**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное автономное образовательное учреждение**

**высшего образования**

**ДАЛЬНЕВОСТОЧНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ**

ШКОЛА ЕСТЕСТВЕННЫХ НАУК

**Кафедра прикладной математики, механики, управления и программного обеспечения**

АВТОМАТИЗАЦИЯ УЧЕТА ТОВАРОВ

**КУРСОВОЙ ПРОЕКТ**

по дисциплине «Технология коллективной разработки информационных систем»

по образовательной программе подготовки бакалавров по направлению 02.03.03 "Математическое обеспечение и администрирование информационных систем"

Студент группы Б8117-02.03.03 \_\_\_\_\_\_\_\_\_/Магалиш В.С.

(подпись)

Студент группы Б8117-02.03.03 \_\_\_\_\_\_\_\_\_/Скопина А.А.

(подпись)

Студент группы Б8117-02.03.03 \_\_\_\_\_\_\_\_\_/Ковальская М.А.

(подпись)

Руководитель

к.т.н. \_\_\_\_\_\_\_\_\_В.М. Гриняк

(подпись)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

(подпись)

Защищен оценкой \_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись) И.О. Фамилия

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

г. Владивосток

2021 г.

**Оглавление**

[1 Регламент инспекции рабочих продуктов 4](#_Toc61433858)

[1.1 Критерии отнесения к формальной и неформальной инспекции различных видов продуктов 4](#_Toc61433859)

[1.2 Перечень ролей участников формальных инспекций и их обязанности 4](#_Toc61433860)

[1.3 Этапы инспекции 5](#_Toc61433861)

[1.4 Порядок организации 5](#_Toc61433862)

[1.5 Порядок подготовки к инспекции 6](#_Toc61433863)

[1.6 Порядок проведения инспекции 7](#_Toc61433864)

[1.7 Перечень статусов и степеней важности замечаний 8](#_Toc61433865)

[1.8 Порядок верификации учета замечаний 9](#_Toc61433866)

[1.9 Метрики, характеризующие эффективность инспекций (предполагаемые) 9](#_Toc61433867)

[2 Модель состояний задач 11](#_Toc61433868)

[2.1 Перечень возможных состояний задачи и их интерпретация 11](#_Toc61433869)

[2.2 Правила создания новой задачи 11](#_Toc61433870)

[2.3 Правила перехода задачи из состояния в состояние 12](#_Toc61433871)

[3 Требования к продукту 14](#_Toc61433872)

[3.1 Требования к подсистеме Перечисления 14](#_Toc61433873)

[3.2 Требования к подсистеме Справочники 16](#_Toc61433874)

[3.3 Требования к подсистеме Документы 26](#_Toc61433875)

[3.4 Требования к подсистеме Регистры накопления 36](#_Toc61433876)

[3.5 Требования к подсистеме Регистры сведений 37](#_Toc61433877)

[4 Дизайн продукта 40](#_Toc61433878)

[4.1 Архитектурно-контекстная диаграмма системы 40](#_Toc61433879)

[4.2 Диаграмма связей объектов 40](#_Toc61433880)

[5 Система измерений проекта 42](#_Toc61433881)

[6 Задачи проекта 44](#_Toc61433882)

[6.1 Задачи для подсистемы Перечисления 44](#_Toc61433883)

[6.2 Задачи для подсистемы Справочники 44](#_Toc61433884)

[6.3 Задачи для подсистемы Документы 45](#_Toc61433885)

[6.4 Задачи для подсистемы Регистры накопления 46](#_Toc61433886)

[6.5 Задачи для подсистемы Регистры сведений 46](#_Toc61433887)

[7 Рекомендации по кодированию 47](#_Toc61433888)

[8 История изменений документа 50](#_Toc61433889)

# **1 Регламент инспекции рабочих продуктов**

## **1.1 Критерии отнесения к формальной и неформальной инспекции различных видов продуктов**

|  |  |  |
| --- | --- | --- |
| Рабочий продукт | Размер продукта или изменения в нем, не менее | |
| формальные инспекции | неформальные инспекции |
| Требования | 5 страниц или 10% рабочего продукта | 2 страницы или 5% рабочего продукта |
| Документы по дизайну | 5 страниц или 10% рабочего продукта | 2 страницы или 5% рабочего продукта |
| Не комментированный код | 50 NCLOC | 20 NCLOC |
| Тесты | 5 страниц или 10% рабочего продукта | 2 страницы или 5% рабочего продукта |

## **1.2 Перечень ролей участников формальных инспекций и их обязанности**

|  |  |
| --- | --- |
| Участник | Роль |
| Инспекция требований | |
| Магалиш В. С. | Автор |
| Ковальская М. А. | Председатель, Ведущий |
| Скопина А. А. | Секретарь, Инспектор |
| Инспекция документов по дизайну | |
| Магалиш В. С. | Автор |
| Ковальская М. А. | Секретарь, Инспектор |
| Скопина А. А. | Председатель, Ведущий |
| Инспекция кода (1) | |
| Магалиш В. С. | Председатель, Ведущий |
| Ковальская М. А. | Секретарь, Инспектор |
| Скопина А. А. | Автор |
| Инспекция кода (2) | |
| Магалиш В. С. | Председатель, Ведущий |
| Ковальская М. А. | Автор |
| Скопина А. А. | Секретарь, Инспектор |
| Инспекция тестов | |
| Магалиш В. С. | Автор |
| Ковальская М. А. | Председатель, Инспектор |
| Скопина А. А. | Секретарь, Инспектор, Ведущий |

## **1.3 Этапы инспекции**

Формальная инспекция состоит из нескольких этапов:

* + - * Планирование инспекции.
* Назначение инспекции.
* Обзорное собрание (опционально).
* Подготовка к инспекции.
* Собрания по инспекции.
* Завершение инспекции (распространение результатов, переработка рабочего продукта, проверка исправления недостатков в рабочем продукте).

