

## ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ НА ПРОИЗВОДСТВЕННУЮ (ПРЕДДИПЛОМНУЮ) ПРАКТИКУ

В объеме 144 часа

Абдулзагирова Исы Османовича

ФИО обучающегося полностью

специальность 09.02.07 Информационные системы и программирование

код и наименование

курса 4 семестра 8 группа 4ИС-9-19

Место прохождения практики

«Биор умней»

наименование организации

109029, город Москва, ул. Нижегородская 34 стр. 4

адрес организации места прохождения практики

Срок прохождения практики с 23.03.2023 г по 19.04.2023 г.

### Виды выполняемых работ

| Перечень работ (задание)  | Количество часов |
|---|------------------|
| Ознакомление с целями и задачами практики   | 6                |
| Знакомство с правилами внутреннего распорядка, рабочим местом   | 6                |
| Практическое изучение предмета проектирования   | 26               |
| Поиск дополнительной информации, необходимость в которой возникла для решения вопросов, возникших в ходе знакомства с предметной областью дипломного проектирования | 26               |
| Подготовка данных для реализации автоматизированной системы   | 26               |
| Подготовка данных для реализации предмета проектирования  | 26               |
| Анализ собранного материала по программным средствам  | 28               |
| <b>Итого</b>  | <b>144</b>       |

Руководитель практики \_\_\_\_\_

подпись

А.К. Морозов

ФИО

Дата выдачи задания 20 апреля 2023 г.

Дата сдачи отчета 17 мая 2023 г.

Задание принял к исполнению \_\_\_\_\_

подпись

И.О. Абдулзагиров

ФИО студента



Частное профессиональное образовательное учреждение  
**«КОЛЛЕДЖ СОВРЕМЕННЫХ ТЕХНОЛОГИЙ И  
МЕДИЦИНЫ»**

---

**ДНЕВНИК**  
**ПРОХОЖДЕНИЯ ПРОИЗВОДСТВЕННОЙ**  
**(ПРЕДДИПЛОМНОЙ) ПРАКТИКИ**

Абдулзагиров Иса Османович

---

Ф.И.О. студента

курс 4, группа 4ИС-9-19, очной формы обучения

специальность 09.02.07 Информационные системы и программирование

---

код и наименование

Место прохождения практики

«Биор умней»

---

наименование организации

Руководитель от колледжа \_\_\_\_\_ А.К. Морозов \_\_\_\_\_

Руководитель от организации \_\_\_\_\_

В объеме   144   часов

г. Москва, 2023 г.

**Запись о работах, выполненных во время прохождения  
производственной (преддипломной) практики**

| Дата                       | Выполняемы работы   |
|----------------------------|---|
| 20.04.2023                 | Ознакомление с целями и задачами практики   |
| 21.04.2023                 | Знакомство с правилами внутреннего распорядка, рабочим местом   |
| 22.04.2023 –<br>25.04.2023 | Практическое изучение предмета проектирования   |
| 26.04.2023 –<br>02.05.2023 | Поиск дополнительной информации, необходимость в которой возникла для решения вопросов, возникших в ходе знакомства с предметной областью дипломного проектирования |
| 03.05.2023 –<br>10.05.2023 | Подготовка данных для реализации автоматизированной системы   |
| 11.05.2023 –<br>13.05.2023 | Подготовка данных для реализации предмета проектирования  |
| 15.05.2023 –<br>16.05.2023 | Анализ собранного материала по программным средствам  |
| 17.05.2023                 | Сдача отчета руководителю от колледжа   |

Руководитель от организации практики

\_\_\_\_\_

подпись

\_\_\_\_\_

ФИО

М.П.



Частное профессиональное образовательное учреждение  
**«КОЛЛЕДЖ СОВРЕМЕННЫХ ТЕХНОЛОГИЙ И  
МЕДИЦИНЫ»**

**Аттестационный лист производственной (преддипломной) практики**

ФИО обучающегося Абдулзагирова Исы Османовича  
Курс 4 группа 4ИС-9-19  
Специальность 09.02.07 Информационные системы и программирование  
Тема дипломного проекта «Разработка программного обеспечения для автоматизации учета клиентов»  
Место практики «Биор умней»  
Период практики 20.04.2023 г. – 17.05.2023 г.

Виды работ, выполненные студентом во время практики

| Формируемые ОК и ПК  | Перечень работ (задание)  | Количество часов, отведенное на выполнение работ |
|--|---|--|
| ОК 01., ОК 02.<br>ОК 03., ОК 04.<br>ОК 05., ОК 06.<br>ОК 07., ОК 08.<br>ОК 09., ОК 10.<br>ОК 11. | Ознакомление с целями и задачами практики   | 6  |
|  | Знакомство с правилами внутреннего распорядка, рабочим местом   | 6  |
|  | Практическое изучение предмета проектирования   | 26   |
|  | Поиск дополнительной информации, необходимость в которой возникла для решения вопросов, возникших в ходе знакомства с предметной областью дипломного проектирования | 26   |
|  | Подготовка данных для реализации автоматизированной системы   | 26   |
|  | Подготовка данных для реализации предмета проектирования  | 26   |
|  | Анализ собранного материала по программным средствам  | 28   |
| <b>Итого</b>   |   | <b>144</b>                                       |

Итоговая оценка \_\_\_\_\_  
дифференцированный зачет

17.05.2023 г. Руководитель практики \_\_\_\_\_ / А.К. Морозов  
подпись



## ОТЧЕТ О ПРОХОЖДЕНИИ ПРАКТИКИ

Производственная (преддипломная) практика

наименование вида практики

Абдулзагиров Иса Османович

Ф.И.О. студента

курс 4, группа 4ИС-9-19, очной формы обучения

специальность 09.02.07 Информационные системы и программирование

код и наименование

Место прохождения практики

«Биор умней»

наименование организации

Срок прохождения практики с 20.04.2023 г по 17.05.2023 г.

В объеме 144 часа

Отчет принят 17.05.2023 г.

Оценка \_\_\_\_\_

Преподаватель, Морозов А.К.

Дата, должность, ФИО руководителя практики

г. Москва, 2023 г.



## ХАРАКТЕРИСТИКА

Выдана Абдулгагирову Исе Османовичу, проходившему производственную практику в «Биор умней», по специальности 09.02.06 Сетевое и системное администрирование, с 2023 г. по 2023 г.

Руководитель ответственный за прохождение практики студентом удостоверяет следующие сведения об обучающемся:

1. Прошёл инструктаж по охране труда и технике безопасности на рабочем месте – 2023 г.
2. Выполняемые работы во время прохождения практики, их качество:
  - Степень соответствия уровня теоретической подготовки требованиям организации – Низкий / Средний / Высокий; соответствует / не соответствует.
  - Степень соответствия уровня практической подготовки требованиям организации – Низкий / Средний / Высокий; соответствует / не соответствует.
  - Готовность к применению теоретических знаний в практической деятельности – Частично готов / Готов.
  - Выполнение поручений руководителя практики – Добросовестно.
  - Нарушения трудовой дисциплины, пропуски рабочих дней без уважительной причины – Не допускал.
3. Трудовая дисциплина – Удовлетворительная / Хорошая;
4. Наличие поощрений, взысканий – Не имеется или имеются, то какие;
5. Оценка за практику – Удовлетворительно / Хорошо / Отлично;

Должность \_\_\_\_\_ /



## **Оглавление**

**ВВЕДЕНИЕ**

**ГЛАВА 1. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ**

1.1 Ознакомление с целями и задачами практики 10

1.2 Знакомство с правилами внутреннего распорядка, рабочим местом

**ГЛАВА 2. ПРАКТИЧЕСКАЯ ЧАСТЬ**

2.1 Практическое изучение предмета проектирования

2.2 Поиск дополнительной информации, необходимость в которой возникла для решения вопросов, возникших в ходе знакомства с предметной областью дипломного проектирования

