Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина»
Институт радиоэлектроники и информационных технологий – РТФ
Школа профессионального и академического образования

ДОПУСТИТЬ К ЗАЩИТЕ ПЕРЕД ГЭК

Директор ШПиАО
_____ Д.В. Денисов
(подпись) (Ф.И.О.)
«03» июня 2024 г.

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

ИССЛЕДОВАНИЕ ВОЗМОЖНОСТЕЙ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ В ЗАДАЧАХ УПРАВЛЕНИЯ И ОБМЕНА ДАННЫМИ В МЕЖДУНАРОДНОЙ ЛОГИСТИКЕ

Научный руководитель: Коломыцева Анна Олеговна
к. э. н. доцент
_____
подпись

Нормоконтролер: Огуренко Егор Владимирович
_____
подпись

Студент группы: РИМ-220963 Мухачев Иван Александрович
_____
подпись

Екатеринбург
2024

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего образования
**«Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина»**

Институт радиоэлектроники и информационных технологий – РТФ
Школа профессионального и академического образования
Направление подготовки 09.04.01 Информатика и вычислительная техника
Образовательная программа 09.04.01/33.03 Инженерия машинного обучения

ЗАДАНИЕ

на выполнение выпускной квалификационной работы

студента *Мухачева Ивана Александровича* **группы** *РИМ-220963*
(фамилия, имя, отчество)

1. Тема выпускной квалификационной работы
Исследование возможностей методов машинного обучения в задачах управления и обмена данными в международной логистике
Утверждена распоряжением по институту от «4» декабря 2023 г. № 33.02-05/298

2. Научный руководитель
Коломыцева Анна Олеговна к.э.н, доцент
(Ф.И.О., должность, ученая степень, ученое звание)

3. Исходные данные к работе
Техническое задание, описание работы сервиса, научная статья

4. Перечень демонстрационных материалов
Презентация, архитектура прототипа системы.

5. Календарный план

№ п/п	Наименование этапов выполнения работы	Срок выполнения этапов работы	Отметка о выполнении
1.	*1 раздел (глава)*	до 23.03.2024 г.	выполнено
2.	*2 раздел (глава)*	до 29.04.2024 г.	выполнено
3.	*3–4 раздел (глава)*	до 20.05.2024 г.	выполнено
4.	*ВКР в целом*	до 24.05.2024 г.	выполнено

Научный руководитель Коломыцева Анна Олеговна
Ф.И.О.
(подпись)

Студент задание принял к исполнению 01 декабря 2023 г.
дата
(подпись)

6. Допустить Мухачева Ивана Александровича к защите выпускной квалификационной работы в экзаменационной комиссии

Директор ШПиАО (подпись) Д.В. Денисов
Ф.И.О.

РЕФЕРАТ

Выпускная квалификационная работа магистра 70 стр., 5 рис., 30 источник, 3 прил.

Ключевые слова: международная логистика, управление данными, информационные системы, автоматизация, интеграция данных,

Цель работы – разработка прототипа системы управления и обменом данными в международной логистике.

Предметом исследования является модели интеграции и обмена данными в интегрированных логистических процессах.

Объектом исследования является управление и интеграция данными в логистике.

Результатом проведенного исследования стала разработка прототипа системы.

СОДЕРЖАНИЕ

РЕФЕРАТ 4

ВВЕДЕНИЕ 7

1.1 Обзор существующих систем управления процессами и обменом данными операций в международной логистике 10

1.2 Систематизация, классификация основных этапов логистических операций в международной логистике, в интегрированных логистических процессах 13

1.3 Основные преимущества и недостатки для управления процессами и обменом данными в интегрированных логистических процессах 19

1.4 Основные потребности и требования для управления процессами и обменом данными в интегрированных логистических процессах 21

1.5 Технологии и методы машинного обучения, применяемые в логистике 22

1.6 Классификация методов управления и обменом данными в международной логистике 26

2.1 Анализ текущей системы управления и обменом данными в международной логистике 28

2.2 Сравнение существующих систем и прототипа системы 32

2.3 Требования к прототипу ILDMES 35

2.4 Архитектура ILDMES 38

2.5 Описание технической части архитектуры 40

2.6 Разработка микросервиса в ILDMES 43

2.6.1 Требования к микросервису 44

2.6.2 Архитектура системы 45

2.6.3 Компоненты и технологии 45

2.6.4 Детали реализации 46

3.1 Архитектура программно-аппаратного комплекса. 47

3.2 Существующие аналоги и их характеристики 47

3.3 Эксперимент на независимых данных 49
3.4 Тестирование прототипа 50
3.5 Перспективы совершенствования инструмента 52
3.6 Условия для остановки автоматизированного решения 54
3.7 Экономическое обоснование внедрения системы-прототипа 55
ЗАКЛЮЧЕНИЕ 57
БИБЛИОГРАФИЧЕСКИЙ СПИСОК 61
Приложение А 64
Приложение В 67
Приложение С 69

ВВЕДЕНИЕ

Международная логистика является одной из ключевых областей современной экономики, которая обеспечивает эффективное перемещение товаров и услуг между странами. В настоящее время сфера международной логистики становится все более важной и сложной, требуя эффективных инструментов управления и обмена данными для обеспечения эффективности и гибкости логистических процессов. Оптимизация процессов управления и интеграции данных в логистике напрямую влияет на конкурентоспособность компаний на рынке. Однако, управление и обмен данными в этой области часто сталкиваются с проблемами, связанными с разнообразием форматов данных, языковыми барьерами. Различные системы автоматизации логистических процессов сложно интегрируют друг с другом, что отражается на быстродействии всей логистической цепочки.

Во все более глобализирующемся мире современного бизнеса международная логистика стала важнейшим компонентом успеха международных компаний. Эффективное перемещение товаров через государственные границы требует высокой степени координации и сотрудничества между многочисленными сторонами, включая грузоотправителей, перевозчиков, агентов, экспедиционных компаний, таможенные службы и получателей. В этой сложной и динамичной среде потребность в надежной и гибкой системе управления данными и обмена ими имеет первостепенное значение.

Актуальность темы исследования обусловлена необходимостью разработки эффективной системы управления и обмена данными в международной логистике, которая сможет эффективно интегрировать данные от различных участников глобальной цепочки поставок, обеспечивая оперативное принятие решений и сокращение времени доставки товаров.

Целью данной дипломной работы является разработка прототипа системы управления и обменом данными в международной логистике. Это позволит улучшить эффективность и точность обработки данных, а также повысить качество принимаемых решений в области международной логистики.

Для достижения поставленной цели необходимо выполнить следующие задачи:

1. Изучить существующие системы управления и обмена данными в международной логистике, провести анализ их функциональности и возможностей.
2. Разработать концепцию архитектуры системы управления и обмена данными, определить основные модули и принципы взаимодействия между ними.
3. Анализ полученных результатов.

Объектом исследования является: управление и интеграция данными в логистике.

Предметом: модели интеграции и обмена данными в интегрированных логистических процессах.

Научная новизна: предложен прототип системы управления и обменом данными в международной логистике, который позволяет оптимизировать процессы и повысить эффективность логистических операций.

Практическая значимость разработки прототипа системы управления и обменом данными в международной логистике заключается в возможности улучшить эффективность и прозрачность логистических процессов. Использование современных технологий и разработка интегрированных систем управления данными позволят сократить временные затраты, уменьшить риски ошибок и упростить взаимодействие между участниками логистической цепи. Это повысит качество обслуживания клиентов, сократит издержки на складирование и транспортировку товаров, а также улучшит контроль над передвижением грузов и управлением запасами.

Экономическая эффективность данной работы проявляется в увеличении операционной прибыли компаний за счет оптимизации логистических процессов, снижении издержек и улучшении управления данными. Благодаря использованию новых технологий и оптимизации процессов, компании смогут повысить свою конкурентоспособность на рынке, привлечь новых клиентов и партнеров, а также повысить уровень автоматизации и эффективности своих операций. Это позволит увеличить доходы компании, снизить издержки и обеспечить стабильное развитие бизнеса в условиях быстро меняющегося рынка и конкуренции.

1.1 Обзор существующих систем управления процессами и обменом данными операций в международной логистике

В зарубежной литературе понятие логистики чаще всего связано с планированием как основной функцией управления. Американские ученые, например, считают логистику структурой планирования. Они определили логистику, прежде всего, как механизм экономии затрат. Совет логистического менеджмента (Councilof Logistics Management — CLM), например, определил логистику как «процесс планирования, внедрения и контроля рационального и эффективного движения товаров, услуг и связанной с этим информации от исходной точки до конечного потребителя с целью удовлетворения требований клиента» [1].

Определения логистики в разных источниках:

— Искусство управления перемещением войск как вдали, так и вблизи от неприятеля, организация их тылового обеспечения [2]

— Учение о системном планировании и контроле материальных, энергетических, информационных и пассажирских потоков, а также об управлении ими [3]

— Процесс планирования, выполнения и контроля с точки зрения снижения затрат на содержание запасов сырья, материалов, незавершенного производства, готовой продукции, сервиса и связанной информации от точки его зарождения до точки потребления (включая импорт, экспорт, внутренние и внешние перемещения) для полного удовлетворения потребителей [4]

— Организация, планирование, контроль и выполнение товарного потока от проектирования и закупок через производство и распределение до конечного потребителя с целью удовлетворения требований рынка с минимальными операционными и капитальными затратами [5]

— Наука о рациональной организации производства, транспорта и распределения, которая комплексно, с системных позиций, охватывает

вопросы снабжения, организации промышленного производства, распределения (товародвижения) и сбыта готовой продукции [6]

— Наука об управлении материальными потоками, связанной с ними информацией, финансами и сервисом в определенной микро-, мезо- или макроэкономической системе для достижения поставленных перед нею целей с оптимальными затратами ресурсов [7]

— Интегральный инструмент менеджмента, способствующий достижению стратегических, тактических или оперативных целей организации бизнеса за счет эффективного (с точки зрения снижения общих затрат и удовлетворения требований конечных потребителей к качеству продуктов и услуг) управления материальными и (или) сервисными, а также сопутствующими им потоками (финансовыми, информационными и т.п.) [8]

— Наука и искусство управления материалопотоком (выполнение основных функций менеджмента: планирования, организации, мотивации и контроля за его продвижением) [9]

— Функция, отвечающая за материальный поток, идущий от поставщиков в организацию, проходящий через операции внутри организации и затем уходящий к потребителям [10]

— Система управления потоковыми процессами, в том числе процессом товародвижения в различных сферах деятельности субъектов хозяйствования [11]

— Наука об управлении потоками в больших системах [12]

В общем виде ЛОГИСТИКА — это управление процессами материальных, информационных и людских потоков на основе их оптимизации. Логистика, по сути, относится к управлению потоком товаров и услуг от пункта происхождения до пункта потребления. Она обеспечивает эффективную транспортировку и хранение товаров, включая услуги и связанную с ними информацию. Таким образом, сектор логистики является основой мировой торговли, перемещая товары по континентам, от производителей к конечным потребителям.

По данным отчета[13], объем мирового рынка логистики достиг 9407,5 млрд долларов США в 2023 году, по прогнозам, в период с 2023 по 2032 год рынок будет расти в среднем на 6,4% и достигнет 15978,2 млрд долларов США к 2032 году.

Одним из движущих фактором роста рынка логистики является растущая глобализация отраслей. По мере того, как предприятия расширяют свое присутствие за рубежом, постоянно растет потребность в комплексных логистических решениях, которые позволяют беспрепятственно управлять трансграничными транзакциями и поставками. Более того, рынок логистики также испытывает попутный ветер от внедрения передовых технологий, таких как искусственный интеллект (AI), машинное обучение (ML) и Интернет вещей (IoT). Эти технологии улучшают оптимизацию маршрутов, управление складом и прогнозную аналитику, еще больше совершенствуя логистический процесс. К таким технологиям относятся системы управления и обменом данными, такие как Logistics Management System (LMS). LMS позволяют управлять транспортировкой и перевозками, флотом (или парком) средств перевозок, цепочками поставок, а также планировать маршруты поставок и решать специализированные задачи логистики.

