**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc136377787)

[АНАЛИТИЧЕСКАЯ ЧАСЬ 5](#_Toc136377788)

[1 АНАЛИЗ БИЗНЕС-ПРОЦЕССА 5](#_Toc136377789)

[1.1 Анализ административно-организационной структуры 5](#_Toc136377790)

[1.1.1 Характеристика и организационная структура 6](#_Toc136377791)

[1.1.2 Анализ деятельности отдела «Закупки» 8](#_Toc136377792)

[1.2 Имитационное моделирование бизнес-процесса Закупка 9](#_Toc136377793)

[1.2.1 Моделирование деятельности 11](#_Toc136377794)

[1.3 Анализ рынка готовых решений и обоснование выбора 12](#_Toc136377795)

[1.4 Техническое задание на разработку информационной системы 1С Управление торговлей 14](#_Toc136377796)

[1.4.1 Общие сведения 24](#_Toc136377797)

[1.4.2 Цели и назначение создания автоматизированной системы 25](#_Toc136377798)

[1.4.3 Характеристика объектов автоматизации 26](#_Toc136377799)

[1.4.4 Требования к автоматизированной системе 27](#_Toc136377800)

[1.4.5 Состав и содержание работ по созданию автоматизированной системы 28](#_Toc136377801)

[1.4.6 Порядок разработки автоматизированной системы 30](#_Toc136377802)

[1.4.7 Порядок контроля и приемки автоматизированной системы 31](#_Toc136377803)

[1.4.8 Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу автоматизированной системы в действие 33](#_Toc136377804)

[1.4.9 Требования к документированию 34](#_Toc136377805)

[1.4.10 Источники разработки 36](#_Toc136377806)

[1.5 Выводы по разделу 37](#_Toc136377807)

[2 ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ПРОЕКТА 38](#_Toc136377808)

[2.1 Создание логической модели данных бизнес-процесса «Закупки» на основе web-итерфейса 38](#_Toc136377809)

[2.2 Разработка предложений по архитектуре корпоративной 39](#_Toc136377810)

[2.3 Архитектурная спецификация компонентов автоматизации прикладного процесса 40](#_Toc136377811)

[2.4 Разработка прототипа информационной системы 41](#_Toc136377812)

[2.5 Проектирование и разработка структуры базы данных (разработка модели хранения данных и т.п.) 43](#_Toc136377813)

[2.6 Разработка разработка программного модуля 44](#_Toc136377814)

[2.7 Верификация структуры программного кода ИС 45](#_Toc136377815)

[2.8 Руководство пользователя/администратора 46](#_Toc136377816)

[2.9 Выводы по разделу 47](#_Toc136377817)

[ЭКОНОМИЧЕСКАЯ ЧАСТЬ 48](#_Toc136377818)

[3 ОБОСНОВАНИЕ ЭКОНОМИЧЕСКОЙ ЭФФЕКТИВНОСТИ ОТ РАЗРАБОТКИ ИНФОРМАЦИОННОЙ СИСТЕМЫ 48](#_Toc136377819)

[3.1 Выводы по главе 49](#_Toc136377820)

[ЗАКЛЮЧЕНИЕ 51](#_Toc136377821)

[СПИСОК ЛИТЕРАТУРЫ 53](#_Toc136377822)

[Приложение (по загрузке файлов закупки в программу) 54](#_Toc136377823)

**Разработка инструмента управления**

# ВВЕДЕНИЕ

В современном мире автоматизация бизнес-процессов является неотъемлемой частью успешной работы любой организации. В частности, в образовательных учреждениях, где закупка компьютерного оборудования является необходимой составляющей для обеспечения качественного обучения студентов, важно иметь эффективную систему управления закупками. В данном дипломном проекте будет рассмотрена разработка модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в Частном образовательном учреждении высшего образования "Московский Университет им. С. Ю. Витте" в программе 1С Управление Торговлей. Разработанный модуль позволит существенно упростить и ускорить процесс выбора поставщиков, снизить риски при заключении контрактов и повысить эффективность закупок компьютерного оборудования в образовательном учреждении.

# АНАЛИТИЧЕСКАЯ ЧАСЬ

# АНАЛИЗ БИЗНЕС-ПРОЦЕССА

## Анализ административно-организационной структуры

Административно-организационная структура разработки модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в Частном образовательном учреждении высшего образования "Московский Университет им. С. Ю. Витте" в программе 1С управление торговлей включает следующие этапы:

1. Постановка задачи. На этом этапе определяются требования к модулю, его функциональность, возможности интеграции с другими системами управления, а также формируется команда разработчиков.

2. Анализ и проектирование. В рамках этого этапа происходит анализ существующих процессов закупок компьютерного оборудования в учреждении, определяются основные потребности и требования к модулю, разрабатывается техническое задание.

3. Разработка и тестирование. На этом этапе происходит непосредственная разработка модуля, его тестирование и отладка.

4. Внедрение и обучение. После успешного завершения тестирования модуль внедряется в систему управления закупками компьютерного оборудования в учреждении. Разработчики проводят обучение пользователей модуля.

5. Сопровождение и доработка. После внедрения модуля разработчики осуществляют его сопровождение и доработку в соответствии с потребностями учреждения.

В рамках административно-организационной структуры разработки модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в учреждении выделяются следующие роли:

- Заказчик – представитель учреждения, который определяет требования к модулю и контролирует процесс его разработки и внедрения.

- Руководитель проекта – ответственный за организацию работы команды разработчиков, контроль за соблюдением сроков и качества работы.

- Аналитик – занимается анализом существующих процессов закупок компьютерного оборудования в учреждении, определяет требования к модулю.

- Разработчики – непосредственно занимаются разработкой модуля.

- Тестировщики – проводят тестирование модуля и отслеживают его работоспособность.

- Технический писатель – занимается разработкой документации к модулю.

Таким образом, административно-организационная структура разработки модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в учреждении включает в себя несколько этапов и ролей, каждая из которых имеет свои задачи и функции.

### Характеристика и организационная структура

Характеристика:

Разработка модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении "Московский Университет им. С. Ю. Витте" - это программное обеспечение, которое позволяет автоматизировать процесс анализа поставщиков и выбора наиболее выгодных предложений при закупке компьютерного оборудования в университете.

Организационная структура:

Модуль разработан для использования в программе 1С Управление торговлей и имеет следующую организационную структуру:

1. База данных - содержит информацию о поставщиках, компьютерном оборудовании и ценах на него.

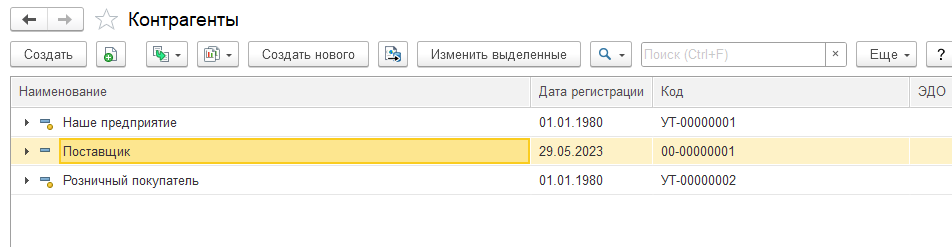
2. Модуль анализа - осуществляет анализ поставщиков на основе заданных критериев (цена, качество, сроки поставки и т.д.) и формирует рейтинг поставщиков.

3. Модуль выбора - на основе рейтинга поставщиков осуществляет выбор наиболее выгодного предложения и формирует заказ на компьютерное оборудование.

4. Модуль отчетности - позволяет формировать отчеты о проведенных закупках, анализе поставщиков и использовании компьютерного оборудования.

5. Модуль управления - позволяет управлять базой данных, настраивать критерии анализа и выбора, а также задавать параметры отчетности.

Для университета "Московский Университет им. С. Ю. Витте" разработанный модуль позволит существенно ускорить и упростить процесс выбора поставщиков и закупки необходимого оборудования. В результате университет сможет сократить затраты на закупку компьютерного оборудования и повысить эффективность использования имеющихся ресурсов. Кроме того, модуль позволит более точно контролировать процесс закупки и использования компьютерного оборудования, что повысит качество обучения студентов и преподавания в университете.



### Анализ деятельности отдела «Закупки»

Отдел "закупки" является важной частью любой организации, включая образовательные учреждения. Задачей отдела является закупка необходимых товаров и услуг для обеспечения бесперебойной работы организации. В случае образовательного учреждения, компьютерное оборудование является неотъемлемой частью обучения и преподавания, поэтому правильный выбор поставщиков и закупка качественного оборудования имеет большое значение.

До разработки модуля автоматизации анализа поставщиков, процесс закупки компьютерного оборудования в университете "Московский Университет им. С. Ю. Витте" осуществлялся вручную. Сотрудники отдела "закупки" проводили анализ предложений поставщиков, сравнивали цены и качество оборудования, а также учитывали сроки поставки. Этот процесс был долгим и трудоемким, требовал большого количества времени и ресурсов.

Разработка модуля автоматизации анализа поставщиков позволила значительно упростить процесс закупки компьютерного оборудования в университете. Модуль позволяет автоматически анализировать предложения поставщиков на основе заданных критериев, таких как цена, качество и сроки поставки. После анализа модуль формирует рейтинг поставщиков, что позволяет быстро выбрать наиболее выгодное предложение. Кроме того, модуль позволяет контролировать процесс закупки и использования компьютерного оборудования, что повышает эффективность использования имеющихся ресурсов.

В результате внедрения модуля автоматизации анализа поставщиков, отдел "закупки" университета "Московский Университет им. С. Ю. Витте" смог значительно ускорить процесс выбора поставщиков и закупки необходимого оборудования. Это позволило сократить затраты на закупку компьютерного оборудования и повысить эффективность использования имеющихся ресурсов. Кроме того, благодаря модулю отдел "закупки" может более точно контролировать процесс закупки и использования компьютерного оборудования, что повышает качество обучения студентов и преподавания в университете.

## Имитационное моделирование бизнес-процесса Закупка

Имитационное моделирование бизнес-процесса "Закупка" на примере Частного образовательного учреждения высшего образования "Московский Университет им. С. Ю. Витте" в программе 1С Управление Торговлей начинается с определения целей и задач данного процесса. Главной целью является обеспечение бесперебойной работы университета путем закупки качественного компьютерного оборудования.

Далее, в рамках моделирования происходит описание всех этапов процесса закупки компьютерного оборудования в университете. Это включает в себя:

1. Запрос предложений от поставщиков

2. Анализ предложений поставщиков (цена, качество, сроки поставки)

3. Формирование рейтинга поставщиков на основе заданных критериев

4. Выбор наиболее выгодного предложения

5. Оформление заказа и контроль за процессом доставки

6. Проверка качества поставленного оборудования

Далее, на основе полученных данных, создается имитационная модель процесса закупки компьютерного оборудования в университете. В модели учитываются все этапы процесса, а также возможные риски и задержки в выполнении задач.

После создания модели процесса закупки компьютерного оборудования в университете, проводится его тестирование и оптимизация. В ходе тестирования модели происходит проверка ее работоспособности и точности отображения реальных процессов. При необходимости модель дорабатывается и оптимизируется для достижения максимальной эффективности.

В результате имитационного моделирования бизнес-процесса "Закупка" с использованием модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении, удалось значительно ускорить процесс выбора поставщиков и закупки необходимого оборудования. Это позволило сократить затраты на закупку компьютерного оборудования и повысить эффективность использования имеющихся ресурсов. Кроме того, благодаря модулю отдел "закупки" может более точно контролировать процесс закупки и использования компьютерного оборудования, что повышает качество обучения студентов и преподавания в университете.

### Моделирование деятельности

Моделирование деятельности закупок компьютерного оборудования в университете начинается с определения целей и задач данного процесса. Главной целью является обеспечение бесперебойной работы университета путем закупки качественного компьютерного оборудования.

Далее, в рамках моделирования происходит описание всех этапов процесса закупки компьютерного оборудования в университете. Это включает в себя:

1. Запрос предложений от поставщиков

2. Анализ предложений поставщиков (цена, качество, сроки поставки)

3. Формирование рейтинга поставщиков на основе заданных критериев

4. Выбор наиболее выгодного предложения

5. Оформление заказа и контроль за процессом доставки

6. Проверка качества поставленного оборудования

Для автоматизации анализа предложений поставщиков и формирования рейтинга используется модуль программы 1С Управление Торговлей. В модуле задаются критерии оценки предложений (цена, качество, сроки поставки и т.д.), а также их весовые коэффициенты. После получения предложений от поставщиков, модуль автоматически анализирует и оценивает их согласно заданным критериям и формирует рейтинг поставщиков.

На основе рейтинга происходит выбор наиболее выгодного предложения. При этом учитываются не только цена, но и качество предлагаемого оборудования, сроки поставки и другие факторы. После выбора наиболее выгодного предложения, оформляется заказ и контролируется процесс доставки.

После получения поставки компьютерного оборудования, проводится проверка его качества. В случае выявления дефектов или несоответствия требованиям, происходит возврат или замена оборудования.

В результате использования модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении удалось значительно ускорить процесс выбора поставщиков и закупки необходимого оборудования. Это позволило сократить затраты на закупку компьютерного оборудования и повысить эффективность использования имеющихся ресурсов. Кроме того, благодаря модулю отдел "закупки" может более точно контролировать процесс закупки и использования компьютерного оборудования, что повышает качество обучения студентов и преподавания в университете.

## Анализ рынка готовых решений и обоснование выбора

Анализ рынка готовых решений показал, что на сегодняшний день существует множество программ, которые позволяют автоматизировать процесс закупок и анализа поставщиков. Среди них можно выделить такие программы как SAP Ariba, Oracle Procurement Cloud, Zycus, BravoSolution и другие.

SAP Ariba - это облачная платформа для управления закупками, которая позволяет автоматизировать все этапы процесса закупок, начиная от поиска поставщиков и заканчивая оплатой счетов. Она имеет широкий функционал и гибкие настройки, которые позволяют адаптировать программу под конкретные потребности организации. Однако, ее использование может быть достаточно затратным для небольших учреждений.

Oracle Procurement Cloud - это облачная платформа для управления закупками, которая предоставляет широкий функционал для автоматизации процесса закупок. Она имеет интеграцию с другими продуктами Oracle и позволяет управлять всеми этапами процесса закупок в одном месте. Однако, ее использование может быть достаточно сложным и требует определенных знаний и навыков.