**ГЛАВА 3. Работы выполненные в ходе практики**

3.1 Подготовка данных для реализации автоматизированной системы

3.2 Подготовка данных для реализации предмета проектирования

3.3 Анализ собранного материала по программным средствам

**ЗАКЛЮЧЕНИЕ**

**СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

## Введение

В процессе дипломного проектирования очень важно не только разработать работающую систему, но и провести некоторый комплекс работ, которые могут помочь в ее реализации. Одной из таких важных работ является практика, которая включает в себя несколько этапов. В начале, студент должен ознакомиться с целями и задачами практики. Он должен понимать, какие результаты будут ожидать, и какие задачи необходимо решить в процессе работы. Затем он знакомится с правилами внутреннего распорядка, правилами использования инструментов и технологий, правилами работы с документацией и прочими правилами, которые могут возникнуть в рамках работы. Следующим шагом является практическое изучение предмета проектирования. Это может включать в себя изучение технологий и платформ, изучение предметной области, в которой разрабатывается система, и понимание базовых концепций и принципов, связанных с проектированием систем. Практический опыт, полученный на этом этапе, может быть очень полезен при последующих этапах проекта.

В дальнейшем необходимо найти дополнительную информацию, которая понадобится для решения возникших вопросов, связанных с предметной областью проекта. Поиск дополнительных материалов, литературы, экспертных мнений и других источников может быть очень полезным в процессе проектирования и разработки системы. Важным этапом является подготовка данных для реализации автоматизированной системы. Это может включать в себя сбор, обработку и организацию данных, необходимых для разработки и реализации системы. В этом процессе может использоваться различное программное обеспечение, инструменты и технологии. Также студент должен подготовить данные для реализации предмета проектирования. Этот процесс включает в себя изучение требований проекта и подготовку данных, необходимых для проектирования

системы. Это может включать создание схем, моделей и прототипов, анализ и выбор используемых технологий и многое другое.

И наконец, очень важным этапом является анализ собранных материалов по программным средствам, который позволяет проанализировать собранную информацию, проверить ее релевантность и значимость. Это может помочь определить наилучшие практики, тренды и прогресс в данной предметной области.

## **ГЛАВА 1. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ**

### **1.1 Ознакомление с целями и задачами практики**

Ознакомление с целями и задачами практики является ключевым этапом процесса дипломного проектирования по разработке программного обеспечения для автоматизации учета клиентов. Главной целью этой программы является создание системы, которая позволит эффективно управлять данными и процессами, связанными с учетом клиентов.

В рамках такой практики студент должен ознакомиться с поставленными задачами и целями для решения проблем, связанных с учетом клиентов и управлением этими данными. Он должен понимать, какие результаты будут ожидать от проекта, и какие задачи необходимо решить в процессе работы.

Кроме того, студент должен ознакомиться с правилами внутреннего распорядка и правилами работы с инструментами и технологиями, которые будут использоваться в процессе разработки программного обеспечения. Он также должен знакомиться с рабочим местом и процессами, связанными с работой над проектом.

В целом, цель этой практики заключается в том, чтобы подготовить студента к разработке программного обеспечения для автоматизации учета клиентов, а также обучить его решать различные задачи, связанные с управлением данными и процессами, связанными с учетом клиентов. Он должен уметь проектировать, разрабатывать и тестировать программное обеспечение, которое отвечает требованиям и потребностям бизнеса.

### **1.2 Знакомство с правилами внутреннего распорядка, рабочим местом**

Разработка программного обеспечения для автоматизации учета клиентов является серьезной задачей, необходимо строго следить за правилами внутреннего распорядка, чтобы обеспечить эффективность работы и сохранность информации.

В рабочем месте, где происходит работа по разработке ПО, сотрудники должны соблюдать следующие правила:

1. Каждый сотрудник должен знать и соблюдать правила и политику компании относительно безопасности информации. Он должен обеспечить сохранность конфиденциальных данных клиентов, которые он использует в своей работе.

2. Сотрудник должен следить за чистотой своего рабочего места. Все документы и бумаги должны быть упорядочены и иметь ярлыки, чтобы было легко найти нужную информацию при необходимости. Программное обеспечение должно быть установлено на рабочее место только с разрешения руководства.

3. В соответствии с политикой компании, на рабочем месте не допускается употребление алкоголя, наркотиков и курение.

4. Сотрудники должны приходить на работу вовремя и готовыми к работе. Из команды могут быть исключены сотрудники, которые постоянно опаздывают или не выполняют свои обязанности.

5. Сотрудники должны соблюдать правила охраны труда и безопасности при работе с компьютерной техникой.

6. На рабочем месте запрещается устанавливать не лицензионное программное обеспечение, поскольку это может негативно отразиться на эффективности работы.

7. Сотрудники должны поддерживать чистоту своих устройств передачи информации, используемых в работе, и следить за тем, чтобы они не были инфицированы вредоносным программным обеспечением.

Эти правила помогут эффективно работать вместе и достигать целей по разработке ПО для автоматизации учета клиентов.

## ГЛАВА 2. ПРАКТИЧЕСКАЯ ЧАСТЬ

### 2.1 Практическое изучение предмета проектирования

При практическом изучении предмета проектирования программного обеспечения для автоматизации учета клиентов можно использовать следующий подход:

- Определение требований к функциональности системы. Необходимо в первую очередь определить, какие задачи должна выполнять система и какие функции ей необходимы.
- Разработка архитектуры системы. На этом этапе определяются компоненты, которые будут необходимы для создания системы, в том числе базы данных, интерфейса с пользователем и алгоритмов обработки данных.
- Создание дизайна пользовательского интерфейса. На этом этапе происходит создание макетов и проектирование пользовательского интерфейса, чтобы обеспечить удобство использования системы.
- Разработка кода и тестирование. Разработка кода происходит на основе спецификаций и требований к системе. Также происходит тестирование системы, чтобы проверить, соответствует ли она требованиям.
- Интеграция компонентов. На этом этапе компоненты системы интегрируются друг с другом, чтобы создать единую систему.
- Тестирование системы в реальных условиях. После интеграции компонентов и создания единой системы, необходимо провести тестирование в реальных условиях, чтобы проверить работоспособность и соответствие требованиям.
- Запуск системы и обучение персонала. После успешного тестирования система запускается в эксплуатацию, а персонал обучается ее использованию.
- Сопровождение системы. Разработчики должны следить за работоспособностью системы и обновлять ее при необходимости, чтобы обеспечить эффективность ее работы.

При разработке программного обеспечения для автоматизации учета клиентов также необходимо соблюдать правила и политику компании, относительно безопасности информации.

## **2.2 Поиск дополнительной информации, необходимость в которой возникла для решения вопросов, возникших в ходе знакомства с предметной областью дипломного проектирования**

Для поиска дополнительной информации, необходима более конкретная формулировка вопросов или проблем, которые возникли в ходе знакомства с предметной областью. Например:

- Какие наиболее распространенные методы автоматизации учета клиентов используются в современных компаниях?
- Какие требования к защите информации необходимо учитывать при разработке программного обеспечения для автоматизации учета клиентов?
- Какие методы тестирования ПО можно применять при разработке программного обеспечения для автоматизации учета клиентов?
- Какие инструменты и технологии разработки ПО следует использовать при создании системы автоматизации учета клиентов?

Для того, чтобы найти ответы на эти вопросы, можно использовать различные источники, например, научные статьи, книги, онлайн-курсы и другие материалы, посвященные разработке программного обеспечения для автоматизации учета клиентов. И сейчас ответим на каждый вопрос:

1. Какие наиболее распространенные методы автоматизации учета клиентов используются в современных компаниях?