Рисунок 1 – Составные части LMS (Logistics Management System)

Таблица 1 – Самые популярные системы управления и обмена данными в международной логистике:

Название	Описание
Enterprise Resource Planning (ERP)	это системы, которые интегрируют управление заказами, инвентаризацию, финансы и другие бизнес-процессы в одну систему
Transportation Management Systems (TMS)	это системы, которые помогают организовать, контролировать и оптимизировать поставки грузов по всему миру
Warehouse Management Systems (WMS)	это системы, которые управляют складом, инвентаризацией и процессами отгрузки и получения товаров
Electronic Data Interchange (EDI)	это стандартизированный формат обмена данных между компаниями, который позволяет автоматизировать процессы обработки заказов и счетов
Cloud-based Logistics Management Systems	это системы, которые предоставляют онлайн-платформы для управления логистическими процессами и данными

1.2 Систематизация, классификация основных этапов логистических операций в международной логистике, в интегрированных логистических процессах

Для начала определим основные понятия, которые будут использоваться в данной главе. Международная логистика — это стратегическое управление международной цепью поставок, понимаемое как объединение логистической деятельности организаций-звеньев цепи поставок, а именно, операционной, финансовой и маркетинговой функций и контроля за материальным потоком товаров, денежных средств и информации через границы и над границами различных государств [14]. Интегрированная логистика или SCM (Supply Chain Management) — это

логистика, направленная на решение задач комплексного управления снабжением, производством, распределением, устойчивого, взаимовыгодного сотрудничества организации с поставщиками ресурсов, посредниками, потребителями, субъектами инфраструктуры[15].

Рисунок 2 –Цепочка поставок (Supply Chain Management, SCM)[16]

Исходя из основных понятий, можно выделить основные этапы логистических операций в международной логистике:

Рисунок 3 – Этапы логистических операций.

Таблица 1.2 – Этапы логистических операций [17]

Этап	Описание
Планирование	на этом этапе основную роль играют системы управления запасами, прогнозирования спроса и планирования производства. Такие системы позволяют оптимизировать запасы, учитывать спрос на товары и услуги, а также планировать производственные мощности в соответствии с потребностями рынка
Закупки и снабжение	для эффективного закупа материалов и компонентов используются системы управления поставками (Supply Chain Management), которые позволяют автоматизировать процесс выбора поставщиков, контролировать сроки поставок и качество поступающих материалов
Производство	на этапе производства важную роль играют системы управления производственными процессами (Manufacturing Execution Systems), которые обеспечивают контроль и управление производством, позволяют оптимизировать использование ресурсов и повышать производительность
Хранение	для эффективного управления складскими запасами и складированием товаров используются системы управления складом (Warehouse Management Systems), которые

	обеспечивают контроль за приемом, хранением и отгрузкой товаров, а также оптимизируют процессы комплектации заказов
Транспортировка	на этапе транспортировки основную роль играют системы управления транспортом (Transportation Management Systems), которые позволяют планировать и отслеживать транспортные перевозки, выбирать оптимальные маршруты доставки и контролировать исполнение логистических операций
Дистрибуция	для эффективного управления распределением товаров и услуг между точками продажи используются системы управления дистрибуцией (Distribution Management Systems), которые позволяют оптимизировать процессы доставки, управлять запасами на складах и контролировать выполнение заказов
Обратная логистика	для управления обратными логистическими процессами используются системы обратной логистики, которые позволяют отслеживать и управлять возвратом товаров, их переработкой или утилизацией, а также повторным использованием в производственных циклах

Таким образом, для эффективной реализации каждого этапа логистических операций в международной логистике и интегрированных логистических процессах необходимо использовать специализированные

системы управления, которые помогут оптимизировать процессы и достичь лучших результатов.

Использование специализированных систем управления международной логистикой позволяет более эффективно и точно контролировать все этапы транспортировки грузов, от закупки и хранения до доставки потребителю. Это позволяет повысить эффективность работы логистической цепи, сократить издержки и улучшить качество обслуживания клиентов. Такие системы также помогают управлять сложными процессами, учитывать все необходимые параметры и быстро реагировать на изменения в ситуации. В целом, цель использования специализированных систем управления международной логистикой заключается в повышении эффективности работы и снижении рисков при организации движения грузов через международные границы.

Таблица 1.3 – Концепции и принципы систем управления логистическими процессами [18]

Принцип	Описание
Интеграция	система должна обеспечивать интеграцию различных компонентов цепочки поставок, таких как склады, транспорт, заказы и персонал, для синхронизации и оптимизации процессов
Международная совместимость	система должна быть способна работать с множеством различных стран, учитывая различия в законодательстве, валютах, языках и прочих аспектах
Масштабируемость	система должна быть способна масштабироваться в соответствии с ростом бизнеса и объемом операций, а также поддерживать работу в различных географических регионах

Аналитика и отчетность	система должна обеспечивать возможность анализа данных о производительности, затратах, времени доставки и других показателях для принятия обоснованных решений и оптимизации процессов
Безопасность	система должна обладать надежными мерами защиты данных и конфиденциальности, учитывая возможные риски хакерских атак и утечки информации
Мобильность	система должна быть доступна на различных устройствах и позволять работать удаленно, что особенно важно для международных операций логистики
Гибкость и адаптивность	система должна быть гибкой и способной адаптироваться к изменяющимся условиям рынка, требованиям клиентов и новым технологиям

Эти концепции и принципы помогают компаниям сделать свои логистические процессы более эффективными, прозрачными и надежными, что является ключевым для успешной работы в международной логистике.

1.3 Основные преимущества и недостатки для управления процессами и обменом данными в интегрированных логистических процессах

Каждая система имеет как свои плюсы, так и минусы. Разберем подробнее основные преимущества существующих систем управления в международной логистике.

Преимущества:

- ✓ **Повышение эффективности:** системы помогают автоматизировать и оптимизировать логистические процессы, что позволяет сократить время выполнения задач, улучшить управление ресурсами и уменьшить издержки.
- ✓ **Улучшение прозрачности и контроля:** благодаря системам управления логистикой можно отслеживать все шаги логистической цепочки в режиме реального времени, что обеспечивает более точный контроль над выполнением заказов и процессами.
- ✓ **Улучшение обслуживания клиентов:** возможность оперативной обработки заказов, отслеживания статусов грузов и обмена данными с партнерами позволяет предоставить более качественное обслуживание клиентам.
- ✓ **Аналитика и оптимизация:** системы предоставляют данные и отчеты для анализа производительности, выявления узких мест и оптимизации процессов логистики.
- ✓ **Улучшенное управление запасами:** Интегрированные системы позволяют более точно прогнозировать потребности в запасах, оптимизировать их уровень и рационально распределять товары по всей цепочке поставок.

- ✓ **Интеграция:** возможность интегрировать различные системы и платформы упрощает обмен данными между различными отделами компании и сторонними партнерами.
- ✓ **Снижение издержек:** Оптимизация процессов в рамках интегрированных систем помогает сократить издержки на складирование, транспортировку и управление запасами.

Недостатки:

- ✓ **Высокие затраты на внедрение и обновление:** внедрение и поддержка систем управления логистикой требует значительных инвестиций как в финансовом, так и в человеческом плане.
- ✓ **Сложность настройки и обучения персонала:** многие системы требуют специализированных знаний и опыта для настройки и эффективного использования, что может потребовать времени и усилий для обучения персонала.
- ✓ **Риски безопасности:** системы управления логистикой могут быть уязвимы для кибератак и утечек данных, поэтому необходимы меры по обеспечению безопасности информации.
- ✓ **Необходимость постоянного обновления и адаптации:** с учетом быстрого развития технологий и требований рынка, системы логистики должны постоянно обновляться и адаптироваться к изменениям.
- ✓ **Зависимость от поставщиков систем:** использование систем сторонних поставщиков подразумевает зависимость от их надежности, качества обслуживания и способности предоставить актуальные обновления.

Несмотря на эти недостатки, существующие системы управления логистикой предоставляют значительные преимущества компаниям, помогая им сделать свои логистические процессы более эффективными, прозрачными и конкурентоспособными

1.4 Основные потребности и требования для управления процессами и обменом данными в интегрированных логистических процессах

- Интеграция данных: необходимость собирать, хранить и обрабатывать данные из различных источников (поставщики, производители, транспортные компании и т.д.) для создания полной картины логистической цепи.
- Автоматизация процессов: внедрение автоматизированных систем управления, которые могут управлять процессами без необходимости вручную контролировать каждый этап.
- Обмен данными в реальном времени: возможность обмена информацией между различными участниками цепи поставок в режиме реального времени для оперативного реагирования на изменения и сокращения времени на принятие решений.
- Управление рисками и предотвращение простоев: необходимость иметь систему предупреждения и реагирования на потенциальные риски и проблемы в логистической цепи для минимизации простоев и задержек.
- Повышение прозрачности: возможность отслеживать статус и местоположение груза на каждом этапе его перемещения, что способствует повышению прозрачности и контроля над логистическими процессами.
- Соблюдение стандартов и законодательных требований: важно учитывать соответствие существующим нормативам и стандартам для обмена и хранения данных, а также правовым требованиям в области логистики.
- Обучение и поддержка персонала: необходимость обучения сотрудников по работе с интегрированными системами и обеспечение технической поддержки для эффективной работы системы.

Изучив все преимущества и недостатки логистических систем управления данными, определив потребности и требования для них, можно сделать вывод, что необходимость внедрения интегрированных логистических систем управления данными становится очевидной для того, чтобы повысить эффективность, прозрачность и контролируемость логистических процессов. Осознание преимуществ и недостатков подчеркивает важность баланса между затратами на внедрение и ожидаемой прибылью от оптимизации логистической цепи, а понимание потребностей и требований к логистическим системам управления данными позволяет сделать правильный выбор при выборе подходящего решения для конкретной логистической цепи.

Важно помнить, что успешная реализация интегрированных логистических систем требует не только технологических инноваций, но и организационных изменений, обучения персонала и согласования с партнерами в цепи поставок.

В целом, анализ преимуществ, недостатков, потребностей и требований к логистическим системам управления данными позволяет составить четкий план действий по оптимизации логистических процессов и повышению эффективности всей цепи поставок.

1.5 Технологии и методы машинного обучения, применяемые в логистике

В современном мире с каждым днем объем и сложность логистических процессов стремительно растет, требуя более эффективных и инновационных подходов к их управлению. Одним из таких подходов является использование технологий и методов машинного обучения, которые позволяют автоматизировать и оптимизировать логистические операции, улучшить предсказательные аналитики и управление запасами, снизить издержки и повысить общую производительность цепи поставок.

Современные технологии машинного обучения стали важной составляющей в поддержке и оптимизации логистических процессов [19]. В данной части мы рассмотрим основные технологии и методы машинного обучения, применяемые в логистике, и их влияние на современные логистические процессы.

Рисунок 4 – ML в логистике [20]

Рассмотрим некоторые из наиболее значимых технологий и методов машинного обучения, используемых в логистике:

- ❖ Прогнозирование спроса: Модели машинного обучения, такие как регрессия, временные ряды, случайные леса и нейронные сети, могут использоваться для прогнозирования спроса на товары и услуги. Это помогает компаниям оптимизировать запасы, улучшить планирование производства и транспортные операции.
- ❖ Маршрутизация и оптимизация транспортных сетей: Методы оптимизации маршрутов, такие как генетические алгоритмы, жадные алгоритмы и алгоритмы Монте-Карло, используются для оптимизации

маршрутов доставки грузов и транспортировки грузов в различные точки. Это помогает уменьшить временные и транспортные издержки.

