**Федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«РОССИЙСКАЯ АКАДЕМИЯ НАРОДНОГО ХОЗЯЙСТВА   
И ГОСУДАРСТВЕННОЙ СЛУЖБЫ**

**ПРИ ПРЕЗИДЕНТЕ РОССИЙСКОЙ ФЕДЕРАЦИИ»**

**НИЖЕГОРОДСКИЙ ИНСТИТУТ УПРАВЛЕНИЯ – филиал РАНХиГС**

Факультет управления

Кафедра информатики и информационных технологий

Направление подготовки: 09.03.03 Прикладная информатика

Направленность (профиль): Корпоративные информационные системы управления

**КУРСОВАЯ РАБОТА (ПРОЕКТ)**

|  |  |
| --- | --- |
| по дисциплине (модулю) | Проектирование информационных систем |
| на тему: | «Проектирование информационной системы для для ритейл агенства-поставщика «ИнтерТех» с использованием CASE-технологий» |
|  |  |
|  |  |

|  |  |  |
| --- | --- | --- |
|  |  | **АВТОР** |
|  |  | Обучающаяся 3 курса группы Иб-331 |
|  |  | очной формы обучения |
|  |  |  |
|  |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Дубков М.О. |
|  |  | *(подпись) (фамилия, инициалы)* |
|  |  |  |
|  |  | **НАУЧНЫЙ РУКОВОДИТЕЛЬ** |
|  |  | оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  |  | «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2025 г. |
|  |  | *(дата защиты)* |
|  |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Миронов А. Ю. |
|  |  | *(подпись) (фамилия, инициалы)* |

Нижний Новгород, 2025г.

Оглавление

[I. Теоретическая часть 5](#_Toc198069503)

[1.1 Характеристика предметной области 5](#_Toc198069504)

[Матрица функциональной ответственности 7](#_Toc198069505)

[Анализ бизнес-процессов "Как есть" (AS-IS) 8](#_Toc198069506)

[Работа компании «ИнтерТех» 8](#_Toc198069507)

[Описание процессов диаграммы A0 (AS-IS) 10](#_Toc198069508)

[UML Диаграмма прецедентов (Use Case Diagram) в состоянии "Как есть" (AS-IS) 12](#_Toc198069509)

[Диаграммы последовательности в состоянии "Как есть" (AS-IS) 15](#_Toc198069510)

[Обработка заявки A2 (AS-IS) 18](#_Toc198069511)

[Анализ программного и аппаратного обеспечения 19](#_Toc198069512)

[1.2 Проблемная область 21](#_Toc198069513)

[1.3 Предложения по решению 22](#_Toc198069514)

[1.4 Анализ средств проектирования и разработки 24](#_Toc198069515)

[II. Проектирование и реализация информационной системы «Один» 26](#_Toc198069516)

[2.1 Целевая модель бизнес-процессов компании «ИнтерТех» после внедрения системы «Один» 26](#_Toc198069517)

[UML Диаграмма прецедентов в состоянии "Как должно быть" (TO-BE) 29](#_Toc198069518)

[UML Диаграмма последовательности (Sequence Diagram) в состоянии "Как должно быть" (TO-BE) 30](#_Toc198069519)

[2.2 Проектирование базы данных информационной системы «Один» 31](#_Toc198069520)

[2.2.1 Логическое проектирование базы данных 31](#_Toc198069521)

[2.2.2 Физическое проектирование базы данных 33](#_Toc198069522)

[2.3 Создание моделей экранных форм информационной системы «Один» 35](#_Toc198069523)

[2.3.1 Модель формы авторизации 35](#_Toc198069524)

[2.3.2 Модель главного окна (Основной интерфейс системы «Один») 36](#_Toc198069525)

[2.3.3 Другие формы и шаблоны HTML 37](#_Toc198069526)

[2.5 Программная реализация информационной системы «Один» 39](#_Toc198069527)

[Заключение 42](#_Toc198069528)

[Польза от реализации предложенного проекта 43](#_Toc198069529)

[Направления дальнейшего развития проекта 44](#_Toc198069530)

[Список использованных источников 46](#_Toc198069531)

[Приложения 47](#_Toc198069532)

**Введение**

Компания **«***ИнтерТех***»** занимается комплексным снабжением и поставкой промышленного оборудования для различных отраслей. В её ассортимент входят современные импортные оборудования мировых производителей и запасные части. Компания предоставляет услуги высокого качества, гарантируя соблюдение сроков поставок, а для постоянных клиентов предлагает программу лояльности.

Несмотря на успешность компании, существуют трудности в управлении клиентами, заказами и внутренними коммуникациями. Основные проблемы включают:

1. Отсутствие централизованной базы данных клиентов, что приводит к задержкам в обработке запросов и потере информации.
2. Ручное управление заказами, что увеличивает вероятность ошибок и затрудняет контроль их статусов.
3. Замедленные коммуникации между отделами, из-за чего клиенты получают ответы с задержкой.

Для устранения этих проблем предлагается внедрение системы «*Один*». Это позволит:

1. Централизовать данные о клиентах и заказах;

2. Автоматизировать процессы управления заказами;

3. Улучшить коммуникации между отделами и ускорить взаимодействие с клиентами.

**Цель работы:** разработка и внедрение системы «*Один*», которая повысит эффективность управления клиентами и заказами, сократит время на обработку запросов и минимизирует ошибки.

**Задачи работы:**

1. Провести анализ текущих бизнес-процессов компании.
2. Определить функциональные требования к системе.
3. Разработать архитектуру и базу данных для новой системы.
4. Подготовить макеты интерфейса и отчётности.
5. Оценить результаты внедрения системы.

# Глава I. Теоретическая часть

## **Характеристика предметной области**

Компания «ИнтерТех» специализируется на комплексном снабжении и поставке промышленного оборудования и запасных частей для предприятий различных отраслей. Основная деятельность компании направлена на обеспечение своевременной и качественной доставки оборудования, соответствующего потребностям заказчиков.

Работа компании включает несколько ключевых этапов. Сначала клиенты отправляют свои запросы через электронную почту, телефон или мессенджеры. Менеджеры по продажам принимают заявки, фиксируют их и проверяют наличие продукции по базе данных. На этапе обработки уточняются детали заказа, такие как ассортимент, наличие продукции, сроки поставки и стоимость. Затем формируются окончательная заявка и счёт на оплату. После согласования условий заказа с клиентом выставляется счёт, а клиент производит оплату. Данные об оплате фиксируются в таблицах. После подтверждения оплаты логист организует доставку в соответствии с оплаченной заявкой. Складские сотрудники подготавливают продукцию к отправке, сверяя её с деталями заказа. В завершение менеджеры подготавливают итоговые отчёты о выполнении заказов, фиксируя сроки, объёмы и качество.

**Организационная структура** компании включает менеджеров по продажам, логистов и складских сотрудников. Менеджеры принимают заявки, консультируют клиентов, ведут базу данных. Логисты отвечают за координацию доставки, контролируют выполнение заказов. Складские сотрудники занимаются обработкой и подготовкой оборудования к отправке. Для работы используются таблицы Excel для учёта клиентов, заказов и складских остатков, а также такие каналы связи, как телефон, электронная почта и мессенджеры.

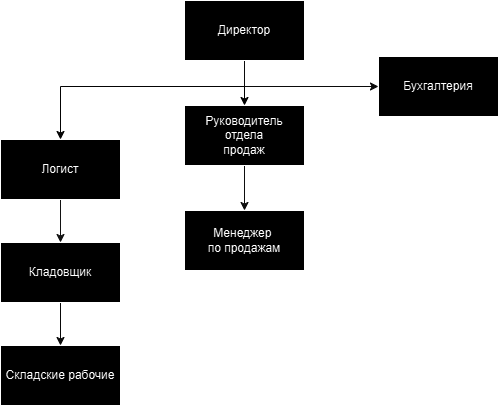
В текущей работе компании «ИнтерТех» наблюдаются значительные проблемы, которые снижают её общую эффективность. Основной проблемой является отсутствие централизованной базы данных. Информация о клиентах, заказах и взаимодействиях хранится в разрозненных таблицах Excel, что приводит к трудностям в поиске и анализе данных, потере информации о клиентах и их запросах, увеличению времени на обработку заказов. История взаимодействий с клиентами практически не фиксируется, что снижает уровень обслуживания.

Рисунок 1. Организационная структура

Ещё одной проблемой является нерациональная организация работы с заказами. Заказы обрабатываются вручную через электронную почту и телефон, что увеличивает вероятность ошибок и усложняет контроль статусов выполнения заказов.

Отсутствует автоматическое отслеживание статусов заказов, что требует дополнительных временных затрат сотрудников. Также наблюдается низкая эффективность коммуникаций. Взаимодействие между отделами осуществляется через электронную почту и телефон без единого стандарта. Сотрудники тратят значительное время на уточнение деталей, что замедляет обработку запросов. Отсутствие аналитики также является серьёзной проблемой.

Нет инструментов для систематического анализа эффективности работы компании, а отчёты формируются вручную, что увеличивает риск ошибок и требует много времени.

**Матрица функциональной ответственности** (таб. 1) описывает распределение ролей и ответственности за выполнение ключевых функций в компании.[[1]](#footnote-1) Для каждого функционального блока определены сотрудники или отделы, которые выполняют задачи, несут ответственность за результат, консультируют по вопросам и информированы о процессе. Эта матрица наглядно показывает, кто и за что отвечает на каждом этапе рабочего процесса в текущем состоянии ("как есть").

### Матрица функциональной ответственности

*Таблица 1*

| **Функциональная область** | **Менеджер по продажам** | **Логист** | **Складской сотрудник** |
| --- | --- | --- | --- |
| **Приём заявки от клиента** | В, О | Н | Н |
| **Обработка заявки** | В, О | К | Н |
| **Ведение базы данных клиентов** | В, О | К | Н |
| **Контроль выполнения заказа** | К | В, О | Н |
| **Отгрузка продукции** | Н | В, О | В, О |
| **Формирование отчёта** | В, О | К | Н |

* **В** — Тот, кто непосредственно выполняет задачу.
* **О** — Тот, кто несёт ответственность за результат и принимает окончательное решение.
* **К** — Тот, кто консультирует по вопросам, связанным с задачей.
* **Н** — Тот, кто информирован о процессе и результатах, но не принимает активного участия.

### ****Анализ бизнес-процессов "Как есть" (AS-IS)****

### ****Работа компании «ИнтерТех»****

Входными данными для главного процесса являются запрос клиента и данные о заказе. Запрос клиента поступает через телефон, электронную почту или другие доступные каналы связи. Данные о заказе включают в себя спецификации товара, объем, сроки и условия поставки.

* Процесс регулируется прайс-листами и различными регламентами:
* Прайс-листы определяют стоимость предлагаемых товаров и услуг.
* Регламенты компании описываюют стандарты работы с заказами и взаимодействия между сотрудниками, и нормативные документы устанавливают требования к процессу доставки и оформлению документации.

Для выполнения процесса используются следующие ресурсы:

* Сотрудники, включая менеджеров по продажам, логистов и складских работников, которые играют ключевую роль в обработке заказов.
* Таблицы Excel, в которых фиксируются данные о клиентах, заказах и выполнении.
* Средства связи, такие как телефон, электронная почта и мессенджеры, которые используются для взаимодействия с клиентами и между сотрудниками.

Выходными данными процесса являются выполненный заказ и итоговый отчёт. Выполненный заказ включает в себя переданную продукцию с подтверждением доставки. Итоговый отчёт содержит данные о процессе выполнения заказа, включая его сроки, стоимость и качество выполнения.

Процесс "Выполнение заказов" включает в себя несколько этапов, начиная от приёма заявки клиента и заканчивая передачей продукции и формированием отчёта. На текущий момент многие этапы выполняются вручную, что снижает общую эффективность системы.