## **1.4 Порядок организации**

Автор:

1. Инициирует формальную инспекцию рабочего продукта и оповещает руководителя проекта о готовности рабочего продукта к формальной инспекции.
2. Запрашивает руководителя проекта об имеющихся ресурсах на роль председателя инспекции.
3. Если необходимо, проводит обзорное собрание, чтобы ввести всех участников инспекции в курс дела.

Председатель:

1. Проверяет рабочий продукт на готовность к формальной инспекции.
2. Определяет необходимость проведения обзорного собрания.
3. Проверяет, что все обязательные участники формальной инспекции приняли приглашение либо прислали отказ. В случае получения хотя бы одного отказа, председатель проверяет, может ли быть проведена инспекция без отказавшегося участника. Если нет, то инспекция не проводится.

Председатель и автор (ответственный - председатель):

1. Определяют место, дату и время проведения инспекции, согласовывая это с участниками.
2. Подбирают команду участников и распределяют роли.
3. Приглашают участников на формальную инспекцию.

Секретарь, ведущий и инспектор:

1. Принимают или отклоняют приглашение на формальную инспекцию.

## **1.5 Порядок подготовки к инспекции**

Инспектор:

1. Изучает предоставленный для инспекции рабочий продукт, используя накопленный опыт, стандарты, руководства, контрольные списки. Срок: 2 недели.
2. Заполняет необходимые поля протокола подготовки к формальной инспекции (в котором указываются время подготовки к инспекции и описание найденных ошибок) и отправляет его председателю формальной инспекции, секретарю и автору. Срок: 2 дня.

Председатель:

1. Решает, провести инспекцию, перенести ее или отменить. Срок: 2 дня.
2. Оповещает всех участников формальной инспекции об изменениях. Срок: 1 день.

Автор:

1. Изучает содержание полученных от инспекторов протоколов подготовки к формальной инспекции и анализирует изложенные в них замечания. Срок: 1 неделя.

Секретарь:

1. Анализирует замечания, зафиксированные инспекторами в протоколах подготовки к формальной инспекции, выявляет повторяющиеся. На данном этапе секретарь собирает полный список проблем рабочего продукта, объединив все замечания, изложенные в протоколах подготовки к формальной инспекции каждого инспектора. Срок: 1 неделя.

## **1.6 Порядок проведения инспекции**

Этот этап наступает только при условии успешного завершения двух предыдущих этапов (когда ни на одном из предыдущих этапов не было принято решение не проводить формальную инспекцию).

Председатель:

1. Оглашает цель данного собрания, представляет участников инспекции (если это необходимо) и уточняет их роли.
2. Проверяет, что у каждого инспектора есть его замечания, и просит ведущего начать представление инспектируемого рабочего продукта.
3. Следит за ходом и темпом собрания: пресекает попытки решения выявленной в рабочем продукте проблемы и просит ведущего продолжать представление рабочего продукта; разрешает конфликты; пресекает попытки ухода от темы собрания; следит, чтобы внимание было сосредоточено на рабочем продукте и выявлении его недостатков.
4. По ходу собрания решает, действительно ли конкретное замечание является проблемой в случае, если инспекторы не могут прийти к единому мнению.
5. Может прервать собрание, если его длительность превысила запланированное время.
6. Принимает решение о переносе собрания в случае нарушения участниками инспекции установленного приглашением порядка проведения инспекции или возникновения непредвиденных обстоятельств.
7. Принимает окончательное решение о назначении проверяющего.

Ведущий:

1. Представляет рабочий продукт участникам собрания.

Секретарь:

1. Оглашает для всех участников окончательную формулировку каждого замечания, его статус и местоположение.
2. Ведет протокол собрания. Формулирует обнаруженные в ходе собрания недостатки рабочего продукта в виде замечаний и заносит их в протокол инспекции.

Инспектор:

1. Задает вопросы и оглашает найденные в рабочем продукте проблемы.
2. По ходу инспекции высказывает свое мнение о статусе замечания.
3. В конце инспекции выражает свое мнение по поводу необходимости проведения повторной формальной инспекции рабочего продукта.

Автор:

1. Отвечает на вопросы инспекторов, не пытаясь оценить корректность рабочего продукта.

## **1.7 Перечень статусов и степеней важности замечаний**

Допустимые значения статуса замечания:

1. Ошибка (Error) - проблема, которая найдена на той же фазе, на которой внесена.
2. Дефект (Defect) - проблема, которая найдена на фазе, отличной от той, на которой внесена.
3. Комментарий (Comment) - наблюдение, предложение, рекомендация или предложенное улучшение, или вопрос, требующий разъяснения.
4. Замечание для исследования (Investigate) - проблема, природа которой не может быть определена на собрании и требует дополнительного исследования.

Допустимые значения степени серьезности замечания:

1. Критическое (Critical)
2. Особо важное (Major)
3. Среднее (Moderate)
4. Незначительное (Minor)
5. Другое (Other)

## **1.8 Порядок верификации учета замечаний**

1. Автор фиксирует решение по каждому замечанию в протоколе инспекции, анализирует все замечания, исправляет все недостатки рабочего продукта.
2. Секретарь рассылает всем участникам протокол формальной инспекции.
3. Председатель должен убедиться в правильности и полноте заполнения протокола, а также что протокол разослан и все участники формальной инспекции оповещены.
4. Лицо, назначенное во время инспекции на роль проверяющего следит, чтобы все замечания были учтены в обновленном рабочем продукте. Именно после вердикта проверяющего формальная инспекция считается завершенной.

## **1.9 Метрики, характеризующие эффективность инспекций (предполагаемые)**

Набор метрик по инспекциям, собираемых на конкретном проекте, определяется нормативными документами по процессу и программой измерений предприятия и планируется на фазе планирования проекта. Метрики используются для принятия решения об эффективности инспекции и принятия решения о необходимости реинспекции. К типичным метрикам по инспекциям относятся:

1. Inspection Fault Density (IFD)

IFD = (Количество найденных ошибок / Размер рабочего продукта) Стратегическая цель метрики: повысить качество разрабатываемого ПО.