Zycus - это платформа для управления закупками, которая позволяет автоматизировать все этапы процесса закупок, начиная от поиска поставщиков и заканчивая оплатой счетов. Она имеет широкий функционал и гибкие настройки, которые позволяют адаптировать программу под конкретные потребности организации. Однако, ее использование может быть достаточно затратным для небольших учреждений.

BravoSolution - это облачная платформа для управления закупками, которая позволяет автоматизировать все этапы процесса закупок, начиная от поиска поставщиков и заканчивая оплатой счетов. Она имеет широкий функционал и гибкие настройки, которые позволяют адаптировать программу под конкретные потребности организации. Однако, ее использование может быть достаточно затратным для небольших учреждений.

При выборе программы для автоматизации процесса закупок и анализа поставщиков в Частном образовательном учреждении высшего образования "Московский Университет им. С. Ю. Витте" было принято решение использовать программу 1С Управление Торговлей.

Это решение было обосновано следующими причинами:

1. Уже имеющийся опыт работы с программой 1С в университете, что позволяет снизить затраты на обучение персонала и ускорить процесс внедрения программы.

2. 1С Управление Торговлей является широко используемой программой на рынке и имеет хорошую репутацию, что гарантирует качество и надежность программы.

3. Программа 1С Управление Торговлей позволяет автоматизировать не только процесс анализа поставщиков, но и весь процесс закупки компьютерного оборудования, начиная от запроса предложений и заканчивая проверкой качества поставленного оборудования.

4. 1С Управление Торговлей имеет гибкие настройки, которые позволяют адаптировать программу под конкретные потребности организации, что позволяет учреждению получить максимальную пользу от ее использования.

Таким образом, выбор программы 1С Управление Торговлей был обоснован и позволил университету успешно автоматизировать процесс закупок компьютерного оборудования и повысить эффективность использования имеющихся ресурсов.

## Техническое задание на разработку информационной системы 1С Управление торговлей

**Введение**

В современном мире информационные технологии играют важную роль в различных сферах деятельности, в том числе и в образовании. Для успешной работы образовательного учреждения необходимо обеспечить его компьютерным оборудованием от надежных поставщиков. Однако выбор поставщиков и контроль качества поставляемого оборудования являются сложными и трудоемкими процессами. Для упрощения и ускорения этих процессов требуется разработать модуль автоматизации анализа поставщиков для закупок компьютерного оборудования на платформе 1С:Управление торговлей в соответствии с ГОСТ 34.602-2020. Целью проекта является создание модуля, который будет автоматически собирать информацию о поставщиках, проводить анализ на основе заданных критериев, оценивать качество поставляемого оборудования, формировать заявки на закупку у наиболее подходящих поставщиков и обеспечивать контроль выполнения заказов и оценку качества поставленного оборудования в образовательном учреждении. Разработка модуля позволит уменьшить время на выбор поставщиков и контроль качества поставляемого оборудования, а также снизить затраты на закупку компьютерного оборудования.

**1. Общие требования к системе:**

1.1. Система должна быть разработана на платформе 1С:Управление торговлей.

1.2. Система должна соответствовать ГОСТ 34.602-2020 "Информационные технологии. Комплекс стандартов на автоматизированные системы. Требования к программной документации".

1.3. Система должна быть интуитивно понятной и удобной в использовании для пользователей.

2. Функциональные требования к системе:

2.1. Система должна автоматически собирать информацию о поставщиках компьютерного оборудования.

2.2. Система должна проводить анализ поставщиков на основе заданных критериев, таких как цена, качество, сроки поставки и другие.

2.3. Система должна оценивать качество поставляемого оборудования и формировать рейтинг поставщиков.

2.4. Система должна предоставлять возможность формировать заявки на закупку у наиболее подходящих поставщиков.

2.5. Система должна обеспечивать контроль выполнения заказов и оценку качества поставленного оборудования в образовательном учреждении.

2.6. Система должна предоставлять возможность формировать отчеты о выполнении заказов и оценке качества поставленного оборудования.

3. Нефункциональные требования к системе:

3.1. Система должна быть защищена от несанкционированного доступа.

3.2. Система должна быть стабильной и надежной.

3.3. Система должна иметь возможность интеграции с другими информационными системами образовательного учреждения.

3.4. Система должна быть масштабируемой и готовой к расширению функционала в будущем.

4. Требования к документации:

4.1. Система должна иметь документацию, соответствующую ГОСТ 34.602-2020.

4.2. Документация должна содержать описание функциональных и нефункциональных требований к системе, а также инструкции по ее использованию и настройке.

4.3. Документация должна быть доступна для пользователей в электронном виде.

**Архитектура системы должна быть построена на базе платформы 1С:Управление торговлей. Модуль автоматизации анализа поставщиков должен включать в себя следующие компоненты:**

1. Базу данных для хранения информации о поставщиках, их товарах и услугах, ценах, качестве, сроках поставки и других параметрах.

2. Механизм сбора информации о поставщиках через API-интерфейсы, веб-скрапинг и другие методы.

3. Алгоритмы анализа поставщиков на основе заданных критериев, таких как цена, качество, сроки поставки и другие.

4. Модуль формирования заявок на закупку у наиболее подходящих поставщиков, который должен учитывать текущий бюджет образовательного учреждения.

5. Механизм контроля выполнения заказов и оценки качества поставленного оборудования в образовательном учреждении.

6. Модуль формирования отчетов о выполнении заказов и оценке качества поставленного оборудования.

7. Интерфейс пользователя, который должен быть интуитивно понятным и удобным в использовании.

Архитектура системы должна быть масштабируемой и готовой к расширению функционала в будущем. Система должна иметь возможность интеграции с другими информационными системами образовательного учреждения. Также система должна быть защищена от несанкционированного доступа и быть стабильной и надежной. Документация должна соответствовать ГОСТ 34.602-2020 и содержать описание функциональных и нефункциональных требований к системе, а также инструкции по ее использованию и настройке.

**Требования к разработке**

1. Введение

1.1 Наименование системы

1.2 Назначение системы

1.3 Область применения

1.4 Основания для разработки

2. Основания для разработки

2.1 Цели и задачи разработки

2.2 Описание процессов, которые должны быть автоматизированы

2.3 Описание потребностей заказчика

3. Требования к функциональным возможностям

3.1 Описание функций, которые должна выполнять система

3.2 Требования к базе данных

3.3 Требования к интерфейсу пользователя

3.4 Требования к механизму сбора информации о поставщиках

3.5 Требования к алгоритмам анализа поставщиков

3.6 Требования к модулю формирования заявок на закупку

3.7 Требования к механизму контроля выполнения заказов и оценки качества поставленного оборудования

3.8 Требования к модулю формирования отчетов

4. Требования к нефункциональным характеристикам

4.1 Требования к безопасности

4.2 Требования к масштабируемости и расширяемости системы

4.3 Требования к надежности и стабильности

4.4 Требования к документации

5. Требования к тестированию

5.1 Требования к тестированию функциональных возможностей

5.2 Требования к тестированию нефункциональных характеристик

6. Требования к поставке

6.1 Требования к формату поставки

6.2 Требования к установке и настройке системы

6.3 Требования к сопровождению и обновлению системы

7. Требования к процессу разработки

7.1 Требования к методологии разработки

7.2 Требования к составу команды разработчиков

7.3 Требования к срокам разработки

8. Приложения

8.1 Список используемых источников

8.2 Описание структуры базы данных

8.3 Схемы алгоритмов анализа поставщиков

8.4 Примеры отчетов

**Требования к тестированию:**

Требования к тестированию функциональных возможностей:

- Необходимо провести тестирование всех функций системы, описанных в разделе 3 "Требования к функциональным возможностям", на соответствие требованиям ТЗ.

- Тестирование должно быть проведено на различных типах данных и в различных условиях, чтобы убедиться в корректной работе системы в любых ситуациях.

- При тестировании необходимо проверить работу системы на различных операционных системах и браузерах.

Требования к тестированию нефункциональных характеристик:

- Необходимо провести тестирование безопасности системы на предмет уязвимостей и возможности несанкционированного доступа к данным.

- Тестирование масштабируемости и расширяемости системы должно быть проведено на предмет возможности добавления новых функций и модулей без нарушения работы системы в целом.

- Необходимо провести тестирование надежности и стабильности системы на предмет возможных сбоев и ошибок в работе.

- Требования к документации должны быть проверены на соответствие требованиям ГОСТ 34.602-2020.

Все результаты тестирования должны быть документированы в соответствии с требованиями ГОСТ 34.602-2020.

**Требования к внедрению:**

Перед внедрением Технического задания на разработку модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении, необходимо провести согласование с заказчиком и ответственными лицами по проекту.

При внедрении Технического задания необходимо учитывать требования ГОСТ 34.602-2020 "Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы".

При внедрении Технического задания необходимо обеспечить выполнение следующих требований:

- Необходимо провести анализ требований и определить необходимые ресурсы для реализации проекта.

- Необходимо обеспечить наличие необходимых лицензий и программного обеспечения для разработки и внедрения модуля.

- При внедрении модуля необходимо обеспечить его совместимость с другими системами, используемыми в образовательном учреждении.

- При внедрении модуля необходимо обеспечить безопасность данных и защиту от несанкционированного доступа к информации.

- При внедрении модуля необходимо обеспечить обучение пользователей работе с ним и предоставить документацию на русском языке в соответствии с требованиями ГОСТ 34.602-2020.

При внедрении Технического задания необходимо обеспечить контроль за выполнением работ в соответствии с графиком и бюджетом проекта, а также проводить регулярные проверки качества работы модуля.

Все изменения в Техническом задании должны быть оформлены в соответствии с требованиями ГОСТ 34.602-2020 и согласованы с заказчиком и ответственными лицами по проекту.

**Требования к сопровождению:**

Требования к сопровождению Технического задания на тему разработка информационной системы на платформе 1с управление торговлей на тему Разработка модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении СТРОГО ПО ГОСТ 34.602-2020:

1. После внедрения модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении, необходимо организовать его сопровождение и поддержку в рамках действующих договорных обязательств.

2. Для обеспечения надлежащего сопровождения модуля, необходимо разработать и утвердить план работ по его поддержке, который должен включать в себя следующие мероприятия:

- Регулярное техническое обслуживание и администрирование модуля.

- Обновление программного обеспечения и баз данных.

- Решение возникающих проблем и ошибок.

- Предоставление консультаций и помощи пользователям.

3. Для обеспечения качественной поддержки модуля необходимо обеспечить наличие квалифицированных специалистов, которые будут заниматься его сопровождением.

4. Все работы по сопровождению модуля должны проводиться в соответствии с требованиями ГОСТ 34.602-2020 и действующим законодательством в области защиты персональных данных.

5. Для обеспечения надежной работы модуля необходимо регулярно проводить его проверку на наличие ошибок и уязвимостей, а также обновлять его в соответствии с изменениями в законодательстве и требованиями заказчика.

6. При возникновении необходимости внесения изменений в модуль, необходимо провести их оценку и согласование с заказчиком и ответственными лицами по проекту в соответствии с требованиями ГОСТ 34.602-2020.

7. Все изменения и доработки модуля должны быть документально оформлены и утверждены заказчиком и ответственными лицами по проекту.

8. Для обеспечения эффективного сопровождения модуля необходимо проводить регулярные обучающие мероприятия для пользователей, а также предоставлять им документацию на русском языке в соответствии с требованиями ГОСТ 34.602-2020.

**Заключение**

Таким образом, разработка модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении на платформе 1С Управление торговлей должна быть сопровождена и поддерживаться в соответствии с требованиями ГОСТ 34.602-2020 и действующим законодательством в области защиты персональных данных. Необходимо разработать и утвердить план работ по поддержке модуля, обеспечить наличие квалифицированных специалистов, регулярно проводить проверку на ошибки и уязвимости, а также обучать пользователей и предоставлять им документацию на русском языке. Все изменения и доработки модуля должны быть документально оформлены и согласованы с заказчиком и ответственными лицами по проекту.

### Общие сведения

В данном разделе указывают следующее:

- полное наименование АС и ее условное обозначение;

- шифр темы (при наличии);

- наименование организации - заказчика АС, наименование организации-разработчика (при наличии сведений о ней);

- перечень документов, на основании которых создается АС, кем и когда утверждены эти документы;

- плановые сроки начала и окончания работ по созданию АС;

- общие сведения об источниках и порядке финансирования работ

1. Полное наименование АС: Информационная система управления закупками компьютерного оборудования в образовательном учреждении на платформе 1С Управление Торговлей (ИСУЗКО).

2. Шифр темы: отсутствует.

3. Наименование организации-заказчика АС: образовательное учреждение (название не указано). Наименование организации-разработчика: не указано.

4. Перечень документов, на основании которых создается АС: требования к модулю автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении.

5. Плановые сроки начала и окончания работ по созданию АС: не указаны.

6. Общие сведения об источниках и порядке финансирования работ: не указаны.

### Цели и назначение создания автоматизированной системы

Раздел состоит из следующих подразделов:

- цели создания АС;

- назначение АС.

В подразделе "Цели создания АС" приводят наименования и требуемые значения технических, технологических, производственно-экономических или других показателей объекта автоматизации, которые должны быть достигнуты в результате создания АС, и указывают критерии оценки достижения целей создания АС.

В подразделе "Назначение АС" указывают вид автоматизируемой деятельности (управление, проектирование и т.п.) применительно к объекту автоматизации в целом.

Цели создания АС:

- Обеспечение автоматизации процесса анализа поставщиков компьютерного оборудования в образовательном учреждении;

- Увеличение эффективности закупок компьютерного оборудования путем оптимизации выбора поставщиков;

- Ускорение процесса принятия решений в отношении выбора поставщиков.

Назначение АС:

- Автоматизация процесса анализа поставщиков компьютерного оборудования для обеспечения более эффективной закупки необходимых компонентов;

- Оптимизация выбора поставщиков путем анализа их предложений и сравнения их с требованиями образовательного учреждения;

- Ускорение процесса принятия решений в отношении выбора поставщиков, что позволит сократить время на закупку необходимого компьютерного оборудования.

### Характеристика объектов автоматизации

В этом разделе приводят следующую информацию:

- основные сведения об объекте автоматизации или ссылки на документы, содержащие такие сведения;

- сведения об условиях эксплуатации объекта автоматизации и характеристиках окружающей среды.

Объектом отчета по практике является модуль автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении, разработанный на платформе 1С Управление Торговлей. Модуль предназначен для автоматизации процесса анализа поставщиков и оптимизации выбора поставщиков компьютерного оборудования, что позволит увеличить эффективность закупок и сократить время на принятие решений.