1) Существует множество методов автоматизации учета клиентов, которые позволяют улучшить бизнес-процессы и повысить эффективность бизнеса в целом. Некоторые из наиболее распространенных методов автоматизации учета клиентов в современных компаниях включают в себя:

2) Использование специального программного обеспечения для учета клиентов – это мощный инструмент, который позволяет вести учет клиентов, контактов и их взаимодействия с компанией, автоматизировать маркетинг и продажи, проводить анализ и статистические исследования, и

многое другое.

3) Реализация системы управления взаимоотношениями с клиентами (CRM) – это метод управления и организации взаимодействия компании с клиентами. CRM-системы позволяют собирать, обрабатывать и анализировать информацию, связанную с клиентами, и совершать действия, нацеленные на укрепление связи с клиентами.

4) Применение автоматических систем отправки электронной почты и SMS-рассылок – эти системы позволяют автоматизировать рассылку целевой информации, например, рекламных материалов или оповещений о новых услугах или товарах. Это помогает сократить временные затраты на сбор и отправку информации, а также повышает эффективность взаимодействия с клиентами.

5) Использование социальных медиа-платформ и интернет ресурсов – социальные медиа и различные интернет ресурсы позволяют вести прямую коммуникацию с клиентами, быстро решать возникающие проблемы и отвечать на вопросы клиентов.

6) Хранение информации об учете клиентов в облачном хранилище – облачные хранилища представляют собой экономически эффективное решение для хранения информации

2. Какие требования к защите информации необходимо учитывать при разработке программного обеспечения для автоматизации учета клиентов?

1) Конфиденциальность данных клиентов. Необходимо обеспечить защиту информации клиентов от несанкционированного доступа.

2) Интегретет данных клиентов. Важно, чтобы данные в системе были правильными и актуальными.

3) Доступность данных. Необходимо гарантировать доступность данных клиентов, чтобы они могли быть использованы в любое время и в любом месте.

4) Аудит и логирование. Необходимо обеспечить возможность отслеживания процессов работы с данными клиентов.

5) Защита от вредоносных программ. Система учёта клиентов



должна быть защищена от вредоносных программ, таких как вирусы, трояны и шпионские приложения.

6) Защита сетевой инфраструктуры. Необходимо обеспечить защиту сетевой инфраструктуры, чтобы предотвратить несанкционированный доступ к информации клиентов через сеть.

7) Шифрование данных. Необходимо использовать шифрование данных, чтобы обеспечить защиту информации клиентов при передаче по сети.

3. Какие методы тестирования ПО можно применять при разработке программного обеспечения для автоматизации учета клиентов?

При разработке программного обеспечения для автоматизации учета клиентов можно использовать следующие методы тестирования:

1) Модульное тестирование – проверка отдельных компонентов программы на соответствие требованиям и возможность их взаимодействия.

2) Интеграционное тестирование – проверка взаимодействия отдельных компонентов программы и их работоспособности вместе.

3) Системное тестирование – проверка работоспособности всей системы как единого целого на соответствие требованиям и ожиданию конечного пользователя.

4) Функциональное тестирование – проверка соответствия функциональных требований программы.

5) Нагрузочное тестирование – проверка работоспособности программы при заданной ей нагрузке.

6) Тестирование безопасности – проверка защиты программы от несанкционированного доступа.

7) Тестирование совместимости – проверка работоспособности программы на различных операционных системах и платформах.

8) Тестирование производительности – проверка скорости и эффективности работы программы.

9) Тестирование пользовательского интерфейса – проверка удобства и понятности интерфейса программы для конечного пользователя.

4. Какие инструменты и технологии разработки ПО следует использовать при создании системы автоматизации учета клиентов?

При создании системы автоматизации учета клиентов уместно использовать следующие инструменты и технологии:

1) Языки программирования, такие как Java, Python, C#, PHP, которые могут быть использованы для разработки приложений для автоматизации учета.

2) СУБД, такие как MySQL, PostgreSQL, Oracle, которые могут использоваться для хранения данных клиентов.

3) Фреймворки и библиотеки, такие как Django, Ruby on Rails, React, Angular, которые могут облегчить процесс разработки и сделать код более организованным.

4) Инструменты контроля версий, такие как Git, Mercurial, SVN, которые могут помочь структурировать процесс разработки и координацию работы в команде разработчиков.

5) Инструменты автоматизированного тестирования, такие как Selenium, Appium, PyUnit, которые могут ускорить и оптимизировать тестирование.

6) Инструменты для обеспечения безопасности, такие как Kaspersky Antivirus, Norton Security, которые можно использовать для защиты системы от вирусов.

7) Инструменты для мониторинга работы приложения, такие как New Relic, AppDynamics, которые помогают отслеживать производительность и обнаруживать возможные ошибки и узкие места в работе приложения.

8) Среды разработки, такие как Visual Studio, IntelliJ IDEA, PyCharm, которые могут облегчить процесс написания кода и увеличить производительность разработчика.

## **ГЛАВА 3. Работы выполненные в ходе практики**

### **3.1 Подготовка данных для реализации автоматизированной системы**

Подготовка данных для реализации программного обеспечения для автоматизации учёта клиентов включает в себя следующие пункты:

1. Бизнес-процессы, связанные с учетом клиентов, могут быть разных видов, но общий принцип работы с клиентом заключается в том, чтобы собирать и хранить информацию о клиентах, анализировать данные и предоставлять предложения, которые могут помочь улучшить организацию работы с клиентами. Задачи, выполняемые сотрудниками организации, начинаются с того, что они получают первичную информацию о клиентах — это может быть контактная информация, анкетные данные, интересы и потребности клиентов. Далее, сотрудники организации обрабатывают эту информацию, чтобы она была доступна для использования в дальнейшей работе.

Одна из важных задач в учете клиентов — обеспечение своевременности контакта с клиентами. Это может быть сделано путем установки системы напоминаний, которая будет очень полезна для менеджеров по продажам и специалистов в области обслуживания клиентов. Также важной задачей в учете клиентов является анализ информации о клиентах. Он может помочь организации понять, какие предложения и какую стратегию следует применять в работе с клиентами. Это могут быть как изменения в существующей стратегии, так и пересмотр взаимоотношений с клиентами в целом. Взаимодействие с информационной системой также важно в рамках процессов учета клиентов. Система должна быть такой, чтобы была возможность быстро получать данные о клиенте, настраивать механизмы управления предложениями и процессы обслуживания клиентов.

В целом, учет клиентов – это очень важный аспект бизнеса, который помогает организации оптимизировать процессы работы и оказания услуг. Важно убедиться в том, что все процессы происходят автоматически и рационально, в наилучшей манере.

2. Конкретное описание требований к информационной системе зависит от конкретной организации и её потребностей. Однако, в общем виде, можно охарактеризовать следующие функциональные и не функциональные требования к информационной системе, связанной с учётом клиентов:

Функциональные требования:

- Учёт информации о клиентах: базовая информация (имя, номер телефона, адрес), детальная информация (семейное положение, профессия, предпочтения, интересы), история взаимодействия с организацией.
- Система учёта и обработки заказов: отслеживание статуса заказа, хранение информации о заказах.
- Автоматизированная рассылка рекламных сообщений и текстов писем клиентам.
- Аналитические инструменты для анализа данных о клиентах и продажах.

Не функциональные требования:

- Высокая стабильность и надёжность сервиса, обеспечение защиты данных клиентов.
- Гибкость системы в адаптации под изменяющиеся потребности бизнеса и внедрение новых функциональностей.
- Интуитивно понятный интерфейс и удобство использования системы.
- Высокая масштабируемость системы.
- Поддержка технической консультации и сопровождения системы.
- Интеграция с другими информационными системами уже используемыми организацией.