- ❖ Обработка естественного языка (Natural Language Processing, NLP): NLP используется для анализа и интерпретации текстовых данных, таких как электронные письма, чаты с клиентами, отзывы и комментарии. Это помогает компаниям быстрее реагировать на запросы клиентов и улучшать общее обслуживание.
- ❖ Обнаружение аномалий и предотвращение мошенничества: Методы машинного обучения, такие как алгоритмы кластеризации, классификации и анализа выбросов, помогают обнаруживать аномалии в данных логистических операций, что позволяет предотвращать мошенничество и уменьшать потери.
- ❖ Автоматизация складских операций: Машинное обучение применяется для автоматизации складских процессов, включая распределение товаров по складу, оптимизацию маршрутов сборки товаров и управление инвентаризацией.
- ❖ Управление цепочками поставок: Методы машинного обучения помогают улучшить прогнозирование спроса, оптимизировать запасы и управлять поставками в цепочке поставок, что способствует снижению издержек и улучшению общей эффективности.

Эти технологии и методы машинного обучения играют ключевую роль в современной логистике, помогая компаниям улучшить операционную эффективность, повысить качество обслуживания клиентов и снизить издержки.

В современной логистике данные играют ключевую роль в принятии бизнес-решений и оптимизации логистических процессов. Данные становятся основой для применения технологий машинного обучения, которые позволяют автоматизировать и улучшить процессы принятия решений, прогнозирования спроса, оптимизации маршрутов доставки, управления запасами и других аспектов логистики. Рассмотрим методы

сбора, анализа и обработки данных для управления логистическими процессами.

- ✓ Системы автоматического идентификации (AID): такие как штрихкоды, RFID (Radio-Frequency Identification), QR-коды, используются для идентификации и отслеживания грузов, товаров и оборудования в различных точках логистической цепочки.
- ✓ Системы управления запасами (Inventory Management Systems): позволяют отслеживать уровень запасов, управлять закупками и оптимизировать запасы для минимизации издержек и улучшения эффективности.
- ✓ Системы управления транспортом (Transportation Management Systems, TMS): обеспечивают автоматизацию управления транспортными процессами, включая маршрутизацию, отслеживание грузов и оптимизацию транспортных затрат.
- ✓ Системы управления складом (Warehouse Management Systems, WMS): предоставляют инструменты для эффективного управления складскими операциями, включая прием, хранение, отбор и отгрузку товаров.
- ✓ Системы управления заказами (Order Management Systems): помогают автоматизировать обработку заказов, отслеживать статус заказов и оптимизировать процессы выполнения заказов.
- ✓ Системы аналитики данных (Data Analytics Systems): используются для анализа и интерпретации данных, полученных из различных источников, с целью выявления тенденций, прогнозирования спроса, оптимизации процессов и принятия стратегических решений.
- ✓ Системы управления производством (Manufacturing Execution Systems, MES): помогают синхронизировать производственные процессы с логистическими операциями, обеспечивая планирование

производства, отслеживание статусов производства и управление производственными ресурсами.

✓ Интернет вещей (Internet of Things, IoT): позволяет собирать данные с помощью датчиков и устройств, установленных на оборудовании, транспорте и товарах, для отслеживания и мониторинга различных параметров в реальном времени.

Эти методы сбора, анализа и обработки данных играют ключевую роль в улучшении эффективности и прозрачности логистических процессов, обеспечивая компаниям возможность принимать информированные решения и улучшать свою операционную деятельность.

1.6 Классификация методов управления и обменом данными в международной логистике

Методы управления и обмена данными в международной логистике могут быть классифицированы по различным критериям [21], таким как способ передачи данных, стандарты обмена информацией, уровень автоматизации и т.д. Ниже мы рассмотрим основные методы и классифицируем их по нескольким критериям:

- Электронные документы и электронные данные:

— Электронная почта (email): подходит для обмена документами и информацией между партнерами в международной логистике, но имеет ограничения в отношении безопасности и автоматизации обработки.

— Электронные документы (EDI): Стандартизованный метод обмена данных между компаниями, использование которого требует согласованных форматов и протоколов.

- Веб-сервисы и API [22]:

— RESTful API: позволяет программным образом обмениваться данными между системами, обеспечивая скорость и гибкость.

— SOAP API: Стандартизованный протокол для обмена структурированными данными, более тяжеловесный и менее гибкий по сравнению с RESTful API.

- Системы управления цепочкой поставок (Supply Chain Management, SCM)[23]:

— ERP-системы: Интегрированные системы управления предприятием позволяют управлять всей цепочкой поставок, включая управление складами, транспортировкой и производством.

— WMS (Warehouse Management System): Системы управления складом обеспечивают эффективное управление складскими операциями и инвентаризацией.

- Технологии IoT и RFID[24]:

— RFID (Radio-Frequency Identification): используется для отслеживания грузов и контейнеров в реальном времени, обеспечивая точность и автоматизацию.

— IoT (Internet of Things): позволяет связывать физические объекты с интернетом, улучшая мониторинг и контроль в цепочке поставок.

- Облачные технологии [25]:

— Облачные хранилища и вычисления: позволяют хранить и обрабатывать данные в облаке, обеспечивая доступность и масштабируемость.

— SaaS (Software as a Service): предоставляет программное обеспечение как сервис через интернет, упрощая доступность и управление.

Различные методы управления и обмена данными в международной логистике могут быть комбинированы и адаптированы в зависимости от конкретных потребностей компании и требований цепочки поставок. Они помогают улучшить прозрачность, эффективность и управляемость логистических процессов.

2.1 Анализ текущей системы управления и обменом данными в международной логистике

Существующую систему управления и обмена данными в международной логистике [26] можно охарактеризовать как сложную, фрагментированную и не интероперабельную, поскольку различные заинтересованные стороны используют разные системы и форматы данных для обмена. В существующей системе обмена логистическими данными также существуют проблемы с безопасностью и конфиденциальностью данных. Поскольку в логистический процесс вовлечено так много различных сторон, существует риск того, что конфиденциальные данные будут скомпрометированы или попадут не в те руки. Это может привести к юридическим и финансовым последствиям для всех вовлеченных сторон.

В последние годы предпринимаются усилия по стандартизации обмена данными, улучшению безопасности, электронизации в международной логистике с помощью инициатив организаций, таких как: Ассоциация цифровых контейнерных перевозок (DCSA) [27], Международная ассоциация воздушного транспорта (IATA) [28] и Международная федерация транспортных компаний (FIATA) [29]. Однако, несмотря на эти усилия, существующая система управления и обмена данными в международной логистике остается несовершенной, сложной и фрагментированной. Существует потребность в более стандартизированной и безопасной системе обмена данными, а также в более эффективном способе управления логистическими данными.

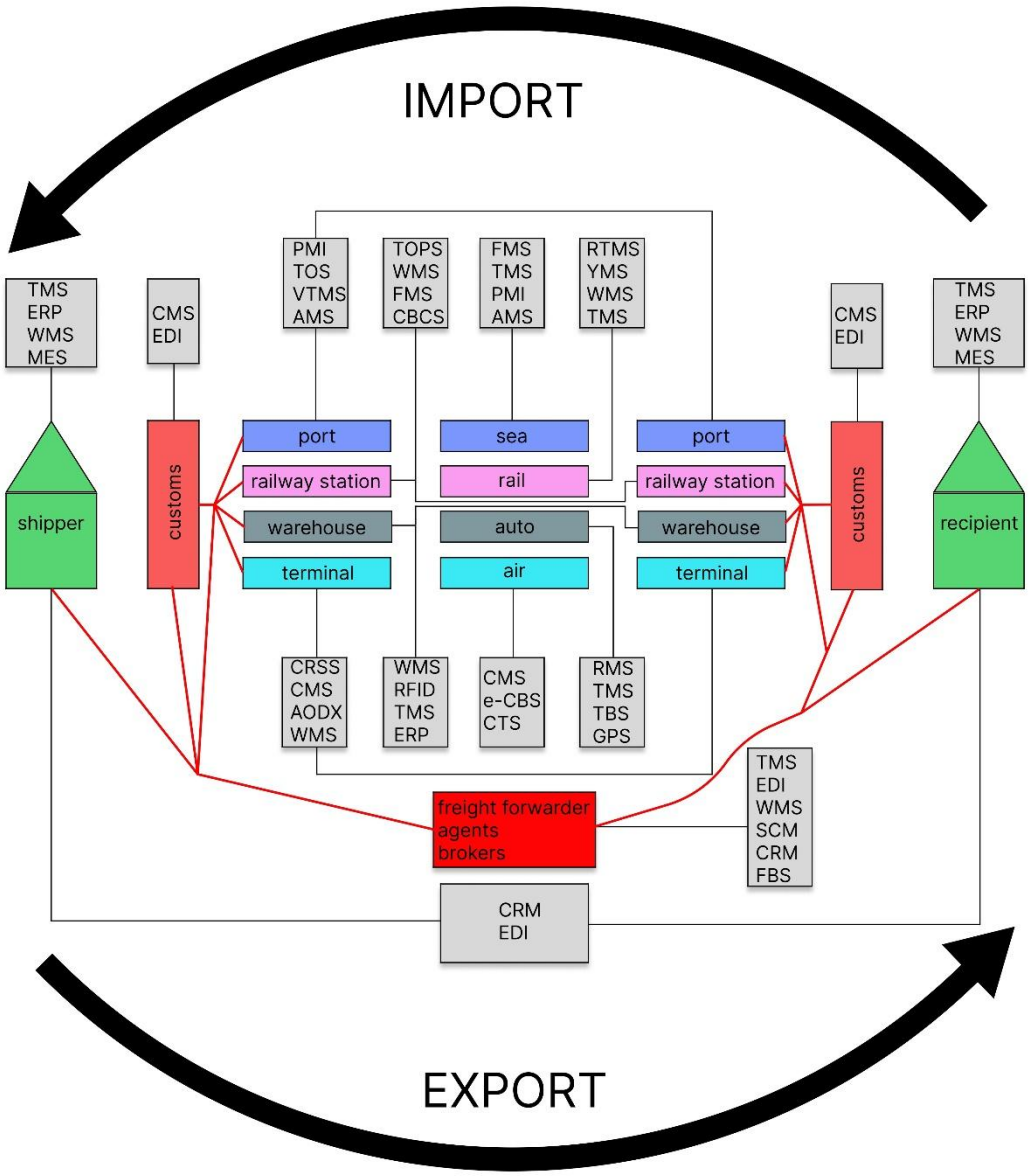

Рисунок 3 – схема текущего взаимодействия участников логистической цепочки поставок.

Система управления и обмена данными в международной логистике нуждается в пересмотре и предложении новой системы потому, что:

- Увеличение сложности: международная логистика становится все более сложной, и старые системы не могут обеспечить необходимую гибкость и адаптивность.