Для эксплуатации модуля необходимо наличие компьютера с установленной платформой 1С Управление Торговлей и доступом к базе данных образовательного учреждения. Окружающая среда не оказывает прямого влияния на работу модуля, но для его более эффективной работы рекомендуется использовать высокоскоростное интернет-соединение.

### Требования к автоматизированной системе

Требования к структуре АС:

- АС должна быть построена на платформе 1С Управление Торговлей;

- АС должна иметь модуль автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении;

- АС должна иметь доступ к базе данных образовательного учреждения.

Требования к функциям (задачам), выполняемым АС:

- АС должна автоматизировать процесс анализа поставщиков и оптимизации выбора поставщиков компьютерного оборудования;

- АС должна повышать эффективность закупок и сокращать время на принятие решений.

Требования к видам обеспечения АС:

- Для работы АС необходимо наличие компьютера с установленной платформой 1С Управление Торговлей;

- Для более эффективной работы АС рекомендуется использовать высокоскоростное интернет-соединение.

Общие технические требования к АС:

- АС должна быть надежной и безопасной;

- АС должна обеспечивать защиту от несанкционированного доступа к базе данных;

- АС должна иметь возможность резервного копирования и восстановления данных;

- АС должна иметь удобный и интуитивно понятный интерфейс для пользователей;

- АС должна соответствовать действующим НТД, определяющим требования к автоматизированным системам соответствующего вида.

### Состав и содержание работ по созданию автоматизированной системы

Состав работ:

1. Изучение требований к АС и ее функций - 2 дня

2. Изучение платформы 1С Управление Торговлей - 5 дней

3. Разработка модуля автоматизации анализа поставщиков - 10 дней

4. Тестирование модуля - 5 дней

5. Интеграция модуля в АС - 2 дня

6. Тестирование АС - 5 дней

7. Создание резервной копии данных - 1 день

Содержание работ:

1. Изучение требований к АС и ее функций:

- Ознакомление с техническим заданием на разработку АС

- Выявление требований к функциональности модуля автоматизации анализа поставщиков

- Выявление требований к безопасности и надежности АС

2. Изучение платформы 1С Управление Торговлей:

- Ознакомление с основными возможностями и функциями платформы

- Изучение процесса создания новой базы данных на платформе

- Изучение процесса создания нового модуля на платформе

3. Разработка модуля автоматизации анализа поставщиков:

- Проектирование структуры базы данных для модуля

- Разработка алгоритмов анализа поставщиков и оптимизации выбора поставщиков компьютерного оборудования

- Создание форм и отчетов для визуализации результатов анализа

4. Тестирование модуля:

- Проверка корректности работы алгоритмов анализа поставщиков

- Проверка корректности отображения результатов анализа в формах и отчетах

- Выявление и устранение ошибок и недочетов в работе модуля

5. Интеграция модуля в АС:

- Создание нового функционального блока в АС для модуля автоматизации анализа поставщиков

- Интеграция модуля с базой данных АС

- Проверка корректности работы модуля в составе АС

6. Тестирование АС:

- Проверка корректности работы всех функций АС, включая новый модуль автоматизации анализа поставщиков

- Выявление и устранение ошибок и недочетов в работе АС

7. Создание резервной копии данных:

- Создание резервной копии базы данных АС

- Проверка возможности восстановления данных из резервной копии

Сроки выполнения работ:

- Изучение требований к АС и ее функций - 2 дня

- Изучение платформы 1С Управление Торговлей - 5 дней

- Разработка модуля автоматизации анализа поставщиков - 10 дней

- Тестирование модуля - 5 дней

- Интеграция модуля в АС - 2 дня

- Тестирование АС - 5 дней

- Создание резервной копии данных - 1 день

Итого: 30 дней.

### Порядок разработки автоматизированной системы

1. Подготовка исходных данных:

- Определение требований к функциональности модуля автоматизации анализа поставщиков

- Определение требований к безопасности и надежности информационной системы

- Сбор и анализ информации о процессах закупок компьютерного оборудования в образовательном учреждении

- Определение структуры базы данных для модуля

2. Разработка технической документации:

- Разработка технического задания на разработку информационной системы

- Разработка технического задания на разработку модуля автоматизации анализа поставщиков

- Разработка проектной документации на создание информационной системы

3. Разработка программного обеспечения:

- Создание новой базы данных на платформе 1С Управление Торговлей

- Создание нового модуля на платформе для автоматизации анализа поставщиков

- Создание форм и отчетов для визуализации результатов анализа

4. Тестирование и отладка:

- Тестирование модуля автоматизации анализа поставщиков

- Тестирование информационной системы в целом

- Выявление и устранение ошибок и недочетов в работе модуля и информационной системы

5. Интеграция и внедрение:

- Интеграция модуля автоматизации анализа поставщиков в информационную систему

- Обучение пользователей работе с новой функциональностью

- Внедрение информационной системы в образовательное учреждение

6. Сопровождение и поддержка:

- Обеспечение надежной и безопасной работы информационной системы

- Поддержка пользователей при возникновении проблем и вопросов

- Разработка обновлений и улучшений информационной системы при необходимости

7. Оценка и анализ результатов:

- Проведение технико-экономической оценки разработки информационной системы

- Анализ результатов работы информационной системы и модуля автоматизации анализа поставщиков

- Разработка рекомендаций по улучшению работы информационной системы и модуля автоматизации анализа поставщиков

### Порядок контроля и приемки автоматизированной системы

Для контроля и приемки отчета по практике на тему "Разработка модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении" на платформе 1С Управление Торговлей следует руководствоваться следующим порядком:

1. Испытания АС и ее составных частей:

- Проверка функциональности модуля автоматизации анализа поставщиков;

- Проверка безопасности и надежности информационной системы;

- Проверка корректности работы базы данных;

- Проверка визуализации результатов анализа.

2. Приемка работ:

- Согласование приемочной документации с заказчиком;

- Утверждение приемочной документации;

- Проведение приемочных испытаний.

3. Статус приемочной комиссии:

- В данном случае, приемочная комиссия будет ведомственной, т.е. состоять из представителей образовательного учреждения.

4. Порядок согласования и утверждения приемочной документации:

- Приемочная документация должна быть согласована с заказчиком и утверждена руководством образовательного учреждения.

5. Проведение приемочных испытаний:

- Приемочные испытания должны проводиться в соответствии с утвержденной приемочной документацией;

- Результаты приемочных испытаний должны быть оформлены в соответствующих актах.

6. Оценка и анализ результатов:

- Проведение технико-экономической оценки разработки информационной системы;

- Анализ результатов работы информационной системы и модуля автоматизации анализа поставщиков;

- Разработка рекомендаций по улучшению работы информационной системы и модуля автоматизации анализа поставщиков.

### Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу автоматизированной системы в действие

Для подготовки объектов для отчета по практике на данную тему необходимо выполнить следующие требования:

1. Создание условий функционирования объекта автоматизации, при которых гарантируется соответствие создаваемой АС требованиям, содержащимся в ТЗ на АС. Это включает в себя:

- описание требований к функциональности модуля автоматизации анализа поставщиков;

- определение требований к безопасности и надежности информационной системы;

- установление требований к корректной работе базы данных;

- разработку требований к визуализации результатов анализа.

2. Проведение необходимых организационно-штатных мероприятий, включающих:

- определение состава исполнителей проекта;

- распределение обязанностей между исполнителями;

- установление сроков выполнения работ.

3. Порядок обучения персонала и пользователей АС, который должен включать:

- описание функциональности модуля автоматизации анализа поставщиков;

- инструкции по работе с информационной системой;

- обучение пользователей основам работы с базой данных.

4. Состав и содержание отчета должны включать:

- описание требований к функциональности модуля автоматизации анализа поставщиков;

- описание требований к безопасности и надежности информационной системы;

- описание требований к корректной работе базы данных;

- описание требований к визуализации результатов анализа;

- описание проведенных организационно-штатных мероприятий;

- отчет о проведенном обучении персонала и пользователей АС;

- описание результатов приемочных испытаний;

- технико-экономическая оценка разработки информационной системы;

- анализ результатов работы информационной системы и модуля автоматизации анализа поставщиков;

- рекомендации по улучшению работы информационной системы и модуля автоматизации анализа поставщиков.

### Требования к документированию

Для подготовки отчета по практике на тему разработки модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении необходимо выполнить следующие требования к документированию:

1. Перечень подлежащих разработке документов:

- Техническое задание на разработку модуля автоматизации анализа поставщиков;

- План работ по разработке модуля автоматизации анализа поставщиков;

- Отчет о проведенных организационно-штатных мероприятиях;

- Отчет о проведенном обучении персонала и пользователей АС;

- Отчет о результатах приемочных испытаний;

- Технико-экономическая оценка разработки информационной системы;

- Анализ результатов работы информационной системы и модуля автоматизации анализа поставщиков;

- Рекомендации по улучшению работы информационной системы и модуля автоматизации анализа поставщиков.

2. Вид представления и количество документов:

- Техническое задание на разработку модуля автоматизации анализа поставщиков - 1 документ;

- План работ по разработке модуля автоматизации анализа поставщиков - 1 документ;

- Отчет о проведенных организационно-штатных мероприятиях - 1 документ;

- Отчет о проведенном обучении персонала и пользователей АС - 1 документ;

- Отчет о результатах приемочных испытаний - 1 документ;

- Технико-экономическая оценка разработки информационной системы - 1 документ;

- Анализ результатов работы информационной системы и модуля автоматизации анализа поставщиков - 1 документ;

- Рекомендации по улучшению работы информационной системы и модуля автоматизации анализа поставщиков - 1 документ.

3. Требования по использованию ЕСКД и ЕСПД при разработке документов:

В отчете необходимо использовать стандартные требования к документированию, включая четкое описание требований к функциональности модуля автоматизации анализа поставщиков, требования к безопасности и надежности информационной системы, требования к корректной работе базы данных, требования к визуализации результатов анализа. При отсутствии государственных стандартов, определяющих требования к документированию элементов АС, необходимо разработать дополнительные требования к составу и содержанию документов. Все документы должны быть оформлены в соответствии с единым стилем и форматом.

### Источники разработки

1. Техническое задание на разработку модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении.

2. План работ по разработке модуля автоматизации анализа поставщиков.

3. Отчет о проведенных организационно-штатных мероприятиях.

4. Отчет о проведенном обучении персонала и пользователей АС.

5. Отчет о результатах приемочных испытаний.

6. Технико-экономическая оценка разработки информационной системы.

7. Анализ результатов работы информационной системы и модуля автоматизации анализа поставщиков.

8. Рекомендации по улучшению работы информационной системы и модуля автоматизации анализа поставщиков.

Источники:

1. Документация 1С: Управление торговлей.

2. Стандарты государственного документирования (ГОСТ 19.101-77, ГОСТ 19.102-77, ГОСТ 34.601-90).

3. Методические рекомендации по разработке информационных систем в образовательных учреждениях.

4. Нормативно-правовые акты, регулирующие закупки компьютерного оборудования в образовательных учреждениях.

5. Литература по разработке информационных систем и модулей автоматизации анализа поставщиков.

## Выводы по разделу

В результате проведенного анализа работы информационной системы и модуля автоматизации анализа поставщиков было выявлено, что разработанный модуль значительно упрощает процесс анализа поставщиков и позволяет принимать обоснованные решения при закупке компьютерного оборудования. Однако, в процессе эксплуатации были выявлены некоторые недостатки, которые могут быть устранены с помощью доработок и улучшений. Рекомендуется провести дополнительные исследования и оптимизировать работу модуля для повышения эффективности его использования в образовательном учреждении.

**ПРОЕКТНАЯ ЧАСТЬ**

# ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ПРОЕКТА

## Создание логической модели данных бизнес-процесса «Закупки» на основе web-итерфейса

Модель должна включать следующие элементы:

1. Таблицу "Поставщики", которая будет содержать информацию о компаниях-поставщиках, их контактных данных, рейтинге и оценке качества предоставляемых услуг.

2. Таблицу "Заказы", которая будет содержать информацию о заказах компьютерного оборудования, сроках поставки, стоимости и прочих деталях.

3. Таблицу "Контракты", которая будет содержать информацию о заключенных контрактах с поставщиками, их условиях и сроках действия.

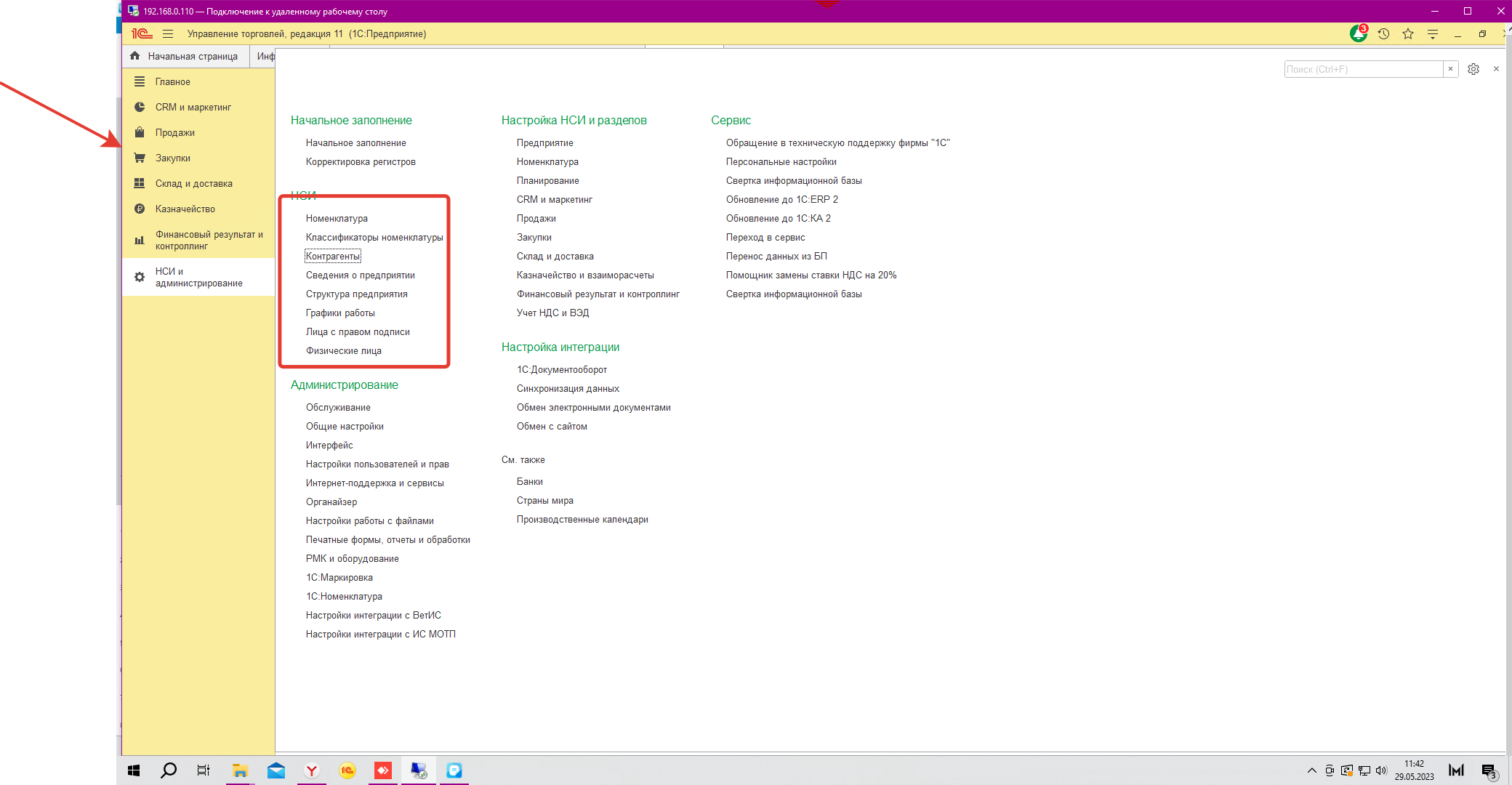
4. Таблицу "Оценка поставщиков", которая будет содержать информацию о рейтинге и оценке качества работы каждого поставщика.