Рисунок 2. Контекстная диаграмма A0

### ****Описание процессов диаграммы A0 (AS-IS)****

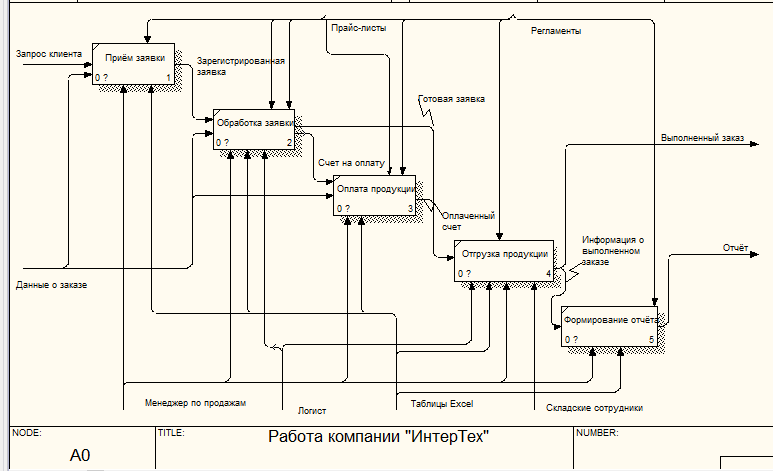


Рисунок 3. Контекстная диаграмма – IDEF0

**A1: Приём заявки**

Приём заявки является отправной точкой в цепочке бизнес-процессов компании «ИнтерТех». Клиенты отправляют свои запросы через различные каналы связи, такие как электронная почта, телефон или мессенджеры, предоставляя основные данные о желаемом заказе. Менеджер по продажам принимает этот запрос и фиксирует его в таблицах Excel, сверяясь с прайс-листами для проверки актуальности цен. В процессе приёма заявки менеджер руководствуется внутренними регламентами компании, которые определяют порядок и стандарты обработки поступающей информации. Результатом этого этапа становится зарегистрированная заявка, готовая к дальнейшему рассмотрению.

**A2: Обработка заявки**

После регистрации заявки начинается её обработка, в ходе которой уточняются детали заказа, такие как ассортимент, наличие продукции, сроки поставки и стоимость. Менеджер по продажам и логист совместно анализируют заявку, используя актуальные прайс-листы и соблюдая установленные регламенты компании. На этом этапе формируются готовая заявка и счёт на оплату, которые передаются клиенту для подтверждения и дальнейших действий. Все данные фиксируются в таблицах Excel, что обеспечивает прозрачность обработки и возможность последующего анализа информации.

**A3: Оплата продукции**

Этап оплаты продукции начинается с выставления счёта клиенту, сформированного на предыдущем этапе. После получения счёта клиент производит оплату, что подтверждается бухгалтерией компании. Менеджер по продажам контролирует поступление оплаты, сверяя данные через таблицы Excel. Для корректного выполнения этого процесса используются прайс-листы и регламенты компании, которые позволяют удостовериться в точности и соответствии всей документации. Завершением этого этапа становится оплаченный счёт, который становится основанием для перехода к этапу отгрузки продукции.

**A4: Отгрузка продукции**

После подтверждения оплаты продукции начинается её отгрузка. На основании готовой заявки и оплаченного счёта логист организует процесс доставки. Складские сотрудники подготавливают продукцию к отправке, сверяя её с деталями заказа. Менеджер по продажам и логист совместно следят за выполнением этого процесса, руководствуясь внутренними регламентами компании. Для учёта перемещений продукции и контроля используются таблицы Excel. Завершением этого этапа является выполненный заказ, информация о котором передаётся на заключительный этап формирования отчётов.

**A5: Формирование отчётов**

Финальный этап цепочки процессов компании заключается в формировании отчётов, которые фиксируют результаты выполненных заказов. На основе информации о доставленной продукции менеджер по продажам подготавливает отчёты, включающие данные о сроках выполнения, объёмах поставок и состоянии складов. Этот процесс осуществляется в соответствии с регламентами компании, с использованием таблиц Excel для систематизации и анализа данных.

Итогом работы становится готовый отчёт, который позволяет оценить эффективность работы и подготовить информацию для стратегического планирования.

### UML Диаграмма прецедентов (Use Case Diagram) в состоянии "Как есть" (AS-IS)

Для анализа текущего состояния бизнес-процессов компании «ИнтерТех» в рамках управления клиентами и заказами была построена UML Диаграмма прецедентов (рис. 4), отражающая функциональные требования к деятельности организации с точки зрения ее участников (акторов). На данной диаграмме граница системы представляет собой совокупность текущих операционных процессов компании, а не программное обеспечение.

Главными участниками, взаимодействующими с этими процессами, являются:

**Клиент**: Внешний участник, который инициирует взаимодействие с компанией, отправляя запросы на необходимую продукцию, а также завершает финансовую сторону сделки, производя оплату по выставленным счетам.

**Менеджер по продажам**: Ключевой внутренний актор, который находится в центре процесса работы с клиентами и заказами. В его обязанности входит принятие входящих запросов от клиентов, комплексная обработка этих запросов, внесение информации о заказах, контроль статуса оплаты и формирование различных отчетов о своей деятельности.

**Логист**: Внутренний актор, ответственный за физическое перемещение товаров. Он занимается организацией доставки продукции до клиента и осуществляет контроль за выполнением этой доставки.

**Складской сотрудник**: Внутренний актор, непосредственно работающий с товаром на складе. Его функции включают подготовку продукции к отгрузке согласно заказам и выполнение самой операции отгрузки.

**Бухгалтер**: Внутренний актор, участвующий в финансовом контуре процессов, в частности, осуществляющий получение и контроль счетов, а также фиксирующий факт оплаты.

Диаграмма прецедентов детализирует основные задачи (прецеденты), выполняемые этими акторами внутри бизнес-процесса:

Клиент может выполнить два ключевых действия: "Отправить запрос на продукцию" (инициируя сделку) и "Произвести оплату" (завершая финансовую часть).

Менеджер по продажам имеет наиболее широкий набор функций. Он выполняет "Принять запрос" от клиента, а затем приступает к центральному и наиболее сложному прецеденту – "Обработать заявку". Этот прецедент "Обработать заявку" включает в себя (связь <<include>>, указывающая на обязательность выполнения этих под-действий в рамках обработки):

* "Проверить наличие продукции (Excel)" – поиск нужных товаров в текущих учетных таблицах.
* "Проверить данные клиента в БД (Excel)" – поиск информации о клиенте, хранящейся в разрозненных источниках. Если в процессе этой проверки выясняется, что данных о клиенте нет или они требуют обновления, выполняется расширяющий прецедент (связь <<extend>>) "Внести/обновить данные клиента".
* "Рассчитать стоимость заказа" – определение итоговой цены вручную или с использованием неавтоматизированных инструментов (прайс-листы, калькулятор).
* "Сформировать счет" – создание документа для оплаты.
* "Согласовать сроки и условия" – взаимодействие с клиентом для уточнения деталей доставки и прочих условий.

Помимо обработки заявки, Менеджер также отвечает за "Внести данные о заказе" (добавление информации в учетные формы), "Контроль оплаты" (отслеживание поступления средств, вероятно, в сверке с бухгалтерией) и "Формирование отчета" (подготовка сводной информации о продажах/заказах).

Логист осуществляет прецеденты "Организация доставки" и "Контроль выполнения доставки".

Складской сотрудник выполняет действия "Подготовки продукции к отгрузке" и непосредственно "Отгрузку".

Бухгалтер выполняет прецедент "Получение и контроль счетов", подтверждая финансовые операции.

**Анализ проблем через призму диаграммы "Как есть":**

Данная Диаграмма прецедентов наглядно демонстрирует ключевые проблемы, существующие в текущей деятельности компании, что подтверждает необходимость предлагаемых улучшений. Сложность и "раздутость" центрального прецедента "Обработать заявку" у Менеджера, а также множество включенных в него шагов, явно указывают на высокую долю ручного труда и отсутствие автоматизации. Особо проблемными являются шаги, прямо указывающие на использование неинтегрированных инструментов, например, "Проверить наличие продукции (Excel)" и "Проверить данные клиента в БД (Excel)". Это прямое следствие отсутствия централизованной базы данных и автоматизированного поиска, что приводит к задержкам, ошибкам и потере информации.

Множество связей между различными акторами и разнесение последовательных этапов (например, обработка менеджером, организация доставки логистом, отгрузка складом, контроль оплаты бухгалтером) по разным акторам без централизованного управления потоками информации и статусами, иллюстрирует проблему низкой эффективности коммуникаций и ручного контроля. Передача информации между отделами требует дополнительных усилий и времени, увеличивает вероятность ошибок и затрудняет оперативное отслеживание всего жизненного цикла заказа.

Необходимость выполнения отдельных прецедентов для ручного "Внесения данных о заказе" и "Формирования отчета" подчеркивает отсутствие автоматизированного сбора и анализа данных, что замедляет работу и снижает точность отчетности.

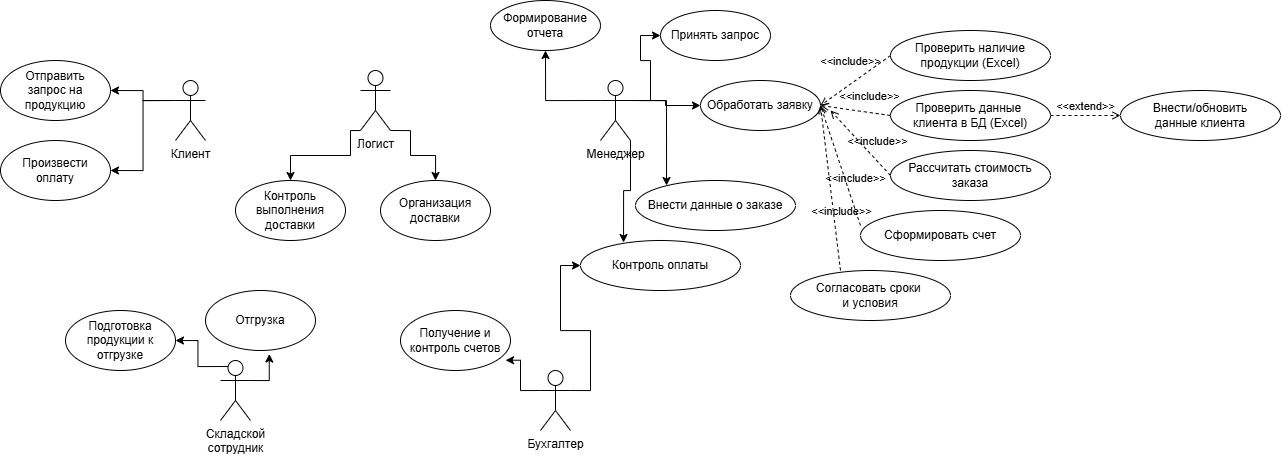


Рисунок 4. Диаграмма прецедентов (Use Case Diagram) в состоянии "Как есть" (AS-IS)

### Диаграммы последовательности в состоянии "Как есть" (AS-IS)

Процесс начинается с того, что клиент инициирует взаимодействие, отправляя менеджеру по продажам "Запрос на продукцию".

Получив запрос, менеджер по продажам приступает к его обработке. Первым делом ему необходимо проверить информацию о клиенте и наличии запрошенных товаров. Для этого менеджер вручную обращается к "Базе данных клиентов (Excel)", осуществляя поиск по имеющимся данным клиента. В ответ на этот поиск он получает "Данные клиента" из файла (если они найдены). Параллельно или последовательно, менеджер также вручную обращается к "Базе данных товаров (Excel)" для проверки наличия интересующей продукции на складе. Из этого файла менеджер получает "Информацию о наличии".

После сбора необходимой информации, менеджер выполняет расчет стоимости заказа. На основании всех этих данных он формирует "Счет на оплату", который затем отправляет клиенту.

