|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования **«МИРЭА − Российский технологический университет»**  **РТУ МИРЭА** |

**Институт информационных технологий (ИИТ)**

**Кафедра практической и прикладной информатики (ППИ)**

ИТОГОВЫЙ ОТЧЕТ

по дисциплине «Анализ и концептуальное моделирование систем»

|  |  |  |  |
| --- | --- | --- | --- |
| Студент группы ИНБО-01-17 | *ИНБО-12-23. Албахтин И.В.* | (подпись) | |
| Преподаватель | *Акатьев Я.А.* | (подпись) | |
|  |  | |  | |

Москва 2025 г.

# Цели и задачи

**Цель работы:** изучить структуру и функционал рассматриваемой

информационной системы.

**Задачи:** Необходимо детально описать функционал системы в соответствии синдивидуальным вариантом учебного проекта.

**Индивидуальный вариант:**

Моделирование работы аптеки.

# 1.1 Предварительная информация

Аптечная сеть «Апрель» — одна из крупнейших в России. Компания работает на рынке фармацевтического ритейла уже 24 года и обслуживает свыше 25 миллионов держателей карт лояльности.

**Ассортимент и услуги:**

**Товары:** лекарственные препараты, медицинская техника, детские товары, средства гигиены и косметические продукты.

**Онлайн-сервисы:** поиск и бронирование товаров через официальный сайт и мобильное приложение.

**Программа лояльности «Апрель + Аптечный клуб»:** предоставляет скидки до 50% и возможность накопления баллов.

**Мобильное приложение:** Доступно для устройств на платформах iOS и Android, приложение «Аптека Апрель» позволяет:

* искать и бронировать товары;
* просматривать историю покупок;
* получать информацию о скидках и акциях;
* оплачивать заказы онлайн.

# 1.2 Описание объекта автоматизации

На данный момент основная проблема – **различие цен между сайтом и физическими аптеками** (рис.1), что создает неудобства для клиентов и требует дополнительной работы персонала.