Изучаемый объект метрики – инспекция, измеряемый атрибут – плотность найденных в ходе инспекции ошибок.

Единица измерения: ошибка / <страница, требование, LOC, тест>.

IFD характеризует эффективность инспекции, а также качество инспектируемого продукта. Целью предприятия является снижение IFD.

1. Inspection Preparation Rate (IPR)

IPR = (Количество инспекторов \* Размер продукта) / Общее время подготовки

Стратегическая цель метрики: повысить качество разрабатываемого ПО.

Изучаемый объект метрики – подготовка к инспекции, измеряемый атрибут – производительность подготовки к инспекции.

Единица измерения: <страница, требование, LOC, тест> / час.

Метрика IPR характеризует эффективность и степень подготовки инспекторов к инспекции (чем меньше IPR, тем эффективнее будет инспекция).

1. Inspection Rate (IR)

IR = Размер продукта / Общее время инспектирования

Стратегическая цель метрики: повысить качество разрабатываемого ПО.

Изучаемый объект метрики – проведение инспекции, измеряемый атрибут – производительность инспектирования.

Единица измерения: <страница, требование, LOC, тест> / час.

Метрика IR характеризует эффективность инспекции (чем меньше IR, тем эффективнее инспекция).

# **2 Модель состояний задач**

## **2.1 Перечень возможных состояний задачи и их интерпретация**

Задача может пребывать в следующих состояниях:

* New – новая подзадача.
* Approved for analysis – допущено для анализа.
* Assigned for analysis – назначено для анализа.
* Analysis – в процессе анализа.
* Waiting for information – в ожидании информации.
* Analysis completed – анализ завершён.
* Terminated – прерванная.
* Forward – передана на дальнейшую разработку.
* Coding – кодирование.
* Inspected – проинспектировано.
* Resolved – проверено.
* Integrated – интегрировано.
* Tested - протестировано.
* Closed – закрыто.

## **2.2 Правила создания новой задачи**

Новые задачи создает руководитель команды (team lead). Новые задачи создаются в следующих случаях:

* Начало работы над проектом.
* При выявлении во время инспекции непредвиденных, неучтенных аспектов разработки, их последующее проектирование и реализация становятся новой задачей.
* Если созданная задача оказывается слишком объемной и ее можно разбить на подзадачи, эти подзадачи становятся новыми задачами.

## **2.3 Правила перехода задачи из состояния в состояние**

1. Approved for analysis

В это состояние подзадача переводится после рассмотрения её контрольной группой CCB.

2. Assigned for analysis

В это состояние подзадача переводится после назначение её на конкретного сотрудника.

3. Analysis

В это состояние подзадачу переводит сотрудник после того, как начнёт её анализ.

4. Waiting for information

В это состояние задача может быть переведена любым участником рабочего процесса в случаях, когда для принятия решения ему необходима дополнительная информация.

5. Analysis completed

Переводится сотрудником после завершения анализа задачи.

6. Terminated

В это состояние задача может быть переведена CCB после рассмотрения и принятия решения о необходимости отмены, прекращения работы над задачей. В это состояние задача может быть переведена, например, сразу после создания новой или по результатам её анализа.

7. Forward

В это состояние задача переводится CCB после анализа при назначении задачи на разработку конкретному сотруднику.

8. Coding

В это состояние задача переводится сотрудником-разработчиком, при начале работы по кодированию, связанному с задачей.

9. Inspected

В это состояние задача переводится сотрудником-разработчиком после завершения кодирования и инспектирования изменений рабочего продукта.

10. Resolved

Переводится после проверки изменений рабочего продукта по результатам инспектирования специалистом-экспертом.

11. Integrated

Переводится сотрудником, осуществляющим интеграцию изменений в основную ветку рабочего продукта после успешной интеграции этих изменений.

12. Tested

Переводится сотрудником, осуществляющим тестирование изменений в рабочий продукт.

13. Closed

В это состояние задача переводится CCB по результатам отчёта о тестировании сделанных изменений.

# **3 Требования к продукту**

Программный продукт предназначен для автоматизации учета товаров в магазине.

Программный продукт состоит из следующих подсистем:

1. Перечисления
   1. Виды номенклатуры
   2. Пол
   3. Виды контактной информации
   4. Виды контрагентов
   5. Виды складов
2. Справочники
   1. Номенклатура
   2. Контрагенты
   3. Физические лица
   4. Склады
3. Документы
   1. Поступление товарок и услуг
   2. Продажа товаров и услуг
   3. Установка цен
4. Регистры накопления
   1. Товары на складах
5. Регистры сведений
   1. Цены
   2. Штрихкоды

## **3.1 Требования к подсистеме Перечисления**

Требование REQ\_ENUM\_VN\_001

Имя: ВидыНоменклатуры

Синоним: Виды номенклатуры

Значения:

* Имя: Услуга

Синоним: Услуга

* Имя: ТабачноеИзделие

Синоним: Табачное изделие

* Имя: АлкогольнаяПродукция

Синоним: Алкогольная продукция

Требование REQ\_ENUM\_P\_001

Имя: Пол

Синоним: Пол

Значения:

* Имя: Женский

Синоним: Женский

* Имя: Мужской

Синоним: Мужской

Требование REQ\_ENUM\_VKI\_001

Имя: ВидыКонтактнойИнформации

Синоним: Виды контактной информации

Значения:

* Имя: ЮридическийАдрес

Синоним: Юридический адрес

* Имя: ФактическийАдрес

Синоним: Фактический адрес

* Имя: Телефон

Синоним: Телефон

* Имя: ЭлектроннаяПочта

Синоним: Элекьронная почта

* Имя: Другое

Синоним: Другое

Требование REQ\_ENUM\_VK\_001

Имя: ВидыКонтрагентов

Синоним: Виды контрагентов

Значения:

* Имя: ФизическоеЛицо

Синоним: Физическое лицо

* Имя: ЮридическоеЛицо

Синоним: Юридическое лицо

* Имя: ИндвидуальныйПредприниматель

Синоним: Индивидуальный предприниматель

Требование REQ\_ENUM\_VS\_001

Имя: ВидыСкладов

Синоним: Виды контрагентов

Значения:

* Имя: Собственный

Синоним: Собственный

* Имя: Арендуемый

Синоним: Арендуемый

## **3.2 Требования к подсистеме Справочники**

Требование REQ\_CTLG\_N\_001

Имя: Номенклатура

Синоним: Номенклатура

Требование REQ\_CTLG\_N\_002

Справочник Номенклатура должен быть иерархическим, вид иерархии – иерархия групп и элементов.