Следующим этапом является оплата. Клиент производит оплату, сам процесс которой происходит вне прямого контроля менеджера и данной конкретной последовательности, но результат оплаты фиксируется. Информация о поступившей оплате поступает к бухгалтеру, который выполняет "Получение и контроль счетов". После подтверждения факта оплаты, бухгалтер сообщает менеджеру об оплате счета. Это критически важный момент, поскольку без этого подтверждения дальнейшие действия по выполнению заказа не начинаются.

Получив переданное "Подтверждение оплаты" от бухгалтера, менеджер по продажам также передает необходимую информацию о заказе (что, кому, куда, когда) логисту (для организации доставки) и складскому сотруднику (для подготовки и отгрузки).

После получения информации, логист приступает к "Организации доставки" и "Контролю выполнения доставки", что включает взаимодействие с транспортными компаниями и отслеживание перемещения груза. Последовательно, складской сотрудник выполняет "Подготовку продукции к отгрузке" и осуществляет саму "Отгрузку" товара клиенту.

В завершение процесса, менеджер по продажам, опираясь на собранную информацию о выполнении заказа, формирует итоговый "Отчет".

**Проблемы и неудобства, иллюстрируемые данной последовательностью:**

Данная диаграмма последовательности наглядно выявляет ключевые проблемы текущей работы компании. Главный недостаток — это высокая степень ручного труда и зависимость процесса от межличностных коммуникаций между разными участниками.

Менеджер вынужден обращаться к отдельным, несвязанным "базам данных" (файлам Excel) для получения информации о клиентах и товарах. Этот ручной поиск занимает много времени, подвержен ошибкам и не гарантирует актуальности данных, поскольку файлы обновляются независимо.

Менеджер по продажам выполняет огромное количество разнородных задач: прием запросов, поиск информации, расчет стоимости, формирование документов, отслеживание оплаты, передача информации в другие отделы, формирование отчетов. Это перегружает менеджера и замедляет весь процесс.

Отсутствие централизованного контроля и статуса: Диаграмма показывает, что нет единого места, где статус выполнения заказа был бы виден всем участникам в реальном времени. Каждый актор (Логист, Склад, Бухгалтер) владеет только своей частью информации. Менеджеру приходится вручную собирать данные для отслеживания и отчетности.

Сложность масштабирования: Увеличение количества заказов приведет к пропорциональному увеличению ручной работы и усугублению всех описанных проблем, делая текущий процесс неэффективным для растущего бизнеса.

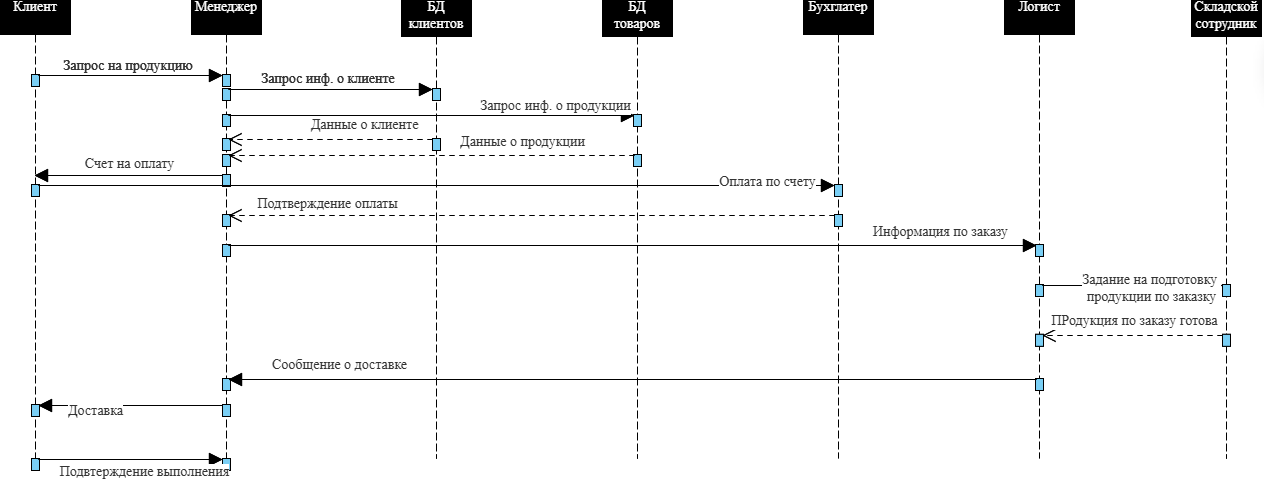


Рисунок 5. UML диаграмма последовательности

### ****Обработка заявки A2 (AS-IS)****

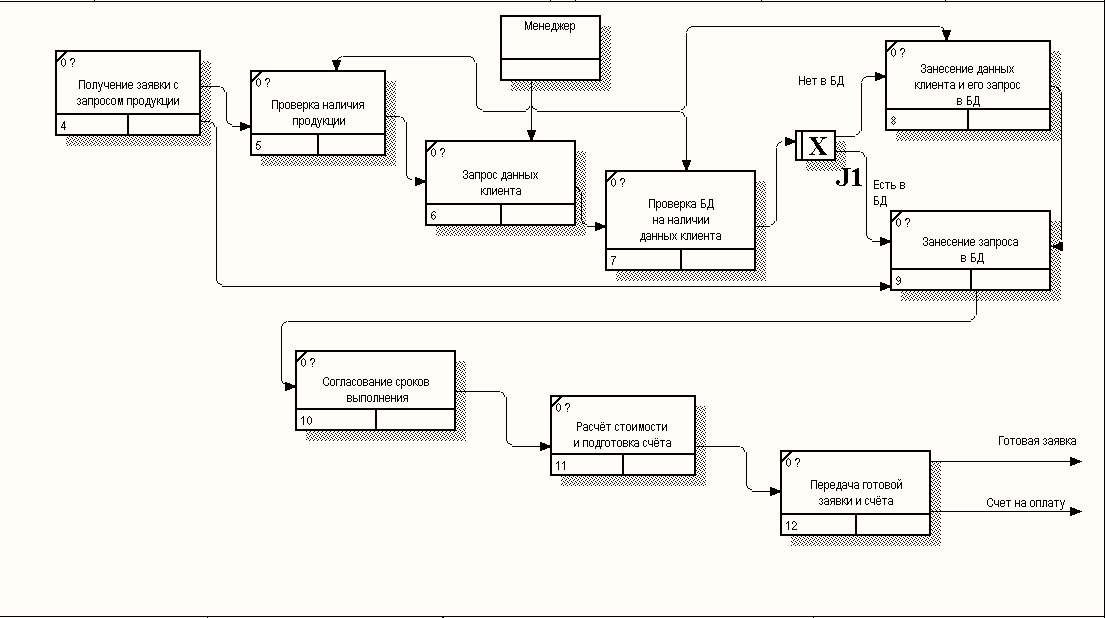


Рисунок 6. Диаграмма – IDEF3

Схема IDEF3 демонстрирует процесс обработки заявки на продукцию в компании "ИнтерТех" от момента получения запроса клиента до передачи готовой заявки и счёта. Работа начинается с получения заявки, где фиксируется информация о требуемой продукции. Далее осуществляется проверка наличия продукции на складе и в базе данных компании.

На следующем этапе происходит запрос информации о клиенте, после чего проводится проверка на наличие данных о клиенте в базе. Если данных о клиенте в системе нет, они добавляются вместе с его запросом. В случае наличия клиента в базе регистрируется только его заявка, что упрощает и ускоряет обработку. После внесения всех данных начинается согласование сроков выполнения, где учитываются доступность продукции и ресурсы компании.

На следующем этапе производится расчёт стоимости заказа и подготовка счёта. На основе этих данных формируется окончательная готовая заявка, которая передаётся клиенту вместе со счётом. Этот процесс завершает цикл обработки заявки, обеспечивая клиента всей необходимой информацией.

### Анализ программного и аппаратного обеспечения

Информационная инфраструктура компании «ИнтерТех» в ее текущем состоянии базируется на стандартном комплексе аппаратных и программных средств, предназначенных для обеспечения повседневной офисной деятельности и базовых коммуникаций.(рис. 7) Эта инфраструктура является основой, на которой развертываются текущие бизнес-процессы, связанные с управлением клиентами и выполнением заказов.

На рабочих местах сотрудников всех основных отделов, включая менеджеров по продажам, логистов, персонал бухгалтерии и руководство, используются персональные компьютер. Конфигурация ПК включает процессор AMD Ryzen 5 6600, с объемом оперативной памяти 8 гигабайт. В качестве основных накопителей используются твердотельные накопители объемом 512-1024 гигабайт. На всех этих компьютерах установлена операционная система Microsoft Windows 10 Pro. Рабочие станции оснащены стандартным набором прикладного программного обеспечения: пакетом Microsoft Office, включающим широко используемые Excel для ведения табличного учета данных по клиентам, заказам и наличию товаров, Word для работы с документами и шаблонами, а также Outlook для корпоративной электронной почты, являющейся одним из основных каналов приема заказов и обмена информацией. Помимо этого, на рабочих местах установлены различные веб-браузеры для доступа к сети Интернет, программы-клиенты популярных мессенджеров для оперативной связи. Сотрудники работают с информацией на мониторах с диагональю от 21.5 до 24 дюймов.

Для обеспечения централизованного доступа к общим в компании установлен сервер. Этот сервер, представляет собой стандартное серверное оборудование с процессором Intel Xeon и объемом оперативной памяти 32 гигабайт. Дисковая подсистема сервера, организована с использованием RAID-массива для повышения надежности хранения данных и имеет общий объем в 12 терабайт. На сервере функционирует серверная операционная система на базе Linux. Основное назначение сервера в текущей конфигурации сводится к функциям файлового сервера, а также к поддержке корпоративной электронной почты и хостингу информационного веб-сайта компании.

Все компьютеры, сервер и прочие сетевые устройства в офисе объединены в единую локальную вычислительную сеть. Физическую связность обеспечивает сетевое оборудование, представленное коммутатор. Доступ компании к глобальной сети Интернет осуществляется через маршрутизатор, который, также выполняет функции базового межсетевого экрана для обеспечения сетевой безопасности.

В состав периферийного оборудования входят многофункциональные устройства (МФУ), расположенные в различных отделах. Эти устройства предоставляют сотрудникам возможность печати документов (счетов, отчетов, договоров) и сканирования бумажных материалов. МФУ являются сетевыми, обеспечивая доступ к функциям печати и сканирования из локальной сети.

Для ведения финансового учета в компании используется специализированное бухгалтерское программное обеспечение, работающее независимо от основных операционных данных о клиентах и заказах.