3. Структура базы данных информационной системы, связанной с учетом клиентов, должна обеспечивать эффективное хранение и доступность к данным о клиентах, заказах и платежах. Основные сущности системы включают клиентов, заказы и платежи.

1) Сущность "Клиенты". База данных должна содержать информацию о каждом клиенте, включая его контактную информацию, детальные данные, такие как персональный номер и пароль, а также историю взаимодействия с организацией. Основные поля, которые могут включать в себя эта сущность:

- Имя и фамилия клиента
- Контактные данные (номер телефона, адрес электронной почты, адрес)
- Персональный номер и пароль
- Дата и время последнего взаимодействия
- История заказов
- История обращений в поддержку клиентов
- Предпочтения и интересы клиента
- Маркетинговые действия, проведенные для клиента

2) Сущность "Заказы". База данных должна содержать информацию о каждом заказе, выполненном клиентом, включая детали заказа, стоимость, дату и статус. Основные поля, которые могут включать в себя эта сущность:

- Уникальный номер заказа
- Дата и время заказа
- Клиент, сделавший заказ
- Перечень заказанных товаров или услуг
- Стоимость заказа
- Статус заказа (например, "новый", "в обработке", "выполнен", "отменен")

3) Сущность "Платежи". База данных должна содержать информацию обо всех платежах, сделанных клиентами, включая детали о платежном счете, сумме платежа и дате. Основные поля, которые могут включать в себя эта сущность:

- Уникальный номер платежа
- Дата и время платежа
- Клиент, совершивший платеж
- Счет на который был сделан платеж

- Сумма платежа
- Статус платежа (например, "обработан", "отклонен")

Отношения между сущностями системы:

- Многие клиенты могут иметь много заказов
- Один заказ может быть сделан многими клиентами, так как заказ может быть оплачен несколькими людьми.
- Один клиент может сделать много платежей
- Один платеж может быть сделан многими клиентами, к примеру, если несколько клиентов разделяют платеж.

При проектировании структуры базы данных следует убедиться в том, что она эффективна, а также учитывать требования безопасности и конфиденциальности, при обработке данных о клиентах, заказах и платежах.

4. Управление доступом к информационной системе и её защите – это процесс обеспечения безопасности данных и защиты системы от несанкционированного доступа. Для достижения этой цели необходимы следующие задачи:

1) Создание пользовательских аккаунтов и управление правами доступа

- Задача: Создание уникальных пользовательских аккаунтов для каждого пользователя в информационной системе и управление их доступом к данным согласно их ролям и функциям.

- Категории пользователей: Администраторы, менеджеры, сотрудники, гости.

- Права доступа: Администраторы имеют полный доступ к системе; менеджеры – доступ к данным своего отдела и части данных других отделов; сотрудники – доступ к данным, необходимым для работы; гости – ограниченный доступ к информации.

- Политика безопасности: Установка сложных паролей, проверка их использования, регулярное изменение паролей, ограничение доступа по IP-адресам и сетевым устройствам, использование системы двухфакторной аутентификации.

## 2) Защита системы от несанкционированного доступа

- **Задача:** Обеспечение защиты системы от взлома и несанкционированного доступа с помощью регулярного мониторинга и анализа активности системы.

- **Категории пользователей:** Доступ к системе должен иметь только зарегистрированные пользователи с соответствующими правами доступа.

- **Права доступа:** Настройка правильных настроек конфигурации сервера, контроль за доступом к системе, мониторинг системы на поиск аномалий и инцидентов безопасности.

- **Политика безопасности:** Установка политик доступа к системе, обучение пользователей безопасной работе в системе, использование средств антивирусной защиты и фильтрации трафика, регулярное обновление программного обеспечения системы.

## 3) Резервное копирование данных

- **Задача:** Создание резервных копий данных на случай потери информации из-за сбоев в работе системы или злонамеренных действий.

- **Категории пользователей:** Администраторы.

- **Права доступа:** Администраторы имеют полный доступ к системе и могут создавать и восстанавливать резервные копии данных.

- **Политика безопасности:** Установка частоты создания резервных копий, проверка работоспособности копий, контроль доступа к копиям.

## 4) Мониторинг безопасности информации

- **Задача:** Проведение мониторинга системы на наличие угроз безопасности информации.

- **Категории пользователей:** Администраторы.

- **Права доступа:** Администраторы имеют полный доступ к системе и могут проводить мониторинг безопасности информации.

- **Политика безопасности:** Установка системы обнаружения вторжений, анализ системных журналов, контроль сетевой активности, обучение пользователей технологиям безопасности данных.

## 5) Обновление системы

- Задача: Регулярное обновление программного обеспечения системы для исправления уязвимостей и улучшения защиты данных.

- Категории пользователей: Администраторы.

- Права доступа: Администраторы имеют полный доступ к системе и могут обновлять программное обеспечение.

- Политика безопасности: Регулярное обновление программного обеспечения, проверка на наличие уязвимостей, тестирование обновлений перед установкой, контроль доступа к установке обновлений.

5. Параметры производительности информационной системы – это набор характеристик, которые определяют её работоспособность и эффективность. Некоторые из основных параметров производительности информационной системы:

1) Количество пользователей – это количество пользователей, которое может использовать систему в определенный момент времени. Чем выше это число, тем больше нагрузка на систему. Для оптимизации производительности системы необходимо учитывать ожидаемое количество пользователей и иметь достаточное количество оборудования для поддержки их работы.

2) Время отклика – это время, которое требуется системе для обработки запроса пользователя и возврата результата. Чем быстрее время отклика, тем лучше производительность системы. Производительность системы может улучшаться с помощью оптимизации кода программного обеспечения и конфигурации оборудования.

3) Контроль нагрузки – это мероприятия и технологии, разработанные для повышения производительности системы. Они включают в себя кэширование, балансировку нагрузки, кластеризацию и параллельную обработку. Вместе эти технологии помогают ускорить процесс обработки запросов и увеличить общую производительность системы.

4) Оптимизация – это процесс улучшения производительности системы путём оптимизации кода программного обеспечения, настройки и конфигурации оборудования, а также оптимизации базы данных.



Оптимизация может помочь ускорить время отклика и увеличить количество пользователей, которые могут использовать систему одновременно.

5) Резервное копирование данных – это процесс создания дублированных копий данных для случаев сбоев в работе системы или намеренного воздействия. Резервные копии помогают обеспечить надежность данных и быстрое восстановление работы системы в случае необходимости. Копии могут использоваться для тестирования обновлений программного обеспечения и бекапа системы, а также для миграции системы на другое оборудование.

6. Схема взаимодействия компонентов системы и её ус тройственной архитектуры зависит от конкретных технических требований и характеристик аппаратного и программного обеспечения. Однако, общая схема может включать следующие компоненты:

1) Клиентские устройства – это устройства пользователя, которые используются для доступа к информационной системе, такие как компьютеры, смартфоны или планшеты. Эти устройства используются для ввода и вывода данных и информации.

2) Серверы – это центральные компоненты информационной системы, которые обеспечивают хранение и обработку всех данных. Серверы могут включать в себя базы данных, веб-серверы, файловые серверы и различные приложения, необходимые для функционирования системы.

3) Сетевое оборудование – это компоненты, которые обеспечивают передачу данных между клиентскими устройствами и серверами. Сетевое оборудование может включать в себя маршрутизаторы, коммутаторы и другие элементы сетевой инфраструктуры.

4) Безопасность системы – это набор компонентов, которые обеспечивают безопасность информации предоставляемой системой. Эта область может включать в себя антивирусы, программы защиты от взлома и угроз безопасности, а также системы резервного копирования.