Требование REQ\_CTLG\_N\_003

Реквизиты:

* Вид
* Синоним: Вид
* Тип: ПеречислениеСсылка.ВидыНоменклатуры
* Проверка заполнения: Не проверять

Требование REQ\_CTLG\_N\_004

Все элементы автонумеруются с контролем уникальности, серии кодов для всего справочника.

Требование REQ\_CTLG\_N\_005

Длина кода: 9.

Тип кода: Строка.

Длина наименования: 25.

Основное представление: В виде наименования.

Требование REQ\_CTLG\_N\_006

Система должна проверять, что вид элемента соответствует виду группы, в который записывается данный элемент. При несоответствии элемент не записывается, и выдается сообщение: "Ошибка: Вид элемента группы должен совпадать с видом группы."

Требование REQ\_CTLG\_N\_007

Форма справочника Номенклатура состоит из следующих элементов:

* Код - уникальный номер элемента, заполняется автоматически.
* Наименование - имя элемента. Пользователь вводит самостоятельно.
* Родитель - группа, которой принадлежит элемент. Пользователь может выбрать из выпадающего списка уже существующую группу или создать новую.
* Вид номенклатуры - значение реквизита “Вид”. Пользователь может выбрать вид номенклатуры из выпадающего списка.
* Штрихкоды - ссылка, открывающая штрихкоды данного элемента.

Пользователь может сохранить изменения, нажав кнопку “Записать и закрыть” или кнопку “Записать”.

Форма справочника Номенклатура представлена на рисунке 1.

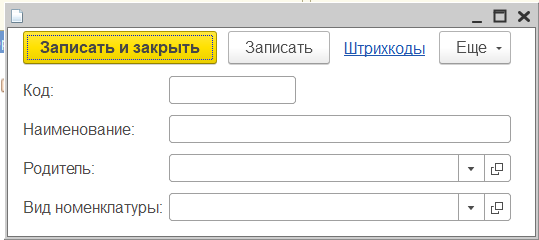


Рисунок 1. Форма справочника Номенклатура

Требование REQ\_CTLG\_K\_001

Имя: Контрагенты

Синоним: Контрагенты

Требование REQ\_CTLG\_K\_002

Справочник Контрагенты должен быть не иерархическим.

Требование REQ\_CTLG\_K\_003

Реквизиты:

* Вид
* Синоним: Вид
* Тип: ПеречислениеСсылка.ВидыКонтрагентов
* Проверка заполнения: Выдавать ошибку
* ИНН
* Синоним: ИНН
* Тип: Строка неограниченной длины
* Проверка заполнения: Не проверять
* КПП
* Синоним: КПП
* Тип: Строка неограниченной длины
* Проверка заполнения: Не проверять

Требование REQ\_CTLG\_K\_004

Табличная часть КонтактнаяИнформация:

* Синоним: Контактная информация
* Реквизиты:
* Вид
* Синоним: Вид
* Тип: ПеречислениеСсылка.ВидыКонтактнойИнформации
* Проверка заполнения: Выдавать ошибку
* Информация
* Информация
* Тип: Строка неограниченной длины
* Проверка заполнения: Выдавать ошибку

Требование REQ\_CTLG\_K\_005

Все элементы автонумеруются с контролем уникальности, серии кодов для всего справочника.

Требование REQ\_CTLG\_N\_006

Длина кода: 9.

Тип кода: Строка.

Длина наименования: 70.

Основное представление: В виде наименования.

Требование REQ\_CTLG\_K\_007

В записи ИНН и КПП допустимы только цифры. В случае, если введены неверные символы, объект не записывается и пользователю выводятся сообщения:

* "Ошибка: В записи КПП допустимы только цифры."
* "Ошибка: В записи ИНН допустимы только цифры."

Требование REQ\_CTLG\_K\_008

Если значение реквизита Вид - ФизическоеЛицо, то реквизиты ИНН и КПП не проверяются на заполнение.  
 Если значение реквизита Вид - ЮридическоеЛицо или ИндивидуальныйПредприниматель, то реквизиты ИНН и КПП проверяются на заполнение. Причем для юридических лиц длина ИНН равняется 10, а для индивидуальных предпринимателей - 12. Длина КПП всегда равняется 9. Если данные введены неверно, объект не записывается и пользователю выводятся соответствующие сообщения:

* "Ошибка: КПП должен состоять из 9 цифр."
* "Ошибка: ИНН должен состоять из 10 цифр."
* "Ошибка: ИНН должен состоять из 12 цифр."

Требование REQ\_CTLG\_K\_009

Форма справочника Контрагенты состоит из следующих элементов:

* Код - уникальный номер элемента, заполняется автоматически.
* Наименование - имя элемента. Пользователь вводит самостоятельно.
* Вид контрагента- значение реквизита “Вид”. Пользователь может выбрать вид контрагента из выпадающего списка.
* ИНН - значение реквизита “ИНН”. Пользователь вводит самостоятельно.
* КПП- значение реквизита “КПП”. Пользователь вводит самостоятельно.
* Табличная часть
* № - порядковый номер строки табличной части. Заполняется автоматически.
* Вид контактной информации - значение реквизита “ВидКонтактнойИнформации”. Пользователь вводит самостоятельно.
* Значение - значение реквизита “Информация”. Пользователь вводит самостоятельно.

Пользователь может сохранить изменения, нажав кнопку “Записать и закрыть” или кнопку “Записать”.

