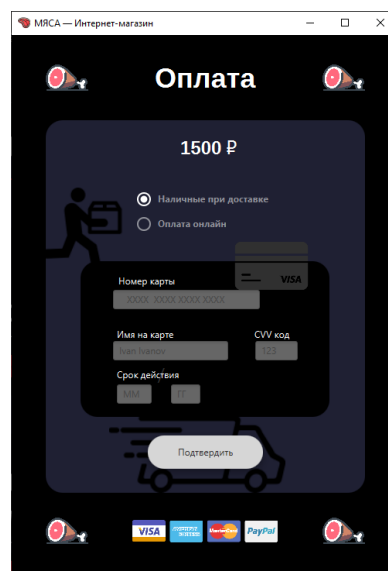


Рис 14. Экран оплаты

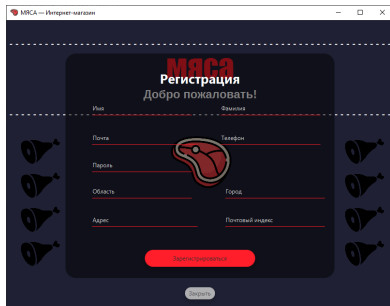


Рис 9. Экран регистрации

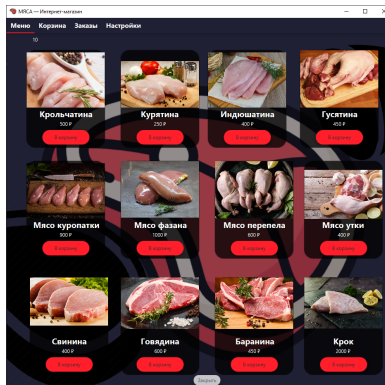


Рис 10. Экран меню

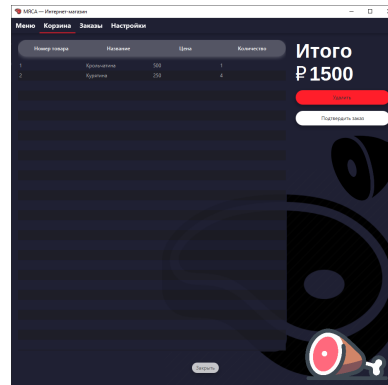


Рис 11. Экран корзины



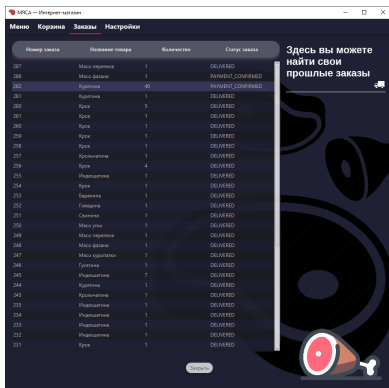


Рис 12. Экран заказов

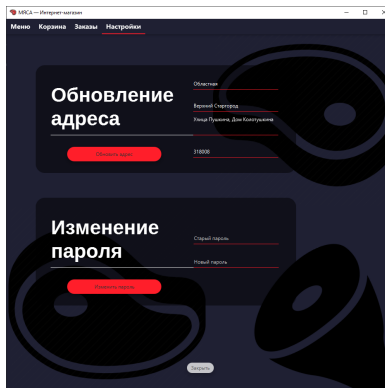


Рис 13. Экран настроек

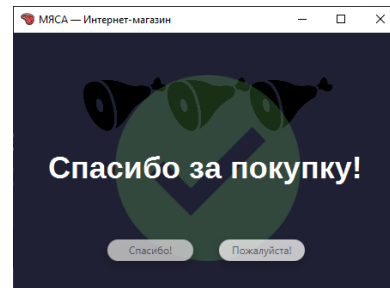


Рис 15. Экран спасибо

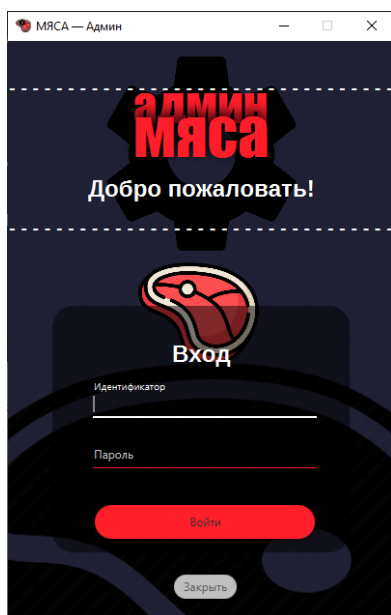


Рис 16. Экран авторизации работника

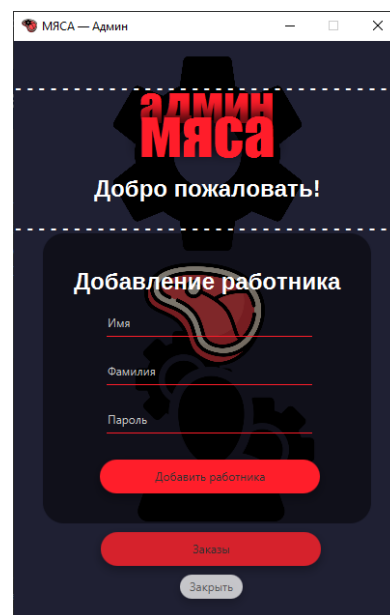


Рис 17. Экран панели админа

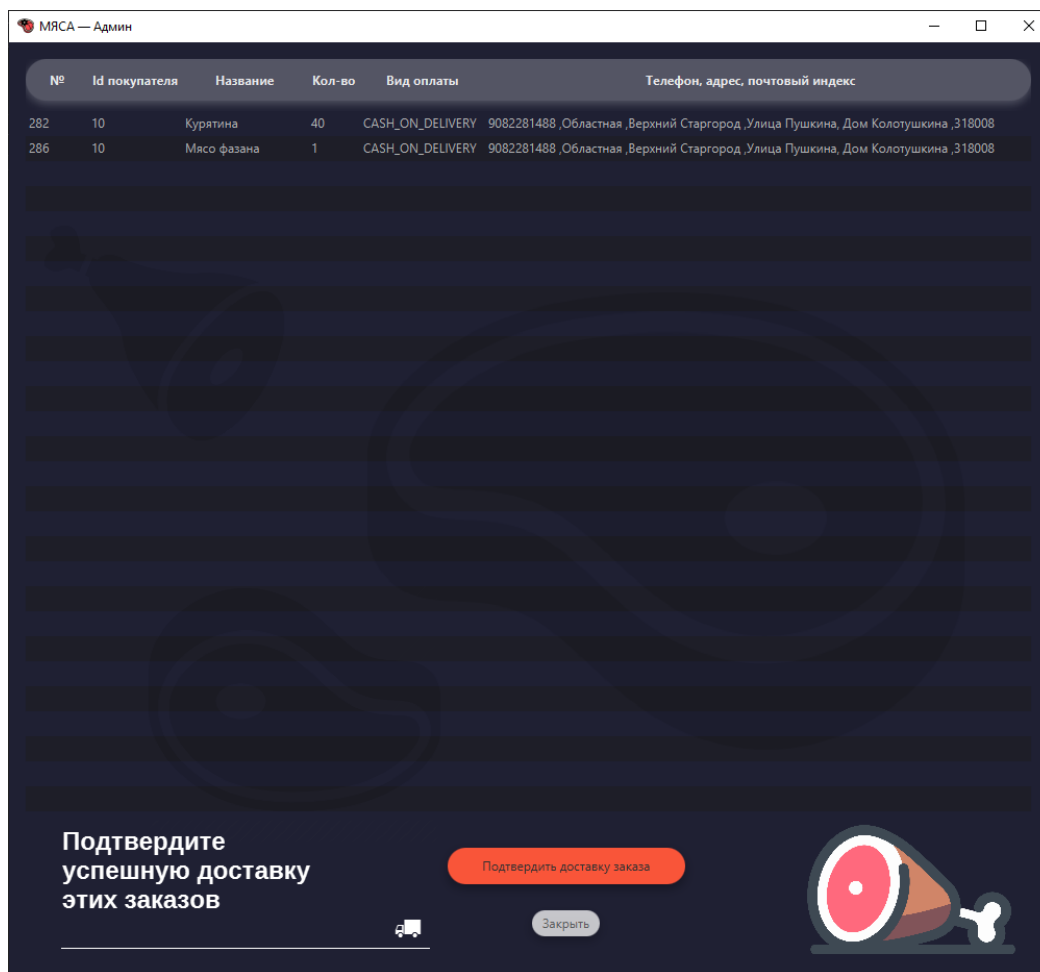


Рис 18. Экран подтверждения доставки заказов

#### 4.3 Разработка структуры системы графического пользовательского интерфейса, логики приложения

Код для разработанной информационной системы расположен по ссылке:

[https://gitlab.nntc.nnov.ru/neugomonov\\_v/meath](https://gitlab.nntc.nnov.ru/neugomonov_v/meath)