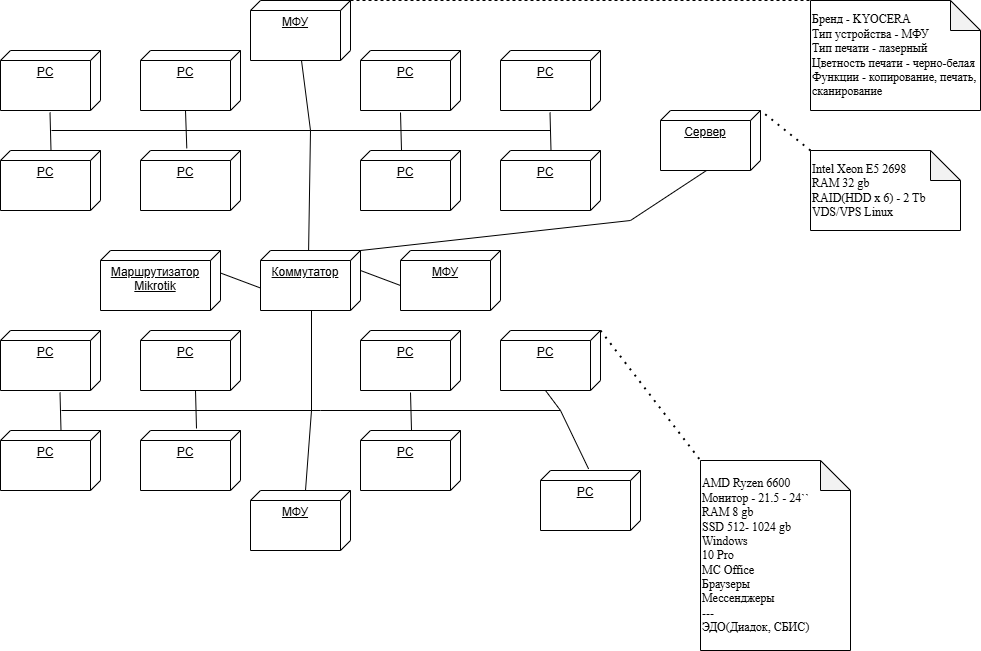


Рисунок 7. Аппаратное обеспечение

## **1.2 Проблемная область**

Проведенный анализ текущей деятельности компании «ИнтерТех» в области управления взаимоотношениями с клиентами и обработки заказов выявил ряд проблем, которые негативно сказываются на общей эффективности и потенциале роста. Основным корнем этих проблем является отсутствие централизованной и структурированной информационной системы. В настоящее время все ключевые данные о клиентах, их запросах, сформированных заказах, статусах выполнения и наличии товаров ведутся в разрозненных электронных таблицах Microsoft Excel и других несвязанных файлах. Как было показано на Диаграммах прецедентов и последовательности в состоянии "Как есть", сотрудникам приходится выполнять множество ручных операций по поиску и сверке информации в этих файлах, что приводит к дублированию, противоречивости данных, их потенциальной потере и значительным временным затратам на обработку даже простых запросов. История взаимодействия с клиентами фрагментирована, что затрудняет предоставление качественного и персонализированного сервиса.

Непосредственным следствием отсутствия централизованной системы является ручное и неэффективное управление полным жизненным циклом заказа. От приема заявки до отгрузки и формирования финальных отчетов процесс зависит от ручного выполнения шагов, подтверждений и передач информации между сотрудниками и отделами. Этот ручной характер операций значительно увеличивает вероятность ошибок на каждом этапе – от внесения данных заказа до расчета стоимости или передачи информации для доставки. Отсутствие автоматизированного отслеживания статусов заказов делает весь процесс непрозрачным; менеджеры вынуждены тратить дополнительное время на выяснение текущего состояния заказа у логистов, склада или бухгалтерии, что усложняет контроль и замедляет выполнение.

Ключевой проблемой также является низкая эффективность внутренних и внешних коммуникаций. Передача критически важной оперативной информации между отделами (например, подтверждение оплаты от бухгалтера менеджеру, или деталей заказа от менеджера логисту и складу) осуществляется через неформализованные каналы. Это приводит к задержкам в распространении информации, увеличивает риск недопонимания или потери данных и существенно замедляет общую скорость реакции на изменения или запросы клиентов. Отсутствие единой платформы для координации действий всех участников процесса делает взаимодействие громоздким и затрудняет оперативное решение возникающих вопросов.

Текущая ситуация характеризуется отсутствием инструментов для автоматизированной аналитики и значительными сложностями в формировании отчетности. Сбор данных для анализа эффективности работы, объема продаж, статусов заказов и других ключевых показателей требует ручного консолидирования информации из множества источников (Excel-файлов). Этот процесс трудоемок, занимает много времени и не позволяет получать актуальные данные для оперативного анализа и принятия своевременных управленческих решений. Ручное формирование отчетов также увеличивает вероятность ошибок и ограничивает возможности для стратегического планирования на основе объективных данных. Совокупность всех этих проблем создает серьезные барьеры для масштабирования бизнеса и повышения его конкурентоспособности.

## **1.3 Предложения по решению**

Для эффективного решения проблем и существенного повышения операционной эффективности компании «ИнтерТех» предлагается разработка и последующее внедрение специализированной информационной системы, получившей условное наименование «Один». Целью данной системы является трансформация текущих ручных и разрозненных процессов в интегрированную, автоматизированную и контролируемую систему управления.

Фундаментальным шагом станет создание централизованной базы данных. Система «Один» объединит всю критически важную информацию о клиентах, их полной истории взаимодействий, деталях заказов и их статусах, а также актуальные данные о наличии и характеристиках продукции в едином, структурированном хранилище. Это позволит исключить дублирование и противоречивость данных, обеспечит их доступность и актуальность для всех авторизованных сотрудников, значительно упростит поиск и анализ, а также предоставит менеджерам полную и целостную картину по каждому клиенту для повышения качества обслуживания.

Внедрение системы «Один» позволит автоматизировать ключевые бизнес-процессы, в первую очередь, связанные с обработкой и управлением заказами. Система обеспечит автоматическую регистрацию и маршрутизацию заявок, расчет стоимости, формирование стандартных документов (например, счетов). Централизованное отслеживание статусов заказов в реальном времени позволит всем участникам процесса видеть актуальное состояние выполнения, что повысит прозрачность, упростит контроль и координацию, а также существенно сократит время на ручные проверки и запросы. Автоматизация рутинных операций позволит минимизировать количество ошибок, связанных с человеческим фактором, и ускорить цикл выполнения заказа.

Внедрение централизованной системы «Один» обеспечит возможности для автоматизированного сбора данных и формирования аналитической отчетности. Хранение всех операционных данных в структурированном виде позволит системе автоматически строить различные аналитические срезы (по продажам, эффективности работы, динамике заказов и т.д.) и генерировать необходимые отчеты в реальном времени. Это предоставит руководству мощный инструмент для оперативного мониторинга, глубокого анализа деятельности компании и принятия обоснованных управленческих и стратегических решений на основе актуальных и достоверных данных. Таким образом, система «Один» призвана стать катализатором повышения эффективности всех ключевых процессов, снижения издержек, улучшения качества обслуживания и укрепления позиций компании «ИнтерТех» на рынке.

**1.4 Анализ средств проектирования и разработки**

Для реализации проекта по разработке информационной системы «Один» необходимо определить конкретные средства, которые будут использованы на этапах проектирования и непосредственной разработки. Выбор инструментария основан на поставленных задачах и целях курсовой работы.

Для проектирования будущей системы, включая моделирование бизнес-процессов в состоянии "Как должно быть" (TO-BE), построения диаграмм архитектуры системы и детальной проработки структуры базы данных (логической и физической модели), я буду использовать специализированное программное обеспечение для моделирования, такое как ERWin Process Modeler[[2]](#footnote-2) и онлайн утилиту Draw.io[[3]](#footnote-3). Проектирование структуры базы данных будет выполнено с использованием инструментов визуального моделирования БД в приложении MySQL Workbench[[4]](#footnote-4).

Для непосредственной разработки программной части системы «Один» я буду использовать следующий стек технологий и инструментарий: языком программирования будет Python, как универсальный и производительный язык, с использованием фреймворка или Flask[[5]](#footnote-5) для быстрой и структурированной разработки веб-приложения. В качестве системы управления базами данных (СУБД) для надежного хранения и управления информацией я буду использовать PostgreSQL, как мощное и распространенные реляционные СУБД. Код будет написан в интегрированной среде разработки PyCharm, которые предоставляют необходимые функции для написания, отладки и тестирования. Управление версиями исходного кода будет осуществляться с использованием системы Git.

Данный выбор средств позволит эффективно выполнить задачи проектирования, разработать функциональную систему «Один», соответствующую определенным требованиям, и обеспечить возможность ее последующего развития.

**Вывод по главе I**

В рамках первого этапа работы был проведен всесторонний анализ деятельности компании «ИнтерТех», связанной с управлением клиентами и заказами. Были детально изучены существующие бизнес-процессы, определены участники и их роли, смоделировано текущее состояние ("Как есть") с помощью диаграмм IDEF0, IDEF3 и UML прецедентов и последовательности. Выявлены ключевые проблемные области, обусловленные отсутствием централизованной информационной базы, ручным характером обработки заказов и неэффективностью коммуникаций, что подтверждается анализом текущего программно-аппаратного обеспечения. На основе этого анализа сформулированы предложения по решению проблем путем внедрения информационной системы «Один», определены основные функции этой системы и выбраны средства, которые будут использованы для ее создания.

На основе проведенного глубокого анализа текущей ситуации и четкого определения требований к будущему решению, сформирована необходимая база для перехода к следующему этапу проекта – непосредственному проектированию и реализации системы.

**Глава II. Проектирование и реализация информационной системы «Один»**

Проведенный в первой главе анализ текущего состояния деятельности компании «ИнтерТех» выявил существенные проблемы, связанные с управлением клиентами и заказами, обусловленные отсутствием централизованной информационной системы, разрозненностью данных и преобладанием ручных операций. Эти проблемы снижают общую эффективность работы, увеличивают операционные издержки и ограничивают возможности масштабирования. В ответ на выявленные недостатки была сформулирована задача по разработке и внедрению специализированной информационной системы «Один», призванной трансформировать существующие процессы.

**2.1 Целевая модель бизнес-процессов компании «ИнтерТех» после внедрения системы «Один»**

Для формализованного описания целевой модели бизнес-процессов были использованы диаграммы моделирования IDEF0. На верхнем уровне, Контекстная диаграмма A0 (TO-BE) отражает взаимодействие обновленной "Работы компании 'ИнтерТех'" с внешним окружением. Как и в модели "Как есть", процесс получает "Запрос клиента" и "Данные о заказе" в качестве входной информации и генерирует "Выполненный заказ" и "Отчёт" как результаты своей деятельности. Управляющие воздействия в виде "Прайс-листов" и "Регламентов" по-прежнему определяют правила выполнения процессов. Однако, ключевое и принципиальное отличие от модели "Как есть" проявляется в блоке механизмов. Если ранее основным инструментом учета и обработки данных являлись разрозненные "Таблицы Excel", то в целевой модели центральным механизмом, используемым всеми участниками процесса, становится "Система ОДИН". Человеческие роли – Менеджер по продажам, Складские сотрудники, Логист – остаются в списке механизмов, поскольку они продолжают выполнять свои функции, но теперь их работа осуществляется при непосредственном использовании и поддержке системы «Один». Отсутствие механизма "Таблицы Excel" на этой диаграмме верхнего уровня символизирует полный отказ от файлового учета как основного метода работы с оперативными данными в пользу централизованной информационной системы.

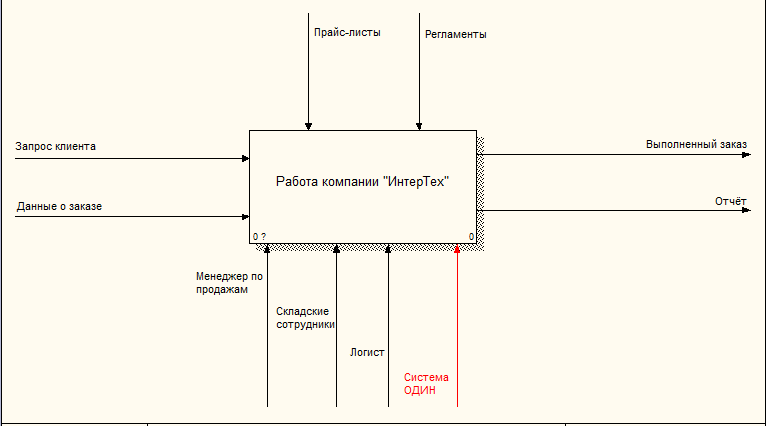


Рисунок 8. IDEF0 TO-BE

Детализация основного процесса A0 в целевой модели представлена на Декомпозиции диаграммы A0 (A1-A5) (TO-BE). Cодержание и способ выполнения каждого из этапов кардинально меняются благодаря системе «Один». На диаграмме это отражено подключением механизма "Система ОДИН" к каждому отдельному блоку подпроцесса. Это подчеркивает, что система является не просто одним из инструментов, а интегрированной платформой, поддерживающей выполнение всех ключевых действий на каждом этапе – от регистрации входящего запроса до формирования итогового отчета. Подобно Контекстной диаграмме, здесь полностью отсутствует механизм "Таблицы Excel", что означает перенос всего оперативного учета и обработки данных в новую систему.