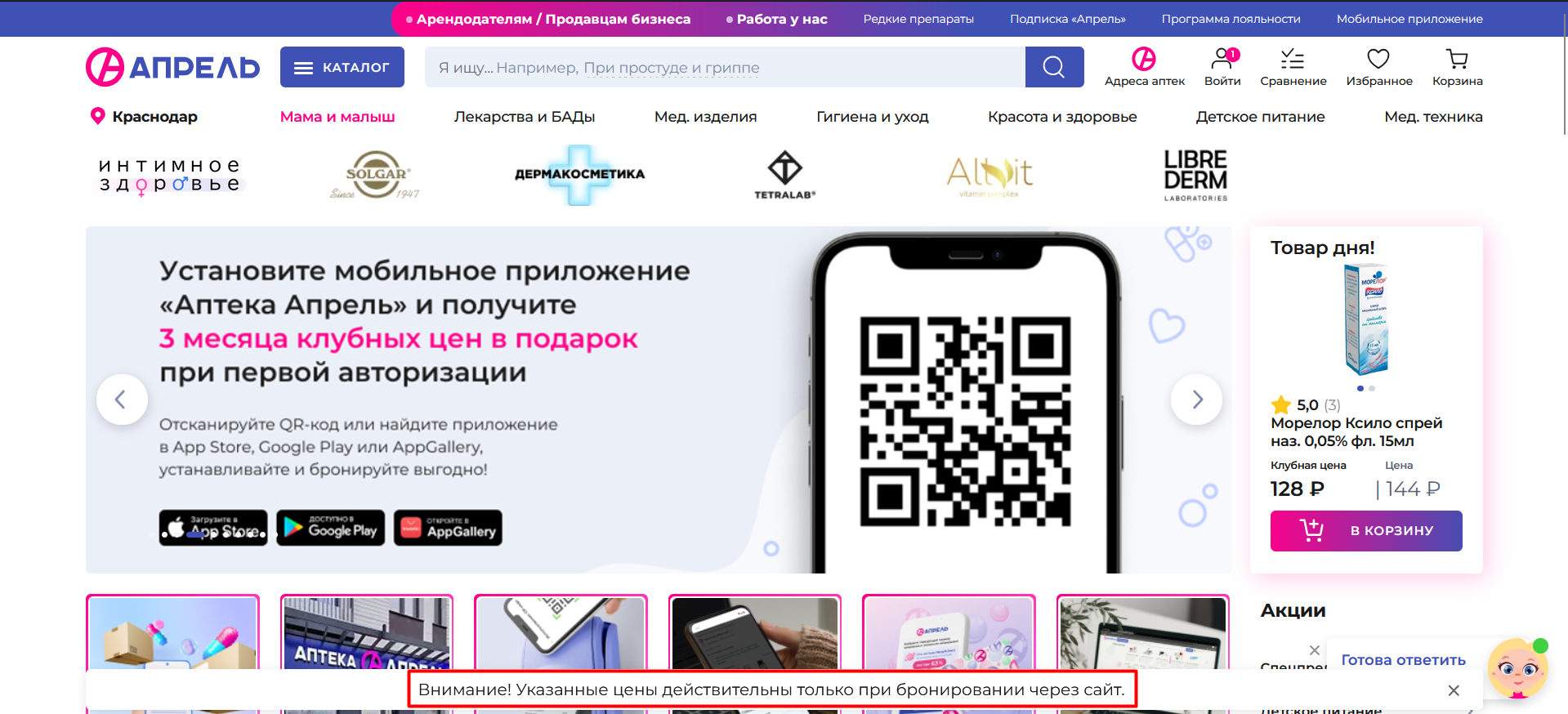


Рисунок 1 – Подтверждение проблемы, с помощью оповещения в нижней части экрана на официальном сайте «Апрель»

Для решения необходимо разработать систему автоматического обновления цен, которая синхронизирует данные в реальном времени.

***Таблица 1 - Описание объекта автоматизации***

|  |  |  |
| --- | --- | --- |
| **Компонент системы** | **Описание** | **Текущие проблемы** |
| **Физическая аптека** | Продажа лекарств и медицинских товаров, оформление заказов, программа лояльности | Различие цен между сайтом и кассами, необходимость ручного обновления цен |
| **Онлайн-сервис (сайт и приложение)** | Поиск, бронирование и оформление заказов, оплата онлайн, скидки и акции | Цены на сайте не всегда актуальны, возможны расхождения с ценами в аптеках |
| **Программа лояльности** | Система скидок и накопления баллов для постоянных клиентов | Требует интеграции с обновленной системой цен для корректной работы |

Продолжение таблицы

|  |  |  |
| --- | --- | --- |
| **Система управления запасами** | Автоматизированный учет товаров, складские остатки, поставки. | Возможны задержки в обновлении данных о наличии товаров в аптеках. |
| **База данных цен** | Хранение информации о стоимости товаров, обновление данных по поставкам. | Различные источники данных, что приводит к несоответствию цен. |

# 1.3 Описание основных функций системы

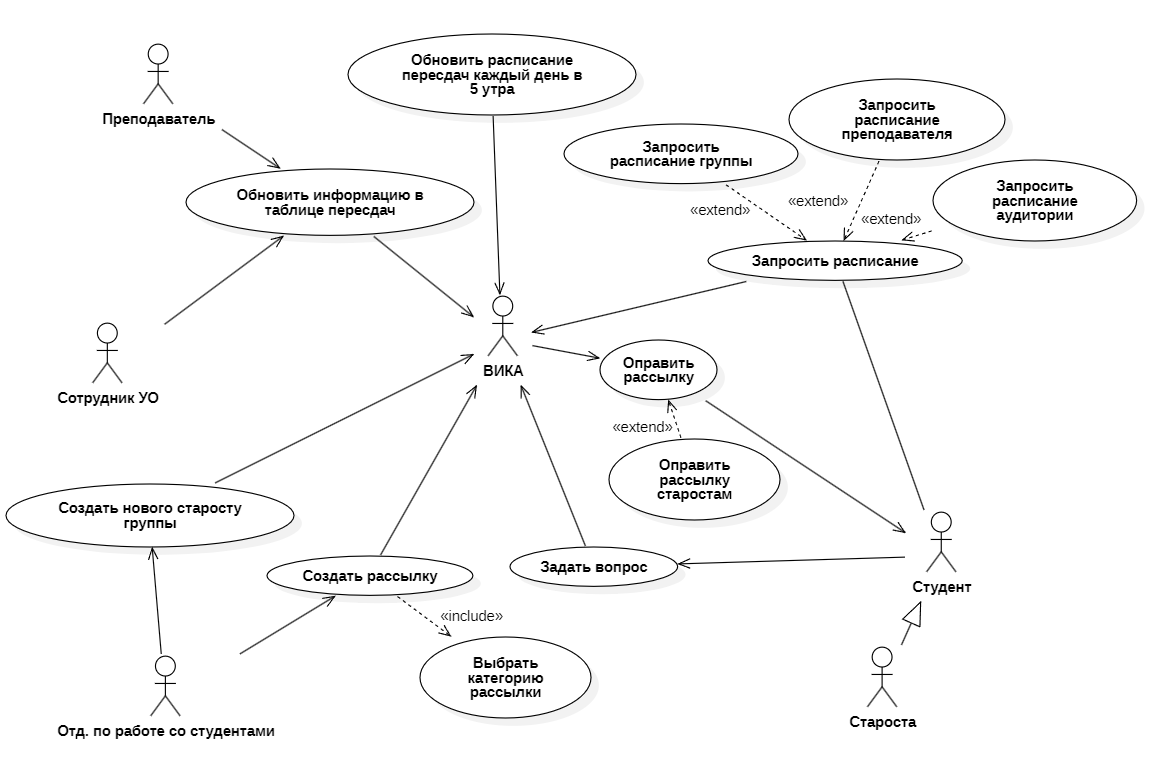
***Таблица 2 - Описание основных функций системы***