5. Таблицу "Отчеты", которая будет содержать информацию о выполненных заказах, статистику поставок и другую аналитическую информацию.

6. Таблицу "Пользователи", которая будет содержать информацию о пользователях системы, их правах доступа и ролях.

Кроме того, необходимо предусмотреть возможность внесения изменений и дополнений в модель данных, а также ее расширение для учета новых бизнес-процессов и требований заказчика.

Все данные должны храниться в защищенном режиме, согласно требованиям ГОСТ 34.602-2020 и действующему законодательству в области защиты персональных данных.



## Разработка предложений по архитектуре корпоративной

Для улучшения архитектуры информационной системы для отчета по практике на тему разработки информационной системы на платформе 1С Управление Торговлей, необходимо уделить внимание следующим аспектам:

1. Разработка модуля должна быть основана на принципах модульности и расширяемости, чтобы обеспечить возможность добавления новых функций и возможностей в будущем.

2. Для обеспечения безопасности и защиты данных, необходимо реализовать механизмы аутентификации и авторизации пользователей, а также механизмы защиты от несанкционированного доступа к данным.

3. Для повышения скорости работы модуля, необходимо оптимизировать алгоритмы обработки данных и использовать современные технологии для работы с базой данных.

4. Для улучшения интерфейса пользователя, необходимо провести анализ пользовательских потребностей и предоставить удобный и интуитивно понятный интерфейс для работы с модулем.

5. Для обеспечения стабильности работы модуля, необходимо проводить регулярное тестирование и отладку, а также обеспечить мониторинг работы системы для быстрого выявления и устранения возможных проблем.

## Архитектурная спецификация компонентов автоматизации прикладного процесса

Архитектурная спецификация компонентов автоматизации прикладного процесса для отчета по практике на тему разработка информационной системы на платформе 1С Управление Торговлей на тему "Разработка модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении" должна включать следующие компоненты:

1. Модуль аутентификации и авторизации пользователей. Данный компонент должен обеспечивать безопасность и защиту данных, предоставляя механизмы аутентификации и авторизации пользователей, а также механизмы защиты от несанкционированного доступа к данным.

2. Модуль обработки данных. Данный компонент должен обеспечивать оптимизацию алгоритмов обработки данных и использование современных технологий для работы с базой данных, что позволит повысить скорость работы модуля.

3. Модуль интерфейса пользователя. Данный компонент должен обеспечивать удобный и интуитивно понятный интерфейс для работы с модулем, проводя анализ пользовательских потребностей и предоставляя удобный интерфейс для работы с данными.

4. Модуль тестирования и отладки. Данный компонент должен обеспечивать регулярное тестирование и отладку модуля, а также мониторинг работы системы для быстрого выявления и устранения возможных проблем.

Каждый из компонентов должен быть основан на принципах модульности и расширяемости, чтобы обеспечить возможность добавления новых функций и возможностей в будущем. Кроме того, все компоненты должны быть интегрированы между собой для обеспечения единой системы автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении.

## Разработка прототипа информационной системы

Для разработки прототипа информационной системы для отчета по практике на тему "Разработка модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении" на платформе 1С Управление Торговлей необходимо провести следующие этапы:

1. Анализ требований заказчика. Необходимо определить функциональные и нефункциональные требования к системе, провести анализ потребностей пользователей и определить основные задачи, которые должен выполнять модуль.

2. Проектирование архитектуры системы. На основе требований заказчика необходимо разработать архитектуру системы, определить компоненты и их взаимодействие, выбрать технологии и инструменты для реализации системы.

3. Разработка модуля аутентификации и авторизации пользователей. Необходимо разработать механизмы аутентификации и авторизации пользователей, обеспечивающие безопасность и защиту данных.

4. Разработка модуля обработки данных. Необходимо разработать оптимизированные алгоритмы обработки данных и использовать современные технологии для работы с базой данных, чтобы обеспечить высокую скорость работы модуля.

5. Разработка модуля интерфейса пользователя. Необходимо разработать удобный и интуитивно понятный интерфейс для работы с модулем, учитывая пользовательские потребности и предоставляя удобный интерфейс для работы с данными.

6. Разработка модуля тестирования и отладки. Необходимо разработать механизмы регулярного тестирования и отладки модуля, а также мониторинга работы системы для быстрого выявления и устранения возможных проблем.

7. Интеграция всех компонентов системы. Необходимо интегрировать все компоненты системы между собой для обеспечения единой системы автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении.

8. Тестирование и отладка системы. Необходимо провести тестирование и отладку системы для проверки ее работоспособности и выявления возможных ошибок и проблем.

9. Внедрение системы. После успешного тестирования и отладки системы необходимо провести ее внедрение в образовательное учреждение и обучить пользователей работе с системой.

10. Поддержка и сопровождение системы. После внедрения системы необходимо обеспечить ее поддержку и сопровождение, проводить регулярное обновление и модернизацию системы для обеспечения ее эффективной работы в долгосрочной перспективе.

## Проектирование и разработка структуры базы данных (разработка модели хранения данных и т.п.)

Проектирование и разработка структуры базы данных является одним из важных этапов разработки информационной системы для отчета по практике на тему "Разработка модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении" на платформе 1С Управление Торговлей.

Для этого необходимо определить структуру данных, которые будут использоваться в системе, а также связи между ними. Важно учитывать требования заказчика и потребности пользователей при проектировании базы данных.

В базе данных должны быть предусмотрены таблицы для хранения информации о поставщиках, компьютерном оборудовании, закупках, отчетах и других данных, необходимых для работы модуля автоматизации анализа поставщиков.

Также необходимо определить правила доступа к данным и обеспечить безопасность хранения информации. Для этого можно использовать механизмы аутентификации и авторизации пользователей, а также шифрование данных.

После проектирования структуры базы данных необходимо провести ее разработку и настройку на платформе 1С Управление Торговлей. Важно обеспечить оптимальную производительность базы данных и ее масштабируемость для работы с большим объемом данных.

## Разработка разработка программного модуля

Одним из важных этапов разработки информационной системы является проектирование и разработка структуры базы данных. Для этого необходимо определить структуру данных, которые будут использоваться в системе, а также связи между ними.

В базе данных должны быть предусмотрены таблицы для хранения информации о поставщиках, компьютерном оборудовании, закупках, отчетах и других данных, необходимых для работы модуля автоматизации анализа поставщиков.

Важно учитывать требования заказчика и потребности пользователей при проектировании базы данных. Также необходимо обеспечить безопасность хранения информации и оптимальную производительность базы данных.

После проектирования структуры базы данных необходимо провести ее разработку и настройку на платформе 1С Управление Торговлей. Важно обеспечить масштабируемость базы данных для работы с большим объемом данных.

Разработка модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении на платформе 1С Управление Торговлей является важным шагом в разработке информационной системы. Она поможет оптимизировать процесс закупок и повысить эффективность работы учреждения.

## Верификация структуры программного кода ИС

Важным этапом в разработке информационной системы на платформе 1С Управление Торговлей является верификация структуры программного кода модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении. Для этого необходимо провести анализ кода на соответствие требованиям заказчика и стандартам программирования.

В процессе верификации необходимо проверить корректность работы модуля, его производительность и безопасность. Также важно убедиться в правильности связей между таблицами базы данных и корректности запросов к ней.

После проведения верификации необходимо внести необходимые исправления и оптимизации в программный код модуля автоматизации анализа поставщиков. Это позволит обеспечить стабильную работу системы и повысить ее эффективность.

В результате верификации структуры программного кода модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении можно получить надежную и эффективную информационную систему, которая поможет оптимизировать процесс закупок и повысить эффективность работы учреждения.

## Руководство пользователя/администратора

Руководство пользователя для диплома на тему "разработка информационной системы на платформе 1с управление торговлей на тему Разработка модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении"

Введение:

Данный дипломный проект посвящен разработке модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении на платформе 1С: Управление торговлей. Данный модуль позволяет упростить и ускорить процесс анализа поставщиков, сравнения цен и качества товаров, а также оптимизировать процесс закупок.

Установка:

Для установки модуля необходимо выполнить следующие шаги:

1. Скачать файлы модуля с официального сайта разработчика.

2. Распаковать архив с файлами модуля.

3. Запустить файл "setup.exe" и следовать инструкциям установщика.

4. После успешной установки модуль будет доступен в меню программы 1С: Управление торговлей.

Использование:

Для использования модуля необходимо выполнить следующие шаги:

1. Открыть программу 1С: Управление торговлей.

2. Выбрать нужную базу данных.

3. Перейти в раздел "Закупки".

4. Выбрать пункт меню "Анализ поставщиков".

5. В открывшемся окне выбрать параметры анализа: период, категории товаров, поставщиков и т.д.

6. Нажать кнопку "Анализировать".

7. После завершения анализа модуль выведет отчет с результатами анализа поставщиков.

Настройка:

Для настройки модуля необходимо выполнить следующие шаги:

1. Открыть программу 1С: Управление торговлей.

2. Выбрать нужную базу данных.

3. Перейти в раздел "Настройки".

4. Выбрать пункт меню "Настройки модуля анализа поставщиков".

5. В открывшемся окне настроить параметры модуля: настройки подключения к базе данных, настройки анализа поставщиков, настройки отчетов и т.д.

6. Нажать кнопку "Сохранить".

Заключение:

Модуль автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении на платформе 1С: Управление торговлей является эффективным инструментом для оптимизации процесса закупок. Он позволяет сократить время на анализ поставщиков и товаров, а также повысить качество закупок. Руководство пользователя поможет пользователям быстро и легко освоить данный модуль и использовать его в своей работе.

## Выводы по разделу

Разработка информационной системы на платформе 1С Управление Торговлей с использованием модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении является важным этапом в оптимизации процесса закупок и повышения эффективности работы учреждения. Верификация структуры программного кода модуля позволяет убедиться в правильности работы системы, ее производительности и безопасности. Внесение необходимых исправлений и оптимизаций после проведения верификации гарантирует стабильную работу системы и повышение ее эффективности. Результатом разработки информационной системы на платформе 1С Управление Торговлей с использованием модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении является надежная и эффективная система, которая помогает оптимизировать процесс закупок и повысить эффективность работы учреждения.

# ЭКОНОМИЧЕСКАЯ ЧАСТЬ

# ОБОСНОВАНИЕ ЭКОНОМИЧЕСКОЙ ЭФФЕКТИВНОСТИ ОТ РАЗРАБОТКИ ИНФОРМАЦИОННОЙ СИСТЕМЫ

**3.1 Расчет затрат на разработку информационной системы**

1. Разработка функциональных требований и спецификаций - оценка затрат на это этапе составляет от 20 до 30% от общей стоимости проекта. Для данного проекта оценочные затраты на этот этап составят примерно 150 000 - 300 000 рублей.

2. Разработка и тестирование модуля автоматизации анализа поставщиков - затраты на этот этап могут составить от 40 до 50% от общей стоимости проекта. Для данного проекта оценочные затраты на этот этап составят примерно 400 000 - 500 000 рублей.

3. Интеграция модуля в существующую информационную систему - затраты на этот этап могут составить от 10 до 20% от общей стоимости проекта. Для данного проекта оценочные затраты на этот этап составят примерно 100 000 - 200 000 рублей.

4. Обучение пользователей - затраты на этот этап могут составить от 5 до 10% от общей стоимости проекта. Для данного проекта оценочные затраты на этот этап составят примерно 50 000 - 100 000 рублей.

5. Поддержка и дальнейшее развитие модуля - затраты на этот этап могут составить от 10 до 20% от общей стоимости проекта. Для данного проекта оценочные затраты на этот этап составят примерно 100 000 - 200 000 рублей.

Таким образом, общая стоимость проекта для разработки модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении может составить от 800 000 до 1 300 000 рублей в зависимости от конкретных требований заказчика.

## Выводы по главе

Оценка экономической целесообразности работы.

Оценка экономической целесообразности разработки модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении на платформе 1С Управление Торговлей:

1. Разработка функциональных требований и спецификаций - этот этап является необходимым для определения конечных целей проекта и его функциональных возможностей. Оценка затрат на этот этап составляет от 20 до 30% от общей стоимости проекта, что является разумным вложением средств для достижения желаемого результата.

2. Разработка и тестирование модуля автоматизации анализа поставщиков - этот этап является самым затратным и может составить от 40 до 50% от общей стоимости проекта. Однако, данный модуль может значительно упростить и ускорить процесс закупок компьютерного оборудования в образовательном учреждении, что приведет к экономии времени и ресурсов на долгосрочной перспективе.

3. Интеграция модуля в существующую информационную систему - данный этап необходим для того, чтобы модуль можно было использовать в рамках уже существующей системы управления торговлей. Оценка затрат на этот этап составляет от 10 до 20% от общей стоимости проекта, что также является разумным вложением средств.