Наиболее значимые изменения в Декомпозиции A0 (TO-BE) касаются информационных потоков, связывающих подпроцессы. Стрелки, идущие между блоками в целевой модели представляют собой не ручную передачу документов или файлов, а потоки данных и статусов, которые хранятся в централизованной базе данных системы «Один» и автоматически становятся доступными для следующих этапов процесса.

Например, после завершения обработки заявки в блоке A2, вся информация о готовом заказе и сформированном счете мгновенно фиксируется в системе и доступна для процесса A3 "Оплата продукции" без необходимости ручной передачи документов. Подтверждение оплаты в A3 обновляет статус заказа в системе, автоматически сигнализируя о готовности к отгрузке в A4. Информация об отгрузке, внесенная в систему в рамках A4, служит основой для автоматического формирования отчетов в A5. Управляющие воздействия, такие как "Прайс-листы" и "Регламенты", теперь также оказывают влияние непосредственно на логику работы системы «Один», обеспечивая автоматическое применение правил и актуальных цен.

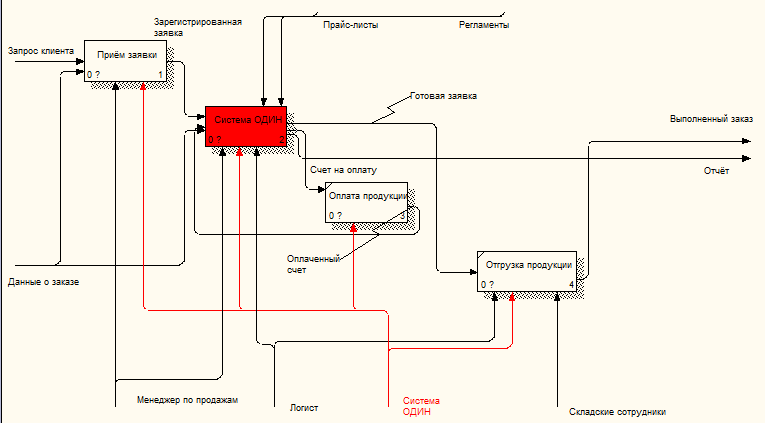
Представленные IDEF0 диаграммы целевой модели "Как должно быть" наглядно демонстрируют трансформацию бизнес-процессов из ручного, фрагментированного и основанного на файловом обмене состояния в интегрированную, автоматизированную и централизованную систему, где «Один» выступает в роли единого центра управления данными и координации рабочих потоков, что является ключевым условием для повышения эффективности и решения проблем, выявленных в первой главе.

Рисунок 9. Декомпозиции диаграммы A0 (A1-A5) (TO-BE)

**UML Диаграмма прецедентов в состоянии "Как должно быть" (TO-BE)**

В отличие от Диаграммы прецедентов "Как есть", которая отображала разрозненные ручные действия и взаимодействия между людьми с использованием различных инструментов, Диаграмма прецедентов "Как должно быть" демонстрирует чистый набор функций, предоставляемых самой системой. Среди ключевых прецедентов, реализованных внутри «ОДИН», выделяются: "Принять запрос" для формализованной регистрации входящих обращений; "Формирование отчета" для автоматического создания аналитических сводок; а центральным элементом является комплексный прецедент "Обработать заявку". Этот прецедент включает ряд автоматизированных под-функций: "Наличие продукции" (проверка по централизованным данным системы), "Данные клиента" (получение информации из единой базы), "Расчеты" (автоматический расчет стоимости), "Сформировать счет" (автоматическая генерация документа), "Согласовать сроки и условия" (фиксация деталей в системе). При необходимости, в процессе обработки заявки может быть расширяющий прецедент "Внести/обновить данные клиента", позволяющий корректировать информацию в единой базе. Дополнительные прецеденты системы включают "Внести данные о заказе", "Контроль оплаты" (для отслеживания), а также функции, связанные с этапами выполнения заказа: "Отслеживание изменения статуса", "Организация доставки" и "Контроль выполнения доставки" (для Логиста), "Подготовка продукции к отгрузке" и "Отгрузка" (для Складского сотрудника), "Получение и контроль счетов" (для Бухгалтера). Значительным улучшением, показанным этой диаграммой, является централизация всех операционных функций внутри системы «ОДИН», что заменяет ручные операции и поиск данных в несвязанных файлах (как было в AS-IS), делая взаимодействие пользователя с бизнес-процессом прямым и стандартизированным через единый интерфейс системы.

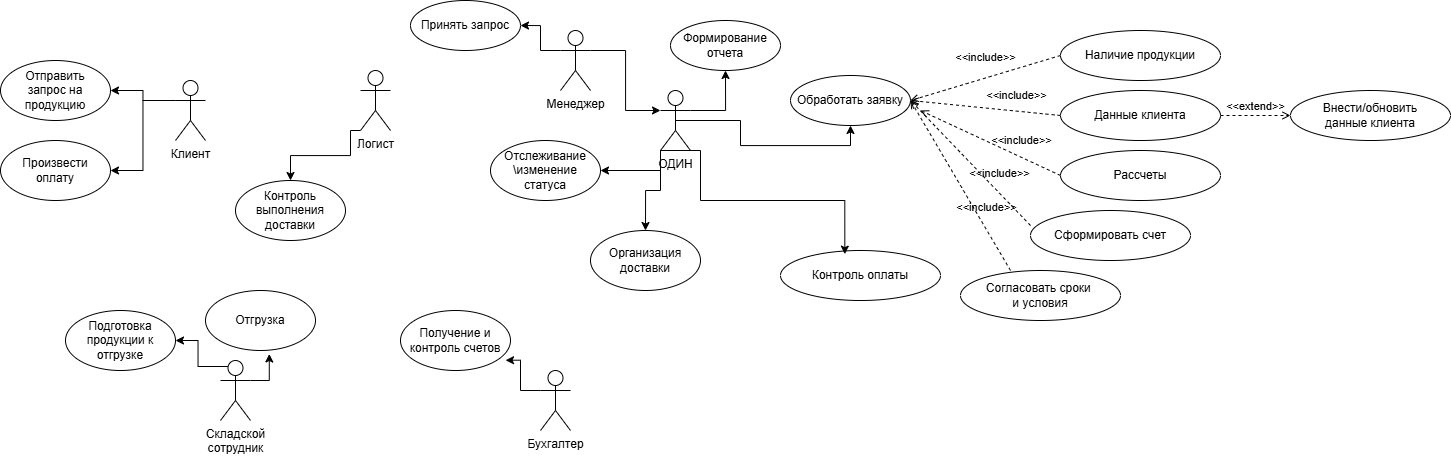


Рисунок 10. Диаграмма прецедентов в состоянии "Как должно быть"

**UML Диаграмма последовательности (Sequence Diagram) в состоянии "Как должно быть" (TO-BE)**

Сценарий обработки заказа в целевой модели демонстрирует радикальные изменения по сравнению с последовательностью "Как есть". После получения внешнего "Запроса на продукцию" от Клиента, Менеджер начинает работу, открывая пользовательский интерфейс и инициируя "Создание нового заказа" в системе «ОДИН». Система берет на себя большую часть рутинных операций: автоматически выполняет "занесение данных о заказе и о клиенте" в централизованную базу, производит "расчет" стоимости и "создание счета". Затем система передает готовый "Счет на оплату" Менеджеру (или напрямую Клиенту через интегрированный канал). После внешнего действия "Оплата" со стороны Клиента, информация о поступлении средств регистрируется Бухгалтером и система «ОДИН» получает "Подтверждение статуса оплаты". Система самостоятельно выполняет "Изменение статуса оплаты" в базе данных и, основываясь на этом, автоматически генерирует задачи или уведомления, передавая необходимую информация Логисту и Складскому сотруднику. Складской сотрудник и Логист выполняют свои действия ("Отгрузка"), фиксируя их выполнение в системе. Система «ОДИН» автоматически обновляет "Статус отгрузки", информирует Менеджера ("Информация об успешно выполненном заказе"), который завершает процесс, меняя "Статус заказа на Выполнен" в системе. Наконец, система автоматически выполняет "Формирование отчета" на основе собранных данных.

Главные улучшения, показанные этой диаграммой последовательности, заключаются в автоматизации переходов между этапами, исключении ручных "передач" информации между участниками и централизации актуального статуса заказа в системе «ОДИН». Система выступает как координатор, обеспечивая бесшовное протекание процесса и доступность информации для всех участников в реальном времени, что сокращает время выполнения заказа и минимизирует возможность ошибок.

**2.2 Проектирование базы данных информационной системы «Один»**

Внедрение информационной системы «Один» предполагает создание единого, надежного и структурированного хранилища всей операционной информации компании – данных о клиентах, товарах, заказах, поставках, платежах и взаимодействиях. Проектирование базы данных является фундаментальным шагом на пути к реализации системы, обеспечивающим целостность, актуальность и доступность данных для всех ее компонентов и пользователей. Этот процесс включает два основных этапа: логическое и физическое проектирование.

**2.2.1 Логическое проектирование базы данных**

Логическое проектирование базы данных фокусируется на определении структуры данных, необходимой для поддержки бизнес-процессов компании «ИнтерТех», а также на установлении связей между различными типами данных».

Согласно представленной логической модели, основными сущностями для системы «Один» являются:

* Клиенты: Сущность, представляющая компании или физические лица, с которыми работает «ИнтерТех». В таблице "Клиенты" хранятся основные атрибуты клиента: client\_id (уникальный идентификатор клиента), client\_name (название компании или ФИО), client\_type (тип клиента, например, юридическое или физическое лицо), phone (контактный телефон) и email (адрес электронной почты).
* Продукция: Сущность, описывающая товары – промышленное оборудование и запасные части. В таблице "Продукция" фиксируются атрибуты товара: article (артикул, вероятно, являющийся уникальным идентификатором продукции), name (наименование товара), current\_price (текущая цена за единицу) и current\_stock (текущий остаток на складе).
* Менеджеры: Сущность, представляющая сотрудников компании, ответственных за работу с клиентами и заказами (менеджеров по продажам). Таблица "Менеджеры" содержит атрибуты: manager\_id (уникальный идентификатор менеджера) и manager\_name (ФИО менеджера).
* Заказы: Центральная сущность, представляющая конкретный заказ клиента. Таблица "Заказы" хранит информацию о заказе: order\_number (номер заказа), created\_at (дата создания заказа) и total\_amount (общая сумма заказа). Также в таблице "Заказы" присутствуют атрибуты, реализующие связи с другими сущностями: client\_id, manager\_id и status\_id. В представленной модели, таблица "Заказы" также содержит атрибуты article и qty, что указывает на прямую связь заказа с одним артикулом продукции и заказанным количеством в этой упрощенной схеме. Атрибут status\_id говориь о наличие отдельной сущности "Статус заказа.