Форма справочника Контрагенты представлена на рисунке 2.

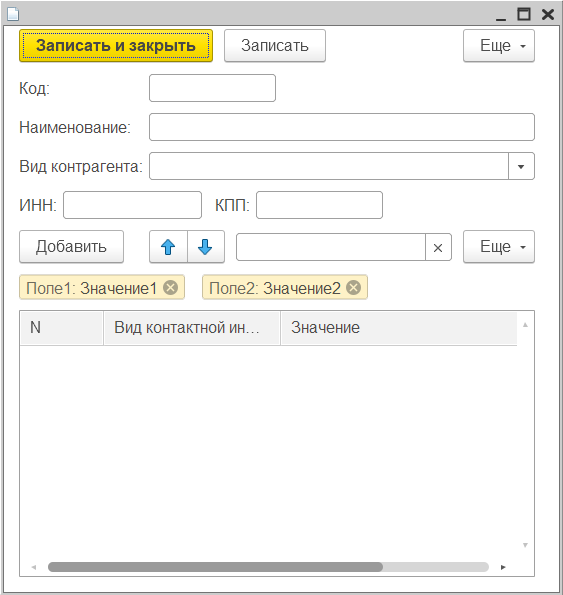


Рисунок 2. Форма справочника Контрагенты

Требование REQ\_CTLG\_FL\_001

Имя: ФизическиеЛица

Синоним: Физические лица

Требование REQ\_CTLG\_FL\_002

Справочник ФизическиеЛица должен быть не иерархическим.

Требование REQ\_CTLG\_FL\_003

Реквизиты:

* Пол
* Синоним: Пол
* Тип: ПеречислениеСсылка.Пол
* Проверка заполнения: Не проверять
* Фамилия
* Синоним: Фамилия
* Тип: Строка не более 20 символов
* Проверка заполнения: Не проверять
* Имя
* Синоним: Фамилия
* Тип: Строка не более 20 символов
* Проверка заполнения: Не проверять
* Отчество
* Синоним: Фамилия
* Тип: Строка не более 20 символов
* Проверка заполнения: Не проверять

Требование REQ\_CTLG\_FL\_004

Все элементы автонумеруются с контролем уникальности, серии кодов для всего справочника.

Требование REQ\_CTLG\_FL\_005

Длина кода: 9.

Тип кода: Строка.

Длина наименования: 100.

Основное представление: В виде наименования.

Требование REQ\_CTLG\_FL\_006

Форма справочника ФизическиеЛица состоит из следующих элементов:

* Код - уникальный номер элемента, заполняется автоматически.
* Наименование - имя элемента. Заполняется автоматически.
* Фамилия - значение реквизита “Фамилия”. Пользователь вводит самостоятельно.
* Имя - значение реквизита “Имя”. Пользователь вводит самостоятельно.
* Отчество - значение реквизита “Отчество”. Пользователь вводит самостоятельно.
* Пол - значение реквизита “Пол”. Пользователь может выбрать из выпадающего списка.

Пользователь может сохранить изменения, нажав кнопку “Записать и закрыть” или кнопку “Записать”.

Форма справочника ФизическиеЛица представлена на рисунке 3.



Рисунок 3. Форма справочника ФизическиеЛица

Требование REQ\_CTLG\_FL\_007

Поле “Наименование” формы справочника ФизическиеЛица должно быть неизменяемым и должно составляться из заполненных пользователем полей “Фамилия”, “Имя” и “Отчество”.

Требование REQ\_CTLG\_S\_001

Имя: Склады

Синоним: Склады

Требование REQ\_CTLG\_S\_002

Справочник Склады должен быть не иерархическим.

Требование REQ\_CTLG\_S\_003

Реквизиты:

* ОтветственноеЛицо
* Синоним: Ответственное лицо
* Тип: ПеречислениеСсылка.Контрагенты, ПеречислениеСсылка.ФизическиеЛица
* Проверка заполнения: Выдавать ошибку
* ВидСклада
* Синоним: Вид склада
* Тип: ПеречислениеСсылка.ВидыСкладов,
* Проверка заполнения: Не проверять

Требование REQ\_CTLG\_S\_004

Все элементы автонумеруются с контролем уникальности, серии кодов для всего справочника.

Требование REQ\_CTLG\_S\_005

Длина кода: 9.

Тип кода: Строка.

Длина наименования: 25.

Основное представление: В виде наименования.

Требование REQ\_CTLG\_S\_006

При копировании элемента реквизит ОтветственноеЛицо должен очищаться.

Требование REQ\_CTLG\_S\_007

Если ответственным лицом назначен Контрагент, то вид склада должен быть Арендуемый. Если выбрано другое значение для вида склада, объект не записывается и пользователю выводится сообщение: "Ошибка: Вид склада для ответственных лиц-контрагентов должен быть ""Арендуемый"".

Если ответственным лицом назначено физическое лицо, то вид склада должен быть Собственный. Если выбрано другое значение для вида склада, объект не записывается и пользователю выводится сообщение: "Ошибка: Вид склада для офизических ответственных лиц должен быть ""Собственный"".

Требование REQ\_CTLG\_S\_008

Форма справочника Склады состоит из следующих элементов:

* Код - уникальный номер элемента, заполняется автоматически.
* Наименование - имя элемента. Пользователь вводит самостоятельно.
* Ответственное лицо - значение реквизита “ОтветственноеЛицо”. Пользователь может выбрать из выпадающего списка.
* Вид склада - значение реквизита “ВидСклада”. Пользователь может выбрать из выпадающего списка.

Пользователь может сохранить изменения, нажав кнопку “Записать и закрыть” или кнопку “Записать”.

Форма справочника Склады представлена на рисунке 4 .