- Технологические изменения: появляются новые технологии, такие как блокчейн, машинное обучение, искусственный интеллект и интернет вещей (IoT). Развитие происходит очень быстро, и старые системы не успевают соответствовать новым требованиям и возможностям для улучшения логистических операций
- Масштабируемость. Существующая система может оказаться не в состоянии обрабатывать растущий объем данных и транзакций, что приводит к неэффективности и ошибкам.
- Стандартизация: Важный элемент эффективного взаимодействия это возможность принимать и обрабатывать разные форматы данных от множества различных систем. В противном случает это приводит к проблемам совместимости и ошибкам
- Увеличение безопасности: международная логистика требует обеспечения высокой безопасности данных и процессов, и старые системы не могут обеспечить это.
- Увеличение прозрачности: международная логистика требует обеспечения прозрачности процессов и доступа к информации для всех заинтересованных сторон, и старые системы не могут обеспечить это.
- Увеличение требования к гибкости: международная логистика требует обеспечения гибкости и адаптивности к изменяющимся условиям и требованиям, и старые системы не могут обеспечить это.
- Увеличение требования к автоматизации: международная логистика требует автоматизации процессов, и старые системы не могут обеспечить это.
- Увеличение требования к интероперабельности: международная логистика требует обеспечения взаимодействия между различными системами и организациями, и старые системы не могут обеспечить это.

В целом, старая система управления и обмена данными в международной логистике не может полноценно соответствовать новым

требованиям, вызовам и возможностям, и поэтому, необходима новая система, которая будет обеспечивать безопасность, прозрачность, гибкость, автоматизацию и интероперабельность.

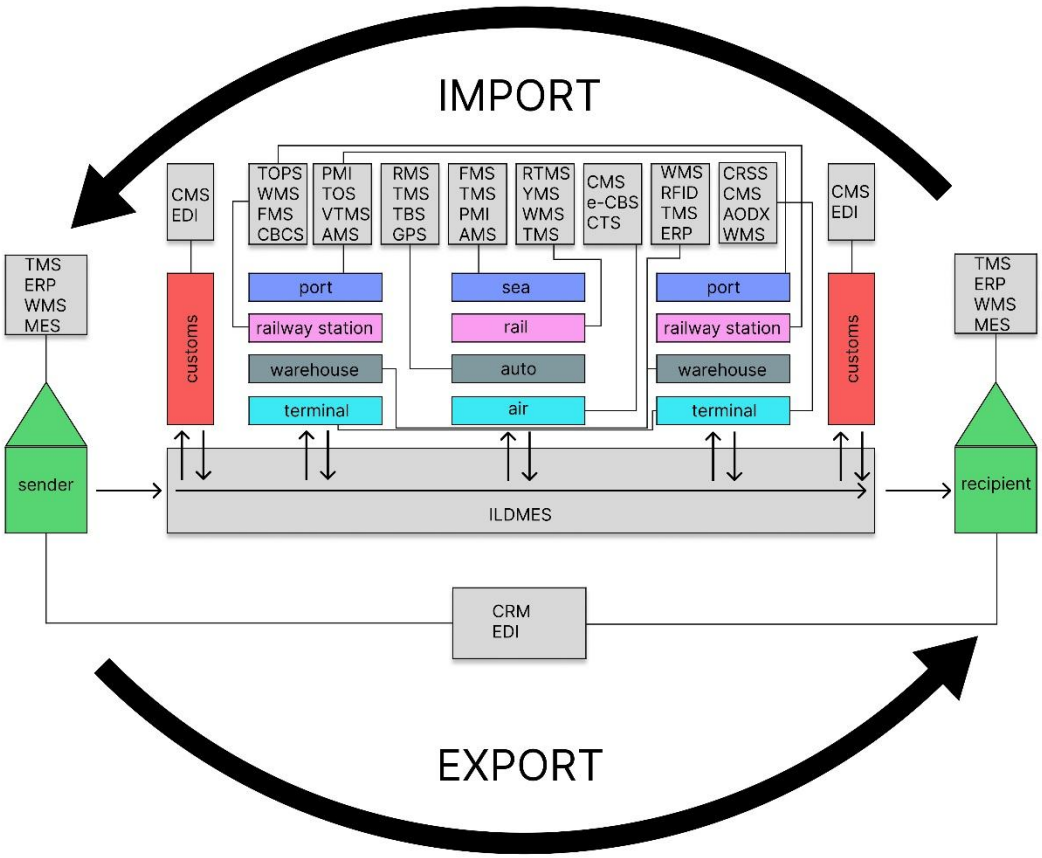

Рисунок 4 – схема взаимодействия участников логистической цепочки поставок с помощью ILDMES (The International Logistics Data Management and Exchange System).

Прототип системы управления и обмена данными в международной логистике будет обеспечивать безопасность, прозрачность, гибкость, автоматизацию, интероперабельность, доступность, масштабируемость и экономию ресурсов.

2.2 Сравнение существующих систем и прототипа системы

Современные системы, используемые в международной логистике для управления и обмена данными, часто фрагментированы, управляются вручную и подвержены ошибкам. В этих системах обычно задействовано множество заинтересованных сторон, каждая из которых имеет свои собственные системы и процессы, что может привести к неэффективности и неточностям.

Предлагаемая система The International Logistics Data Management and Exchange System (ILDMES) предназначена для решения этих проблем путем предоставления централизованной, автоматизированной и стандартизированной платформы для управления и обмена данными.

Таблица 2 – Сравнение существующих систем и прототипа системы

операция	текущая система взаимодействия	прототип системы
Ввод данных	Современные системы часто требуют ручного ввода данных, что может привести к ошибкам и неточностям	Обеспечивает автоматизированный обмен данными между заинтересованными сторонами, сокращая необходимость в ручном вводе данных и сводя к минимуму количество ошибок.
Фрагментированные системы	У каждой заинтересованной стороны есть своя собственная система, что может привести к	Предоставляет централизованную платформу для управления данными и обмена ими, что позволяет легко

	неэффективности и трудностям при обмене и интеграции данных	интегрировать и стандартизировать данные
Ограниченная стандартизация	В существующих системах часто отсутствует стандартизация, что может привести к несогласованности и трудностям при обмене данными	Использует стандартизированные форматы данных, обеспечивая согласованность и точность обмена данными.
Проблемы безопасности	В существующих системах часто отсутствуют надежные меры безопасности, соответствующие развитию технологий, что может привести к утечке данных и несанкционированному доступу	Предоставляет надежные меры безопасности, включая шифрование, аутентификацию и авторизацию, для обеспечения целостности и конфиденциальности данных
Отслеживание статуса груза	Ограниченный доступ к логистическим операциям в режиме реального времени	Полный доступ к статусу и местоположению груза в режиме реального времени

Преимущества ILDMES:

— Повышенная точность данных: ILDMES сокращает количество ошибок и неточностей за счет автоматизации обмена данными и использования стандартизированных форматов данных.

— Повышенная эффективность: ILDMES упрощает обмен и управление данными, сокращая необходимость в ручном вмешательстве и повышая эффективность.

— Повышенная безопасность: ILDMES обеспечивает надежные меры безопасности, гарантирующие целостность и конфиденциальность данных.

— Улучшение взаимодействия: ILDMES обеспечивает бесперебойное взаимодействие между заинтересованными сторонами, улучшая коммуникацию и уменьшая количество недоразумений.

Недостатки ILDMES:

— Первоначальные инвестиции: Внедрение ILDMES требует значительных первоначальных инвестиций в инфраструктуру, программное обеспечение и обучение.

— Управление изменениями: ILDMES требует от заинтересованных сторон адаптации к новым процессам и системам, что может быть сложной задачей. Потребуется внесение изменений в существующие бизнес-процессы и технологическое сопровождение.

— Зависимость от технологий: ILDMES полагается на технологии, которые могут быть уязвимы к сбоям.

— Интеграция данных: ILDMES требует интеграции данных из нескольких источников, что может быть сложным и трудоемким процессом.

В заключение, ILDMES обладает рядом преимуществ по сравнению с существующими системами, включая повышенную точность данных, эффективность, безопасность, интеграцию, стандартизацию и улучшенную совместную работу. Однако это также сопряжено с некоторыми недостатками, включая первоначальные инвестиции, управление изменениями, зависимость от технологий и интеграцию данных. Тщательно оценив эти факторы, заинтересованные стороны могут определить, является ли ILDMES подходящим решением для их потребностей в международной логистике.

2.3 Требования к прототипу ILDMES

Чтобы гарантировать, что прототип системы управления и обмена данными в международной логистике отвечает современным вызовам и требованиям логистической отрасли, могут быть применены следующие требования:

- Масштабируемость: Система должна быть способна обрабатывать большие объемы данных и быстро расти, что делает ее пригодной для крупных и сложных логистических операций.
- Безопасность: Система должна обеспечивать надежные меры безопасности для защиты данных и предотвращения несанкционированного доступа, включая шифрование, брандмауэры и средства контроля доступа.
- Обмен данными в режиме реального времени: Система должна обеспечивать обмен данными в режиме реального времени между всеми заинтересованными сторонами, включая грузоотправителей, перевозчиков и поставщиков логистических услуг.
- Управление данными: Система должна позволять создавать, хранить и извлекать данные, связанные с логистикой, включая информацию об отгрузке, уровнях запасов и статусе доставки.
- Производительность: Система должна быть способна обрабатывать большие объемы данных и транзакций без ущерба для производительности.
- Точность данных: Система должна обеспечивать точность и надежность данных, сокращая количество ошибок и неточностей в логистическом процессе.
- Интероперабельность: Система должна быть способна интегрироваться с различными системами и платформами, обеспечивая

бесперебойный обмен данными и коммуникацию между всеми заинтересованными сторонами.

- Гибкость: Система должна быть способна адаптироваться к меняющимся бизнес-требованиям и логистическим операциям, обеспечивая простоту настройки.
- Прозрачность: Система должна обеспечивать видимость в режиме реального времени для всех заинтересованных сторон, повышая прозрачность и сотрудничество в логистическом процессе.
- Автоматизированные процессы: Система должна автоматизировать как можно больше процессов, сокращая количество ошибок, совершаемых вручную, и повышая эффективность.
- Интеграция с устройствами Интернета вещей: Система должна быть способна интегрироваться с устройствами Интернета вещей, такими как устройства GPS-слежения и датчики, для предоставления данных в режиме реального времени и улучшения сервиса логистических операций.
- Облачная система: Система должна быть облачной, обеспечивающей легкий доступ и масштабируемость, а также снижающей потребность в локальной инфраструктуре.
- Искусственный интеллект и машинное обучение: Система должна включать алгоритмы искусственного интеллекта и машинного обучения для улучшения прогнозной аналитики, оптимизации маршрутов и наглядности цепочки поставок.
- Интеграция API: Система должна иметь интеграцию API с другими системами и платформами.
- Аппаратное обеспечение: Система должна быть способна работать на различных аппаратных платформах, включая серверы, рабочие станции и мобильные устройства.

- Программное обеспечение: Система должна быть способна работать на различных программных платформах, включая Windows, Linux и мобильные операционные системы.
- Хранение данных: В системе должно использоваться надежное решение для хранения данных. Система должна быть способна хранить большие объемы данных в базе данных, включая реляционные базы данных, базы данных NoSQL и управлять ими.
- Удобный интерфейс: Система должна иметь удобный интерфейс, позволяющий всем заинтересованным сторонам легко пользоваться ею и ориентироваться в ней.
- Поддержка нескольких языков: Система должна поддерживать несколько языков, что обеспечивает глобальную коммуникацию и сотрудничество.
- Соблюдение нормативных требований: Система должна соответствовать всем соответствующим положениям и стандартам, таким как GDPR, HIPAA и таможенные правила.
- Настройка: Система должна быть адаптирована для удовлетворения конкретных потребностей различных предприятий и отраслей промышленности.
- Постоянный мониторинг и совершенствование: Система должна быть спроектирована таким образом, чтобы постоянно отслеживать и улучшать свою работу, используя аналитику данных и обратную связь от пользователей.

Применяя эти требования, прототип системы управления и обмена данными в международной логистике может обеспечить соответствие современным вызовам и требованиям логистической отрасли и обеспечить надежное и масштабируемое решение для логистических операций.