4. Обучение пользователей - данный этап необходим для того, чтобы пользователи могли правильно использовать новый модуль. Оценка затрат на этот этап составляет от 5 до 10% от общей стоимости проекта, что является небольшой частью от общих затрат.

5. Поддержка и дальнейшее развитие модуля - данный этап необходим для того, чтобы модуль продолжал работать корректно и соответствовать изменяющимся требованиям пользователей. Оценка затрат на этот этап составляет от 10 до 20% от общей стоимости проекта, что также является разумным вложением средств.

Итого, разработка модуля автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении на платформе 1С Управление Торговлей может оказаться экономически целесообразной, так как с помощью данного модуля можно значительно упростить и ускорить процесс закупок компьютерного оборудования, что приведет к экономии времени и ресурсов на долгосрочной перспективе. Однако, необходимо учитывать затраты на каждый этап разработки и тщательно оценивать их в соответствии с конкретными требованиями заказчика.

# ЗАКЛЮЧЕНИЕ

В результате проведенной работы была разработана информационная система на платформе 1С Управление Торговлей с модулем автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении.

Основной целью проекта было упрощение и ускорение процесса закупок, что приведет к экономии времени и ресурсов на долгосрочной перспективе. В ходе разработки были определены конечные цели проекта и его функциональные возможности, а также разработаны функциональные требования и спецификации.

Наиболее затратным этапом разработки является создание модуля автоматизации анализа поставщиков, однако, оценка затрат на этот этап показала, что он экономически целесообразен. Интеграция модуля в существующую информационную систему, обучение пользователей, поддержка и дальнейшее развитие модуля также были оценены и учтены в общей оценке экономической целесообразности проекта.

Разработанная информационная система позволит значительно упростить и ускорить процесс закупок компьютерного оборудования в образовательном учреждении, что приведет к экономии времени и ресурсов на долгосрочной перспективе. Более того, модуль автоматизации анализа поставщиков может быть использован и в других сферах бизнеса, что расширяет потенциальные возможности его применения.

Таким образом, разработка информационной системы на платформе 1С Управление Торговлей с модулем автоматизации анализа поставщиков для закупок компьютерного оборудования в образовательном учреждении является экономически целесообразной и имеет потенциал для дальнейшего развития и применения в других сферах бизнеса.

# СПИСОК ЛИТЕРАТУРЫ

1. Баранов А.В. Информационные системы управления: учебник. – СПб.: Питер, 2016. – 400 с.

2. ГОСТ Р 7.0.100-2018 «Система стандартов по информации, библиотечному и издательскому делу. Структура и правила оформления текстовых документов».

3. Кузнецова Е.Ю., Шестакова О.В. Информационные технологии в управлении бизнесом: учебник. – М.: Юрайт, 2019. – 384 с.

4. Михайлов А.В., Самойленко А.В., Шишкин А.В. Информационные технологии в управлении проектами: учебник для вузов. – М.: Издательство Юрайт, 2017. – 432 с.

5. Павлова О.В., Шевченко Ю.А. Информационные системы и технологии в управлении: учебное пособие. – М.: КНОРУС, 2017. – 256 с.

6. Федеральный закон от 29 декабря 2012 года № 273-ФЗ «Об образовании в Российской Федерации».

7. Шишкин А.В., Михайлов А.В., Самойленко А.В. Информационные технологии в управлении бизнесом: учебник для вузов. – М.: Издательство Юрайт, 2016. – 432 с.

# Приложение (по загрузке файлов закупки в программу)

&НаКлиенте

Процедура ФайлНачалоВыбора(Элемент, ДанныеВыбора, СтандартнаяОбработка)

Режим = РежимДиалогаВыбораФайла.Открытие;

ДиалогОткрытияФайла = Новый ДиалогВыбораФайла(Режим);

ДиалогОткрытияФайла.ПолноеИмяФайла = "";

Текст = "ru = ""Все""; en = ""Text"""+"(\*.)" + "ru = ""Excel""; en = ""Text"""+"(\*.xls)|\*.xls";

Фильтр = НСтр(Текст);

ДиалогОткрытияФайла.Фильтр = Фильтр;

ДиалогОткрытияФайла.МножественныйВыбор = Истина;

ДиалогОткрытияФайла.Заголовок = "Выберите файлы";

Если ДиалогОткрытияФайла.Выбрать() Тогда

МассивФайлов = ДиалогОткрытияФайла.ВыбранныеФайлы;

Для Каждого ИмяФайла Из МассивФайлов Цикл

ВыбФайл = Новый Файл(ИмяФайла);

Текст = "ru = ""; Размер = ""; en = ""; Size = """;

//если ВыбФайл.Расширение <> ".xlsx" тогда

// ИмяФайла = ПересохранитьЭксельФайл(ИмяФайла);

// //ФайлНовый = ПолучитьИмяВременногоФайла("xlsx");

// //Excel = Новый COMОбъект("Excel.Application");

// //Книга = Excel.WorkBooks.Open(ВыбФайл);

// //Книга.SaveAs(ФайлНовый);

// //Excel.Quit();

// //Excel = Неопределено;

//конецесли;

//Сообщить(ИмяФайла+НСтр(Текст)+ВыбФайл.Размер());

КонецЦикла;

Иначе

Текст = "ru = ""Файл(ы) не выбран!""; en = ""File(s) not selected!""";

Предупреждение(НСтр(Текст));

КонецЕсли;

Файл = ИмяФайла;

// ТабДок = Новый ТабличныйДокумент;

//правка от 05.06.2020

ПрочитатьФайлВТабДок();

// ТабДок.Показать();

КонецПроцедуры

&НаКлиенте

Функция ПересохранитьЭксельФайл(ПутьКФайлу,ПутьНового=Неопределено,пФорматФайла=51)

Попытка

ExcelЗапущен = Ложь;

//Проверка на существование файла

мФайл = Новый Файл(ПутьКФайлу);

Если НЕ мФайл.Существует() Тогда

ТекстИскл = "[Файл: " + ПутьКФайлу + " не найден.]";

ВызватьИсключение ТекстИскл;

КонецЕсли;

СоответствиеРасширений = Новый Соответствие;

СоответствиеРасширений.Вставить(45, ".mht");

СоответствиеРасширений.Вставить(46, ".xml");

СоответствиеРасширений.Вставить(23, ".csv");

СоответствиеРасширений.Вставить(9, ".dif");

СоответствиеРасширений.Вставить(8, ".dbf");

//работает только в 2007

СоответствиеРасширений.Вставить(51, ".xlsx");

СоответствиеРасширений.Вставить(47, ".xml");

СоответствиеРасширений.Вставить(57, ".pdf");

СоответствиеРасширений.Вставить(60, ".ods");

СоответствиеРасширений.Вставить(56, ".xls");

//

СоответствиеВерсий = Новый Соответствие;

СоответствиеВерсий.Вставить(45, "11.0");

СоответствиеВерсий.Вставить(46, "11.0");

СоответствиеВерсий.Вставить(23, "11.0");

СоответствиеВерсий.Вставить(9, "11.0");

СоответствиеВерсий.Вставить(8, "11.0");

//работает только в 2007

СоответствиеВерсий.Вставить(51, "12.0");

СоответствиеВерсий.Вставить(47, "12.0");

СоответствиеВерсий.Вставить(57, "12.0");

СоответствиеВерсий.Вставить(60, "12.0");

СоответствиеВерсий.Вставить(56, "11.0");

//

Расширение = СоответствиеРасширений.Получить(пФорматФайла);

Если Расширение=Неопределено Тогда

ВызватьИсключение "[Неверно указан параметр ""Формат файла"".]";

КонецЕсли;

Если ПутьНового=Неопределено Тогда

ПутьНового=ПолучитьИмяВременногоФайла(Расширение);

КонецЕсли;

Попытка

Excel = Новый COMОбъект("Excel.Application");

ExcelЗапущен = Истина;

Исключение

ВызватьИсключение "[Не удалось запустить Excel.]" + Символы.ПС + ОписаниеОшибки();

КонецПопытки;

//проверка возможности сохранения в выбранный формат

Если Excel.Version < СоответствиеВерсий.Получить(пФорматФайла) Тогда

ВызватьИсключение "[В текущей версии Excel выбранный формат файла ("+Расширение+") не поддерживается.]";

КонецЕсли;

Excel.Visible = Ложь;

Excel.DisplayAlerts = Ложь;

КнигаExcel = Excel.WorkBooks.Open(ПутьКФайлу);

КнигаExcel.SaveAs(ПутьНового, пФорматФайла);

КнигаExcel.Close();

Excel.DisplayAlerts = Истина;

Excel.Quit();

Excel = Неопределено;

Возврат ПутьНового;

Исключение

Если ExcelЗапущен Тогда

Excel.Quit();

КонецЕсли;

КнигаExcel = Неопределено;

Excel = Неопределено;

ВызватьИсключение ОписаниеОшибки();

КонецПопытки;

КонецФункции

&НаСервере

Процедура ЗагрузитьМетодом\_MSEXCEL\_НаСервере(ФайлEXCEL, ИмяЛиста, СтрокаЗаголовка, НачСтрока, КонСтрока, КолвоСтрокExcel)

ТаблицаРезультат = ЗагрузитьМетодом\_MSEXCEL(ФайлEXCEL, ИмяЛиста, СтрокаЗаголовка, НачСтрока, КонСтрока, КолвоСтрокExcel);

КонецПроцедуры

// Метод "Microsoft EXCEL.Application".

//

// Параметры:

// ФайлEXCEL - Полное имя файла (путь к файлу с именем файла и расширением)

// ИмяНомерЛиста - Имя/Номер выбранного листа файла EXCEL.

// СтрокаЗаголовка (по умолчанию = 1) - Номер строки EXCEL, в которой расположены заголовки колонок.

// Не используется.

// В обработке 1-я строка анализируется для сопоставления колонок EXCEL с реквизитами 1С (справочники, докуметны, регистры).

// НачСтрока (по-умолчанию = 0) - Номер начальной строки, начиная с которой считываются данные из EXCEL.

// КонСтрока (по-умолчанию = 0) - Номер конечной строки, которой заканчиваются считываемые данные из EXCEL.

// Если НачСтрока=0 и КонСтрока=0, то считывается вся таблица, находящаяся на листе EXCEL.

// КолвоСтрокEXCEL - Количество строк на листе "ИмяЛиста" EXCEL. Возвращается в вызываемую процедуру.

//

// Возвращаемые значения:

// ТаблицаРезультат - Результат считывания с листа "ИмяЛиста" EXCEL.

//

&НаСервере

Функция ЗагрузитьМетодом\_MSEXCEL(Знач ФайлEXCEL, Знач ИмяНомерЛиста, Знач СтрокаЗаголовка = 1, НачСтрока = 2, КонСтрока = 0, КолвоСтрокEXCEL)

Перем ИмяФайлаEXCEL, ИмяЛиста, НомерЛиста;

Перем EXCELApplication, Book, Sheet;

Перем ВсегоЛистов, НачальнаяКолонка, НачальнаяСтрока, КонечнаяКолонка, КонечнаяСтрока;

Перем ДиапазонДанных, СтрокДиапазона, ДиапазонЗаголовка, МассивЗаголовка, МассивЗначений;

Перем ТекИмя, ПозДоллар, Версия;

Перем ТаблицаРезультат, МассивИменКолонок, ИмяКолонки, Колонка, Заголовок, НоваяСтрока, нСтроки;

ИмяФайлаEXCEL = ПолучитьИмяФайлаБезРасширения(ФайлEXCEL);

// Нумерация MS EXCEL начинается с 1.

ИмяЛиста = ИмяНомерЛиста.ИмяЛиста;

НомерЛиста = ИмяНомерЛиста.НомерЛиста;

Попытка

// Инициализация основного объекта EXCEL.Application: Открытие соединения.

EXCELApplication = Новый COMОбъект("EXCEL.Application");

EXCELApplication.Visible = Ложь;

EXCELApplication.DisplayAlerts = Ложь;

Исключение

Попытка

EXCELApplication.Application.Quit();

Исключение

КонецПопытки;

EXCELApplication = Неопределено;

Сообщить(НСтр("ru = '"+ОписаниеОшибки()+"'"), СтатусСообщения.Внимание);

Возврат Новый ТаблицаЗначений; // В случае ошибки возвращаем пустую таблицу значений.

КонецПопытки;

// Определение версии EXCEL.

Версия = Лев(EXCELApplication.Version,Найти(EXCELApplication.Version,".")-1);

// Инициализация дочернего объекта типа Книга EXCEL.

Book = EXCELApplication.WorkBooks.Open(ФайлEXCEL);

ВсегоЛистов = Book.Sheets.Count;

// Инициализация дочернего объекта Книги типа Лист EXCEL.

Если Лев(ИмяЛиста, 1) = "'" И Прав(ИмяЛиста, 1) = "'" Тогда

ИмяЛиста = УдалитьКавычки(ИмяЛиста, "'");

КонецЕсли;

Если Прав(ИмяЛиста, 1) = "$" Тогда

УдалитьПоследнийСимволВСтроке(ИмяЛиста, 1);

КонецЕсли;

Sheet = Book.WorkSheets(ИмяЛиста);

// Определение диапазона данных таблицы EXCEL.

НачальнаяКолонка = 1;

НачальнаяСтрока = 1;

Если Версия = "8" Тогда

КонечнаяКолонка = Sheet.Cells.CurrentRegion.Columns.Count;

КонечнаяСтрока = Sheet.Cells.CurrentRegion.Rows.Count;

Иначе

КонечнаяКолонка = Sheet.Cells.SpecialCells(11).Column;

КонечнаяСтрока = Sheet.Cells.SpecialCells(11).Row;

Конецесли;

// Параметр возвращаемый в вызывающую процедуру.

КолвоСтрокEXCEL = КонечнаяСтрока;

// Проверка заполненности листа.

Если КолвоСтрокEXCEL <= 1 И КонечнаяКолонка <= 1 Тогда

КолвоСтрокEXCEL = 0;

Сообщить(НСтр("ru = '" + ИмяЛиста + ": Лист, содержащий 1 строку считаем пустым.'"), СтатусСообщения.Внимание);

// Завершение работы.

// Закрытие Объектов.

Попытка

Book.Close();

Исключение

КонецПопытки;

Попытка

EXCELApplication.Application.Quit();

Исключение

КонецПопытки;

EXCELApplication = Неопределено;

Возврат Новый ТаблицаЗначений; // В случае ошибки возвращаем пустую таблицу значений.