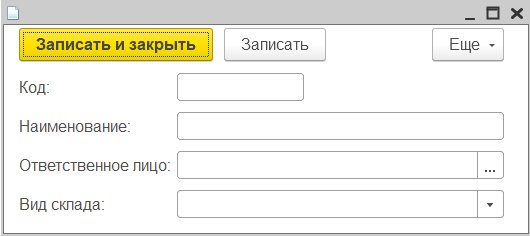


Рисунок 4. Форма справочника Склады

## **3.3 Требования к подсистеме Документы**

Требование REQ\_DOC\_POST\_001

Имя: ПоступлениеТоваровИУслуг

Синоним: Поступление товаров и услуг

Требование REQ\_DOC\_POST\_002

Реквизиты:

* Контрагент
* Синоним: Контрагент
* Тип: СправочникСсылка.Контрагенты
* Проверка заполнения: Выдавать ошибку
* Склад
* Синоним: Склад
* Тип: СправочникСсылка.Склад
* Проверка заполнения: Не проверять
* Комментарий
* Синоним: Комментарий
* Тип: Строка неограниченной длины
* Проверка заполнения: Не проверять

Требование REQ\_DOC\_POST\_003

Табличная часть Товары:

* Синоним: Товары
* Реквизиты:
* Номенклатура
* Синоним: Номенклатура
* Тип: СправочникСсылка.Номенклатура
* Проверка заполнения: Выдавать ошибку
* Количество
* Синоним: Количество
* Тип: Неотрицательное число длины 15 с точностью 3
* Проверка заполнения: Выдавать ошибку
* Цена
* Синоним: Цена
* Тип: Неотрицательное число длины 15 с точностью 2
* Проверка заполнения: Не проверять
* Сумма
* Синоним: Сумма
* Тип: Неотрицательное число длины 15 с точностью 3
* Проверка заполнения: Не проверять
* Содержание
* Синоним: Содержание
* Тип: Строка не более 50 символов
* Проверка заполнения: Не проверять

Требование REQ\_DOC\_POST\_004

Все элементы автонумеруются с контролем уникальности.

Требование REQ\_DOC\_POST\_005

Длина номера: 9.

Тип номера: Строка.

Периодичность: В пределах года.

Требование REQ\_DOC\_POST\_006

Форма документа ПоступлениеТоваровИУслуг состоит из следующих элементов:

* Номер - уникальный номер элемента, заполняется автоматически.
* Дата - день, месяц, год и точное время создания документа, заполняется автоматически.
* Контрагент - значение реквизита “Контрагент”. Пользователь может выбрать из выпадающего списка.
* Склад - значение реквизита “Склад”. Пользователь может выбрать из выпадающего списка.
* Комментарий - значение реквизита “Комментарий”, находится на отдельной вкладке. Пользователь вводит самостоятельно.
* Табличная часть Товары
* № - порядковый номер строки табличной части. Заполняется автоматически.
* Номенклатура - значение реквизита “Номенклатура”. Пользователь может выбрать из выпадающего списка.
* Количество - значение реквизита “Количество”. Пользователь вводит самостоятельно.
* Цена - значение реквизита “Цена”. Пользователь вводит самостоятельно.
* Сумма - значение реквизита “Сумма”. Заполняется автоматически при введенных полях “Цена” и Количество”.

Пользователь может сохранить изменения, нажав кнопку “Провести и закрыть” или кнопку “Записать”.

Форма документа ПоступлениеТоваровИУслуг представлена на рисунке 5.

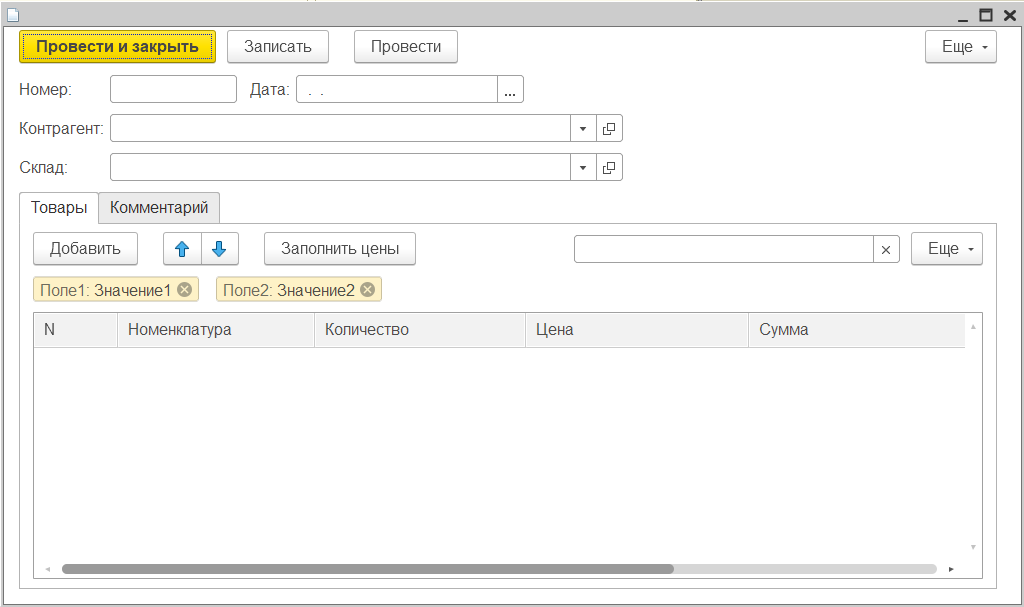


Рисунок 5. Форма документа ПоступлениеТоваровИУслуг

Требование REQ\_DOC\_POST\_007

При открытии формы система должна автоматически заполнять поле “Склад” тем значением, которое было указано в последнем проведенном документе. Пользователь может изменить это поле самостоятельно, выбрав одно из значений из выпадающего списка.

Требование REQ\_DOC\_POST\_008

Система должна проверять поле “Склад” на заполненность во всех случаях, кроме тех, когда в поле “Номенклатура” указана номенклатура вида “Услуга”. В таком случае система должна выдать сообщение об ошибке: “Не указан реквизит Склад”.