|  |  |
| --- | --- |
| **Наименование** | **Краткое описание** |
| Актуализация цен | Автоматическое обновление цен на сайте и в кассовых системах аптек, синхронизация с базой данных |
| Интеграция с базой данных | Централизованное хранение данных о ценах, наличии товаров, скидках и акциях |
| Управление складскими остатками | Учет поступлений, списания и перемещения товаров между аптеками |
| Отчетность и аналитика | Формирование отчетов по продажам, изменениям цен, популярным товарам |
| Уведомления для сотрудников | Оповещение персонала о необходимости обновления цен, изменениях в наличии товаров |

# 1.4 Ожидаемые результаты реализации моделируемой системы

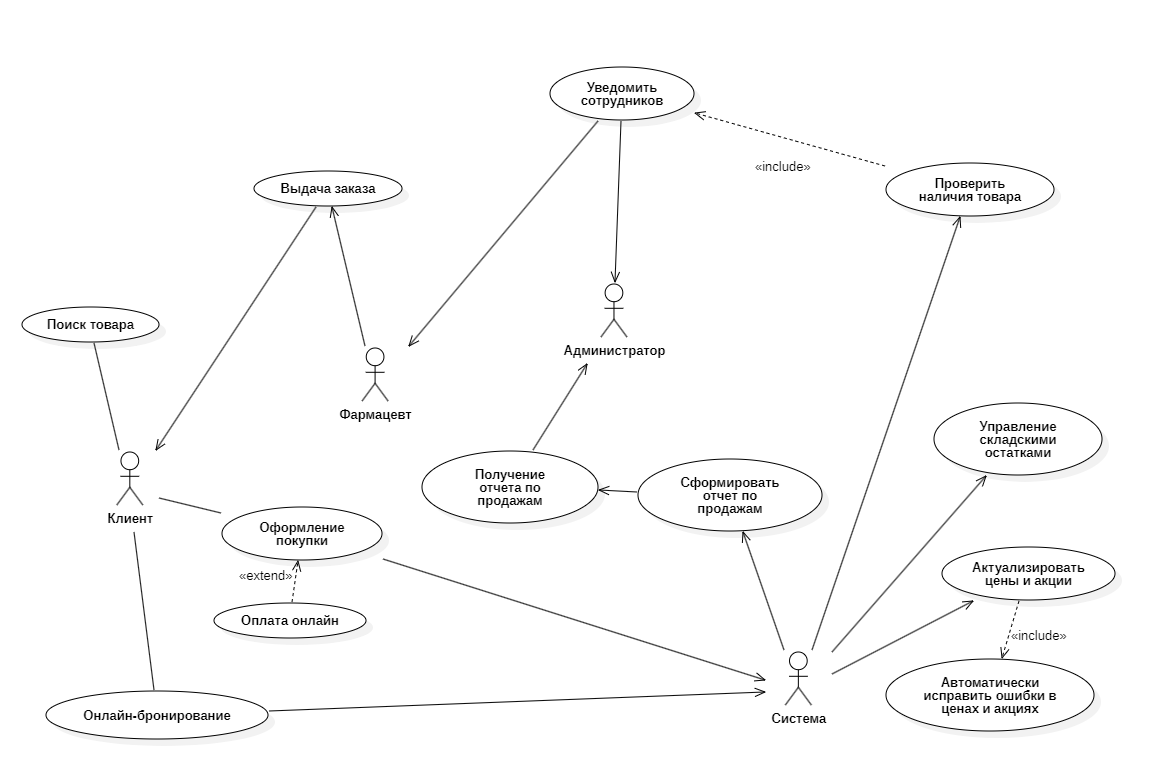
1. Исключение расхождений между ценами на сайте и в физической аптеке, а следовательно и повышение скорости обслуживания в аптеках.
2. Повышение доверия клиентов за счет точной информации о стоимости товаров.
3. Централизованное хранение актуальных данных о товарах, их наличии и ценах.
4. Снижение ошибок, связанных с ручным вводом информации.
5. Повышение эффективности управления запасами.
6. Возможность прогнозирования спроса и корректировки ассортимента.
7. Исключение необходимости ручного обновления цен.
8. Автоматическое уведомление сотрудников об изменениях в системе.