## 5. Тестирование информационной системы

Тестирование – важная часть любой программы контроля качества, поэтому информационная система пройдет этап тестирования, чтобы выявить и искоренить ошибки, мешающие комфортной работе с системой.

Тесты для разработанной информационной системы расположены по ссылке:

[https://gitlab.nntc.nnov.ru/neugomonov\\_v/meath](https://gitlab.nntc.nnov.ru/neugomonov_v/meath)

```
/**
```

```
* Тест проверяет подключение к базе данных
```

```
*/
```

```
@Test
```

```
public void testIsDbConnected() {  
    System.out.println("isDbConnected");  
    LoginModel instance = new LoginModel();  
    boolean expResult = true;  
    boolean result = instance.isDbConnected();  
    assertEquals(expResult, result);  
}
```

```
/**
```

```
* Тест проверяет, получается ли клиенту войти с введенными данными.
```

```
*/
```

```
@Test
```

```
public void testIsLogin() throws Exception {  
    System.out.println("isLogin");
```


```

String email = "egorvoronov228@gmail.com";
String pass = "123123";
LoginModel instance = new LoginModel();
boolean expResult = true;
boolean result = instance.isLogin(email, pass);
assertEquals(expResult, result);
}
/**
 * Тест проверяет, получается ли работнику войти с введёнными данными.
 */
@Test
public void testIsLogin() throws Exception {
    System.out.println("isLogin");
    int userid = 2;
    String pass = "123123";
    LoginModel instance = new LoginModel();
    boolean expResult = true;
    boolean result = instance.isLogin(userid, pass);
    assertEquals(expResult, result);
}
/**
 * Тест проверяет, получается ли пользователю войти как админу.
 */

```

@Test

```
public void testIsAdmin() throws Exception {  
    System.out.println("isAdmin");  
    int userid = 1;  
    String pass = "123123";  
    LoginModel instance = new LoginModel();  
    boolean expResult = true;  
    boolean result = instance.isAdmin(userid, pass);  
    assertEquals(expResult, result);  
}
```

Метод тестирования	Ожидаемый результат	Фактический результат
Ручной метод	<p>Вход в личный кабинет клиента</p> <p>Пользователь нажимает на кнопку “Войти”</p> <p>Пользователь корректно заполняет поля “Почта” и “Пароль”</p> <p>Пользователь нажимает на кнопку “Войти”</p> <p>Пользователь переходит в меню</p>	