Требование REQ\_DOC\_POST\_009

Если в поле “Номенклатура” указана номенклатура вида “Услуга”, и при этом поле “Содержание” не заполнено, то система должна выдать сообщение об ошибке: “Для номенклатуры [наименование номенклатуры] ([номер строки] строка) не заполнено Содержание”.

Требование REQ\_DOC\_PROD\_001

Имя: ПродажаТоваровИУслуг

Синоним: Продажа товаров и услуг

Требование REQ\_DOC\_PROD\_002

Реквизиты:

* Контрагент
* Синоним: Контрагент
* Тип: СправочникСсылка.Контрагенты
* Проверка заполнения: Выдавать ошибку
* Склад
* Синоним: Склад
* Тип: СправочникСсылка.Склад
* Проверка заполнения: Не проверять
* Комментарий
* Синоним: Комментарий
* Тип: Строка неограниченной длины
* Проверка заполнения: Не проверять

Требование REQ\_DOC\_PROD\_003

Табличная часть Товары:

* Синоним: Товары
* Реквизиты:
* Номенклатура
* Синоним: Номенклатура
* Тип: СправочникСсылка.Номенклатура
* Проверка заполнения: Выдавать ошибку
* Количество
* Синоним: Количество
* Тип: Неотрицательное число длины 15 с точностью 3
* Проверка заполнения: Выдавать ошибку
* Цена
* Синоним: Цена
* Тип: Неотрицательное число длины 15 с точностью 2
* Проверка заполнения: Не проверять
* Сумма
* Синоним: Сумма
* Тип: Неотрицательное число длины 15 с точностью 3
* Проверка заполнения: Не проверять
* Содержание
* Синоним: Содержание
* Тип: Строка не более 50 символов
* Проверка заполнения: Не проверять

Требование REQ\_DOC\_PROD\_004

Все элементы автонумеруются с контролем уникальности.

Требование REQ\_DOC\_PROD\_005

Длина номера: 9.

Тип номера: Строка.

Периодичность: В пределах года.

Требование REQ\_DOC\_PROD\_006

Форма документа ПродажаТоваровИУслуг состоит из следующих элементов:

* Номер - уникальный номер элемента, заполняется автоматически.
* Дата - день, месяц, год и точное время создания документа, заполняется автоматически.
* Контрагент - значение реквизита “Контрагент”. Пользователь может выбрать из выпадающего списка.
* Склад - значение реквизита “Склад”. Пользователь может выбрать из выпадающего списка.
* Комментарий - значение реквизита “Комментарий”, находится на отдельной вкладке. Пользователь вводит самостоятельно.
* Табличная часть Товары
* № - порядковый номер строки табличной части. Заполняется автоматически.
* Номенклатура - значение реквизита “Номенклатура”. Пользователь может выбрать из выпадающего списка.
* Количество - значение реквизита “Количество”. Пользователь вводит самостоятельно.
* Цена - значение реквизита “Цена”. Пользователь вводит самостоятельно.
* Сумма - значение реквизита “Сумма”. Заполняется автоматически при введенных полях “Цена” и Количество”.

Пользователь может сохранить изменения, нажав кнопку “Провести и закрыть” или кнопку “Записать”.

Форма документа ПродажаТоваровИУслуг представлена на рисунке 6.

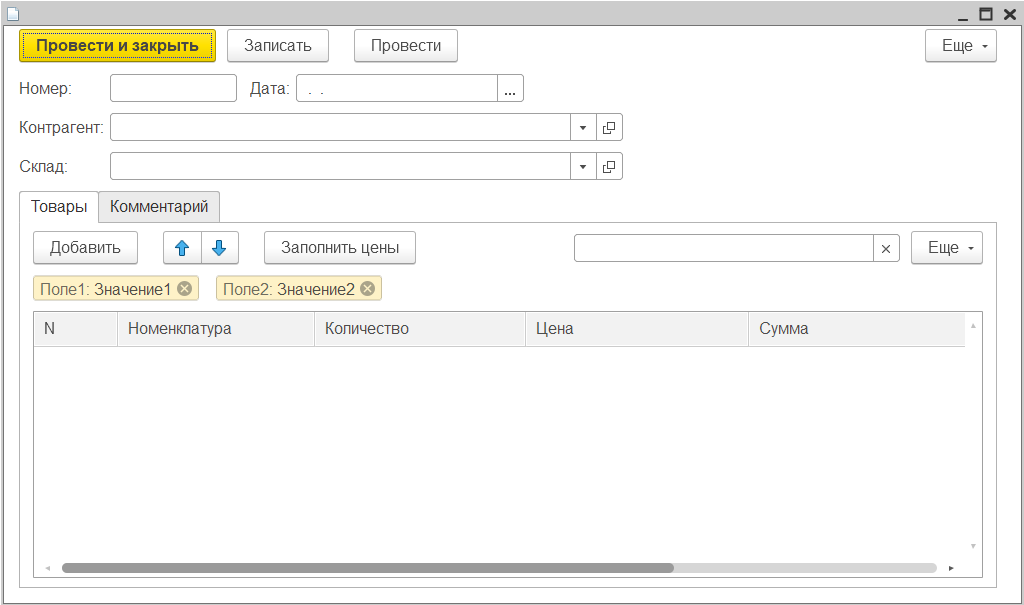


Рисунок 6. Форма документа ПродажаТоваровИУслуг

Требование REQ\_DOC\_PROD\_007

При открытии формы система должна автоматически заполнять поле “Склад” тем значением, которое было указано в последнем проведенном документе. Пользователь может изменить это поле самостоятельно, выбрав одно из значений из выпадающего списка.

Требование REQ\_DOC\_PROD\_008

Система должна проверять поле “Склад” на заполненность во всех случаях, кроме тех, когда в поле “Номенклатура” указана номенклатура вида “Услуга”. В таком случае система должна выдать сообщение об ошибке: “Не указан реквизит Склад”.

Требование REQ\_DOC\_PROD\_009

Если в поле “Номенклатура” указана номенклатура вида “Услуга”, и при этом поле “Содержание” не заполнено, то система должна выдать сообщение об ошибке: “Для номенклатуры [наименование номенклатуры] ([номер строки] строка) не заполнено Содержание”.