2.1 Построение диаграммы UML по вике



**Рисунок 2 – диаграмма UML по помощнику «ВИКА»**

2.2 Построение диаграммы UML по персональному варианту



**Рисунок 3 – диаграмма UML по автоматизации аптеки «Апрель»**

# 2.3 Описания взаимодействий по обеим диаграммам

***Таблица 3 - взаимодействий по UML диаграмме «ВИКА»***

|  |  |  |
| --- | --- | --- |
| **Актер/ВИ** | **Тип связи** | **Вариант использования** |
| Преподаватель | Ассоциация | |  | | --- | |  |   Обновить расписание пересдач каждый день в 5 утра |
| Сотрудник УО | Ассоциация | |  | | --- | |  |   Обновить информацию в таблице пересдач |
| Отдел по работе со студентами | Ассоциация | Создать нового старосту группы |
| Отдел по работе со студентами | Ассоциация | Создать рассылку |
| ВИКА | Ассоциация | Запросить расписание   |  | | --- | |  | |
| ВИКА | Расширение | Запросить расписание группы (расширяет «Запросить расписание») |
| ВИКА | Расширение | Запросить расписание преподавателя (расширяет «Запросить расписание») |
| ВИКА | Расширение | Запросить расписание аудитории (расширяет «Запросить расписание») |
| ВИКА | Ассоциация | Задать вопрос |
| ВИКА | Ассоциация | Отправить рассылку |
| ВИКА | Расширение | Отправить рассылку старостам (расширяет «Отправить рассылку») |
| ВИКА | Включение | Выбрать категорию рассылки (включается в «Создать рассылку») |
| Студент | Ассоциация | Запросить расписание |
| Студент | Ассоциация | Задать вопрос |
| Староста | Обобщение | Наследует функционал студента |

***Таблица 4 - взаимодействий по UML диаграмме «Апрель»***

|  |  |  |
| --- | --- | --- |
| **Актер/ВИ** | **Тип связи** | **Вариант использования** |
| Клиент | Ассоциация | Поиск товара |
| Клиент | Ассоциация | Онлайн-бронирование |
| Клиент | Ассоциация | Оформление покупки |
| Клиент | Расширение | Оплата онлайн (расширяет «Оформление покупки») |
| Фармацевт | Ассоциация | Выдача заказа |
| Администратор | Обобщение | Получение отчета по продажам |
| Система | Ассоциация | Управление складскими остатками |
| Система | Ассоциация | Проверка наличия товара |
| Система | Ассоциация | Актуализация цен и акций |
| Система | Включение | Уведомление сотрудников |
| Система | Включение | Автоматическое исправление ошибок в ценах и акциях(расширяет «Актуализация цен») |
| Система | Ассоциация | Сформировать отчет по продажам |

3.1 Построение тренировочных диаграмм классов UML

Изображение выглядит как диаграмма, линия, шаблон

Контент, сгенерированный ИИ, может содержать ошибки.

**Рисунок 4 – диаграмма классов анализа UML по помощнику «ВИКА»**

Изображение выглядит как текст, диаграмма, План, Технический чертеж

Контент, сгенерированный ИИ, может содержать ошибки.

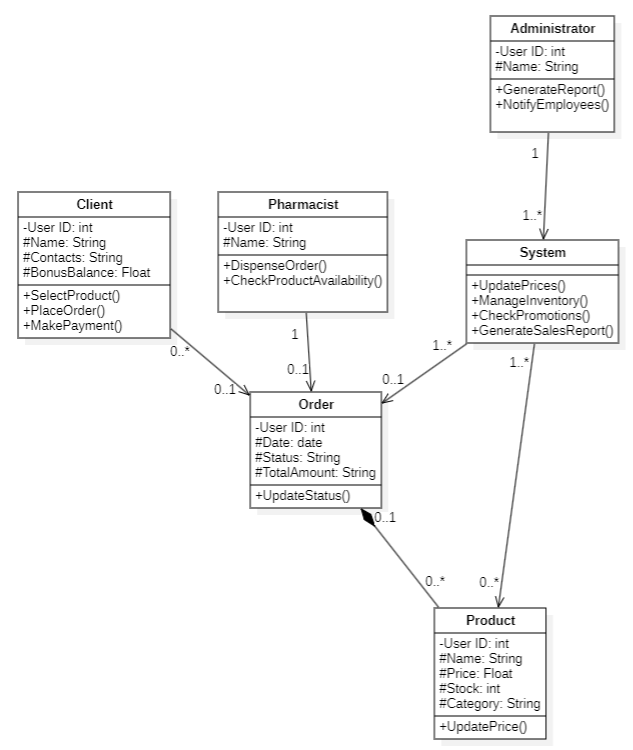
**Рисунок 5 – диаграмма классов UML по помощнику «ВИКА»**

3.2 Построение диаграмм классов по персональному варианту

Изображение выглядит как диаграмма, линия, График

Контент, сгенерированный ИИ, может содержать ошибки.