5) Операционные системы и приложения – это программное обеспечение, которое используется для обеспечения работы системы и её

компонентов. Операционные системы и приложения обеспечивают работоспособность и эффективность информационной системы.

Ус тройственная архитектура системы обычно включает в себя определение аппаратного обеспечения, используемого для каждого компонента. Она может также определять взаимодействие между компонентами, их расположение и другие аппаратные особенности.

## 7. Основные требования к разработке информационной системы:

### 1) Языки программирования:

- Python 3.7 и выше для бекенда
- JavaScript для фронтенда

### 2) Фреймворки:

- Django 3.0 и выше для бекенда
- ReactJS для фронтенда

### 3) Инструменты разработки:

- PyCharm для написания кода на Python
- Visual Studio Code для написания кода на JavaScript и ReactJS
- Git и GitHub для контроля версий
- Docker для создания контейнеров

### 4) База данных:

- PostgreSQL

### 5) Требования к тестированию:

- Необходимо проводить все виды тестирования: модульное, функциональное, интеграционное, системное

- Тестирование должно проводиться на всех уровнях разработки

- В процессе тестирования должны быть обнаружены все ошибки и дефекты

### 6) Требования к документированию:

- Необходимо подготовить документацию на все компоненты, модули и функции системы

- Документация должна содержать описание всех функций и методов, примеры использования

- Документация должна быть обновлена после каждого изменения в системе.

8. Для разработки программного обеспечения для автоматизации учета клиентов необходимо подготовить следующие спецификации и техническую документацию:

1) Техническое задание (ТЗ) - основной документ, содержащий требования к разрабатываемому программному обеспечению. В ТЗ должны быть описаны следующие моменты:

- Описание функционала программы (например, учет контактной информации клиентов, их заказов и оплат, и т.д.).

- Требования к интерфейсам (например, удобства использования, дизайна и т.д.).

- Требования к базе данных (например, тип и структура базы данных, формат хранения данных и т.д.).

- Требования к производительности (например, количество клиентов, которых программа должна обслуживать, время отклика и т.д.).

- Требования к безопасности (например, шифрование конфиденциальной информации клиентов и т.д.).

- Дополнительная информация (например, распределенный доступ к данным, поддержка разных форматов файлов и т.д.).

2) Инструкция по установке и использованию программного обеспечения - этот документ, содержащий сведения о том, как установить и работать с программой. В нем должны содержаться:

- Информация о необходимом окружении программы (операционная система, версия языка программирования, требования к железу и т.д.).

- Шаги по установке и настройке программы.

- Инструкции по работе с программой (например, как добавить нового клиента, как удалить клиента из базы данных и т.д.).

3) Спецификация архитектуры приложения - документ, описывающий структуру и компоненты программы. Он содержит следующую информацию:

- Схему архитектуры приложения (например, блок-схема).

- Описание каждого компонента и связи между ними.

- Информацию о структуре базы данных и логики работы приложения.

4) Тестовая документация - документ, содержащий информацию о проведенных тестах на работоспособность приложения. Она включает в себя:

- Перечень тестов (например, тесты на добавление/удаление/редактирование информации по клиенту и т.д.).

- Результаты каждого теста.

- Информацию о методиках тестирования и используемом оборудовании.

5) Документация по безопасности - документ, описывающий все меры безопасности и защиты данных, применяемые в проекте. Он содержит следующую информацию:

- Информацию о доступе к программе и базе данных.

- Описание шифрования данных и методов защиты передачи данных между клиентом и сервером.

9. Для тестирования функциональности и производительности информационной системы необходимо подготовить различные тестовые данные и сценарии. Ниже представлены возможные примеры тестовых данных и сценариев:

1) Тестовые данные для проверки функциональности:

- Набор данных клиентов (имя, фамилия, адрес, номер телефона, адрес электронной почты и т.д.)

- Набор данных заказов (номер заказа, дата заказа, клиент, описание продукта, цена и т.д.)

- Набор данных оплат (номер заказа, дата оплаты, вид оплаты, сумма оплаты и т.д.)

2) Сценарии проверки функциональности:

- Добавление нового клиента и проверка, что он появился в базе данных

- Добавление нового заказа для клиента и проверка, что он был успешно добавлен
- Добавление новой оплаты для заказа и проверка, что сумма оплаты была корректно учтена
- Редактирование информации о клиенте и проверка, что изменения были успешно сохранены
- Удаление заказа и проверка, что он был удален из базы данных
- Поиск заказа по номеру заказа и проверка, что он был найден

### 3) Тестовые данные для проверки производительности:

- Набор данных клиентов, заказов и оплат малого объема (несколько десятков записей)
  - Набор данных клиентов, заказов и оплат большого объема (несколько тысяч записей)

### 4) Сценарии проверки производительности:

- Добавление новых записей и проверка времени, необходимого для добавления
- Поиск заказа по номеру и проверка времени, необходимого для поиска
- Генерация отчета по заказам за период и проверка времени генерации

Для выполнения тестирования функциональности и производительности системы можно использовать автоматические инструменты тестирования, например, JMeter. Также можно провести ручное тестирование, следующие за предложенными выше сценариями. В результате проверки необходимо создать отчеты с описанием результатов тестирования и информацией о найденных проблемах и ошибках.

10. Процесс развертывания, настройки и обновления программного обеспечения:

#### 1) Развертывание:

- Установка основной системы (операционной системы, PostgreSQL, Python и т.д.).

- Создание виртуальных окружений для бекенда и фронтенда.
- Установка зависимостей бекенда и фронтенда.
- Подготовка базы данных (создание таблиц, заполнение начальных данных)
- Запуск сервера бекенда и фронтенда

## 2) Настройка:

- Настройка параметров базы данных и приложения в файле конфигурации.
- Определение прав доступа для пользователей и групп.
- Настройка системных параметров для оптимизации работы приложения.
- Настройка логирования для сбора данных о работе системы.

## 3) Обновление:

- Получение новой версии приложения из репозитория Git.
- Установка зависимостей новой версии приложения.
- Обновление базы данных (в случае изменений в структуре).
- Перезапуск сервера.

## План реализации проекта:

1. Анализ требований заказчика и составление ТЗ.
2. Разработка архитектуры системы.
3. Выбор языков программирования, фреймворков и инструментов разработки.
4. Разработка базы данных и создание начальных данных.
5. Разработка основной функциональности системы (бекенд и фронтенд).
6. Написание тестового плана и проведение тестирования.
7. Релизный процесс (тестирование, деплой, установка, настройка).
8. Разработка документации.

## План сопровождения созданной системы:

1. Поддержка со стороны команды разработчиков (обновление кода, исправление ошибок и т.д).

2. Предоставление документации, необходимой для обслуживания и поддержки системы.

3. Проведение регулярных тестов производительности и безопасности системы.

4. Определение и исправление ошибок в ходе работы системы.

5. Разработка дополнительных функций и модулей в ответ на запросы клиентов.

### **3.2 Подготовка данных для реализации предмета проектирования**

Подготовка данных для реализации предмета проектирования для автоматизации учёта клиентов включает в себя следующие пункты:

1. Сбор информации о клиентах является одним из важных этапов при разработке программного обеспечения для автоматизации учета клиентов. Ниже представлены возможные данные, которые следует собрать о каждом клиенте:

1) ФИО - полное имя клиента.

2) Контактные данные - телефон, email и другие данные, необходимые для связи с клиентом.

3) Адреса доставки - полный адрес клиента, включая район, улицу и номер дома.

4) Предпочтения по доставке - выбранный метод доставки, удобный график доставки и т.д.