2.4 Архитектура ILDMES

Для обеспечения прочной основы прототипа системы, позволяющую ей соответствовать требованиям и предоставлять масштабируемое, гибкое и безопасное решение для обмена и управления данными в международной логистике предлагается использовать такую архитектуру:

- Система будет построена на основе облачной инфраструктуры с использованием облачного провайдера.
- Масштабируемость, гибкость и устойчивость обеспечивает архитектура микросервисов, при этом каждый микросервис будет отвечать за определенные бизнес-возможности, такие как обмен данными, управление данными и видимость в режиме реального времени.
- Прототип системы будет использовать API-шлюз, который действует как точка входа для всех входящих запросов, обрабатывая аутентификацию, авторизацию и маршрутизацию к соответствующим микросервисам.
- В ILDMES предполагается использовать надежное решение для хранения данных, такое как система управления реляционными базами данных (СУБД) – MySQL или PostgreSQL, для хранения данных, связанных с логистикой.
- Система будет использовать очередь сообщений, такую как Apache Kafka или RabbitMQ, для асинхронной обработки данных и обеспечения видимости логистических операций в режиме реального времени.
- Система будет иметь веб-интерфейс, построенный с использованием веб-фреймворка, такого как React или Angular, который обеспечивает удобный и интуитивно понятный способ взаимодействия пользователей с системой.

- Безопасность: Система будет использовать шифрование и другие меры безопасности для обеспечения безопасности и целостности передаваемых данных.

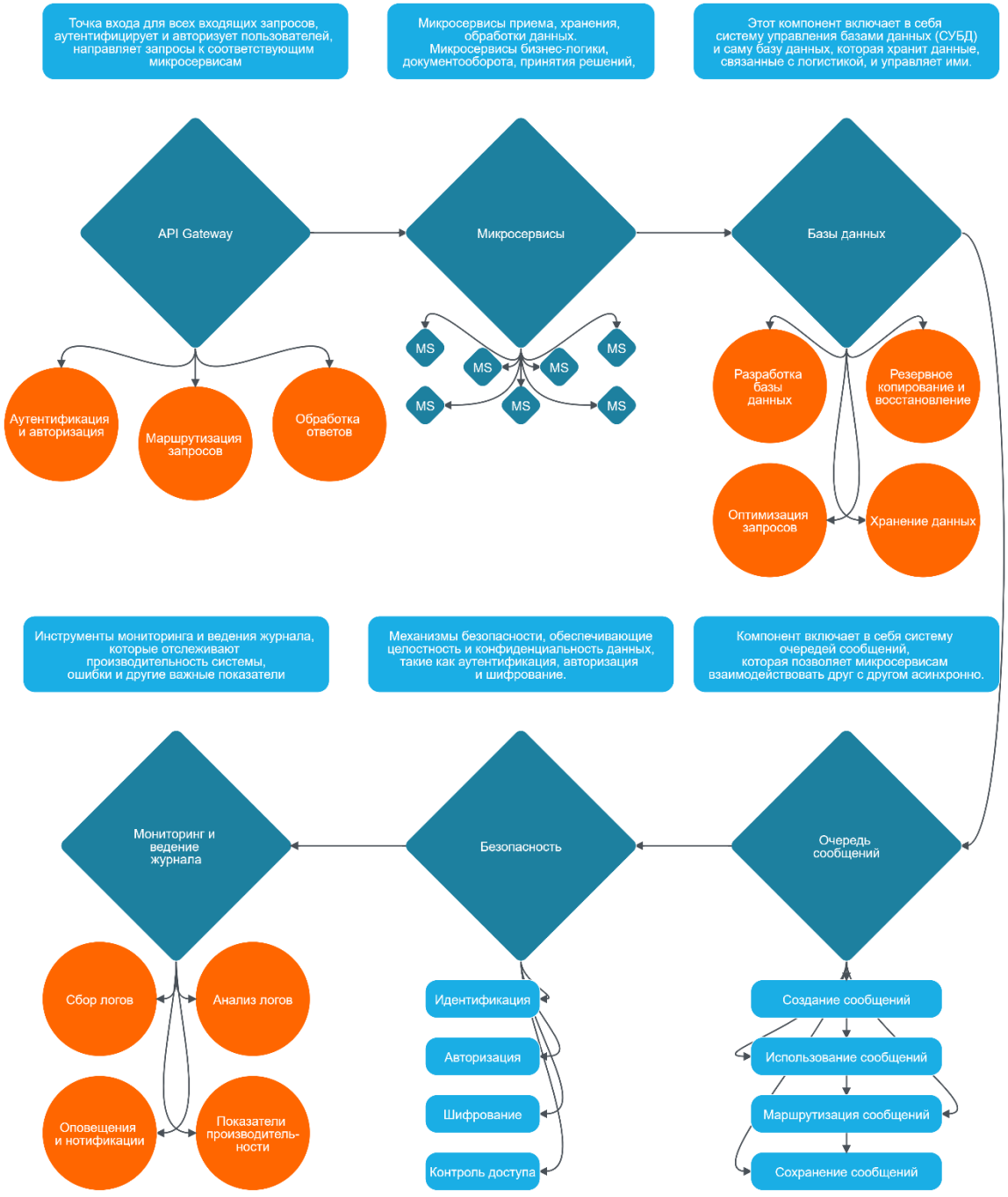

Рисунок 5 – архитектура ILDMES.

2.5 Описание технической части архитектуры

API- Gateway:

Стек технологий: Node.js, Express.js и NGINX.

Аутентификация и авторизация: OAuth 2.0 с токенами JWT, с использованием такой библиотеки, как Passport.js

Маршрутизация запросов: Express.js маршрутизаторов и обратного прокси-сервера NGINX.

Обработка ответов: Express.js объектов ответов и кэширования NGINX.

Балансировка нагрузки: NGINX и HAProxy

Безопасность: шифрование SSL/TLS и рекомендаций по безопасности OWASP.

Микросервисы:

1) Микросервис приема данных: стек технологий – Python, Flask и Apache Kafka.

Интеграция с источниками данных: использование разработчиками и потребителями Apache Kafka для интеграции с различными источниками данных.

Проверка данных: использование библиотек Python, таких как pandas и NumPy, для проверки и очистки данных.

Хранение данных: использование Apache Kafka topics и Apache Cassandra для хранения данных.

2) Микросервис обработки данных: стек технологий – Python, scikit-learn и Apache Spark.

Реализация алгоритма: использование библиотек scikit-learn для алгоритмов машинного обучения.

Преобразование данных: использование Apache Spark для преобразования и обработки данных.

Хранение результатов: использование Apache Cassandra для хранения результатов.

3) Микросервис хранения данных: стек технологий – MySQL или PostgreSQL и Apache Cassandra.

Разработка базы данных: использование шаблонов проектирования баз данных и методов нормализации.

Хранение данных: использование MySQL или PostgreSQL для хранения реляционных данных и Apache Cassandra для хранения данных NoSQL.

Оптимизация запросов: использование методов оптимизации запросов к базе данных и индексации.

4) Микросервис поиска данных: стек технологий – Python, Flask и Apache Cassandra.

Извлечение данных: использование запросов Apache Cassandra и библиотек Python, таких как pandas и NumPy, для извлечения данных.

Преобразование данных: использование Apache Spark для преобразования и обработки данных.

Возврат результатов: использование объектов ответа Flask и кэширования NGINX.

5) Микросервис бизнес-логики: стек технологий – Java, Spring Boot и Apache Camel.

Бизнес-логика: использование классов Java и аннотаций Spring Boot для реализации бизнес-логики.

Поиск данных: использование маршрутов Apache Camel и библиотек Java, таких как Hibernate, для поиска данных.

Возвращаемый результат: использование объектов Spring Boot response и маршрутов Apache Camel.

6) Микросервис рабочего процесса: стек технологий – Python, Flask и Apache Airflow.

Определение рабочего процесса: использование DAG Apache Airflow и библиотек Python, таких как pandas и NumPy, для определения рабочего процесса.

Выполнение задачи: использование задач Apache Airflow и библиотек Python, таких как pandas и NumPy, для выполнения задачи.

Возврат результата: использование объектов ответа Flask и кэширования NGINX.

7) Микросервис поддержки принятия решений: стек технологий: R, Shiny и Apache Spark.

Поддержка принятия решений: использование R-библиотек, таких как dplyr и tidyr, для анализа и визуализации данных.

Поиск данных: использование Apache Spark для поиска и обработки данных.

Возврат результатов: использование Shiny apps и Apache Spark для возврата результатов.

База данных

Стек технологий: MySQL или PostgreSQL и Apache Cassandra.

Проектирование базы данных: использование шаблонов проектирования баз данных и методов нормализации.

Хранение данных: использование MySQL или PostgreSQL для хранения реляционных данных и Apache Cassandra для хранения данных NoSQL.

Оптимизация запросов: использование методов оптимизации запросов к базе данных и индексации.

Резервное копирование и восстановление: использование инструментов и сценариев резервного копирования и восстановления баз данных.

Очередь сообщений

Стек технологий: Apache Kafka, Apache Cassandra и RabbitMQ.

Создание сообщений: использование Apache Kafka producers и RabbitMQ publishers для создания сообщений.

Использование сообщений: использование пользователей Apache Kafka и RabbitMQ для обработки сообщений.

Маршрутизация сообщений: использование тем Apache Kafka и очередей RabbitMQ для маршрутизации сообщений.

Сохранение сообщений: использование Apache Cassandra для сохранения сообщений.

Безопасность

Аутентификация: использование OAuth 2.0 с токенами JWT и библиотекой, такой как Passport.js

Авторизация: с использованием управления доступом на основе ролей (RBAC) и библиотеки, такой как Casper.js

Шифрование: с использованием SSL/TLS-шифрования и библиотеки, такой как OpenSSL.

Управление доступом: с использованием списков ACL Apache Kafka и разрешений RabbitMQ для управления доступом.

Мониторинг и ведение журнала

Сбор логов: использование Apache Kafka logs и RabbitMQ logs для сбора логов.

Анализ логов: использование Apache Spark и библиотек Python, таких как pandas и NumPy для анализа логов.

Оповещения и уведомления об ошибках: использование Apache Kafka alerts и RabbitMQ notifications для оповещений и уведомлений

Показатели производительности: использование Apache Kafka metrics и RabbitMQ metrics для определения показателей производительности.

2.6 Разработка микросервиса в ILDMES

Международная система управления логистическими данными и обмена (ILDMES) - это комплексная платформа, предназначенная для оптимизации логистических операций путем обмена данными между различными участниками цепочки поставок. Один из критических

компонентов ILDMES — это микросервис обработки заказов, который играет важную роль в управлении потоком информации между пользователями и логистическими провайдерами. В этой главе мы будем фокусироваться на разработке этого микросервиса, подчеркивая его архитектуру, функциональность и используемые технологии.

2.6.1 Требования к микросервису

Микросервис обработки заказов отвечает за выполнение следующих задач:

— Прием данных о заказе от пользователей. Программа должна быть способна принимать и валидировать данные о заказе от пользователей, включая детали, такие как место отправления, место назначения, вес и другие релевантные сведения.

— Отправка данных о заказе логистическим провайдерам: Микросервис должен быть способен передавать валидированные данные о заказе соответствующим логистическим провайдерам, обеспечивая безопасную и эффективную передачу данных.

— Прием ответов от логистических провайдеров: Микросервис должен быть способен принимать ответы от логистических провайдеров, включая информацию, такую как номера отслеживания грузов, ожидаемые сроки доставки и другие релевантные сведения.

— Отправка ответов пользователям: Микросервис должен быть способен передавать ответы, полученные от логистических провайдеров, обратно пользователям, обеспечивая им реальное время обновления информации о грузах.