**Рисунок 6 – диаграмма классов анализа UML по автоматизации аптеки «Апрель»**



**Рисунок 7 – диаграмма классов UML по автоматизации аптеки «Апрель»**

3.3 По второй диаграмме (классов, не классов анализа) построить

***Таблица 5 – Взаимодействие между классов на примере VIKA***

|  |  |  |  |
| --- | --- | --- | --- |
| **Класс** | **Кратность** | **Тип отношения** | **Класс** |
| DialogWithVika |  | Обобщение | VKDialogWindow |
| VKGroup |  | Обобщение | VKDialogWindow |
| VKDialogWindow | \*, 1 | Ассоциация | Controller |
| Controller | 1, 0…\* | Ассоциация | Form |
| Controller | 1, 0…\* | Ассоциация | Question |
| Controller | 1…\*, 1 | Ассоциация | Schedule |
| Controller | 1, 0…\* | Ассоциация | Mailing |
| Mailing | 0…\*, 0…\* | Ассоциация | GroupLeader |
| GroupLeader | 0…1, 1 | Ассоциация | Group |

***Таблица 6 – Взаимодействие между классов в персональном варианте***

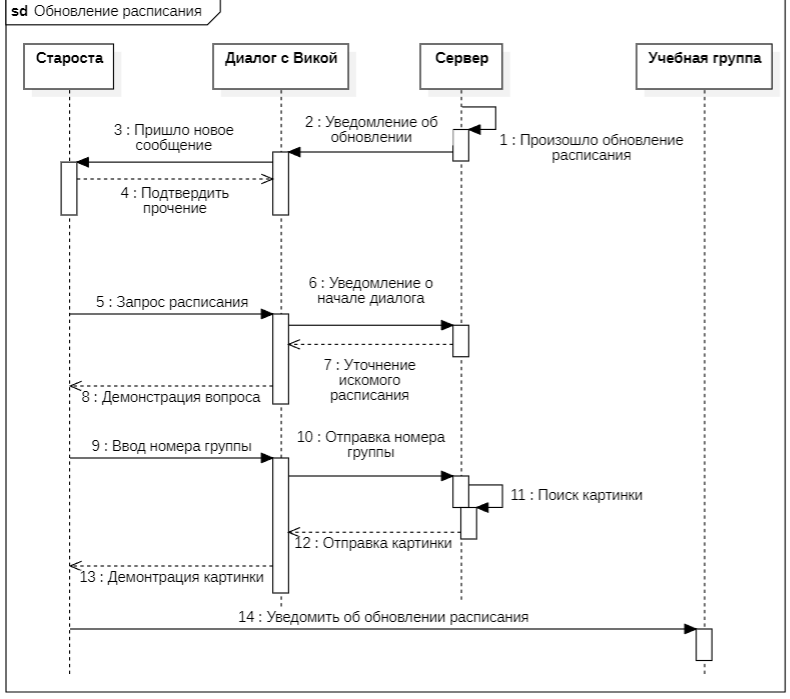
|  |  |  |  |
| --- | --- | --- | --- |
| **Класс** | **Кратность** | **Тип отношения** | **Класс** |
| Client | 0..\* | Ассоциация | Order |
| Pharmacist | 1 | Ассоциация | Order |
| System | 1..\* | Ассоциация | Order |
| Product | 0..\* | Композиция | Order |
| System | 1..\* | Ассоциация | Product |
| Administrotor | 1 | Ассоциация | System |

# 4.1 Тренировочные диаграммы последовательности.

Изображение выглядит как текст, диаграмма, снимок экрана, линия

Контент, сгенерированный ИИ, может содержать ошибки.

**Рисунок 8 – Тренировочная диаграмма последовательности №1**



**Рисунок 9 – Тренировочная диаграмма последовательности №2.**

# 4.2 Персональные диаграммы последовательности.

Изображение выглядит как текст, диаграмма, Параллельный, План

Контент, сгенерированный ИИ, может содержать ошибки.

**Рисунок 10 – Персональная диаграмма последовательности №1**

Изображение выглядит как текст, диаграмма, снимок экрана, План

Контент, сгенерированный ИИ, может содержать ошибки.

**Рисунок 11 – Персональная диаграмма последовательности №2**

Изображение выглядит как текст, снимок экрана, Параллельный, диаграмма

Контент, сгенерированный ИИ, может содержать ошибки.

**Рисунок 12 – Персональная диаграмма последовательности №3**

# 4.3 Персональная диаграмма коммуникации.

Изображение выглядит как текст, диаграмма, линия, Параллельный

Контент, сгенерированный ИИ, может содержать ошибки.