## 6 Руководство пользователя

### 6.1 Страница приветствия и регистрации клиента

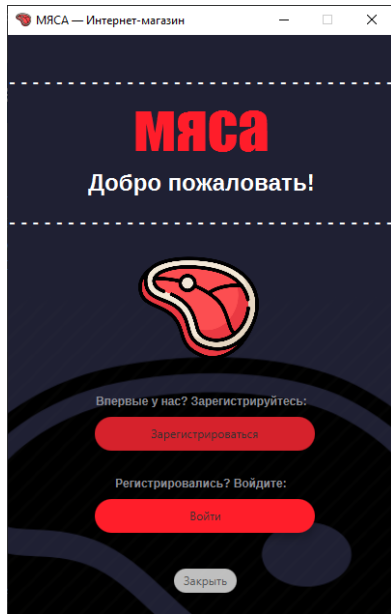


Рис 1. Экран приветствия

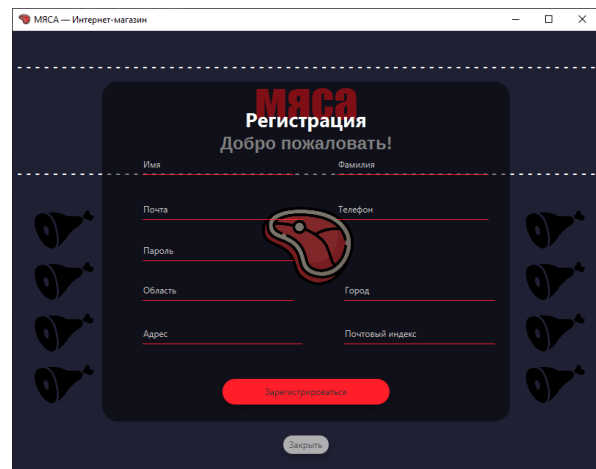


Рис 2. Экран регистрации

На этапе регистрации необходимо заполнить все предоставленные поля на латинице, город - без пробелов (однако в дальнейшем адрес можно изменить, используя кириллицу, и город написать отдельно). Почта и пароль будут использоваться при дальнейшем входе.

## 6.2 Страница входа клиента

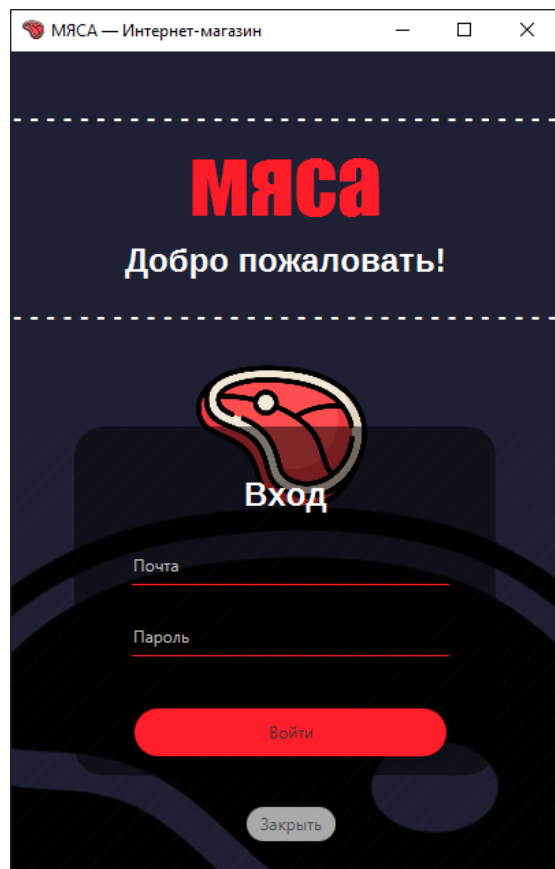


Рис 3. Экран авторизации

На этапе авторизации необходимо заполнить предоставленные поля “Почта” и “Пароль”, указанные при регистрации.