Между этими сущностями установлены связи (отношения), отражающие логические взаимосвязи в предметной области:

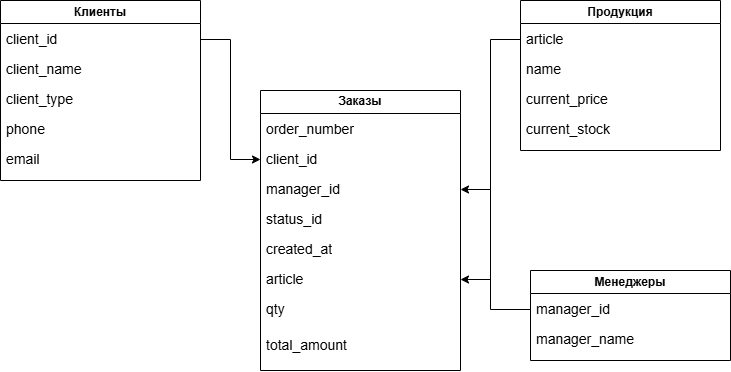
* Клиенты связаны с Заказами: Один Клиент может иметь много Заказов. Эта связь реализуется через наличие атрибута client\_id (идентификатор клиента) в таблице "Заказы", который ссылается на уникальный идентификатор в таблице "Клиенты".
* Менеджеры связаны с Заказами: Один Менеджер может отвечать за много Заказов. Эта связь реализуется через наличие атрибута manager\_id (идентификатор менеджера) в таблице "Заказы", который ссылается на уникальный идентификатор в таблице "Менеджеры".
* Продукция связана с Заказами: В представленной модели связь между Продукцией и Заказами реализована через включение атрибутов article (артикул продукции) и qty (количество) непосредственно в таблицу "Заказы".

Рисунок 11. Диаграмма классов

В результате логического проектирования формируется концептуальная схема данных, которая служит основой для следующего этапа – физического проектирования.

**2.2.2 Физическое проектирование базы данных**

На основе логической модели, каждая сущность трансформируется в таблицу в базе данных PostgreSQL. Атрибуты сущностей становятся столбцами в соответствующих таблицах с назначенными им типами данных PostgreSQL. Для обеспечения уникальной идентификации записей в каждой таблице определяется первичный ключ (PRIMARY KEY). Внешние ключи (FOREIGN KEY) используются для реализации связей между таблицами, ссылаясь на первичные ключи "родительских" таблиц и обеспечивая тем самым ссылочную целостность.

В качестве иллюстрации физической структуры базы данных, ниже представлена таблица с описанием ключевых таблиц, их основных столбцов, ролей ключей и типов данных, используемых в СУБД PostgreSQL:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Таблица | Столбец | Роль ключа | Тип данных (PostgreSQL) | Описание |
| clients | client\_id | PRIMARY KEY | SERIAL | Уникальный идентификатор клиента |
|  | client\_name |  | VARCHAR(255) / TEXT | Название компании или ФИО клиента |
|  | client\_type |  | VARCHAR(50) | Тип клиента |
|  | phone |  | VARCHAR(50) | Контактный телефон |
|  | email |  | VARCHAR(100) | Электронная почта |
| products | product\_id | PRIMARY KEY | SERIAL | Уникальный идентификатор товара |
|  | article | UNIQUE | VARCHAR(50) | Артикул товара |
|  | name |  | VARCHAR(255) | Наименование товара |
|  | current\_price |  | NUMERIC(10, 2) | Текущая цена товара |
|  | current\_stock |  | INTEGER / NUMERIC | Текущий остаток на складе |
| managers | manager\_id | PRIMARY KEY | SERIAL | Уникальный идентификатор менеджера |
|  | manager\_name |  | VARCHAR(255) | ФИО менеджера |
| orders | order\_id | PRIMARY KEY | SERIAL | Уникальный идентификатор заказа |
|  | order\_number | UNIQUE | VARCHAR(50) | Номер заказа |
|  | client\_id | FOREIGN KEY | INTEGER | Ссылка на клиента |
|  | manager\_id | FOREIGN KEY | INTEGER | Ссылка на менеджера |
|  | status\_id | FOREIGN KEY | INTEGER | Ссылка на текущий статус |
|  | created\_at |  | TIMESTAMP | Дата и время создания заказа |
|  | order\_article |  | VARCHAR(50) | Артикул продукции |
|  | order\_qty |  | NUMERIC(10, 2) | Заказанное количество |
|  | total\_amount |  | NUMERIC(10, 2) | Общая сумма заказа |

Физическая структура базы данных в PostgreSQL будет включать представленные таблицы и другие, необходимые для хранения всей требуемой информации с детально определенными столбцами, типами данных, ключами, индексами и ограничениями, обеспечивающими эффективное и надежное функционирование информационной системы «Один». Этот детальный план реализации базы данных служит непосредственным руководством для создания схемы базы данных и последующей разработки программной логики системы «Один», которая будет взаимодействовать с этим хранилищем данных.

**2.3 Создание моделей экранных форм информационной системы «Один»**

На основе разработанной логической модели базы данных и детализированных бизнес-процессов "Как должно быть", выполняется проектирование пользовательского интерфейса информационной системы «Один». Этот процесс включает создание макетов основных экранных форм, через которые пользователи будут получать информацию из системы, вводить новые данные, редактировать существующие и управлять процессами.

Цель – создать удобный, эффективный и интуитивно понятный интерфейс, который минимизирует время на выполнение операций и снизит вероятность ошибок, связанных с вводом данных или навигацией. Особое внимание уделяется проектированию ключевых форм, являющихся точками входа в систему и основными рабочими областями.

**2.3.1 Модель формы авторизации**

Форма авторизации является первой точкой взаимодействия пользователя с системой «Один». Ее основное назначение – обеспечить безопасный доступ к системе, идентифицировав пользователя и подтвердив его права. Модель формы авторизации (рис. 12) имеет простой и понятный дизайн, сфокусированный на выполнении единственной задачи – входа в систему.

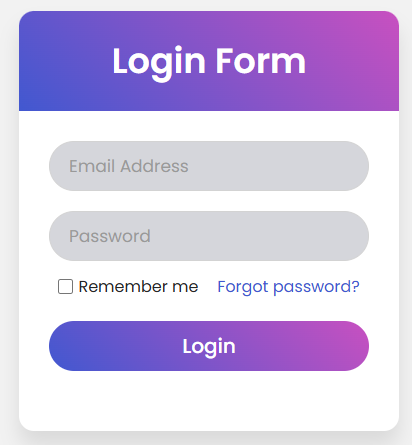
**2.3.2 Модель главного окна (Основной интерфейс системы «Один»)**

Рисунок 12. форма входа

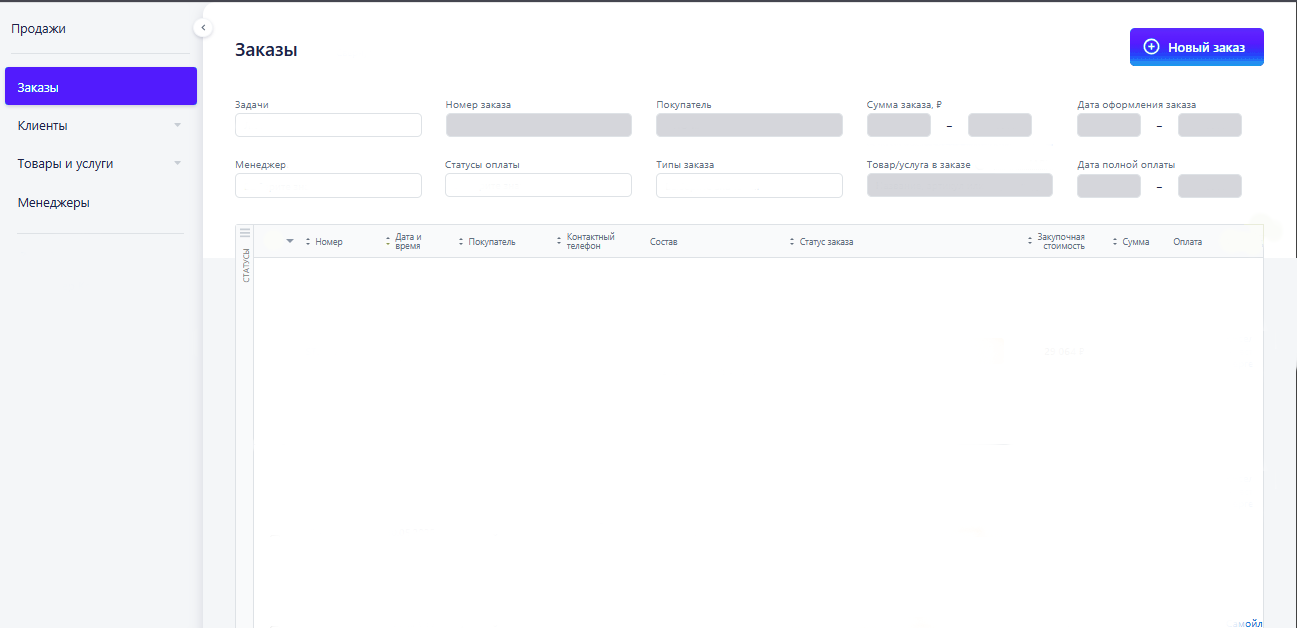
После успешной авторизации пользователь попадает в главное окно информационной системы «Один». Это основной рабочий интерфейс (рис. 13) через который осуществляется доступ ко всей функциональности системы, связанной с управлением клиентами, заказами, товарами и отчетностью. Главное окно системы имеет структурированный и, как правило, единообразный для всех разделов макет, что облегчает навигацию и освоение системы.

Рисунок 13. Модель главного окна

**2.3.3 Другие формы и шаблоны HTML**

Для реализации прецедента "Обработать заявку" и функции "Внести данные о заказе" в системе «Один» разработана модель экранной формы для создания нового заказа (рис. 14). Эта форма предназначена для Менеджеров и позволяет быстро ввести основную информацию о заказе, полученном от клиента.



Рисунок 14. HYML-аблон создание заказа

Для реализации прецедента "Формирование отчетности" и обеспечения руководства и менеджеров необходимой аналитической информацией разработана модель визуальной формы экрана для аналитики продаж (рис. 15) Данная форма представляет собой пример дашборда, сфокусированного на анализе источников привлечения клиентов и эффективности этапов бизнес-процесса.

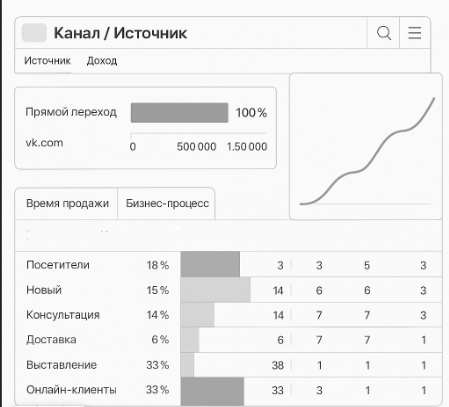


Рисунок 15. HTML новой формы обработки данных

**2.5 Программная реализация информационной системы «Один»**

На основе разработанной целевой модели бизнес-процессов, спроектированной структуры базы данных и моделей пользовательского интерфейса, была выполнена программная реализация основных компонентов информационной системы «Один». Целью этого этапа является создание рабочего прототипа или базовой версии системы, демонстрирующей возможность реализации ключевой функциональности и подтверждающей применимость выбранных средств и архитектурных решений. Часть исходного код программной реализации представлен в Приложениях 1-5 к данной курсовой работе. Полный проект в репозитории на GitHub.

Для реализации системы был выбран технологический стек, описанный в разделе 1.4, который включает:

* Python – как основной язык для серверной логики.
* Веб-фреймворк Flask – для быстрой разработки веб-приложения, управления маршрутизацией запросов, обработкой сессий и взаимодействием с HTTP-запросами.
* PostgreSQL – для надежного хранения всей операционной информации, спроектированной в разделе 2.2.
* HTML, CSS, JavaScript – для создания пользовательского интерфейса, его стилизации и добавления интерактивности в веб-браузере пользователя.

**Архитектура и структура приложения:**

Система имеет классическую клиент-серверную архитектуру. Клиентская часть реализована в виде веб-страниц, отображаемых в браузере пользователя. Эти страницы содержат HTML-структуру, стилизованную с помощью CSS, и JavaScript для динамических элементов интерфейса. Серверная часть написана на Python с использованием фреймворка Flask. Сервер обрабатывает запросы от браузера, выполняет необходимую бизнес-логику (включая взаимодействие с базой данных) и отправляет сформированные HTML-страницы или данные обратно клиенту. База данных PostgreSQL является центральным хранилищем данных, к которому обращается серверное приложение.