2.6.2 Архитектура системы

Микросервис обработки заказов спроектирован как RESTful API с использованием Python и веб-фреймворка Flask. Микросервис контейнеризован с помощью Docker и развернут в кластере Kubernetes, обеспечивая масштабируемость, надежность и высокую доступность.

2.6.3 Компоненты и технологии

Микросервис обработки заказов состоит из следующих компонентов:
API-шлюз: Построенный с использованием Flask, API-шлюз отвечает за прием и валидацию данных о заказе от пользователей. Он также обрабатывает аутентификацию и авторизацию с помощью OAuth 2.0.

Брокер сообщений: RabbitMQ используется в качестве брокера сообщений для обеспечения связи между API-шлюзом и интерфейсами логистических провайдеров. RabbitMQ обеспечивает надежную и последовательную доставку сообщений.

Интерфейсы логистических провайдеров: Эти интерфейсы отвечают за отправку данных о заказе логистическим провайдерам и прием ответов. Они построены с использованием Python и общаются с логистическими провайдерами с помощью API или других механизмов интеграции.

База данных: PostgreSQL используется в качестве базы данных для хранения данных о заказе и ответов от логистических провайдеров.

Контейнеризация: Docker используется для контейнеризации микросервиса, обеспечивая легкую развертку и управление в кластере Kubernetes.

Оркестрация: Kubernetes используется для оркестрации развертывания и масштабирования микросервиса, обеспечивая высокую доступность и масштабируемость.

2.6.4 Детали реализации

Реализация микросервиса обработки заказов включает следующие шаги:

— Реализация API-шлюза: API-шлюз реализуется с использованием Flask, с конечными точками для приема данных о заказе от пользователей и отправки ответов обратно пользователям. OAuth 2.0 используется для аутентификации и авторизации.

— Конфигурация брокера сообщений: RabbitMQ конфигурируется для обработки очередей сообщений для связи между API-шлюзом и интерфейсами логистических провайдеров.

— Реализация интерфейсов логистических провайдеров. Интерфейсы логистических провайдеров реализуются с использованием Python, с API или другими механизмами интеграции для связи с логистическими провайдерами.

— Проектирование базы данных. База данных PostgreSQL проектируется для хранения данных о заказе и ответов от логистических провайдеров, с таблицами для заказов, грузов и ответов логистических провайдеров.

— Контейнеризация и оркестрация. Микросервис контейнеризуется с помощью Docker и развертывается в кластере Kubernetes, обеспечивая масштабируемость, надежность и высокую доступность.

Микросервис обработки заказов является критическим компонентом ILDMES, отвечающим за управление потоком информации между пользователями и логистическими провайдерами. Используя архитектуру микросервисов и технологии, такие как Python, Flask, RabbitMQ, Docker, Kubernetes и PostgreSQL, микросервис обеспечивает масштабируемое, надежное и безопасное решение для обработки заказов. Детали реализации,

описанные в этой главе, предоставляют полное представление о архитектуре и функциональности микросервиса.

3.1 Архитектура программно-аппаратного комплекса.

Архитектура программно-аппаратного комплекса [30] является важнейшим компонентом проекта, поскольку она определяет общую производительность, масштабируемость и надежность системы. Архитектура должна быть разработана с учетом специфических требований проекта, включая обработку, хранение и передачу данных.

Предлагаемая архитектура состоит из следующих компонентов:

Front-end: веб-интерфейс, который позволяет пользователям взаимодействовать с системой, включая ввод, извлечение и анализ данных.

Серверная часть: серверный компонент, который обрабатывает и хранит данные, включая обработку, хранение и передачу данных.

База данных: централизованное хранилище, в котором хранятся все данные, включая данные из различных источников и форматов.

Сетевая инфраструктура: высокоскоростная сеть, обеспечивающая передачу данных между компонентами.

3.2 Существующие аналоги и их характеристики

Было выявлено несколько существующих аналогов предлагаемой системы, в том числе системы управления логистикой, такие как: Manhattan Associates, JDA и SAP, предоставляют комплексные возможности для управления логистикой, включая управление данными, управление транспортировкой и управление складом. Системы управления цепочками

поставок, такие как Oracle, Infor и Epicor, предоставляют комплексные возможности для управления цепочками поставок, включая закупки, управление запасами и дистрибуцию. Платформы для анализа данных, такие как Tableau, Power BI и QlikView, предоставляют возможности визуализации и аналитики данных, включая интеграцию, обработку и создание отчетов.

Характеристики этих аналогов включают в себя:

Масштабируемость: способность обрабатывать большие объемы данных и пользователей.

Гибкость: способность адаптироваться к изменяющимся бизнес-требованиям и процессам.

Интеграция: возможность интеграции с существующими системами и приложениями.

Безопасность: способность обеспечивать конфиденциальность, целостность и доступность данных.

Характеристики предлагаемой системы аналогичны характеристикам систем управления логистикой и цепочками поставок, с акцентом на масштабируемость, гибкость и интеграцию. Функции безопасности системы сопоставимы с функциями платформ для анализа данных.

Таким образом архитектура и характеристики предлагаемой системы разработаны с учетом специфических требований проекта, включая обработку, хранение и передачу данных. Масштабируемость, гибкость и возможности интеграции системы сопоставимы с возможностями существующих аналогов, с акцентом на управление логистикой и цепочками поставок. Функции безопасности системы предназначены для обеспечения конфиденциальности, целостности и доступности данных.

3.3 Эксперимент на независимых данных

Для обеспечения объективной оценки качества предлагаемого инструмента необходимо провести экспериментальное применение системы. Это позволит нам оценить производительность, удобство использования и эффективность инструмента в реальных сценариях.

Экспериментальный проект будет включать в себя следующие этапы:

— Настройка системы: Предлагаемая система будет установлена и сконфигурирована в соответствии со спецификациями проекта.

— Сбор данных: Набор данных будет собран на основе реального логистического сценария, включая данные об отгрузках, запасах и транспортировке.

Экспериментальное тестирование: Система будет протестирована группой специалистов по логистике, которые будут использовать систему для выполнения различных задач, таких как ввод данных, анализ данных и принятие решений.

Показатели оценки: Производительность системы будет оцениваться с использованием набора показателей, включая:

— Точность: точность возможностей системы по обработке и анализу данных.

— Скорость: Скорость обработки и анализа данных, предоставляемых системой.

— Удобство использования: простота использования системы, включая пользовательский интерфейс и навигацию.

— Эффективность: Эффективность системы в поддержке принятия решений и улучшении логистических операций.

Анализ данных: Данные, собранные в ходе экспериментального тестирования, будут проанализированы для оценки производительности системы и определения областей для улучшения.

Ожидается, что экспериментальное применение предлагаемого инструмента приведет к следующим результатам:

— Повышенная точность: Система продемонстрирует высокую точность обработки и анализа данных, что сократит количество ошибок и улучшит процесс принятия решений.

— Повышенная скорость: Система продемонстрирует возможности быстрой обработки и анализа данных, что позволит специалистам в области логистики принимать своевременные решения.

— Повышенная удобство использования: Система будет обладать интуитивно понятным и дружественным интерфейсом, что упростит использование и навигацию для специалистов в области логистики.

— Повышение эффективности: Система продемонстрирует свою способность поддерживать процесс принятия решений и улучшать логистические операции, что приведет к повышению эффективности и снижению затрат.

Вывод: экспериментальное применение предложенного инструмента является решающим шагом в оценке его качества и эффективности. Проведя тщательное экспериментальное тестирование, мы можем гарантировать, что система соответствует требованиям профессионалов в области логистики и обеспечивает высокий уровень производительности, удобства использования и результативности.

3.4 Тестирование прототипа

После завершения этапа разработки прототип будет подвергнут всестороннему тестированию для обеспечения его качества, надежности и удобства использования. План тестирования будет включать следующие этапы:

— Шаг 1: Модульное тестирование

Прототип будет разделен на более мелкие компоненты, и каждый компонент будет протестирован индивидуально, чтобы убедиться в его правильной работе. Тестирование будет сосредоточено на функциональности отдельных компонентов, обработке и хранении данных.

— Шаг 2: Интеграционное тестирование

Отдельные компоненты будут интегрированы в единую систему, и система будет протестирована в целом. Тестирование будет сосредоточено на общей функциональности системы, потоке данных и целостности данных.

— Шаг 3: Тестирование системы

Прототип будет протестирован в имитируемой среде, имитирующей реальные сценарии. Тестирование будет сосредоточено на производительности, масштабируемости и надежности системы.

— Шаг 4: Приемочное тестирование пользователя

Для тестирования прототипа в реальных условиях будет отобрана группа специалистов по логистике. Тестирование будет сосредоточено на удобстве использования системы, пользовательском интерфейсе и общем опыте работы с пользователем.

— Шаг 5: Регрессионное тестирование

После внесения каких-либо изменений или обновлений в прототип будет проведено регрессионное тестирование, чтобы убедиться, что изменения не привели к появлению новых ошибок или не повлияли на общую функциональность системы.

— Шаг 6: Финальное тестирование

Прототип пройдет заключительный этап тестирования, чтобы убедиться, что он соответствует всем требованиям и спецификациям. Тестирование будет сосредоточено на общей производительности, надежности и удобстве использования системы.

Среда тестирования будет настроена таким образом, чтобы имитировать реальные сценарии, включая:

— Моделирование данных

— Моделирование пользователей

— Моделирование сетевых условий

— Моделирование конфигураций аппаратного и программного обеспечения

Будут использоваться следующие инструменты тестирования:

— Автоматизированные инструменты тестирования (например, Selenium, Appium)

— Инструменты ручного тестирования (например, тестовые примеры, контрольные списки)

— Инструменты отладки (например, инструкции для печати, отладчики)

3.5 Перспективы совершенствования инструмента

Инструмент был разработан в условиях нехватки времени и ресурсов, что могло повлиять на качество результатов. Чтобы оценить потенциальное влияние этих ограничений, мы определим их и представим оценки потенциального повышения эффективности, если эти ограничения будут устранены.

Ограничения и потенциальное воздействие. Временные ограничения: Ограниченное время, отведенное на разработку, могло привести к созданию менее всеобъемлющего или усовершенствованного инструмента. Устранение этих ограничений могло бы привести к созданию более надежного и удобного в использовании инструмента.

Предполагаемый прирост эффективности: 10-15%

Ограниченность ресурсов: Ограниченность ресурсов, доступных для разработки, возможно, ограничила возможности и функциональность инструмента. Ослабление этих ограничений может привести к созданию более комплексного и многофункционального инструмента.

Предполагаемый прирост эффективности: 15-20%

Качество и доступность данных: Качество и доступность данных, используемых для разработки, могли быть ограниченными, что могло повлиять на точность и надежность инструмента. Повышение качества и доступности данных может привести к созданию более точного и надежного инструмента.

Предполагаемый прирост эффективности: 5-10%

Предлагаемые улучшения

Исходя из потенциального эффекта от ослабления ограничений, мы предлагаем следующие улучшения:

Улучшение пользовательского интерфейса и взаимодействия с пользователем: улучшите пользовательский интерфейс и взаимодействие с пользователем путем учета отзывов пользователей и проведения тестирования удобства использования.

Расширьте набор функций и функционал: Расширьте набор функций и функционал инструмента, включив в него дополнительные функции и возможности, которые были неосуществимы в первоначальном объеме.

Повышение качества и доступности данных: Повышение качества и доступности данных, используемых для разработки, путем включения дополнительных источников данных и проведения проверок качества данных.

Интеграция с другими системами и инструментами: Интеграция инструмента с другими системами и инструментами для повышения его функциональности и удобства использования.

Проведите тщательное тестирование и валидацию: Проведите тщательное тестирование и валидацию инструмента, чтобы обеспечить его точность, надежность и производительность.