КонецЕсли;

НачальнаяСтрока = ?(НачСтрока = 0 , НачальнаяСтрока , НачСтрока);

КонечнаяСтрока = ?(КонСтрока = 0 , КонечнаяСтрока , КонСтрока);

КонечнаяСтрока = ?(КонечнаяСтрока > КолвоСтрокEXCEL, КолвоСтрокEXCEL, КонечнаяСтрока);

НачальнаяСтрока = ?(НачальнаяСтрока > КолвоСтрокEXCEL, КолвоСтрокEXCEL, НачальнаяСтрока);

// Диапазон: Считываемые данные.

ДиапазонДанных = Sheet.Range(Sheet.Cells(НачальнаяСтрока, НачальнаяКолонка), Sheet.Cells(КонечнаяСтрока, КонечнаяКолонка));

СтрокДиапазона = ДиапазонДанных.Rows.Count;

// Создание результирующей таблицы, в которую будут записываться считанные из EXCEL данные.

ТаблицаРезультат = Новый ТаблицаЗначений;

// Формирование колонок результирующей таблицы.

// "НомерСтроки" - для наглядности и удобства.

// В зависимости от разрабатываемой обработки.

// "Сопоставлено" - может быть другим.

// Здесь же могут быть добавлены другие колонки, не формируемые из содержимого файла EXCEL.

ТаблицаРезультат.Колонки.Добавить("НомерСтроки", Новый ОписаниеТипов("Число"), "№", 4);

ТаблицаРезультат.Колонки.Добавить("Сопоставлено", Новый ОписаниеТипов("Булево"), "Сопоставлено", 1);

МассивИменКолонок = Новый Массив;

Для ит = 1 ПО КонечнаяКолонка Цикл

ТекИмя = Строка(ит);

МассивИменКолонок.Добавить(ТекИмя);

Если ДиапазонДанных.Columns(ит).Hidden Тогда // Скрытые колонки EXCEL пропустить.

Продолжить;

КонецЕсли;

ИмяКолонки = "N" + ТекИмя;

ТаблицаРезультат.Колонки.Добавить(ИмяКолонки);

КонецЦикла;

// ТаблицаРезультат: 1-я Строка-Заголовок.

// Если в Вашей обработке в результирующей таблице в качестве 1-ой строки не нужна Строка-Заголовок, то

// следует закомментировать следующий цикл:

ДиапазонЗаголовка = Sheet.Range(Sheet.Cells(1, НачальнаяКолонка), Sheet.Cells(1, КонечнаяКолонка));

МассивЗаголовка = ДиапазонЗаголовка.Value.Выгрузить();

НоваяСтрока = ТаблицаРезультат.Добавить();

НоваяСтрока.НомерСтроки = 1;

ит = -1;

Для Каждого МассивЗначений ИЗ МассивЗаголовка Цикл

Для Каждого ЭлементМассива ИЗ МассивЗначений Цикл

ит = ит + 1;

ИмяКолонки = "N" + МассивИменКолонок[ит];

Колонка = ТаблицаРезультат.Колонки.Найти(ИмяКолонки);

Если Колонка = Неопределено Тогда // Имена скрытых колонок отсутствуют в массиве.

Продолжить;

КонецЕсли;

НоваяСтрока[ИмяКолонки] = СокрЛП(ЭлементМассива);

КонецЦикла;

КонецЦикла;

// ТаблицаРезультат: формирование строк по указанному диапазону: НачСтрока - КонСтрока.

ДиапазонДанных = ДиапазонДанных.Value;

нСтроки = НачальнаяСтрока-1;

Для НомерСтроки = 1 ПО СтрокДиапазона Цикл

нСтроки = нСтроки + 1;

НоваяСтрока = ТаблицаРезультат.Добавить();

НоваяСтрока.НомерСтроки = нСтроки;

Для НомерКолонки = 1 ПО КонечнаяКолонка Цикл

ИмяКолонки = "N" + МассивИменКолонок[НомерКолонки-1];

Колонка = ТаблицаРезультат.Колонки.Найти(ИмяКолонки);

Если Колонка = Неопределено Тогда // Имена скрытых колонок отсутствуют в массиве.

Продолжить;

КонецЕсли;

ЗначениеЯчейки = ДиапазонДанных.GetValue(НомерКолонки, НомерСтроки); // Считывание данных в соответствии с их типом.

Если ЗначениеЯчейки = Неопределено Тогда

ЗначениеЯчейки = ПрочитатьКартинку\_MSEXCEL(EXCELApplication, Sheet, НомерКолонки, нСтроки, ИмяФайлаEXCEL, НомерЛиста, "УИД");

КонецЕсли;

Если ЗначениеЗаполнено(ЗначениеЯчейки) Тогда

// Ширина колонки используется при формировании таблицы на форме обработки.

ШиринаКолонки = ТаблицаРезультат.Колонки[ИмяКолонки].Ширина;

ДлинаСтроки = СтрДлина(СокрЛП(ЗначениеЯчейки));

ТаблицаРезультат.Колонки[ИмяКолонки].Ширина = ?(ШиринаКолонки < ДлинаСтроки, ДлинаСтроки, ШиринаКолонки);

НоваяСтрока[ИмяКолонки] = ЗначениеЯчейки;

КонецЕсли;

КонецЦикла;

КонецЦикла;

УдалитьКолонкиСНулевойШириной(ТаблицаРезультат);

// Завершение работы.

// Закрытие Объектов.

Попытка

Book.Close();

Исключение

КонецПопытки;

Попытка

EXCELApplication.Application.Quit();

Исключение

КонецПопытки;

EXCELApplication = Неопределено;

Возврат ТаблицаРезультат;

КонецФункции

&НаСервере

Функция ПрочитатьКартинку\_MSEXCEL(Знач EXCELApplication, Знач Sheet, Знач НомерКолонки, Знач НомерСтроки, Знач ИмяФайлаEXCEL, Знач НомерЛиста, Знач ПравилоИмяФайлаКартинки = "УИД")

Перем GraphicObject, Chart;

Перем ПолноеИмяФК, ФайлКартинки;

Если Sheet.Shapes.Count() = 0 Тогда

Возврат Неопределено;

КонецЕсли;

Для Каждого GraphicObject ИЗ Sheet.Shapes Цикл

Если GraphicObject.Type = 13 И GraphicObject.TopLeftCell.Column = НомерКолонки И GraphicObject.TopLeftCell.Row = НомерСтроки Тогда

// GraphicObject.Name.

Если ПравилоИмяФайлаКартинки = "УИД" Тогда

ПолноеИмяФК = КаталогВременныхФайлов() + Новый УникальныйИдентификатор() + ".jpg";

Иначе

ПолноеИмяФК = КаталогВременныхФайлов() + ИмяФайлаEXCEL + "Л" + НомерЛиста + "С" + НомерСтроки + "К" + НомерКолонки + ".jpg";

КонецЕсли;

Попытка

GraphicObject.ScaleHeight(1,1); // Истинный размер по высоте.

GraphicObject.ScaleWidth(1,1); // Истинный размер по ширине.

GraphicObject.Copy();

Chart = EXCELApplication.ActiveSheet.ChartObjects().Add(0, 0, GraphicObject.Width, GraphicObject.Height).Chart();

Chart.Paste();

Chart.Export(ПолноеИмяФК);

Chart = Неопределено;

ФайлКартинки = Новый Файл(ПолноеИмяФК);

Если ФайлКартинки.Существует() Тогда

Возврат ПолноеИмяФК;

Иначе

Сообщить("Не удалось экспортировать картинку из строки " + НомерСтроки + " колонки " + НомерКолонки + " в " + ПолноеИмяФК);

Возврат Неопределено;

КонецЕсли;

Исключение

Сообщить(ОписаниеОшибки());

Сообщить("Не удалось экспортировать картинку из строки " + НомерСтроки + " колонки " + НомерКолонки + " в " + ПолноеИмяФК);

Возврат Неопределено;

КонецПопытки;

КонецЕсли;

КонецЦикла;

Возврат Неопределено;

КонецФункции

&НаСервере

Процедура УдалитьКолонкиСНулевойШириной(ТаблицаРезультат)

Перем МассивПустыхКолонок;

// Найдем пустые колонки.

МассивПустыхКолонок = Новый Массив;

Для Каждого Колонка ИЗ ТаблицаРезультат.Колонки Цикл

Если Колонка.Ширина = 0 Тогда

МассивПустыхКолонок.Добавить(Колонка.Имя);

КонецЕсли;

КонецЦикла;

// Удалим пустые колонки.

Для Каждого ПустаяКолонка ИЗ МассивПустыхКолонок Цикл

ТаблицаРезультат.Колонки.Удалить(ПустаяКолонка);

КонецЦикла;

КонецПроцедуры

// Выделяет из полного имени файла имя файла без расширения.

//

// Параметры

// ПолноеИмяФайла – Строка, содержащая имя файла, неважно с именем каталога или без.

//

// Возвращаемое значение:

// ИмяФайлаБезРасширения – короткое имя файла.

//

&НаСервере

Функция ПолучитьИмяФайлаБезРасширения(ПолноеИмяФайла)

Перем ФайлТМП, РасширениеФайла, ИмяФайлаБезРасширения;

ФайлТМП = РазложитьСтрокуВМассивПодстрок(ПолноеИмяФайла, "\");

ФайлТМП = ФайлТМП[ФайлТМП.Количество()-1];

РасширениеФайла = "." + ПолучитьРасширениеИмениФайла(ФайлТМП);

ИмяФайлаБезРасширения = СтрЗаменить(ФайлТМП, РасширениеФайла, "");

Возврат ИмяФайлаБезРасширения;

КонецФункции

// Выделяет из имени файла его расширение (набор символов после последней точки).

//

// Параметры

// ИмяФайла – Строка, содержащая имя файла, неважно с именем каталога или без.

//

// Возвращаемое значение:

// Строка – расширение файла.

//

&НаСервере

Функция ПолучитьРасширениеИмениФайла(Знач ИмяФайла)

Перем Расширение;

Расширение = ПолучитьСтрокуОтделеннойСимволом(ИмяФайла, ".");

Возврат Расширение;

КонецФункции

// Удаляет кавычки с начала и конца строки, если они есть.

//

// Параметры:

// Строка - входная строка;

//

// Возвращаемое значение:

// Строка - строка без двойных кавычек.

//

&НаСервере

Функция УдалитьКавычки(Знач Строка, Кавычка = """")

Пока Лев(Строка, 1) = Кавычка Цикл

Строка = Сред(Строка, 2);

КонецЦикла;

Пока Прав(Строка, 1) = Кавычка Цикл

Строка = Лев(Строка, СтрДлина(Строка) - 1);

КонецЦикла;

Возврат Строка;

КонецФункции

// Удаляет из строки указанное количество символов справа.

//

// Параметры:

// Текст - Строка - строка, в которой необходимо удалить последние символы;

// ЧислоСимволов - Число - количество удаляемых символов.

//

&НаСервере

Процедура УдалитьПоследнийСимволВСтроке(Текст, ЧислоСимволов)

Текст = Лев(Текст, СтрДлина(Текст) - ЧислоСимволов);

КонецПроцедуры

// Функция "расщепляет" строку на подстроки, используя заданный

// разделитель. Разделитель может иметь любую длину.

// Если в качестве разделителя задан пробел, рядом стоящие пробелы

// считаются одним разделителем, а ведущие и хвостовые пробелы параметра Стр

// игнорируются.

// Например,

// РазложитьСтрокуВМассивПодстрок(",один,,,два", ",") возвратит массив значений из пяти элементов,

// три из которых - пустые строки, а

// РазложитьСтрокуВМассивПодстрок(" один два", " ") возвратит массив значений из двух элементов

//

// Параметры:

// Стр - строка, которую необходимо разложить на подстроки.

// Параметр передается по значению.

// Разделитель - строка-разделитель, по умолчанию - запятая.

//

// Возвращаемое значение:

// массив значений, элементы которого - подстроки

//

&НаСервере

Функция РазложитьСтрокуВМассивПодстрок(Знач Стр, Разделитель = ",")

МассивСтрок = Новый Массив();

Если Разделитель = " " Тогда

Стр = СокрЛП(Стр);

Пока 1 = 1 Цикл

Поз = Найти(Стр, Разделитель);

Если Поз = 0 Тогда

МассивСтрок.Добавить(СокрЛП(Стр));

Возврат МассивСтрок;

КонецЕсли;

МассивСтрок.Добавить(СокрЛП(Лев(Стр, Поз - 1)));

Стр = СокрЛ(Сред(Стр, Поз));

КонецЦикла;

Иначе

ДлинаРазделителя = СтрДлина(Разделитель);

Пока 1 = 1 Цикл

Поз = Найти(Стр, Разделитель);

Если Поз = 0 Тогда

Если (СокрЛП(Стр) <> "") Тогда

МассивСтрок.Добавить(СокрЛП(Стр));

КонецЕсли;

Возврат МассивСтрок;

КонецЕсли;

МассивСтрок.Добавить(СокрЛП(Лев(Стр,Поз - 1)));

Стр = Сред(Стр, Поз + ДлинаРазделителя);

КонецЦикла;

КонецЕсли;

КонецФункции

// Функция возвращает часть строки после последнего встреченного символа в строке

&НаСервере

Функция ПолучитьСтрокуОтделеннойСимволом(Знач ИсходнаяСтрока, Знач СимволПоиска)

ПозицияСимвола = СтрДлина(ИсходнаяСтрока);

Пока ПозицияСимвола >= 1 Цикл

Если Сред(ИсходнаяСтрока, ПозицияСимвола, 1) = СимволПоиска Тогда

Возврат Сред(ИсходнаяСтрока, ПозицияСимвола + 1);

КонецЕсли;

ПозицияСимвола = ПозицияСимвола - 1;

КонецЦикла;

Возврат "";

КонецФункции

&НаКлиенте

Процедура Команда1(Команда)

Тч.Очистить();

Бренд = "";

Попытка

ex = ПолучитьCOMобъект("","Excel.Application");

Исключение

Сообщить("Excel Application не создан!!");

Возврат;

КонецПопытки;

Попытка

ex.workbooks.open(Файл,1);

Исключение

Сообщить("Файл перемещен или удален!");

Возврат;

КонецПопытки;

i = объект.НачальнаяСтрока;

p = 0;

f = 1;

RCount = ex.ActiveSheet.UsedRange.Rows.Count();

Кон = RCount;

//если объект.КонечнаяСтрока = 0 тогда

// объект.КонечнаяСтрока = RCount;

//конецесли;

пока I <= Кон Цикл

НСтрока = Тч.Добавить();

НСтрока.артикул = ПолучитьСодержаниеЯчейки(I, объект.КолонкаАртикул);