**Структура проекта организована логически:**

Основной исполняемый файл приложения (app.py), содержащий конфигурацию Flask, определение маршрутов, и логику обработки запросов.

Папка site (согласно конфигурации template\_folder в app.py) для хранения HTML-шаблонов.

Папка static для хранения статических файлов, таких как CSS-таблицы стилей.

**Реализация клиентской части:**

Пользовательский интерфейс системы реализован с использованием стандартных веб-технологий. HTML-файлы формируют структуру страниц и содержат элементы управления (поля ввода, кнопки, таблицы). Таблицы стилей CSS применяются для оформления этих элементов в соответствии с разработанными моделями экранных форм (раздел 2.3), обеспечивая единообразный внешний вид, цвета, шрифты и расположение элементов (как показано на примере формы авторизации и основной страницы с заказами). JavaScript используется для реализации интерактивных элементов, таких как сворачиваемое боковое меню или показ/скрытие блоков фильтров, что улучшает удобство использования интерфейса. (Приложение 5)

**Реализация серверной части:**

Серверное приложение на Flask обрабатывает входящие HTTP-запросы от веб-браузера. Маршруты приложения (/login, /, /prod, /clients и др.) определяются с помощью декоратора *@app.route*. Логика обработки запросов включает:

Реализацию маршрута /login для обработки GET-запросов и POST-запросов. Используется механизм сессий Flask для отслеживания состояния авторизации пользователя. Декоратор *@login\_required* применяется к защищенным маршрутам, гарантируя, что только аутентифицированные пользователи могут получить к ним доступ, перенаправляя неавторизованных пользователей на страницу логина.

На серверной стороне реализуется логика подключения к базе данных PostgreSQL и выполнения операций с данными (чтение, запись, обновление, удаление) в соответствии со спроектированной структурой БД (раздел 2.2). Например, при отображении списка заказов или клиентов, Flask-приложение выполняет запросы к соответствующим таблицам базы данных, получает данные и передает их в HTML-шаблон для отображения.

Серверное приложение выполняет бизнес-правила, описанные в целевой модели процессов (раздел 2.1). Например, при создании нового заказа, серверная логика проверяет корректность данных, обращается к базе данных клиентов и товаров, рассчитывает стоимость и сохраняет новую запись заказа со всеми деталями и начальным статусом.(Приложение 4)

Документация по проекту приложена в корневом репозитории на GitHub[[6]](#footnote-6).

**Заключение**

В рамках данной курсовой работы был проведен всесторонний анализ деятельности компании «ИнтерТех» в области управления взаимоотношениями с клиентами и обработки заказов. В первой главе была подробно охарактеризована предметная область в ее текущем состоянии, выявлены ключевые бизнес-процессы и участники, а также формализованы с помощью диаграмм. Анализ существующей программно-аппаратной инфраструктуры, а также смоделированные процессы и взаимодействия позволили выявить ряд критических проблем. Основной из них являются отсутствие централизованной системы, приводящее к разрозненности и потере информации, преобладание ручных операций в процессе обработки заказов, замедляющее работу и увеличивающее вероятность ошибок, а также низкая эффективность внутренних коммуникаций и отсутствие автоматизированной аналитики, затрудняющие оперативное управление и принятие решений.

Для решения выявленной проблемы было предложено разработать и внедрить специализированную информационную систему под условным названием «Один». Целью работы стала разработка проекта такой системы, способной повысить эффективность управления клиентами и заказами, сократить время обработки запросов и минимизировать количество ошибок.

Во второй главе была представлена практическая часть работы, посвященная проектированию системы «Один». На основе анализа "Как есть" и сформулированных требований была разработана целевая модель бизнес-процессов, в которой система «Один» занимает центральное место, интегрируя данные и автоматизируя рабочие потоки.

Ключевым этапом проектирования стала разработка структуры базы данных. Была спроектирована логическая модель БД, определяющая сущности предметной области (Клиенты, Продукция, Заказы, Менеджеры, Статусы и др.) и связи между ними. На основе логической модели было выполнено физическое проектирование для СУБД PostgreSQL, определены таблицы, столбцы, типы данных, первичные и внешние ключи, ограничения и индексы, обеспечивающие надежное хранение и целостность данных.

Также были созданы модели ключевых экранных форм пользовательского интерфейса – формы авторизации и основного рабочего окна системы. Это позволило наглядно представить, как пользователи будут взаимодействовать с системой и получать доступ к ее функциональности. В рамках демонстрации практической реализации были представлены базовые компоненты программного кода (SQL-скрипт создания БД, HTML/CSS/JS для интерфейса, основной Flask-сервер), показывающие основу для дальнейшей разработки.

## **Польза от реализации предложенного проекта**

Внедрение информационной системы «Один», спроектированной в данной работе, принесет компании «ИнтерТех» существенные выгоды:

Создание единой базы данных клиентов, заказов и товаров позволит избавиться от разрозненных файлов, обеспечит актуальность и доступность данных для всех сотрудников, сократит время на их поиск и исключит дублирование.

Автоматизация рутинных операций, таких как расчет стоимости заказа, формирование документов и отслеживание статусов, значительно сократит время обработки заявок и выполнения заказов.

Автоматизация процессов и стандартизация ввода данных в системе минимизируют человеческий фактор и связанные с ним ошибки в расчетах и документах.

Система станет единой платформой для обмена информацией и статусами заказа между отделами, повысит прозрачность и ускорит взаимодействие.

Наличие структурированных данных позволит автоматизировать формирование отчетов и проводить оперативный анализ эффективности работы, продаж и выполнения заказов.

Более быстрая и точная обработка запросов, доступность полной истории взаимодействий с клиентами позволят предоставлять более высокий уровень сервиса.

Система создаст надежный фундамент для роста компании, позволяя обрабатывать возрастающее количество заказов и клиентов без пропорционального увеличения ручной нагрузки.

## **Направления дальнейшего развития проекта**

Спроектированная в данной работе система «Один» представляет собой базовый набор функциональности для управления клиентами и заказами. Для развития проекта и расширения его возможностей можно выделить следующие направления:

Добавление новых модулей, таких как управление закупками и поставщиками, более детальный учет складских запасов с историей движений, модуль сервисного обслуживания, аналитика.

Детализация макетов всех экранных форм, разработка удобных дэшбордов для быстрого обзора ключевых показателей, создание мобильного приложения для.

Реализация интеграции с бухгалтерской системой (например, 1С) для автоматической синхронизации финансовых данных и формирования документов, интеграция с системами оплаты и банками, интеграция с сервисами доставки и транспортными компаниями для отслеживания статусов грузов, интеграция с корпоративной электронной почтой и мессенджерами для автоматической регистрации взаимодействий.

Внедрение ролевой модели доступа с детальной настройкой прав пользователей, усиление механизмов аутентификации, регулярное резервное копирование данных, обеспечение безопасности передаваемых данных.

Анализ и оптимизация запросов к базе данных, рефакторинг программной логики для повышения скорости работы при высоких нагрузках, переход на производственный сервер и отказоустойчивую архитектуру для обеспечения стабильной работы.

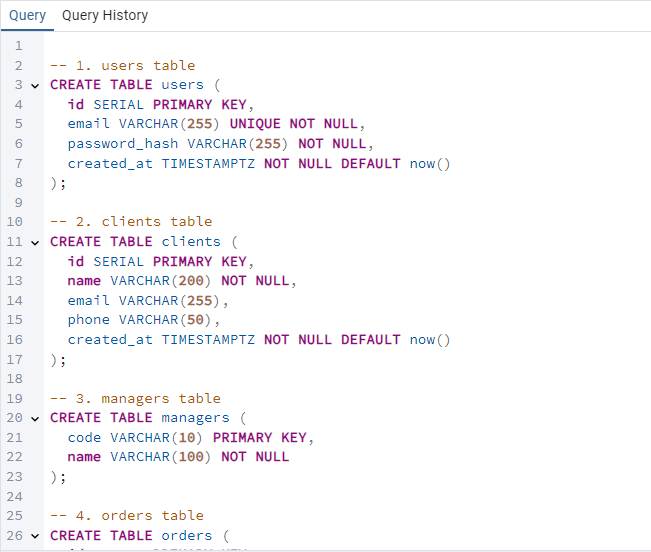
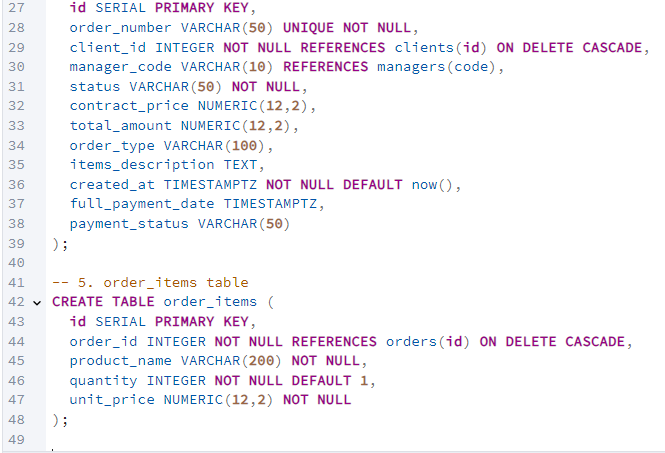
Разработка более сложных аналитических отчетов, инструментов визуализации данных, возможность конструктора отчетов, возможно, внедрение элементов бизнес-анализа (BI).

Создание подробных пользовательских инструкций, руководства администратора, проведение обучающих тренингов для сотрудников.

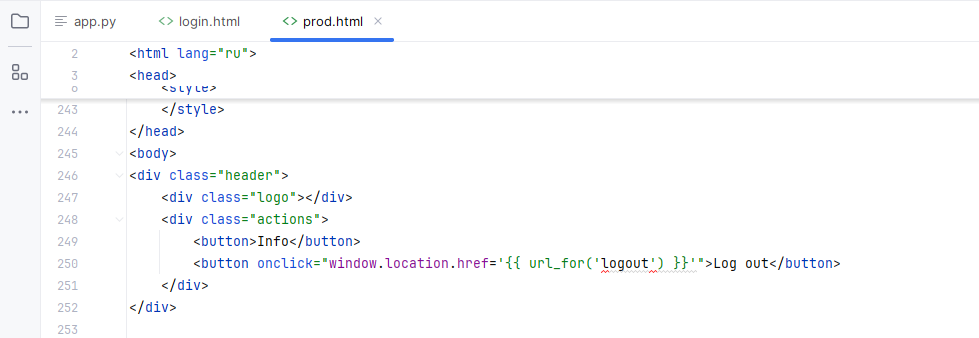
# Список использованных источников

1. Методика RACI: для чего используется и как составить матрицу ролей, задач и ответственности [Электронный ресурс] // developers.sber.ru. — 2024. — URL: <https://developers.sber.ru/help/business-development/raci-methodology>
2. Мега-Учебник Flask. Глава 1: Привет, мир! (издание 2024) [Электронный ресурс] / пер. Alex\_Mer5er // Habr. — 2024. — URL: <https://habr.com/ru/articles/804245/>
3. Долженко, А. И. Управление информационными системами : учебное пособие / А. И. Долженко. — 4-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2025. — 180 c. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/146409.html>
4. Грекул В.И. Управление внедрением информационных систем [Электронный ресурс]: учебное пособие/ Грекул В.И., Денищенко Г.Н., Коровкина Н.Л.— Электрон. текстовые данные.— М.: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2025.— 277 c.
5. Петрова, Е. А. Информационный менеджмент : учебник для вузов / Е. А. Петрова, Е. А. Фокина. — 2-е изд., стер. — Санкт-Петербург : Лань, 2024. — 144 с. — ISBN 978-5-507-49298-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/386036>
6. Граничин, О. Н. Информационные технологии в управлении : учебное пособие / О. Н. Граничин, В. И. Кияев. — 4-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2024. — 400 c.
7. Проектирование информационных систем : учебник и практикум для вузов / Д. В. Чистов, П. П. Мельников, А. В. Золотарюк, Н. Б. Ничепорук. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 273 с
8. Долганова, О. И.  Моделирование бизнес-процессов : учебник и практикум для вузов / О. И. Долганова, Е. В. Виноградова, А. М. Лобанова ; под редакцией О. И. Долгановой. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 322 с. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/560175> (дата обращения: 13.05.2025).