5) История заказов - информация о предыдущих заказах, включая дату, номер заказа, товары, сумму заказа и т.д.

6) Комментарии к заказам - дополнительная информация о заказе, которую может оставить клиент, например, пожелания по упаковке или доставке.

Все данные о клиентах должны быть сохранены в базе данных для дальнейшей работы с ними в программном обеспечении.

2. Сбор требований к функционалу программного обеспечения для автоматизации учета клиентов включает следующие пункты:

- Возможность добавления, редактирования и удаления клиентов,

включая сведения о ФИО, контактных данных и адресах доставки.

- Возможность просмотра и анализа заказов и их статусов, включая информацию о дате заказа, номере заказа, сумме заказа, составе товаров, способах доставки и оплаты.

- Управление данными о клиентах, включая возможность просмотра и редактирования информации, связанной с каждым клиентом, а также создание отчетов о заказах клиента.

- Возможность фильтрации и поиска информации, включая поиск клиентов по ФИО, адресу, номеру телефона, поиск заказов по дате, номеру заказа или статусу.

- Наличие автоматического оповещения клиента о состоянии его заказа и сроках доставки.

- Наличие системы соответствующих скидок и бонусов для постоянных клиентов.

- Наличие системы учета и просмотра истории заказов клиентов, с возможностью просмотра информации за определенный период времени.

- Система контроля доступа и защиты данных от несанкционированного доступа к информации о клиентах и заказах.

- Наличие системы резервного копирования и восстановления данных для обеспечения надежности системы в случае сбоев и ошибок.

3. Структура и формат данных в базе данных определяют организацию и типы информации, которые будут храниться в таблицах. Обычно, в базе данных для электронной коммерции могут быть следующие таблицы:

- Таблица клиентов: содержит информацию о клиентах, таких как имя, фамилия, адрес электронной почты, телефонный номер и т.д.

- Таблица заказов: содержит информацию о заказах, такую как дата заказа, номер заказа, статус заказа и т.д.

- Таблица товаров: содержит информацию о товарах, такую как название, описание, цена, количество на складе и т.д.

- Таблица адресов доставки: содержит информацию о городе или регионе доставки, а также деталях доставки.



Каждая таблица содержит поля для хранения соответствующей информации, и каждое поле имеет определенный тип данных, такой как целые числа, даты, строки и др. Также таблицы в базе данных могут быть связаны между собой ключом, что позволяет связывать данные в разных таблицах. За счет связи таблиц может быть получена информация, включающая данные из разных таблиц.

4. Подготовка данных для импорта в базу данных - это процесс, включающий в себя следующие шаги:

1) Сбор данных из различных источников. Источниками данных могут быть электронные таблицы, базы данных, CSV-файлы, XML-файлы и другие файлы или источники.

2) Анализ данных. Перед импортом необходимо проанализировать данные и убедиться в их соответствии структуре и формату базы данных. Дополнительные шаги можно включать в проверку целостности данных, такую как проверка наличия дубликатов, отсутствие некорректных значений и др.

3) Сопоставление форматов данных. Если данные имеют разные форматы, необходимо их привести к общему формату. Например, даты могут быть представлены в разных форматах в разных источниках, и их необходимо привести к одному формату. Это может потребовать преобразования данных вручную или с использованием специального программного обеспечения.

4) Создание таблиц базы данных. После анализа данных и их сопоставления форматов необходимо создать таблицы в базе данных, соответствующие импортируемому данным.

5) Импорт данных. После создания таблиц необходимо произвести импорт данных в базу данных, используя специальное программное обеспечение или SQL-запросы.

6) Проверка и обновление данных. После импорта данных необходимо проанализировать их и убедиться в их соответствии требованиям и структуре

базы данных. При необходимости можно произвести обновление или исправление импортированных данных.

5. Разработка четких правил и процедур для заполнения и обработки данных - это важный элемент в поддержании точности и целостности данных в системе.

Процедуры для заполнения данных включают в себя следующие шаги:

- Задать ясные и понятные правила для заполнения данных в различных полях таблицы.
- Проверка наличия ошибок при вводе данных с помощью программного обеспечения или вручную.
- Проверка наличия дубликатов и детальный контроль ошибок.

Процедуры для обработки данных включают в себя следующие шаги:

- Задать ясные и понятные правила для обработки данных в различных полях таблицы.
- Ограничение доступа к важным функциям системы только на основании необходимости.
- Регулярное резервное копирование данных и сохранение их на отдельном носителе для предотвращения возможных потерь данных.
- Защита данных с помощью шифрования, фильтраций или других соответствующих методов - для предотвращения неблагонадежного доступа к данным в базе данных.
- Все эти процедуры и правила необходимо регулярно обновлять и пересматривать для предотвращения возможных отклонений и проблем в жизненном цикле базы данных.

6. Тестирование данных - это процесс проверки информации на ее правильность, полноту и соответствие заданным требованиям перед началом работы с ними в программном обеспечении. Это важный этап в разработке любой системы, который позволяет обеспечить качество данных и предотвратить ошибки в работе приложения.

При тестировании данных следует руководствоваться следующими правилами и процедурами:

1) Определение требований к данным - необходимо определить требования к данным, например, формат, тип, размер, структуру и т.д. Это поможет гарантировать соответствие входных данных стандартам и требованиям.

2) Создание тестовых данных - тестирование должно проводиться на специально созданных тестовых данных, которые содержат всевозможные комбинации данных, которые могут быть использованы в продукте. Тестовые данные должны быть максимально разнообразными, чтобы проверить все возможные сценарии использования.

3) Проверка целостности данных - следует проверить целостность данных, чтобы убедиться, что данные не повреждены и не содержат ошибок. Ошибки могут возникнуть в результате искаженной передачи данных, их перезаписи, обрезки, перекодировки и других причин.

4) Проверка правильности данных - важно проверить правильность данных на соответствие их значениям, которые они должны представлять. Например, проверка даты рождения на правильность формата, соответствие возраста данным и т.д.

5) Проверка полноты данных - тестирование должно включать проверку полноты данных, чтобы убедиться, что не пропущены какие-либо данные, такие как некоторые записи или значения.

6) Проверка системных требований - тестирование должно включать проверку соответствия данных системным требованиям внешних систем, которые используются в продукте.

7. Интеграция системы учета клиентов с другими соответствующими системами - это очень важный процесс, который помогает автоматизировать и упростить бизнес-процессы и повысить производительность. Для обеспечения этого процесса следует руководствоваться следующими правилами и процедурами:

1) Определение потребностей и целей - предварительный анализ всех систем, с которыми необходимо интегрировать систему учета клиентов. Необходимо определить целевые функции, данные и процессы, которые должны быть интегрированы, а также сопоставить их друг с другом.

2) Идентификация методов интеграции - определение методов, которые будут использоваться для интеграции систем, например, API, файловые обмены, базы данных и другие.

3) Создание плана интеграции - разработка детального плана интеграции, который включает в себя задачи на каждом этапе, оценку рисков и временные рамки.

4) Интеграция данных - интеграция данных может включать перенос данных из одной системы в другую, обновление данных в режиме реального времени, установку связей между системами, автоматизацию процесса обмена данными и другие методы.

5) Тестирование и отладка - тестирование процесса интеграции, чтобы убедиться в правильности и эффективности интеграции систем. В случае обнаружения ошибок следует провести отладку и внести необходимые изменения.

6) Обучение персонала - важно предоставить обучение персоналу по работе с интегрированными системами, чтобы убедиться, что все работники знают, как использовать системы в рамках их обязанностей.

Интеграция системы учета клиентов с другими соответствующими системами может значительно упростить работу и повысить производительность. Для этого необходимо разработать план интеграции, использовать правильные методы и тестировать процесс интеграции для обеспечения высокого качества именно в тех системах, где он необходим.