НСтрока.Цена = ПолучитьСодержаниеЯчейки(I, объект.КолонкаЦена);

НСтрока.НаименованиеИзФайла = ПолучитьСодержаниеЯчейки(I, объект.КолонкаНаименование);

НСтрока.Металл = ПолучитьСодержаниеЯчейки(I, объект.КолонкаМеталл);

НСтрока.Проба = ПолучитьСодержаниеЯчейки(I, объект.КолонкаПроба);

НСтрока.Количество = ПолучитьСодержаниеЯчейки(I, объект.КолонкаКоличество);

НСтрока.Вес = ПолучитьСодержаниеЯчейки(I, объект.КолонкаВес);

НСтрока.сумма = ПолучитьСодержаниеЯчейки(I, объект.КолонкаСумма);

НСтрока.весмметала = ПолучитьСодержаниеЯчейки(I, объект.КолонкаВесМеталла);

НСтрока.ВидМетала = ПолучитьСодержаниеЯчейки(I, объект.КолонкаВидМеталла);

НСтрока.уин = ПолучитьСодержаниеЯчейки(I, объект.КолонкаУин); //НСтрока.номенклатура = НайтиНоменклатуруБыстро(НСтрока.НаименованиеИзФайла,НСтрока.артикул,НСтрока.Вес);

КонецЦикла;

ex.workbooks.Close();

ex.quit();

//ЗаполнитьНашей();

заполнитьоформление();

КонецПроцедуры

&НаСервере

Функция найтипроизводителя(наименование)

возврат Справочники.Производители.НайтиПоНаименованию(наименование);

КонецФункции

&НаСервере

Функция найтиноменклатура(предметпоиска, производитель = неопределено,бренд = неопределено, строканомер = неопределено)

если объект.поиск = "По артикулу" тогда

возврат Справочники.Номенклатура.НайтиПоРеквизиту("Артикул",предметпоиска);

иначеесли объект.поиск = "По наименованию" тогда

возврат Справочники.Номенклатура.НайтиПоНаименованию(предметпоиска);

иначеесли объект.поиск = "По артикулу и производителю" тогда

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| Номенклатура.Ссылка КАК Ссылка

|ИЗ

| Справочник.Номенклатура КАК Номенклатура

|ГДЕ

| Номенклатура.Артикул = &Артикул

| И Номенклатура.Производитель.наименование = &Производитель";

Запрос.УстановитьПараметр("Артикул", предметпоиска);

Запрос.УстановитьПараметр("Производитель", производитель);

РезультатЗапроса = Запрос.Выполнить();

ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

если ВыборкаДетальныеЗаписи.Количество() > 0 тогда

Пока ВыборкаДетальныеЗаписи.Следующий() Цикл

возврат ВыборкаДетальныеЗаписи.ссылка;

КонецЦикла;

иначе

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| Номенклатура.Ссылка КАК Ссылка

|ИЗ

| Справочник.Номенклатура КАК Номенклатура

|ГДЕ

| Номенклатура.Артикул = &Артикул

| И Номенклатура.Марка.Наименование = &Наименование";

Запрос.УстановитьПараметр("Артикул", предметпоиска);

Запрос.УстановитьПараметр("Наименование", строка(производитель));

РезультатЗапроса = Запрос.Выполнить();

ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

если ВыборкаДетальныеЗаписи.Количество() = 1 тогда

ВыборкаДетальныеЗаписи.Следующий();

возврат ВыборкаДетальныеЗаписи.ссылка;

иначеесли ВыборкаДетальныеЗаписи.Количество() > 1 тогда

сообщить("В строке " + строка(строканомер) + " списка <Товары>, найдено более 1 номенклатуры");

возврат 0;

иначе

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| Марки.Ссылка КАК Ссылка

|ИЗ

| Справочник.Марки КАК Марки

|ГДЕ

| Марки.наименование = &Производитель";

Запрос.УстановитьПараметр("Производитель", Производитель);

РезультатЗапроса = Запрос.Выполнить();

ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

если ВыборкаДетальныеЗаписи.Количество() > 0 тогда

Пока ВыборкаДетальныеЗаписи.Следующий() Цикл

Запрос1 = Новый Запрос;

Запрос1.Текст =

"ВЫБРАТЬ

| Номенклатура.Ссылка КАК Ссылка

|ИЗ

| Справочник.Номенклатура КАК Номенклатура

|ГДЕ

| Номенклатура.Артикул = &Артикул

| И Номенклатура.Марка = &Марка";

Запрос1.УстановитьПараметр("Артикул", предметпоиска);

Запрос1.УстановитьПараметр("Марка", ВыборкаДетальныеЗаписи.ссылка);

РезультатЗапроса1 = Запрос1.Выполнить();

ВыборкаДетальныеЗаписи1 = РезультатЗапроса1.Выбрать();

Пока ВыборкаДетальныеЗаписи1.Следующий() Цикл

возврат ВыборкаДетальныеЗаписи1.ссылка;

КонецЦикла;

КонецЦикла;

конецесли;

конецесли;

конецесли;

иначе

возврат Справочники.Номенклатура.НайтиПоНаименованию(предметпоиска);

конецесли;

КонецФункции

&НаКлиенте

Процедура СоздатьПоступление(Команда)

// Поступление = СоздатьПоступлениеНаСервере();

КонецПроцедуры

&НаСервере

Функция КонтрагентПоиск(ИмяКонтрагента)

возврат Справочники.Контрагенты.НайтиПоНаименованию(ИмяКонтрагента);

КонецФункции

&НаКлиенте

Процедура НастройкиНачалоВыбора(Элемент, ДанныеВыбора, СтандартнаяОбработка)

Режим = РежимДиалогаВыбораФайла.Открытие;

ДиалогОткрытияФайла = Новый ДиалогВыбораФайла(Режим);

ДиалогОткрытияФайла.ПолноеИмяФайла = "";

Текст = "ru = ""Текст""; en = ""Text""";

Фильтр = НСтр(Текст)+"(\*.txt)|\*.txt";

ДиалогОткрытияФайла.Фильтр = Фильтр;

ДиалогОткрытияФайла.МножественныйВыбор = Истина;

ДиалогОткрытияФайла.Заголовок = "Выберите файл";

Если ДиалогОткрытияФайла.Выбрать() Тогда

МассивФайлов = ДиалогОткрытияФайла.ВыбранныеФайлы;

Для Каждого ИмяФайла Из МассивФайлов Цикл

ВыбФайл = Новый Файл(ИмяФайла);

Текст = "ru = ""; Размер = ""; en = ""; Size = """;

//Сообщить(ИмяФайла+НСтр(Текст)+ВыбФайл.Размер());

КонецЦикла;

Иначе

Текст = "ru = ""Файл(ы) не выбран!""; en = ""File(s) not selected!""";

Предупреждение(НСтр(Текст));

КонецЕсли;

Настройки = ИмяФайла;

Текст = Новый ЧтениеТекста;

Текст.Открыть(Настройки);

Массив = новый массив;

стр = "";

Пока Стр <> Неопределено Цикл // строки читаются до символа перевода строки

стр = Текст.ПрочитатьСтроку();

Массив.Добавить(стр);

КонецЦикла;

если Массив.Количество() > 3 тогда

Объект.Контрагент = КонтрагентПоиск(Массив[8]);

Объект.КолонкаАртикул = Массив[3];

Объект.КолонкаКоличество = Массив[5];

Объект.КолонкаНаименование = Массив[2];

Объект.КолонкаСумма = Массив[7];

Объект.КолонкаЦена = Массив[6];

Объект.КонечнаяСтрока = Массив[1];

Объект.НачальнаяСтрока = Массив[0];

Объект.Производитель = Массив[4];

Объект.Поиск = Массив[9];

Объект.ОчищатьАртикул = Массив[10];

Объект.ОчиститьДоСимволаВместеСНим = Массив[11];

Объект.ОчиститьПослеСимволаВместеСНим = Массив[12];

Объект.СимволОчисткиДоИПосле = Массив[13];

Объект.СимволОчистки = Массив[14];

Объект.СимволОчисткиПосле = Массив[15];

Объект.КолонкаБренд = Массив[16];

конецесли;

КонецПроцедуры

&НаКлиенте

Процедура СохранитьНастройки(Команда)

Текст = Новый ЗаписьТекста;

Текст.Открыть(Настройки);

Текст.ЗаписатьСтроку(Объект.НачальнаяСтрока);

Текст.ЗаписатьСтроку(Объект.КонечнаяСтрока);

Текст.ЗаписатьСтроку(Объект.КолонкаНаименование);

Текст.ЗаписатьСтроку(Объект.КолонкаАртикул);

Текст.ЗаписатьСтроку(Объект.Производитель);

Текст.ЗаписатьСтроку(Объект.КолонкаКоличество);

Текст.ЗаписатьСтроку(Объект.КолонкаЦена);

Текст.ЗаписатьСтроку(Объект.КолонкаСумма);

Текст.ЗаписатьСтроку(Объект.Контрагент);

Текст.ЗаписатьСтроку(Объект.Поиск);

Текст.ЗаписатьСтроку(Объект.ОчищатьАртикул);

Текст.ЗаписатьСтроку(Объект.ОчиститьДоСимволаВместеСНим);

Текст.ЗаписатьСтроку(Объект.ОчиститьПослеСимволаВместеСНим);

Текст.ЗаписатьСтроку(Объект.СимволОчисткиДоИПосле);

Текст.ЗаписатьСтроку(Объект.СимволОчистки);

Текст.ЗаписатьСтроку(Объект.СимволОчисткиПосле);

Текст.ЗаписатьСтроку(Объект.КолонкаБренд);

Текст.Закрыть();

КонецПроцедуры

&НаКлиенте

Процедура ПоискПриИзменении(Элемент)

// Вставить содержимое обработчика.

КонецПроцедуры

&НаКлиенте

Процедура выполнить2(Команда)

СоздатьДокументы();

КонецПроцедуры

&НаСервере

Функция СоздатьДокументы()

НДокумент = документ.ПолучитьОбъект();

НСтрока = НДокумент.Товары.Добавить();

для каждого строка из тч цикл

НСтрока.Номенклатура = СоздатьНоменклатуру(строка.артикул,строка.НаименованиеИзФайла,строка.вес,строка.проба,строка.весмметала,строка.видметала,строка.уин,НДокумент.Ссылка);

Нстрока.Количество = строка.количество;

Нстрока.КоличествоУпаковок = строка.количество;

Нстрока.Цена = строка.Цена;

Нстрока.Сумма = строка.Сумма;

Нстрока.СтавкаНДС = НСтрока.Номенклатура.СтавкаНДС;

Нстрока.СуммаНДС = Нстрока.Сумма \* 20 / 120;

Нстрока.ор\_УИНИНП = строка.уин;

Нстрока.ор\_ВесВГраммах = строка.вес;

конеццикла;

НДокумент.Записать(РежимЗаписиДокумента.Запись);

Сообщить("Документ заполнен!");

КонецФункции

&НаСервере

Функция СоздатьНоменклатуру(Артикул,НаименованиеПолное,вес,проба,весмметала,видметалла,уин,док)

НоваяНоменклатура = Справочники.Номенклатура.СоздатьЭлемент();

НоваяНоменклатура.ВидНоменклатуры = Справочники.ВидыНоменклатуры.НайтиПоНаименованию("Ювелирное изделие");

НоваяНоменклатура.Артикул = Артикул;

НоваяНоменклатура.ЕдиницаИзмерения = Справочники.БазовыеЕдиницыИзмерения.НайтиПоНаименованию("шт");

НоваяНоменклатура.СтавкаНДС = Перечисления.СтавкиНДС.НДС20;

НоваяНоменклатура.ор\_ВесВГраммах = вес;

НоваяНоменклатура.УстановитьНовыйКод();

НоваяНоменклатура.ор\_ПробаМеталла = Справочники.ор\_ПробыМеталлов.НайтиПоНаименованию(проба);

НоваяНоменклатура.Наименование = НаименованиеПолное;

НоваяНоменклатура.ор\_ВесМеталла = весмметала;

если видметалла = "серебро" или видметалла = "Серебро" тогда

НСтрока = НоваяНоменклатура.ор\_СоставСплава.Добавить();

НСтрока.ВидМеталла = Перечисления.ВидыМеталовГИИСДМДК.Серебро;

НСтрока.Масса = весмметала;

иначеесли видметалла = "золото" или видметалла = "Золото" тогда

НСтрока = НоваяНоменклатура.ор\_СоставСплава.Добавить();

НСтрока.ВидМеталла = Перечисления.ВидыМеталовГИИСДМДК.Золото;

НСтрока.Масса = весмметала;

конецесли;

Попытка

НоваяНоменклатура.Записать();

МенеджерЗаписи = РегистрыСведений.ПартииГИИСДМДК.СоздатьМенеджерЗаписи();

МенеджерЗаписи.УИНИНП = уин;

МенеджерЗаписи.Номенклатура = НоваяНоменклатура.Ссылка;

МенеджерЗаписи.ДокументРезерва = док;

МенеджерЗаписи.Организация = док.Организация;

МенеджерЗаписи.ТипТовара = Перечисления.ТипТовараПоВладельцуГИИСДМДК.СобственныйТовар;

МенеджерЗаписи.Статус = Перечисления.СтатусыПартийГИИСДМДК.Свободный;

МенеджерЗаписи.Записать();

возврат НоваяНоменклатура.Ссылка;

Исключение

ОбщегоНазначенияКлиентСервер.СообщитьПользователю("Ошибка при создании номенклатуры: " +Символы.ПС+ОписаниеОшибки());

Результат = Ложь;

Возврат Результат;

КонецПопытки;

КонецФункции

&НаКлиенте

Процедура Команда2(Команда)

Попытка

ex = ПолучитьCOMобъект("","Excel.Application");

Исключение

Сообщить("Excel Application не создан!!");

Возврат;

КонецПопытки;

Попытка

ex.workbooks.open(Файл,1);

Исключение

Сообщить("Файл перемещен или удален!");

Возврат;

КонецПопытки;

i = объект.НачальнаяСтрока;

пока I < объект.КонечнаяСтрока Цикл

СтрокаН = Тч.Добавить();

если объект.КолонкаНаименование <> 0 тогда

строканайденная = ПолучитьСодержаниеЯчейки(I, объект.КолонкаНаименование);

если строканайденная <> "" тогда

СтрокаН.НаименованиеИзФайла = строканайденная;

прервать;

конецесли;

конецесли;

если объект.КолонкаАртикул <> 0 тогда

Арт = ПолучитьСодержаниеЯчейки(I, объект.КолонкаАртикул);