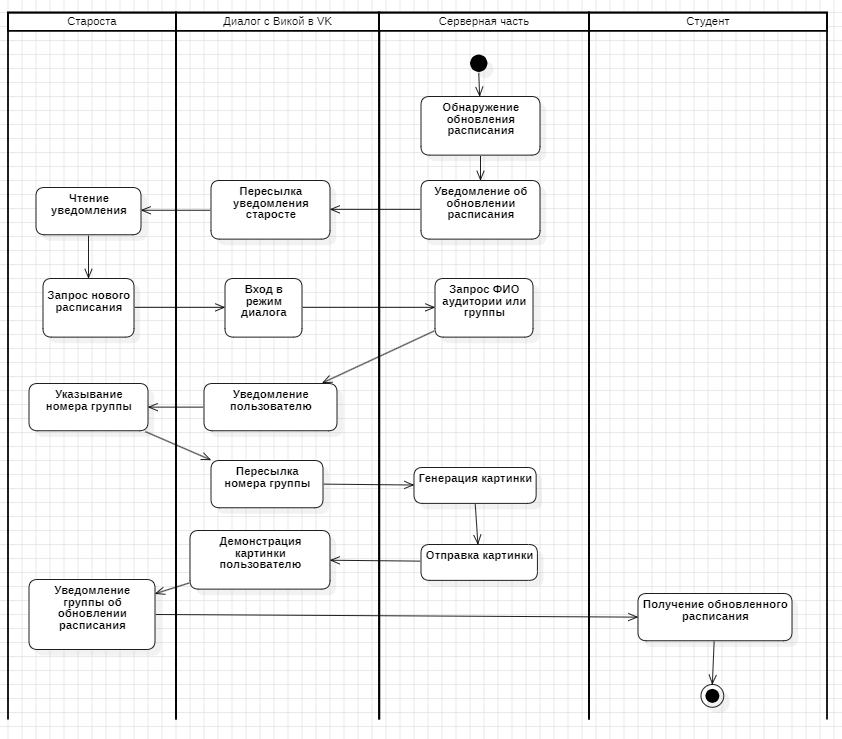
**Рисунок 13 – Персональная диаграмма коммуникации**

# 5.1 Тренировочные диаграммы последовательности.

Изображение выглядит как текст, снимок экрана, диаграмма, линия

Контент, сгенерированный ИИ, может содержать ошибки.

**Рисунок 14 – Тренировочная диаграмма деятельности №1**



**Рисунок 15 – Тренировочная диаграмма деятельности №2.**

# 5.2 Персональные диаграммы последовательности.

Изображение выглядит как текст, диаграмма, Параллельный, линия

Контент, сгенерированный ИИ, может содержать ошибки.

**Рисунок 16 – Персональная диаграмма деятельности №1**

Изображение выглядит как текст, диаграмма, снимок экрана, Параллельный

Контент, сгенерированный ИИ, может содержать ошибки.

**Рисунок 17 – Персональная диаграмма деятельности №2**

Изображение выглядит как текст, диаграмма, План, Параллельный

Контент, сгенерированный ИИ, может содержать ошибки.

**Рисунок 18 – Персональная диаграмма деятельности №3**

Изображение выглядит как текст, диаграмма, линия, Параллельный

Контент, сгенерированный ИИ, может содержать ошибки.

**Рисунок 19 – Персональная диаграмма состояний**

# 6.1 Тренировочная диаграмма компонентов по описанию.

Изображение выглядит как диаграмма, текст, План, Технический чертеж

Контент, сгенерированный ИИ, может содержать ошибки.

**Рисунок 20 – Тренировочная диаграмма компонентов по описанию**

# 6.2 Персональная диаграмма компонентов.

Изображение выглядит как диаграмма, План, Технический чертеж, текст

Контент, сгенерированный ИИ, может содержать ошибки.

**Рисунок 21 – Персональная диаграмма компонентов**

# 6.3 Текстовое описание персональной диаграммы компонентов напарника.

**Предметная область**  
Предметная область охватывает процессы онлайн-покупки лекарств через сайт аптеки "Апрель", автоматическую актуализацию цен и остатков товаров, обработку заказов и управление клиентскими данными. Система автоматизирует взаимодействие клиентов с веб-интерфейсом, складским учётом и бухгалтерией.

1. **Подсистема Pharmacy Website April (Веб-сайт аптеки "Апрель")**Это основная часть системы, через которую клиенты ищут товары, узнают актуальные цены и оформляют заказы.

**Компоненты**:  
**<<component>> :SearchEngine**  
Отвечает за поиск товаров. Через интерфейс Search Inventory отправляет запросы в складскую подсистему для получения данных об остатках.