### **3.3 Анализ собранного материала по программным средствам**

Подготовка анализа собранного материала по программным средствам для автоматизации учёта клиентов включает в себя следующие пункты:

1. Рынок учета клиентов развивается стремительно, ведь каждый бизнес нуждается в системе, которая позволяет следить за клиентами и улучшать качество обслуживания. Существует множество производителей программного обеспечения для учета клиентов, но самые популярные из них – это Salesforce, Microsoft Dynamics CRM, Zoho CRM, HubSpot, Bitrix24.

Salesforce – это один из самых старых и наиболее используемых на рынке решений для учета клиентов. Ее главная особенность – гибкость настройки под конкретные нужды бизнеса. Программа предоставляет широкий функционал, который включает в себя возможность управления контактами, продажами, маркетингом, технической поддержкой и многое другое. Однако, цена на продукт может быть высокой для малых и средних компаний.

Microsoft Dynamics CRM – это решение, которое позволяет управлять клиентами и продажами в более упорядоченной форме. Она работает в сотрудничестве с системами Office 365 и Outlook, а также имеет высокий уровень безопасности. Как и Salesforce, здесь предоставляется широкий функционал, а гибкость настройки поможет адаптироваться различным бизнес-процессам. Стоимость программного обеспечения зависит от версии и пакета, который выберет компания.

Zoho CRM – это облачное решение для учета клиентов, которое позволяет управлять продажами, маркетингом и технической поддержкой. Программа имеет интуитивно понятный интерфейс, что упрощает работу пользователей. Большинство функций, которые предоставляет программа, доступны в базовом пакете, а дополнительные модули можно приобрести в любое время.

HubSpot – это программное обеспечение, которое предлагает возможность управления всеми бизнес-процессами, которые касаются клиентов и продаж. Его главная особенность – это маркетинговый модуль, который помогает управлять лидами, электронной почтой, социальными медиа и другими компонентами маркетинга. Стоимость программы зависит от версии и пакета, который выберет компания.

Bitrix24 – это облачная программа, которая предоставляет возможность управления клиентами, продажами, проектами, задачами и другими аспектами бизнеса. Она имеет широкий функционал и настраивается под конкретные бизнес-потребности. Получить базовый пакет программы можно бесплатно, а дополнительные функции приобретаются за дополнительную плату.

Итак, если говорить о сравнительном анализе, то все рассмотренные программы предоставляют широкий функционал для управления клиентами и продажами. Однако, цены и возможности в каждой из них имеют свои особенности. Salesforce имеет наибольшую цену, но гибкую настройку под бизнес-процессы. Microsoft Dynamics CRM работает наиболее удобно с Office 365 и Outlook. Zoho CRM - облачное решение с интуитивно понятным интерфейсом. HubSpot содержит маркетинговый модуль. Bitrix24 – это облачная программа с бесплатным базовым пакетом.

В каждом случае компания должна оценивать свои потребности, чтобы определить, какая программа наиболее подходит, и какой бюджет может выделиться на ее приобретение.

2. Определение требований – это процесс выяснения бизнес-потребностей компании и определения основных требований к программному обеспечению для учета клиентов. Для этого обычно проводится анализ используемых методов учета клиентов и выявление слабых мест в процессах, а также определение ключевых функциональных требований, которые необходимы для эффективной работы бизнеса.

Кроме того, определение требований включает в себя выяснение возможности интеграции программного обеспечения с другими системами, такими как системы электронной почты, CRM, ERP и прочие. Также необходимо учитывать масштабируемость программы, чтобы она могла расти вместе с бизнесом, и удобство интерфейса, чтобы пользователи могли эффективно работать с программой.

Для определения требований важно провести анализ бизнес-процессов, выявить слабые места и оценить эффективность текущих методов учета

клиентов. На этапе определения требований также советуется вовлечь пользователей и менеджеров компании, чтобы учесть все стороны бизнеса и предоставить оптимальное программное решение для учета клиентов.

3. Оценка стоимости программного обеспечения для учета клиентов может включать в себя несколько элементов затрат.

Приобретение: первое, что нужно учитывать - это стоимость самой программы. Здесь на цену может влиять несколько параметров, таких как комплектация, количество пользователей и дополнительный функционал. Цена может варьироваться от нескольких сотен до нескольких тысяч долларов в год, в зависимости от выбранного решения.

Настройка и внедрение: второй элемент затрат связан с настройкой и внедрением программы. Компании в любом случае необходимо приспособить новую программу под свои бизнес-процессы, подключить ее к нужным базам данных, настроить права пользователей и т.д. В зависимости от сложности настроек, эта процедура может занять от нескольких дней до нескольких месяцев. Расходы на этот этап могут варьироваться от нескольких тысяч до нескольких десятков тысяч долларов.

Обучение: третий элемент затрат связан с обучением пользователей. Новая программа, как правило, требует времени на освоение, поэтому компания должна рассчитывать на расходы на обучение сотрудников. Стоимость обучения обычно зависит от количества сотрудников, которые будут использовать программу, и составляет от нескольких сот до нескольких тысяч долларов.

Поддержка: четвертый элемент затрат - это поддержка программного обеспечения. Практически все программы, которые используются в бизнесе, нуждаются в поддержке и обновлениях. Стоимость обновлений и поддержки может зависеть от выбранного решения. Иногда эта услуга включается в стоимость программы, а иногда она оплачивается отдельно.

Операционные затраты: определенные операционные затраты могут возникнуть в результате использования программы. Это могут быть расходы на техническую поддержку, серверное оборудование или облачные

вычисления, нагрузку на систему электропитания и т.д. Также может возникнуть необходимость в наеме дополнительного персонала для обслуживания программы.

Конечный расход на программное обеспечение для учета клиентов зависит от многих факторов, но также имеет большое значение выбор решения, которое лучше всего подходит под бизнес-процессы и бюджет компании.

4. Интеграция программного обеспечения для учета клиентов с другими системами является важным аспектом реализации эффективного бизнес-процесса в компании. В настоящее время существует множество различных систем, которые используются в бизнесе, таких как бухгалтерские, управление заказами, электронная почта и другие. Поэтому необходимо также учитывать возможность интеграции разрабатываемой программы с существующими системами компании.

Прежде всего, необходимо определить системы, с которыми программа должна интегрироваться. Далее, необходимо рассмотреть различные технологии и стандарты, используемые в этих системах. Некоторые из самых распространенных технологий для интеграции программного обеспечения для учета клиентов с другими системами включают в себя REST API, SOAP, XML и JSON.

REST API - это набор протоколов передачи данных, которые позволяют взаимодействовать с сервером и получать от него нужные данные. REST API является одним из самых распространенных и простых в использовании способов интеграции программного обеспечения с внешними системами.

SOAP - это протокол передачи данных, который предназначен для обмена данными между системами. SOAP использует XML для описания сообщений и обеспечения безопасности передачи данных.

XML - это язык разметки, используемый для создания структурированных документов, данных и других объектов. XML обычно используется в комбинации с другими технологиями, такими как REST API и SOAP, для обмена данными между различными системами.



JSON - это формат обмена данными, который используется для передачи объектов между системами. JSON является легким и удобным способом для обмена данными между системами.

Кроме того, помимо технологий интеграции, важно оценить возможности конкретных программ для интеграции с другими системами. Многие известные программы для учета клиентов, такие как Salesforce, Microsoft Dynamics CRM и Zoho CRM, имеют широкий функционал для интеграции с другими системами, такими как системы управления заказами, бухгалтерские системы и другие.