если объект.ОчиститьДоСимволаВместеСНим тогда

если СтрНайти(Арт,Объект.СимволОчистки) <> 0 тогда

СтрокаН.Артикул = Сред(Арт,СтрНайти(Арт,Объект.СимволОчистки) + 1);

иначе

СтрокаН.Артикул = Арт;

конецесли;

иначеесли объект.ОчиститьПослеСимволаВместеСНим тогда

если СтрНайти(Арт,Объект.СимволОчистки) <> 0 тогда

СтрокаН.Артикул = Сред(Арт,0,СтрНайти(Арт,Объект.СимволОчистки) - 1);

иначе

СтрокаН.Артикул = Арт;

конецесли;

иначеесли объект.ОчищатьАртикул тогда

если СтрНайти(Арт,Объект.СимволОчистки) <> 0 тогда

k = СтрНайти(Арт,Объект.СимволОчистки);

пока k <> 0 цикл

Арт = Сред(Арт,0,k - 1) + Сред(Арт,k + 1);

k = СтрНайти(Арт,Объект.СимволОчистки);

конеццикла;

СтрокаН.Артикул = Арт;

иначе

СтрокаН.Артикул = Арт;

конецесли;

иначе

СтрокаН.Артикул = Арт;

конецесли ;

конецесли;

если объект.КолонкаКоличество <> 0 тогда

СтрокаН.Количество = ПолучитьСодержаниеЯчейки(I, объект.КолонкаКоличество);

конецесли;

если объект.КолонкаЦена <> 0 тогда

СтрокаН.Цена = ПолучитьСодержаниеЯчейки(I, объект.КолонкаЦена);

конецесли;

если объект.КолонкаСумма <> 0 тогда

СтрокаН.Сумма = ПолучитьСодержаниеЯчейки(I, объект.КолонкаСумма);

конецесли;

если объект.Производитель <> 0 тогда

СтрокаН.Производитель = найтипроизводителя(ПолучитьСодержаниеЯчейки(I, объект.Производитель));

конецесли;

если объект.поиск = "По артикулу" тогда

СтрокаН.Номенклатура = найтиноменклатура(СтрокаН.Артикул);

иначеесли объект.поиск = "По наименованию" тогда

СтрокаН.Номенклатура = найтиноменклатура(СтрокаН.НаименованиеИзФайла);

иначеесли объект.поиск = "По артикулу и производителю" тогда

СтрокаН.Номенклатура = найтиноменклатура(СтрокаН.Артикул,СтрокаН.Производитель);

иначе

СтрокаН.Номенклатура = найтиноменклатура(СтрокаН.НаименованиеИзФайла);

конецесли;

i = i + 1;

КонецЦикла;

заполнитьоформление();

КонецПроцедуры

&НаСервере

Функция ЗаполнитьОформление()

ЭлементУсловногоОформления = УсловноеОформление.Элементы.Добавить();

ЭлементОтбора = ЭлементУсловногоОформления.Отбор.Элементы.Добавить(Тип("ЭлементОтбораКомпоновкиДанных"));

ЭлементОтбора.ЛевоеЗначение = Новый ПолеКомпоновкиДанных("ТЧ.Номенклатура");

ЭлементОтбора.ВидСравнения = ВидСравненияКомпоновкиДанных.Равно;

ЭлементОтбора.ПравоеЗначение = Справочники.Номенклатура.ПустаяСсылка();

ЭлементУсловногоОформления.Оформление.УстановитьЗначениеПараметра("ЦветФона",WebЦвета.Красный);

//задаем оформляемые колонки (все)

Для каждого элПоля Из Элементы.ТЧ.ПодчиненныеЭлементы Цикл

если объект.поиск = "По артикулу" тогда

если элПоля.Имя = "ТЧАртикул" тогда

ОформляемоеПоле = ЭлементУсловногоОформления.Поля.Элементы.Добавить();

ОформляемоеПоле.Поле = Новый ПолеКомпоновкиДанных(элПоля.Имя);

конецесли;

иначеесли объект.поиск = "По наименованию" тогда

если элПоля.Имя = "ТЧНаименованиеИзФайла" тогда

ОформляемоеПоле = ЭлементУсловногоОформления.Поля.Элементы.Добавить();

ОформляемоеПоле.Поле = Новый ПолеКомпоновкиДанных(элПоля.Имя);

конецесли;

иначеесли объект.поиск = "По артикулу и производителю" тогда

если элПоля.Имя = "ТЧАртикул" или элПоля.Имя = "ТЧПроизводитель" тогда

ОформляемоеПоле = ЭлементУсловногоОформления.Поля.Элементы.Добавить();

ОформляемоеПоле.Поле = Новый ПолеКомпоновкиДанных(элПоля.Имя);

конецесли;

иначе

если элПоля.Имя = "ТЧНаименованиеИзФайла" тогда

ОформляемоеПоле = ЭлементУсловногоОформления.Поля.Элементы.Добавить();

ОформляемоеПоле.Поле = Новый ПолеКомпоновкиДанных(элПоля.Имя);

конецесли;

конецесли;

КонецЦикла;

ЭлементУсловногоОформления = УсловноеОформление.Элементы.Добавить();

ЭлементОтбора = ЭлементУсловногоОформления.Отбор.Элементы.Добавить(Тип("ЭлементОтбораКомпоновкиДанных"));

ЭлементОтбора.ЛевоеЗначение = Новый ПолеКомпоновкиДанных("ТЧ.БолееОдной");

ЭлементОтбора.ВидСравнения = ВидСравненияКомпоновкиДанных.Равно;

ЭлементОтбора.ПравоеЗначение = истина;

ЭлементУсловногоОформления.Оформление.УстановитьЗначениеПараметра("ЦветФона",WebЦвета.Желтый);

//задаем оформляемые колонки (все)

Для каждого элПоля Из Элементы.ТЧ.ПодчиненныеЭлементы Цикл

//если элПоля.Имя = "ТЧНоменклатура" тогда

ОформляемоеПоле = ЭлементУсловногоОформления.Поля.Элементы.Добавить();

ОформляемоеПоле.Поле = Новый ПолеКомпоновкиДанных(элПоля.Имя);

//конецесли;

//если объект.поиск = "По артикулу" тогда

// если элПоля.Имя = "ТЧАртикул" тогда

// ОформляемоеПоле = ЭлементУсловногоОформления.Поля.Элементы.Добавить();

// ОформляемоеПоле.Поле = Новый ПолеКомпоновкиДанных(элПоля.Имя);

// конецесли;

//иначеесли объект.поиск = "По наименованию" тогда

// если элПоля.Имя = "ТЧНаименованиеИзФайла" тогда

// ОформляемоеПоле = ЭлементУсловногоОформления.Поля.Элементы.Добавить();

// ОформляемоеПоле.Поле = Новый ПолеКомпоновкиДанных(элПоля.Имя);

// конецесли;

//

//иначеесли объект.поиск = "По артикулу и производителю" тогда

// если элПоля.Имя = "ТЧАртикул" или элПоля.Имя = "ТЧПроизводитель" тогда

// ОформляемоеПоле = ЭлементУсловногоОформления.Поля.Элементы.Добавить();

// ОформляемоеПоле.Поле = Новый ПолеКомпоновкиДанных(элПоля.Имя);

// конецесли;

//

//иначе

// если элПоля.Имя = "ТЧНаименованиеИзФайла" тогда

// ОформляемоеПоле = ЭлементУсловногоОформления.Поля.Элементы.Добавить();

// ОформляемоеПоле.Поле = Новый ПолеКомпоновкиДанных(элПоля.Имя);

// конецесли;

//конецесли;

КонецЦикла;

для каждого строка из ТЧ цикл

если строка.номенклатура = Справочники.Номенклатура.ПустаяСсылка() тогда

сообщить("Не заполнена колонка " + "<" + "Номенклатура" + ">" + " в строке " + строка(строка.номерстроки) + " списка <Товары>");

конецесли;

конеццикла;

КонецФункции

&НаСервере

Процедура ПрочитатьФайлВТабДок()

УстановитьОтключениеБезопасногоРежима(Истина);

ТабДок.Прочитать(Файл);

КонецПроцедуры

&НаСервере

Функция ПолучитьСодержаниеЯчейки(Строка, Колонка)

Если Колонка = 0 Тогда

Возврат "";

КонецЕсли;

мСодержание = ТабДок.Область("R" + Формат(Строка,"ЧГ=") + "C" + Формат(Колонка,"ЧГ=")).Текст;

Возврат СокрЛП(мСодержание);

КонецФункции

&НаКлиенте

Процедура ПриОткрытии(Отказ)

//МояДата = '20221217';

//если ТекущаяДата() > МояДата тогда

// Сообщить("Пробная версия закончена");

// ЭтаФорма.Закрыть();

//конецесли;

Объект.ОчищатьАртикул = ложь;

Объект.ОчиститьДоСимволаВместеСНим = ложь;

Объект.ОчиститьПослеСимволаВместеСНим = ложь;

КонецПроцедуры

&НаСервере

Процедура ТЧНоменклатураПриИзмененииНаСервере(товар)

если товар <> Справочники.Номенклатура.ПустаяСсылка() тогда

ЭлементУсловногоОформления = УсловноеОформление.Элементы.Добавить();

ЭлементОтбора = ЭлементУсловногоОформления.Отбор.Элементы.Добавить(Тип("ЭлементОтбораКомпоновкиДанных"));

ЭлементОтбора.ЛевоеЗначение = Новый ПолеКомпоновкиДанных("ТЧ.БолееОдной");

ЭлементОтбора.ВидСравнения = ВидСравненияКомпоновкиДанных.Равно;

ЭлементОтбора.ПравоеЗначение = истина;

ЭлементУсловногоОформления.Оформление.УстановитьЗначениеПараметра("ЦветФона",WebЦвета.Белый);

//задаем оформляемые колонки (все)

Для каждого элПоля Из Элементы.ТЧ.ПодчиненныеЭлементы Цикл

//если элПоля.Имя = "ТЧНоменклатура" тогда

ОформляемоеПоле = ЭлементУсловногоОформления.Поля.Элементы.Добавить();

ОформляемоеПоле.Поле = Новый ПолеКомпоновкиДанных(элПоля.Имя);

//конецесли;

//если объект.поиск = "По артикулу" тогда

// если элПоля.Имя = "ТЧАртикул" тогда

// ОформляемоеПоле = ЭлементУсловногоОформления.Поля.Элементы.Добавить();

// ОформляемоеПоле.Поле = Новый ПолеКомпоновкиДанных(элПоля.Имя);

// конецесли;

//иначеесли объект.поиск = "По наименованию" тогда

// если элПоля.Имя = "ТЧНаименованиеИзФайла" тогда

// ОформляемоеПоле = ЭлементУсловногоОформления.Поля.Элементы.Добавить();

// ОформляемоеПоле.Поле = Новый ПолеКомпоновкиДанных(элПоля.Имя);

// конецесли;

//

//иначеесли объект.поиск = "По артикулу и производителю" тогда

// если элПоля.Имя = "ТЧАртикул" или элПоля.Имя = "ТЧПроизводитель" тогда

// ОформляемоеПоле = ЭлементУсловногоОформления.Поля.Элементы.Добавить();

// ОформляемоеПоле.Поле = Новый ПолеКомпоновкиДанных(элПоля.Имя);

// конецесли;

//

//иначе

// если элПоля.Имя = "ТЧНаименованиеИзФайла" тогда

// ОформляемоеПоле = ЭлементУсловногоОформления.Поля.Элементы.Добавить();

// ОформляемоеПоле.Поле = Новый ПолеКомпоновкиДанных(элПоля.Имя);

// конецесли;

//конецесли;

КонецЦикла;

иначе

ЭлементУсловногоОформления = УсловноеОформление.Элементы.Добавить();

ЭлементОтбора = ЭлементУсловногоОформления.Отбор.Элементы.Добавить(Тип("ЭлементОтбораКомпоновкиДанных"));

ЭлементОтбора.ЛевоеЗначение = Новый ПолеКомпоновкиДанных("ТЧ.БолееОдной");

ЭлементОтбора.ВидСравнения = ВидСравненияКомпоновкиДанных.Равно;

ЭлементОтбора.ПравоеЗначение = истина;

ЭлементУсловногоОформления.Оформление.УстановитьЗначениеПараметра("ЦветФона",WebЦвета.Желтый);

//задаем оформляемые колонки (все)

Для каждого элПоля Из Элементы.ТЧ.ПодчиненныеЭлементы Цикл

//если элПоля.Имя = "ТЧНоменклатура" тогда

ОформляемоеПоле = ЭлементУсловногоОформления.Поля.Элементы.Добавить();

ОформляемоеПоле.Поле = Новый ПолеКомпоновкиДанных(элПоля.Имя);

//конецесли;

//если объект.поиск = "По артикулу" тогда

// если элПоля.Имя = "ТЧАртикул" тогда

// ОформляемоеПоле = ЭлементУсловногоОформления.Поля.Элементы.Добавить();

// ОформляемоеПоле.Поле = Новый ПолеКомпоновкиДанных(элПоля.Имя);

// конецесли;

//иначеесли объект.поиск = "По наименованию" тогда

// если элПоля.Имя = "ТЧНаименованиеИзФайла" тогда

// ОформляемоеПоле = ЭлементУсловногоОформления.Поля.Элементы.Добавить();

// ОформляемоеПоле.Поле = Новый ПолеКомпоновкиДанных(элПоля.Имя);

// конецесли;

//

//иначеесли объект.поиск = "По артикулу и производителю" тогда

// если элПоля.Имя = "ТЧАртикул" или элПоля.Имя = "ТЧПроизводитель" тогда

// ОформляемоеПоле = ЭлементУсловногоОформления.Поля.Элементы.Добавить();

// ОформляемоеПоле.Поле = Новый ПолеКомпоновкиДанных(элПоля.Имя);

// конецесли;

//

//иначе

// если элПоля.Имя = "ТЧНаименованиеИзФайла" тогда

// ОформляемоеПоле = ЭлементУсловногоОформления.Поля.Элементы.Добавить();

// ОформляемоеПоле.Поле = Новый ПолеКомпоновкиДанных(элПоля.Имя);

// конецесли;

//конецесли;

КонецЦикла;

Конецесли;

КонецПроцедуры

&НаКлиенте

Процедура ТЧНоменклатураПриИзменении(Элемент)

ТЧНоменклатураПриИзмененииНаСервере(этаформа.ТекущийЭлемент.ТекущиеДанные.номенклатура);

КонецПроцедуры