Вывод: предлагаемые усовершенствования направлены на устранение ограничений и потенциального влияния нехватки времени и ресурсов. Ослабив эти ограничения, мы сможем создать более всеобъемлющий, многофункциональный и удобный в использовании инструмент, который может принести еще больше пользы специалистам в области логистики.

3.6 Условия для остановки автоматизированного решения

Существуют ограничения на качество изменений, которые могут быть внесены в исследуемый объект, за пределами которых применение предлагаемого инструмента может не дать ожидаемых результатов. Следующие условия указывают, когда необходимо прекратить использование автоматизированного решения:

Сложность объекта: если сложность объекта превышает определенный порог, автоматизированное решение может оказаться неспособным эффективно анализировать и обрабатывать данные, что приведет к неточным или неполным результатам.

Качество данных: если качество данных, используемых для анализа, низкое, автоматизированное решение может не обеспечить получение надежных результатов. В таких случаях может потребоваться ручной анализ или очистка данных.

Динамика объекта: если динамика объекта (например, изменения в поведении, структуре или взаимосвязях) слишком быстрая или непредсказуемая, автоматизированное решение может не успевать за изменениями, что приводит к неточным или устаревшим результатам.

Ограниченное знание предметной области: если автоматизированное решение применяется в области, где разработчики инструмента не обладают достаточным опытом или знаниями, результаты могут быть неточными или неполными.

Недостаточные вычислительные ресурсы: если автоматизированное решение требует недоступных или недостаточных вычислительных ресурсов, результаты могут быть запоздалыми, неточными или неполными.

Размер объекта: если объект слишком велик или сложен, автоматизированное решение может оказаться неспособным эффективно обрабатывать данные, что приведет к снижению производительности или неточным результатам.

Отсутствие четких целей или задач: если цели или задачки анализа неясны или двусмысленны, автоматизированное решение может оказаться неспособным обеспечить значимые результаты.

3.7 Экономическое обоснование внедрения системы-прототипа

Предлагаемая система-прототип для управления данными и обмена ими в международной логистике потенциально может принести значительные экономические выгоды отрасли.

— Повышенная эффективность: автоматизирует многие ручные процессы, сокращая время и усилия, необходимые для управления данными и обмена ими. Это приведет к повышению эффективности и производительности, что приведет к снижению затрат и повышению конкурентоспособности.

— Повышенная точность: снизит риск человеческой ошибки, гарантируя точность и надежность данных. Это приведет к улучшению процесса принятия решений и снижению затрат, связанных с ошибками.

— Расширенная совместная работа: позволит осуществлять совместную работу в режиме реального времени и обмен данными между заинтересованными сторонами, улучшая коммуникацию и снижая риск недопонимания.

— Снижение затрат: уменьшит необходимость в ручном вводе данных, печати и рассылке по почте, что приведет к экономии средств.

— Увеличение выручки: позволит логистическим компаниям предлагать новые услуги и повышать свою конкурентоспособность, что приведет к увеличению выручки.

— Повышение удовлетворенности клиентов: будет предоставлять клиентам обновления в режиме реального времени и улучшенную видимость, что приведет к повышению удовлетворенности клиентов и их лояльности.

— Сокращение сроков выполнения заказа: обеспечит более быстрый и точный обмен данными, сократит сроки выполнения заказа и повысит эффективность цепочки поставок.

— Улучшенное соответствие требованиям: обеспечит соответствие нормативным актам и стандартам, снизив риск штрафов и ущерба репутации.

— Масштабируемость: ILDMES будет разработан с учетом потребностей логистической отрасли, что позволит ей справляться с растущими объемами и сложностью.

ЗАКЛЮЧЕНИЕ

Развитие новой информационной технологии для управления и обмена данными в международной логистике становится все более необходимым в современном мире. Продвинутые технологии помогают улучшить эффективность и прозрачность. Новые информационные технологии, такие как IoT, блокчейн, облачные решения, позволяют компаниям намного эффективнее управлять и отслеживать грузы, контейнеры, склады и транспортные средства в цепочке поставок. Современные системы управления цепочкой поставок и обмена данными обеспечивают более быструю, точную и своевременную передачу данных и информации между участниками цепочки поставок. Новые технологии позволяют автоматизировать многие процессы в цепочке поставок, что способствует снижению издержек и повышению конкурентоспособности компаний на рынке. Развитие новых методов шифрования и защиты данных важно для предотвращения утечек информации и соблюдения правил конфиденциальности, особенно при обмене данными через публичные сети. С помощью новых информационных технологий компании могут создавать более устойчивые и устойчивые цепочки поставок, что помогает им адаптироваться к переменным условиям рынка и ограничениям.

Изучение различных систем управления данными и операциями в логистике, рассмотрение их преимуществ и недостатков, а также анализ применения машинного обучения позволяет нам заключить, что одной из основных проблем в логистике является интеграция различных информационных систем между собой и скорость обработки и обмена данными.

Интеграция информационных систем – критически важный аспект, поскольку различные системы могут работать в изоляции друг от друга, что затрудняет полноценное управление данными и операциями в целом.

Необходима единая платформа, способная интегрировать данные из различных источников для обеспечения единого понимания процессов и принятия более глобальных стратегических решений. Одновременно с этим, важно обратить внимание на скорость обработки и обмена данными. В условиях быстро меняющегося рынка и повышающихся требований к сервису, необходимо обеспечить оперативное получение и анализ данных для быстрого реагирования на изменения в окружающей среде.

Интеграция различных информационных систем в единую платформу повысит прозрачность и связанность логистических операций, обеспечивая единую точку доступа к данным и упрощая процессы принятия решений. Благодаря улучшенной скорости обработки данных можно будет оперативно анализировать информацию, проводить прогнозы, оптимизировать маршруты и улучшать управление запасами.

Таким образом, разработка и внедрение единой платформы, способной эффективно работать с различными типами данных и с различными системами, является важным шагом для современных логистических компаний, помогая им быть конкурентоспособными, адаптивными и успешными на рынке.

В этой работе была изучена существующая система управления и обменом данными в международной логистике, были выявлены проблемы и ограничения. В качестве решения предложена разработка прототипа ILDMES, который направлен на решение текущих проблем и ограничений существующей системы. Система разработана таким образом, чтобы обеспечить бесперебойный и эффективный способ для всех заинтересованных сторон в цепочке поставок обмениваться данными, тем самым повышая общую эффективность и точность логистического процесса, быть гибкой и масштабируемой, позволяя ей адаптироваться к различным форматам и структурам данных и обрабатывать большие объемы данных.
Прототип системы был создан с использованием комбинации технологий, включая языки программирования, базы данных и API.

Система имеет ряд преимуществ по сравнению с существующей системой. Во-первых, она предоставляет стандартизированную и централизованную платформу для обмена и обработки данных, которая устраняет необходимость в ручном вводе данных и снижает риск ошибок. Во-вторых, она обеспечивает обмен и обработку данных в режиме реального времени, что позволяет быстрее и точнее принимать решения. В-третьих, он обеспечивает гибкую и масштабируемую архитектуру, которая позволяет адаптироваться к различным форматам и структурам данных и обрабатывать большие объемы данных.

Следует отметить, что прототип системы, разработанный в этой дипломной работе, является значительным улучшением по сравнению с существующей системой и потенциально может оптимизировать взаимодействие между всеми участниками цепочки поставок в международной логистике. Прототип системы способен существенно преобразовать существующую систему логистических операций, обеспечив новый уровень эффективности, точности и взаимодействия. Автоматизируя многие ручные процессы, система сократит время и усилия, необходимые для управления данными и обмена ими, высвободив ресурсы для более стратегической деятельности.

По мере постепенного внедрения прототипа системы это окажет влияние на всю цепочку поставок, снижая нагрузку на участников и высвобождая ресурсы для компаний. Это приведет к снижению затрат на логистику, что в конечном итоге отразится на стоимости товаров.

Преимущества прототипа системы будут многогранными:

— Снижение затрат: Автоматизируя ручные процессы и уменьшая количество ошибок, компании смогут снизить свои издержки и повысить конкурентоспособность.

— Повышение эффективности: Система обеспечит более быстрый и точный обмен данными, сократит время выполнения заказов и повысит эффективность цепочки поставок.
— Расширенное сотрудничество: Система обеспечит платформу для улучшения коммуникации и взаимодействия между заинтересованными сторонами, что сократит количество недоразумений и ошибок.
— Повышенная прозрачность: Система обеспечит видимость цепочки поставок в режиме реального времени, позволяя компаниям принимать более обоснованные решения и быстро реагировать на изменения.

Внедрение прототипа системы окажет значительное влияние на отрасль логистики, что позволит компаниям повысить свою конкурентоспособность.: Снижая затраты и повышая эффективность, компании смогут повысить свою конкурентоспособность, улучшить свои финансовые показатели и получить конкурентное преимущество. Предоставляя информацию о цепочке поставок в режиме реального времени, компании смогут улучшить качество обслуживания своих клиентов.

В заключение, прототип системы может произвести улучшения в логистической отрасли, выведя ее на новый уровень эффективности, точности и совместной работы. По мере постепенного внедрения она окажет значительное влияние на отрасль, позволяя компаниям снижать издержки, повышать эффективность и улучшать качество обслуживания клиентов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Шумаев, В. А. Основы логистики: учеб. пособие / В. А. Шумаев. М.: Юридический институт МИИТ, 2016. — 314 с.
2. 01 - Введение в логистику - История. – URL: https://studfile.net/preview/5409140/ (дата обращения: 30.05.2024). – Текст : электронный.
3. Логистика в отечественной экономике // Vuzlit. – URL: https://vuzlit.com/256773/razvitie_logistiki_nauchnogo_ekonomicheskogo_napravleniya – Текст: электронный.
4. Тема 1 Возникновение и развитие логистики. – URL: https://studfile.net/preview/9715201/ (дата обращения: 30.05.2024). – Текст: электронный.
5. Европейская логистическая ассоциация (ELA) в 1994 г. // URL: https://www.elalog.eu/ – Текст: электронный.
6. Внешнеторговые транспортные операции и логистика: учеб. пособие / Под ред. проф. Д. С. Николаева. – М.: АНКИЛ, 1998. – 317 с.
7. Корпоративная логистика. 300 ответов на вопросы профессионалов / под общ. и науч. ред. В. И. Сергеева. М.: НИЦ ИНФРА-М, 2004. – 976 с.
8. Логистика: учеб. пособие / под ред. Б. А. Аникина, Т. А. Родкиной. М.: ТК Велби, Проспект, 2007. – 216 с.
9. Неруш Ю. М. Логистика: учебник. 4-е изд., перераб. и доп. М.: ТБ Велби. Проспект, 2007. – 517 с.
10. Уотерс Д. Логистика. Управление цепью поставок / пер. с англ. М.: ЮНИТИ-ДАНА, 2003. – 507 с.
11. Степанов В. И. Логистика: учебник. М.: ТК Велби; Проспект, 2006. – 487 с.