5. Анализ способов миграции данных при переносе программного обеспечения для учета клиентов может быть существенно важен для компании. Данный процесс включает в себя планирование, тестирование, выполнение и окончание миграционного процесса. Следует рассмотреть несколько способов миграции данных для выбора оптимального решения.

1) Ручной импорт данных - предусматривает возможность переноса данных вручную путем экспорта данных из старой системы и импортирования их в новую. С данной опцией будет связана высокая степень риска потери данных, а также затраты на ручной процесс миграции.

2) Электронный обмен данными - осуществляется через использование средств, которые могут отслеживать и импортировать данные из старой системы в новую. С помощью автоматического обмена данными возможно снизить риск ошибок и временные затраты на миграцию.

3) Конвертер данных - в некоторых случаях может применяться специальный конвертер, который осуществляет миграцию данных из старой системы в новую. Однако, возможны проявления несовместимости данных и форматов между старой и новой системами.

4) Обращение к поставщику программного обеспечения - если компания работает с одним и тем же производителем ПО для учета клиентов, можно обратиться к производителю для узнавания о способах миграции данных.

Таким образом, при переносе программного обеспечения для учета клиентов рекомендуется использование одного из вышеперечисленных способов миграции данных с целью минимизации затрат, несовместимостей и потери данных

6. Выбор технологий при разработке программного обеспечения (ПО) является важным этапом, поскольку правильный выбор технологии может повысить производительность, улучшить качество и ускорить разработку. Однако, необходимо также принимать во внимание совместимость выбранных технологий с другими системами, которые уже используются в компании.

Первым шагом при выборе технологий является определение бизнес-требований к будущему продукту. Необходимо проанализировать, что именно требуется от системы, какие функции необходимо реализовать и какие особенности должны быть включены в разрабатываемое ПО. Затем можно определить, какие технологии могут быть использованы для реализации этих требований.

Важно также учитывать, какие технологии уже используются в компании. Если компания уже имеет определенный стек технологий, то может быть целесообразно использовать технологии, хорошо совместимые с уже существующими системами. Например, компания, использующая базу данных Oracle, может захотеть использовать Java, поскольку Java хорошо работает с Oracle.

Следующий шаг - провести исследование рынка технологий и определить наиболее подходящие технологии для решения поставленных задач. Необходимо учитывать, какие технологии лучше всего подходят для определенных видов задач и являются более прогрессивными и актуальными на данный момент.

Также можно обратиться к опыту других компаний, использующих технологии, которые рассматриваются для разработки ПО. Часто в интернете можно найти отзывы и рекомендации по использованию технологий от других разработчиков.

7. Планирование и контроль являются важными этапами при разработке программного обеспечения (ПО). На этапе планирования определяются цели, качество и временные рамки разработки ПО, а также ресурсы и сроки реализации проекта. Контроль качества программного обеспечения проводится на каждом этапе разработки, в целях обеспечения соответствия требованиям заказчика и качественного исполнения проекта.

Определение механизмов контроля качества программного обеспечения в процессе разработки позволяет своевременно выявлять и решать возникающие проблемы, а также снижать риски, связанные с качеством ПО. Это может быть реализовано через систему тестирования, которая позволит проверить работоспособность и соответствие функциональных требований.

Планирование этапов тестирования и внедрения программного обеспечения помогает сократить риски и снизить затраты. Тестирование должно проводиться как на стадии разработки, так и на стадии внедрения, позволяя выявить и устранить ошибки и проблемы, которые могут возникнуть при использовании ПО.

Также важно учитывать международные и отраслевые стандарты качества, такие как ISO 9001, для обеспечения высокого уровня качества программного обеспечения и удовлетворения потребностей клиентов.

Таким образом, планирование и контроль являются важными этапами при разработке ПО, позволяющими снизить риски и затраты, а также обеспечить высокое качество ПО, которое удовлетворит потребности заказчика.

## **Заключение**

Разработка программного обеспечения является важным процессом в современном бизнесе. При разработке ПО следует учитывать множество факторов, таких как выбор технологий, планирование, контроль и миграция данных. Определение совместимости продукта с уже существующими системами компании является обязательным этапом в разработке.

Важно также учитывать международные и отраслевые стандарты качества, для обеспечения высокого уровня качества продукта, которое удовлетворит потребности заказчика и соответствует требованиям рынка.

Миграция данных при переносе программного обеспечения для учета клиентов помогает компании сэкономить время и деньги, используя оптимальный способ миграции данных в новую систему.

Планирование и контроль являются неотъемлемой частью процесса разработки. Это позволяет выявлять и решать возникающие проблемы, а также повышает качество продукта и удовлетворенность клиентов.

Таким образом, все перечисленные выше факторы являются важными и необходимыми для полноценной разработки продукта и удовлетворения потребностей рынка.

## Список литературы

1. ГОСТ 34.602-2018 "Информационная технология. Защита информации. Методы и средства защиты программного обеспечения".
2. ГОСТ Р ИСО/МЭК 12207-2017 "Процессы жизненного цикла программного обеспечения".
3. ГОСТ Р ИСО/МЭК 29110-2017 "Жизненный цикл программных средств для малых организаций".
4. ГОСТ Р 51583-2000. Порядок создания автоматизированных систем в защищенном исполнении. - М.
5. ГОСТ Р 50922-96. Защита информации. Основные термины и определения. - М.: Госстандарт России.
6. ГОСТ 34.320-96 Информационные технологии. Система стандартов по базам данных. Концепции и терминология для концептуальной схемы и информационной базы [Текст]. - Москва: Изд-во стандартов.
7. ГОСТ 34.003-90 Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Термины и определения [Текст]. - Москва: Изд-во стандартов.
8. ГОСТ Р ИСО 9001-2015 "Системы менеджмента качества. Требования".
9. ГОСТ 34.601-90. Автоматизированные системы. Стадии создания. - М.: Госстандарт России.
10. Литвинова К.А. Разработка программного обеспечения для автоматизации учета клиентов / К.А. Литвинова // Информационные технологии и вычислительные системы. – 2017. – № 1. – С. 50-55.
11. Матвеев А.А. Автоматизированные системы учета: разработка программного обеспечения для учета клиентов / А.А. Матвеев, И.А.

Касьянова // Промышленные информационные системы. – 2016. – № 1. – С. 33-38.

12. Гриценко Е.А. Разработка программного обеспечения для автоматизации учета клиентов: проблемы и перспективы / Е.А. Гриценко, А.С. Скрипицкий // Информатика и математическое моделирование. – 2017. – Т. 8. – № 2. – С. 67-74.

13. Колесников А.И. Автоматизация учета клиентов: технологии и инструменты. - М.: Издательство "Горячая линия-Телеком", 2018.

14. Антонова О.А. Программное обеспечение для автоматизации учета клиентов в малом бизнесе. - СПб.: Питер, 2019.

15. Черникова Н.Ю. Разработка программного обеспечения для учета клиентов с использованием баз данных. - М.: Финансы и статистика, 2020.

16. Смирнов Д.А. Анализ и сравнение современных систем учета клиентов. - М.: Интернет-Университет Информационных Технологий, 2021.

17. Иванов С.М. Проектирование и разработка ПО для учета клиентов в сфере дистрибуции. - М.: Издательский дом "Эксмо", 2023.

18. "Программное обеспечение для автоматизации учета клиентов в малом бизнесе", Антонова О.А., 2018.

19. "Автоматизация учета клиентов: технологии и инструменты", Колесников А.И., 2019.

20. "Анализ и сравнение современных систем учета клиентов", Смирнов Д.А., 2021.

21. "Проектирование и разработка ПО для учета клиентов в сфере дистрибуции", Иванов С.М., 2023.

22. "Конструирование программных продуктов", Роберт К. Мартин, 2018.

23. "Программирование на Python", Марк Лутц, 2021.