# Приложения

Приложение 1. Код базы данных

Приложение 2. prod.html(основа web-приложения(часть))

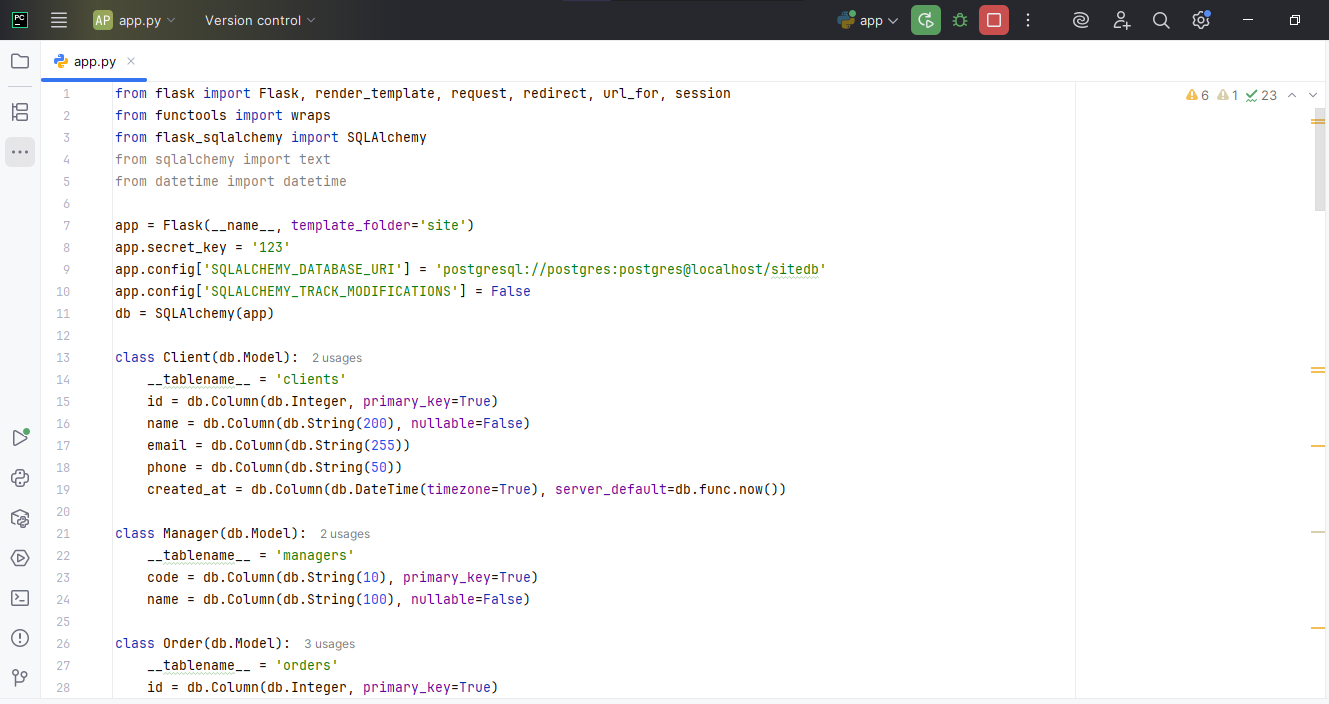




Приложение 3. Login.html



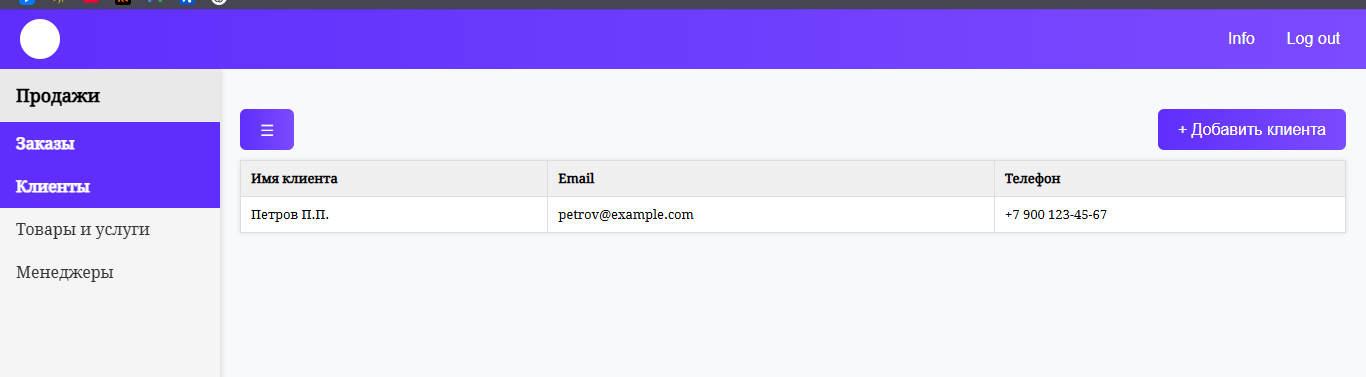
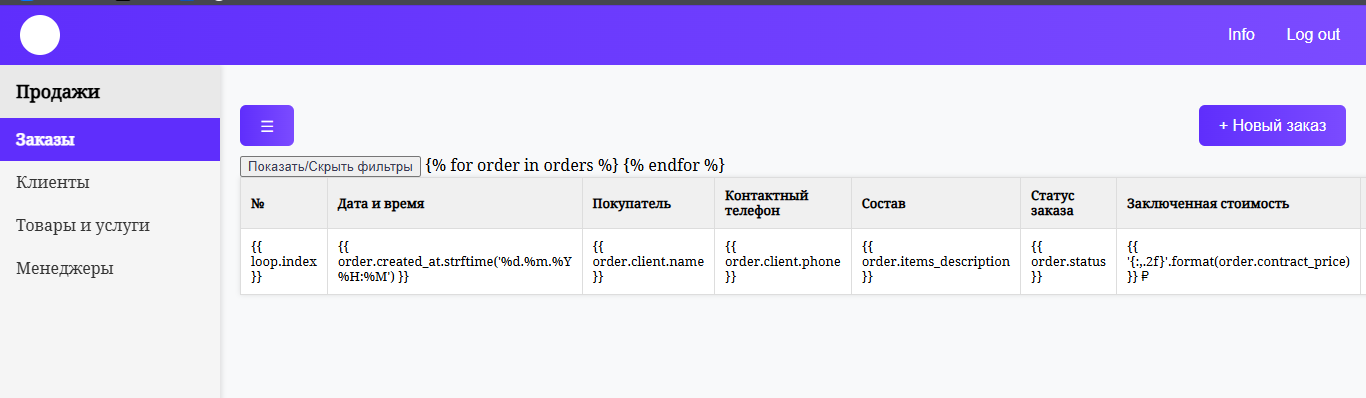
Приложение 4. Серверная часть(python)

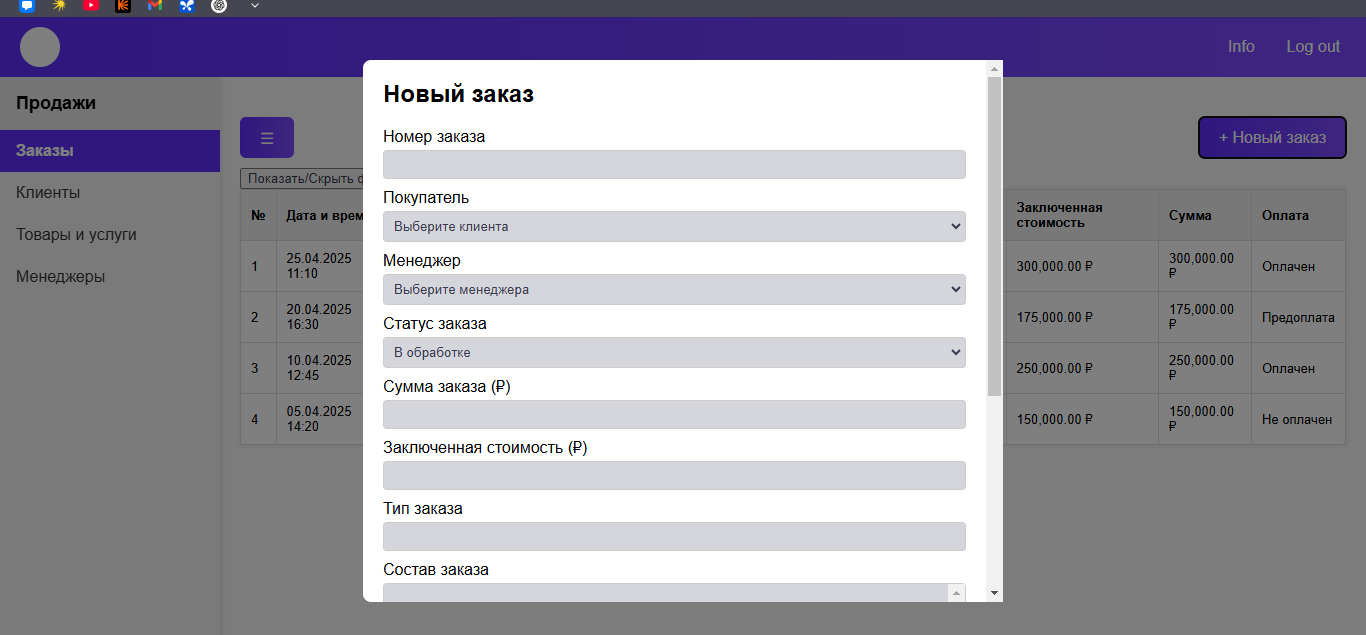


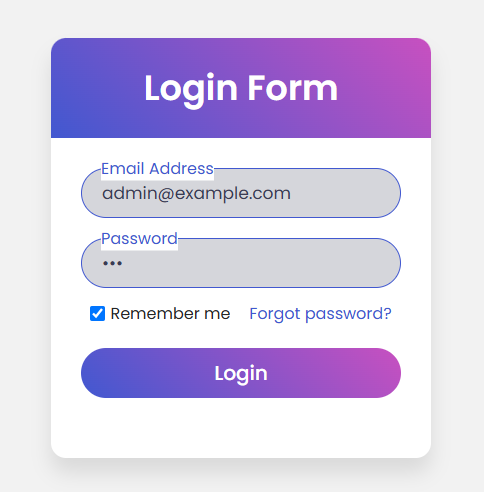




Приложение 5







Приложение 6

**Репозиторий на GitHub - https://github.com/Polubomy/kurs\_three**

1. **Методика RACI: для чего используется и как составить матрицу ролей, задач и ответственности** [Электронный ресурс] // developers.sber.ru. — 2024. — URL: <https://developers.sber.ru/help/business-development/raci-methodology> [↑](#footnote-ref-1)
2. https://softwar.ru/sistemy-dlja-biznesa-ofisa-i-nauchnoj-raboty/379-ca-erwin-process-modeler.html [↑](#footnote-ref-2)
3. https://www.drawio.com/ [↑](#footnote-ref-3)
4. https://www.mysql.com/products/workbench/ [↑](#footnote-ref-4)
5. Мега-Учебник Flask. Глава 1: Привет, мир! (издание 2024) [Электронный ресурс] / пер. Alex\_Mer5er // Habr. — 2024. — URL: <https://habr.com/ru/articles/804245/> [↑](#footnote-ref-5)
6. https://github.com/Polubomy/kurs\_three [↑](#footnote-ref-6)