### 6.3 Основная страница личного кабинета клиента

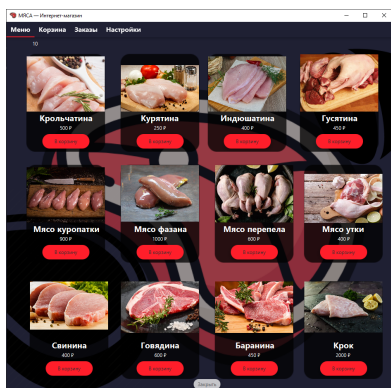


Рис 4. Экран меню

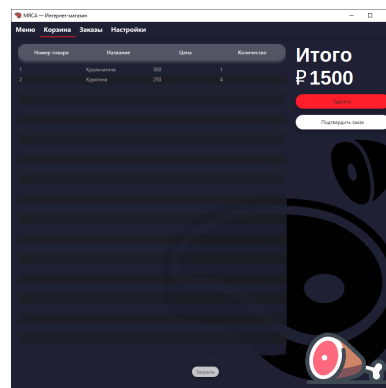


Рис 5. Экран корзины

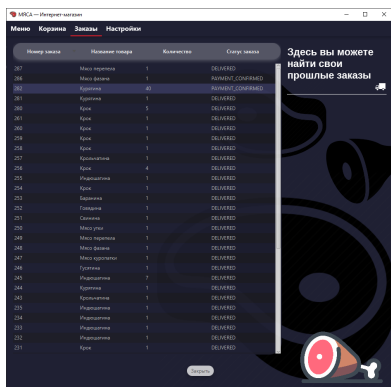


Рис 6. Экран заказов

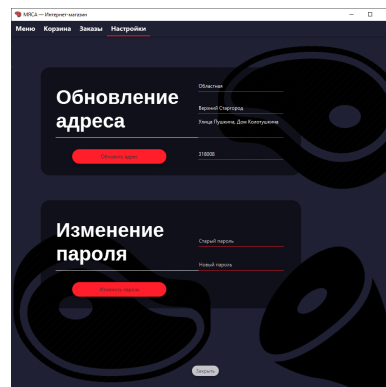


Рис 7. Экран настроек

Ура! Вы успешно зарегистрировались, вошли в интернет-магазин и теперь наконец-то можете поесть.

Перед вами представлено меню со всеми продуктами, что вы можете заказать. Добавьте их в корзину. Во вкладке правее - корзина - оформите заказ или, если передумали, уберите некоторые товары. Ваши оформленные заказы будут находиться во вкладке правее - заказы. Следующая вкладка - настройки - позволяет вам изменить именно те данные, которые вы указывали при регистрации - адрес и пароль. Не забудьте сохранить изменения, нажав на соответствующую кнопку.



## 6.4 Страница оплаты заказа

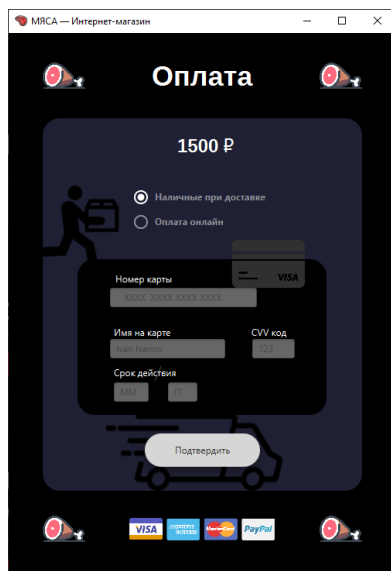


Рис 8. Экран оплаты

Вы можете выбрать между двумя вариантами - оплата заказа при доставке и оплата онлайн. Для оплаты онлайн введите данные своей банковской карты (это безопасно).

## 6.5 Страница подтверждения успешного заказа

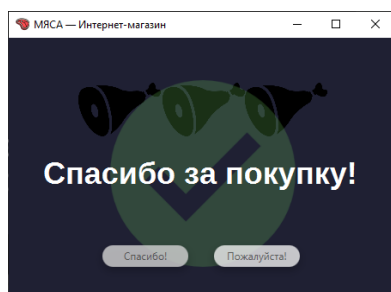


Рис 9. Экран спасибо

Мы благодарим вас за покупку. Ответьте взаимностью!