**<<component>> :Pricing and Stock**  
Отвечает за получение новых цен от поставщиков и обновление информации о товарах. Передаёт новые прайс-листы в складскую подсистему.

**<<component>> :**ShoppingCart  
Обеспечивает работу корзины покупок, позволяет оформить заказ, взаимодействует с подсистемами управления заказами и клиентами через интерфейсы Manage Orders и Manage Customers.

**<<component>> :**Authentication  
Отвечает за регистрацию и авторизацию пользователей. Через интерфейсы Login и Register предоставляет пользователям доступ к системе.

**Основные интерфейсы:**  
ProductSearch – поиск товаров.  
New prices from Supplier – получение новых цен.  
Online Shopping – оформление покупок.  
Registration, Login – регистрация и вход в систему.  
Search Inventory, New Price List, Manage Orders, Manage Customers – взаимодействие с другими подсистемами.

1. **Подсистема Warehouses (Складской учёт)**  
   Отвечает за хранение данных о наличии товаров на складе и актуальные цены.

**Компоненты:**  
**<<component>> :Inventory**  
Обрабатывает запросы на поиск товаров и управляет складскими данными через интерфейс Search Inventory.

**<<component>> :Prices**  
Обновляет и хранит информацию о ценах на товары через интерфейс Price List.

**Основные интерфейсы:**Search Inventory – получение информации о товарах на складе.  
Price List – обновление прайс-листов от внешних источников.  
Manage Inventory – административные операции по учёту товаров.

1. **Подсистема Accounting (Учёт и заказы)**Ведёт обработку заказов клиентов и управление клиентской базой.

**Компоненты**:  
**<<component>> :Orders**  
Отвечает за приём и управление заказами, поступающими с сайта аптеки через интерфейс Manage Orders.

**<<component>> :Customers**  
Ведёт учёт зарегистрированных клиентов, обновляет и предоставляет данные через интерфейс Manage Customers.

**Основные интерфейсы**:  
Manage Orders, Orders – обработка заказов.  
Manage Customers – работа с клиентскими данными.

# 6.4 Текстовое описание персональной диаграммы компонентов.

**Предметная область**

Предметная область охватывает процесс приёма заявок на проверку автомобилей, извлечения и актуализации данных об их истории, проведения юридической оценки, формирования отчётов и информирования пользователей. Система автоматизирует работу менеджеров автосалонов, сотрудников юридического отдела, продавцов и покупателей автомобилей, а также интегрируется с внешними государственными базами данных (ГИБДД, ФНС, ФСПП**).**

1. **Подсистема Website “Rolf”**

Это пользовательский интерфейс, через который клиенты взаимодействуют с системой.

Компоненты:

* **<<component>> Authentication Service**  
  Отвечает за обработку операций входа (Login) и регистрации (Register). Принимает запросы через интерфейсы Login и Registration, выполняет валидацию и выдаёт токен/сессию пользователю.
* **<<component>> Request Handler**  
  Обрабатывает входящие запросы клиентов (Client’s request) на проверку данных об автомобиле. Принимает данные из веб-формы и формирует внутренний объект «Submit Check Request» для передачи в подсистему бизнес-логики.
* **<<component>> Web form**  
  Обеспечивает представление результата проверки отчёта (Viewing the report) в браузере клиента**.**

**Интерфейсы:**

* Публичные (user-facing):
  + Login
  + Registration
  + Request
  + Viewing the report
* Взаимодействие с Application Services:
  + Authenticate User (uses → Authentication Service)
  + Submit Check Request (uses → Request Handler)

1. **Подсистема Application Services**

Содержит бизнес-логику по интеграции с внешними сервисами и генерации отчётов.

**Компоненты:**

* **<<component>> Gov Integration Service**Выполняет запросы к государственным базам данных по интерфейсу Fetch External Data. Принимает аутентифицированный запрос, преобразует его в формат API внешних служб и возвращает сырые данные.
* **<<component>> Report Generator**Обрабатывает поступившие данные (Submit Check Request), запрашивает у Gov Integration Service необходимую информацию и формирует конечный отчёт. Предоставляет готовый отчёт по интерфейсу Read for Reports.

**Интерфейсы:**

* Требуемые от Website “Rolf”:
  + Authenticate User
  + Submit Check Request
* Публичные (для Data Stores и внешних компонентов):
  + Fetch External Data (предоставляет Gov Integration Service)
  + Read for Reports (предоставляет Report Generator)

**3. Подсистема Data Stores**

Хранит и предоставляет данные из внешних государственных баз.