12. Плоткин Б. К. Основы логистики: учеб. пособие. Л.: Изд-во ЛФЭИ, 1991. – 53 с.
13. Global Logistics Market Report and Forecast 2024-2032. – URL: https://www.researchandmarkets.com/reports/5775157/global-logistics-market-report-forecast (дата обращения: 21.03.2024). – Текст: электронный.
14. Основы международной логистики: учеб. пособие [Н.С. Рычихина и др.;] Иван. гос. хим. технол. ун-т. – Иваново, 2017. – 159с.
15. Пустынникова Е. В. Интегрированная логистика / Е.В. Пустынникова. - Санкт-Петербург: Интермедия, 2017. - 312 с. - ISBN 978-5-4383-0151-6.
16. Презентация на тему: «Международные аспекты логистики Преподаватель – БЕЛОЗЕРЦЕВА Наталья Петровна Кафедра маркетинга и коммерции 2007.» – URL: http://www.myshared.ru/slide/838805/ (дата обращения: 31.05.2024). – Текст: электронный.
17. Основы логистики: учебное пособие / С.А. Черникова; М-во с.-х. РФ, федеральное гос. бюджетное образов. учреждение высшего образов. «Пермский гос. аграрно-технолог. ун-т им. акад. Д.Н. Прянишникова». – Пермь: ИПЦ «Прокростъ», 2018. – 240 с. ISBN 978-5-94279-386-9
18. Кристофер М. Логистика и управление цепочками поставок / Пер. с англ. В. Кузин. Питер, год издания: 2004 – 315 с. ISBN: 5-94723-168-9, 0-273-63049-0
19. Середа С.В. Применение машинного зрения в логистике [Текст] / С.В. Середа // Sciences of Europe. – 2021. – № 65-1. – 50 с.
20. Machine learning in logistics: technology breakdown & 10 use cases // Itransition. – URL: https://www.itransition.com/ – Текст: электронный.
21. Классификация методов и моделей управления логистической деятельностью. – URL: https://cyberleninka.ru/article/n/klassifikatsiya-metodov-i-modeley-upravleniya-logisticheskoy-deyatelnostyu/viewer – Текст : электронный.

22. Лоре А. Проектирование веб-API / Пер. с англ. Д. А. Беликова. – М.: ДМК Пресс, 2020.– 440 с. ISBN 978-5-97060-861-6
23. Управление цепью поставок (SCM): учеб. пособие / сост. П. П. Крылатков, М.А. Прилуцкая. — Екатеринбург: Изд-во Урал. ун-та, 2018.— 140 с. ISBN 978-5-7996-2269-5
24. Глушак, Е. В. Введение в Интернет вещей: учебное пособие / Е.В. Глушак, А.В. Куприянов. – Самара: Издательство Самарского университета, 2023. – 104 с.: ил. ISBN 978-5-7883-2010-6
25. Никульчев Е.В., Лукьянчиков О.И., Ильин Д.Ю. Облачные технологии [Электронный ресурс]: учеб. пособие / Никульчев Е.В., Лукьянчиков О.И., Ильин Д.Ю. — М.: РТУ МИРЭА, 2019 – 78 с.
26. Гаррисон А., Ремко В. Г. Логистика. Стратегия управления и конкурирования через цепочки поставок: учебник: перевод третьего английского издания. Москва: Дело и сервис, 2010 – 368 с. ISBN 978-5-8018-0469-9
27. Трансформация индустрии контейнерных перевозок // Digital container shipping association. – URL: https://dcsa.org/ – Текст: электронный.
28. Digital Innovation Ecosystem // IATA. – URL: https://www.iata.org/ – Текст: электронный.
29. FIATA Statutes // International Federation of Freight Forwarders Associations. – URL: https://fiata.org/ – Текст: электронный.
30. Сорокин, С. А. Архитектура программно-аппаратных комплексов: методические указания / Сорокин С. А.,Горшков А. В. - Москва : РТУ МИРЭА, 2023. - 61 с. - Б. ц. - Текст: непосредственный.

Приложение А

app.py

```
from flask import Flask, request, jsonify
from flask import render_template
from flask import send_file
from flask_sqlalchemy import SQLAlchemy
from marshmallow import Schema, fields
import os
from flask.signals import got_request_exception

app = Flask(__name__)
app.config["SQLALCHEMY_DATABASE_URI"] =
"postgresql://postgres:1122@localhost:5432/postgres"
app.config["SQLALCHEMY_TRACK_MODIFICATIONS"] = False

db = SQLAlchemy(app)

class Order(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    sender = db.Column(db.String(100), nullable=False)
    recipient = db.Column(db.String(100), nullable=False)
    type_container = db.Column(db.String(100), nullable=False)
    load_date = db.Column(db.DateTime, nullable=False)
    total_amount = db.Column(db.Float, nullable=False)

    def __repr__(self):
        return f"Order({self.id}, {self.sender}, {self.recipient},
{self.type_container}, {self.load_date}, {self.total_amount})"

class OrderSchema(Schema):
    class Meta:
        model = Order
        sqla_session = db.session

order_schema = OrderSchema()
orders_schema = OrderSchema(many=True)


@app.route("/")
def index():
    return render_template("index.html")

@app.route("/favicon.ico")
def favicon():
    return send_file("static/favicon.ico",
mimetype='image/vnd.microsoft.icon')

@app.route("/orders", methods=["GET"])
def get_orders():
```

```
    orders = Order.query.all()
    return jsonify(orders_schema.dump(orders))

@app.route("/orders", methods=["POST"])
def create_order():
    data = request.get_json()
    order = Order(sender=data["sender"], recipient=data["recipient"],
type_container=data["type_container"], load_date=data["load_date"],
total_amount=data["total_amount"])
    db.session.add(order)
    db.session.commit()
    return jsonify(order_schema.dump(order))

@app.route("/orders/<int:order_id>", methods=["GET"])
def get_order(order_id):
    order = Order.query.get(order_id)
    if order is None:
        return jsonify({"error": "Order not found"}), 404
    return jsonify(order_schema.dump(order))

@app.route("/orders/<int:order_id>", methods=["PUT"])
def update_order(order_id):
    order = Order.query.get(order_id)
    if order is None:
        return jsonify({"error": "Order not found"}), 404
    data = request.get_json()
    order.sender = data["sender"]
    order.recipient = data["recipient"]
    order.type_container = data["type_container"]
    order.load_date = data["load_date"]
    order.total_amount = data["total_amount"]
    db.session.commit()
    return jsonify(order_schema.dump(order))

@app.route("/orders/<int:order_id>", methods=["DELETE"])
def delete_order(order_id):
    order = Order.query.get(order_id)
    if order is None:
        return jsonify({"error": "Order not found"}), 404
    db.session.delete(order)
    db.session.commit()
    return jsonify({"message": "Order deleted"})

if __name__ == "__main__":
    app.run(debug=True)
```

Данный код представляет веб-приложение на основе Flask, которое реализует CRUD-операции (Create, Read, Update, Delete) для сущности Order в базе данных PostgreSQL.

1. Сначала создается Flask-приложение и настраивается подключение к базе данных PostgreSQL с помощью SQLAlchemy. Создается модель Order, которая представляет таблицу Order в базе данных.
2. Далее определяется Schema для сериализации объектов Order и списков Order в JSON. Это делает работу с JSON данными в приложении более удобной.
3. Приложение имеет несколько маршрутов:
— / - возвращает содержимое файла index.html.
— /favicon.ico - возвращает иконку сайта.
— /orders - работает как метод GET для получения списка всех заказов, и метод POST для создания нового заказа.
— /orders/<order_id> - работает как метод GET для получения конкретного заказа, метод PUT для обновления заказа и метод DELETE для удаления заказа.
4. В методах вызываются соответствующие методы SQLAlchemy для выполнения операций на базе данных (например, добавление, обновление, удаление записей).
5. Приложение запускается при условии, что скрипт запущен непосредственно (т.е. это не импорт в другом модуле), в режиме отладки.
Этот код создает RESTful API для работы с заказами в базе данных PostgreSQL с помощью Flask. Разделение логики работы с базой данных и описания протокола маршрутизации и работы с данными (CRUD) позволяет лучшую организацию и сопровождение кода.

Приложение В

models.py

```
from app import db

class Order(db.Model):
    __table_args__ = {'extend_existing': True}
    id = db.Column(db.Integer, primary_key=True)
    sender = db.Column(db.String(100), nullable=False)
    recipient = db.Column(db.String(100), nullable=False)
    type_container = db.Column(db.String(100), nullable=False)
    load_date = db.Column(db.DateTime, nullable=False)
    total_amount = db.Column(db.Float, nullable=False)

    def __repr__(self):
        return f"Order({self.id}, {self.sender}, {self.recipient},
{self.type_container}, {self.load_date}, {self.total_amount})"


db.Model.metadata.create_all(db.engine)


# В этом коде мы создаем модель Order с четырьмя атрибутами:
# id: уникальный идентификатор заказа
# sender: отправитель
# recipient: получатель
# type_container: тип контейнера
# load_date: дата загрузки
# total_amount: общая сумма заказа
# Модель наследуется от db.Model, что позволяет использовать SQLAlchemy для
работы с базой данных.
```

Этот код создает модель данных для объектов "Order" (заказов), которая будет использоваться для взаимодействия с базой данных с помощью SQLAlchemy.

1. Сначала импортируется объект db из app.

2. Определяется класс Order, который является подклассом db.Model, что делает его моделью базы данных.

3. Атрибуты класса Order:

— id: целое число, является первичным ключом (primary key) заказа (идентификатор).

— sender: строка (строковое поле в базе данных), обязательное для заполнения, содержащее информацию об отправителе.

— recipient: строка, обязательное для заполнения, содержащее информацию о получателе.

— typecontainer: строка, обязательное для заполнения, содержащее информацию о типе контейнера.

— loaddate: объект DateTime, обязательное для заполнения, содержащее информацию о дате загрузки.

— totalamount: число с плавающей запятой, обязательное для заполнения, содержащее общую сумму заказа.

4. Метод repr определяет, как объект класса Order будет представлен в виде строки. В данном случае, он возвращает строку с атрибутами объекта.

5. После определения класса Order, вызывается метод createall(db.engine) для создания всех таблиц в базе данных (соответствующих модели Order).

Этот код позволяет создать таблицу заказов в базе данных и предоставляет структуру для хранения информации о заказах.

Приложение С

database.py

```
import os
from sqlalchemy import create_engine
from sqlalchemy.ext.declarative import declarative_base
from sqlalchemy.orm import sessionmaker

# настройки базы данных
DB_HOST = 'localhost'
DB_NAME = 'postgres'
DB_USER = 'postgres'
DB_PASSWORD = '1122'

# создаем строку подключения к базе данных
database_url =
f'postgresql://{DB_USER}:{DB_PASSWORD}@{DB_HOST}:5432/{DB_NAME}'

# создаем движок для работы с базой данных
engine = create_engine(database_url)

# создаем сессию для работы с базой данных
Session = sessionmaker(bind=engine)

# создаем базу данных для моделей
Base = declarative_base()

# функция для создания таблиц в базе данных
def create_tables():
    Base.metadata.create_all(engine)

# функция для создания сессии
def get_session():
    return Session()
```

Этот код создает и настраивает базу данных PostgreSQL для работы с приложением.

1. Он импортирует необходимые модули: os, create_engine из SQLAlchemy для настройки подключения к базе данных, declarative_base и sessionmaker из SQLAlchemy ORM для работы с объектами базы данных.

2. Затем устанавливаются параметры базы данных, такие как хост (DB_HOST), имя базы данных (DB_NAME), имя пользователя (DB_USER) и пароль (DB_PASSWORD).

3. Создается строка подключения к базе данных (database_url) с использованием параметров, указанных выше.

4. Создается движок SQLAlchemy (engine) для работы с базой данных, используя созданную строку подключения.

5. Создается сессия SQLAlchemy (Session) с привязкой к движку.

6. Создается база данных для определения моделей объектов базы данных (Base).

7. Определяются две функции:

— create_tables(): Функция использует метаданные базы данных (Base.metadata) для создания всех таблиц в базе данных, связанных с моделями данных.

— get_session(): Функция возвращает сессию базы данных, созданную выше.

8. В итоге, этот код предоставляет удобный способ настройки базы данных PostgreSQL, создания соединения к ней, создания всех необходимых таблиц и получения сессии для выполнения запросов к базе данных.