## 6.6 Страница авторизации сотрудника

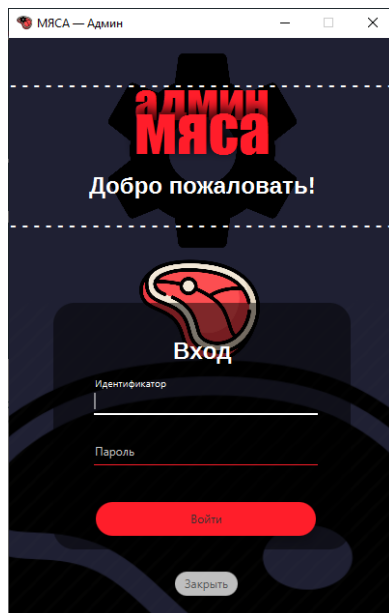


Рис 10. Экран авторизации работника

Ура! Вас приняли на работу. Чтобы начать работать, введите идентификатор и пароль, предоставленные вам админом.

## 6.7 Страница подтверждения доставки заказов

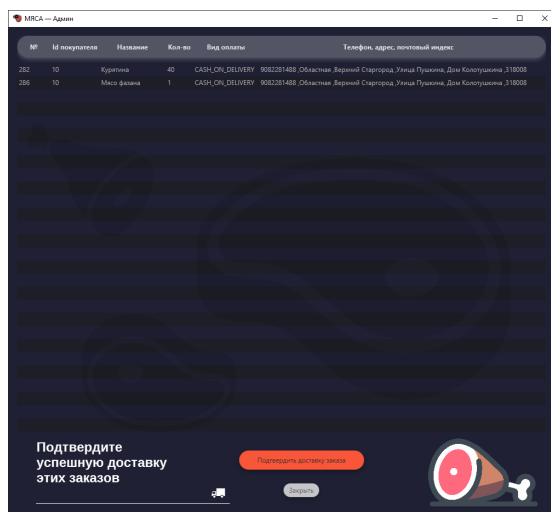


Рис 18. Экран подтверждения доставки заказов

Для подтверждения успешной доставки заказа нажмите на заказ, потом на кнопку “Подтвердить доставку заказа”. Заказ должен исчезнуть из таблицы.

## Заключение

Итогом данного курсового проекта является настольное приложение для клиента и работника компании «МЯСА» информационной системы для магазина мясной продукции.

Разработанное приложение позволяет облегчить работу админа, помочь заказчикам удобнее размещать заказы и отслеживать статус своих заказов.

Проект выполнен в полном соответствии с заданием на курсовое проектирование.

В проекте разработана эксплуатационная документация для подробного ознакомления пользователей с работой в системе.

Таким образом можно сделать вывод, что во время разработки были достигнуты все цели создания системы, и что её внедрение должно привести к повышению производительности и качества работы компании «МЯСА».

## Список использованных источников

1. Буч Г., Рамбо Дж., Джекобсон А. Язык UML. Руководство пользователя.-М.:ДМК ПРЕСС; СПб.:Питер,2004.-429 с.
2. Вязовик, Н.А. Программирование на Java : курс лекций / Вязовик Н.А. — Москва : Интуит НОУ, 2016. — 604 с. — ISBN 978-5-9556-0006-2. — URL: <https://book.ru/book/918118>
3. Кондрашов, Ю.Н. Эффективное использование СУБД MS SQL Server : учебное пособие / Кондрашов Ю.Н. — Москва : Русайнс, 2017. — 128 с. — ISBN 978-5-4365-2293-7. — URL: <https://book.ru/book/927673>
4. Монахов, В.В. Язык программирования Java и среда NetBeans : курс лекций / Монахов В.В. — Москва : Интуит НОУ, 2016. — 451 с. — URL: <https://book.ru/book/918297>
5. Полякова, Л.Н. Основы SQL : курс лекций / Полякова Л.Н. — Москва : Интуит НОУ, 2016. — 274 с. — ISBN 978-5-9556-0101-4. — URL: <https://book.ru/book/917832>
6. Свистунов, А.Н. Построение распределенных систем на Java : курс лекций / Свистунов А.Н. — Москва : Интуит НОУ, 2016. — 318 с. — ISBN 978-5-9963-0444-8. — URL: <https://book.ru/book/917980>