**Компоненты:**

* **<<component>> External Gov DBs**Представляет собой набор удалённых баз данных, содержащих информацию об автомобилях. Предоставляет интерфейс Car Information для отдачи готовых данных по запросу Fetch External Data**.**

**Интерфейсы:**

* Требуемые от Application Services:
  + Fetch External Data
* Публичные (для Sales Manager):
  + Car Information

Изображение выглядит как диаграмма, План, Технический чертеж, зарисовка

Контент, сгенерированный ИИ, может содержать ошибки.

**Рисунок 22 – Диаграмма компонентов, построенная по описанию напарника**

# 6.5 Анализ почему диаграммы получились разные.

Диаграмма компонентов, построенная по текстовому описанию, может отличаться от оригинальной из-за обобщений и неполноты информации в тексте. Некоторые связи между компонентами могут быть интерпретированы по-разному, а детали, такие как дополнительные интерфейсы или вспомогательные компоненты, могут быть упущены. Роли компонентов также могут быть поняты иначе, что приводит к их объединению или разделению. При восстановлении структуры возможны разные решения в группировке подсистем и выборе уровня детализации, что влияет на итоговый вид диаграммы.

# 7.1 Требования к развёртыванию системы

**1. Серверная инфраструктура компании:**

* **Веб-сервер (Web Server)**
  + Назначение: размещение клиентской части веб-приложения (HTML/JS/CSS)
  + Развёртывание: физический сервер в офисе или виртуальная машина
  + Порты: 80 (HTTP), 443 (HTTPS)
* **Сервер приложений (Application Server)**
  + Назначение: обработка логики (модули работы с товарами, ценами, API)
  + Развёртывание: отдельная виртуальная машина или Docker-контейнер
  + Взаимодействие: с веб-сервером, поставщиком и БД
* **Сервер базы данных (Database Server)**
  + Назначение: хранение данных о товарах, ценах, заказах и отчетах
  + База: PostgreSQL / MySQL
  + Развёртывание: локальный сервер или облако (например, Azure Database, Amazon RDS)

**2. Облачные/внешние сервисы:**

* **Поставщик (Supplier Node)**
  + Назначение: отправка обновлённого прайса (JSON, XML, API)
  + Тип подключения: внешняя API-интеграция по HTTPS
  + Пример: автоматическая выгрузка файла раз в сутки
* **Сервис генерации отчетов (Report Generation Service)**
  + Назначение: формирование аналитических отчётов
  + Развёртывание: облачный микросервис или локальное приложение
  + Связь: доступ к базе данных и API

**3. Виртуальные машины / контейнеризация:**

* Все основные компоненты могут быть развернуты:
  + как отдельные **виртуальные машины (VM)** (например, в VMware, Hyper-V)
  + либо как **контейнеры (Docker)** с оркестрацией через Kubernetes
* Предпочтительно использование **облачной платформы** (например, AWS, Azure, Yandex Cloud) для масштабируемости и отказоустойчивости

**4. Клиентская сторона:**

* Пользователь взаимодействует через браузер
* Не требует установки ПО
* Поддержка HTTPS, адаптивная вёрстка

# 7.2 Персональная диаграмма развертывания

Изображение выглядит как текст, диаграмма, План, зарисовка

Контент, сгенерированный ИИ, может содержать ошибки.

**Рисунок 23 – Диаграмма развертывания по персональному варианту**

# 7.3 Персональная диаграмма пакетов.

Изображение выглядит как текст, диаграмма, линия, снимок экрана

Контент, сгенерированный ИИ, может содержать ошибки.

**Рисунок 24 – Диаграмма пакетов по персональному варианту**

# Вывод

В ходе выполнения практических работ были всесторонне изучены структура, функционал и особенности информационной системы организации работы аптеки «Апрель», а также системы-помощника «ВИКА». Основной акцент был сделан на освоении и применении инструментов UML-моделирования, которые доказали свою эффективность в визуализации, анализе и проектировании сложных систем.

Каждая практика способствовала закреплению специфических навыков: от построения диаграмм классов, последовательности, коммуникации, деятельности, компонентов и развертывания до анализа их структуры и интерпретации. В результате были созданы детализированные UML-диаграммы, отражающие ключевые сущности, их взаимосвязи, процессы и архитектуру системы «Апрель», включая серверы, клиентские устройства и внешние сервисы. Эти диаграммы обеспечили наглядное представление системы, что является критически важным для проектирования и разработки программного обеспечения.

Основная проблема системы «Апрель» была выявлена, а полученные навыки моделирования и анализа позволили глубже понять принципы взаимодействия компонентов и узлов. Освоенные техники UML, включая ассоциации, обобщения и композиции, а также правила построения различных типов диаграмм, создали прочную основу для дальнейшей работы со сложными информационными системами. Полученные знания и практические навыки будут полезны для проектирования, оптимизации и анализа систем в профессиональной деятельности, подчеркивая значимость системного подхода и визуализации в разработке программного обеспечения.