Требование REQ\_DOC\_PROD\_010

Система должна проверять, есть ли на указанном складе необходимое количество указанного товара. Если на указанном складе нет указанного товара в достаточном количестве, то система должна выдавать сообщение об ошибке: “Товара [наименование товара] осталось на складе [наименование склада] всего [количество] штук”.

Требование REQ\_DOC\_UST\_001

Имя: УстановкаЦен

Синоним: Установка цен

Требование REQ\_DOC\_UST\_002

Табличная часть Цены:

* Синоним: Цены
* Реквизиты:
* Номенклатура
* Синоним: Номенклатура
* Тип: СправочникСсылка.Номенклатура
* Проверка заполнения: Не проверять
* Цена
* Синоним: Цена
* Тип: Неотрицательное число длины 15 с точностью 2
* Проверка заполнения: Выдавать ошибку

Требование REQ\_DOC\_UST\_003

Все элементы автонумеруются с контролем уникальности.

Требование REQ\_DOC\_UST\_004

Длина номера: 9.

Тип номера: Строка.

Периодичность: Непериодический.

Требование REQ\_DOC\_UST\_005

Форма документа УстановкаЦен состоит из следующих элементов:

* Номер - уникальный номер элемента, заполняется автоматически.
* Дата - день, месяц, год и точное время создания документа, заполняется автоматически.
* Табличная часть
* № - порядковый номер строки табличной части. Заполняется автоматически.
* Номенклатура - значение реквизита “Номенклатура”. Пользователь может выбрать из выпадающего списка.
* Цена - значение реквизита “Цена”. Пользователь вводит самостоятельно.

Пользователь может сохранить изменения, нажав кнопку “Провести и закрыть” или кнопку “Записать”.

Форма документа УстановкаЦен представлена на рисунке 7.

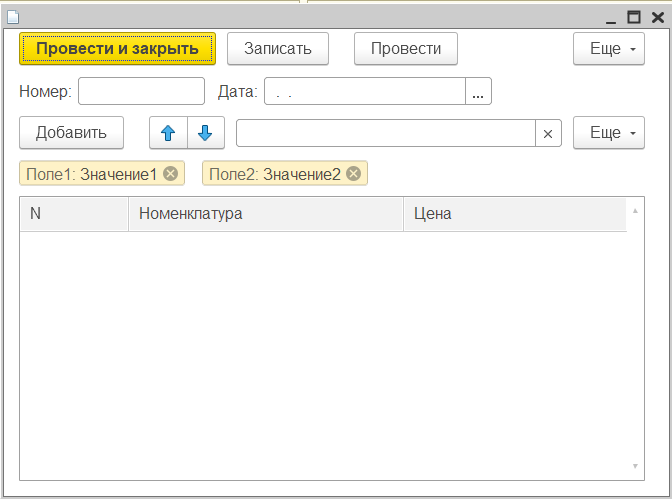


Рисунок 7. Форма документа УстановкаЦен

## **3.4 Требования к подсистеме Регистры накопления**

Требование REQ\_REG\_OST\_001

Имя: ТоварыНаСкладах

Синоним: Товары на складах

Вид регистра: Остатки

Регистр должен накапливать информацию об остатках товаров на складах. Информация поступает при проведении документов ПоступлениеТоваровИУслуг и ПродажаТоваровИУслуг.

Требование REQ\_REG\_OST\_002

Измерения:

* Склад
* Синоним: Склад
* Тип: СправочникСсылка.Склады
* Номенклатура
* Синоним: Номенклатура
* Тип: СправочникСсылка.Номенклатура

Требование REQ\_REG\_OST\_003

Ресурсы:

* Количество
* Синоним: Количество
* Тип: Число длины 15 с точностью 3

Требование REQ\_REG\_OST\_004

Регистраторы:

* ПоступлениеТоваровИУслуг
* ПродажаТоваровИУслуг

## **3.5 Требования к подсистеме Регистры сведений**

Требование REQ\_REG\_C\_001

Имя: Цены

Синоним: Цены

Периодичность: В пределах дня

Режим записи: Подчинение регистратору

Регистр должен хранить информацию о ценах на товары. Информация поступает при проведении документа УстановкаЦен.

Требование REQ\_REG\_C\_002

Измерения:

* Номенклатура
* Синоним: Номенклатура
* Тип: СправочникСсылка.Номенклатура
* Ведущее измерение

Требование REQ\_REG\_C\_003

Ресурсы:

* Цена
* Синоним: Цена
* Тип: Неотрицательное число длины 15 с точностью 2

Требование REQ\_REG\_C\_004

Регистраторы:

* УстановкаЦен

Требование REQ\_REG\_SH\_001

Имя: Штрихкоды

Синоним: Штрихкоды

Периодичность: Непериодический

Режим записи: Независимый

Регистр должен хранить информацию о штрихкодах номенклатуры.У одного объекта номенклатуры может быть несколько штрихкодов, но один штрихкод может относиться только к одному объекту номенклатуры..

Требование REQ\_REG\_SH\_002

Измерения:

* Штрихкод
* Синоним: Номенклатура
* Тип: Строка длины 13

Требование REQ\_REG\_SH\_003

Ресурсы:

* Номенклатура
* Синоним: Номенклатура
* Тип: СправочникСсылка.Номенклатура

# **4 Дизайн продукта**

Программный продукт Автоматизация учета товаров предназначен для систематизации и автоматизации учета товаров.

Настоящий документ содержит спецификации дизайна, включающие в себя:

* Архитектурно-контекстную диаграмму системы в целом
* Диаграмму связей объектов

## **4.1 Архитектурно-контекстная диаграмма системы**

Программный продукт название состоит из следующих подсистем:

1. Справочники
2. Документы
3. Перечисления
4. Регистры накопления
5. Решистры сведений

Архитектурно-контекстная диаграмма системы представлена на Рисунке 1.

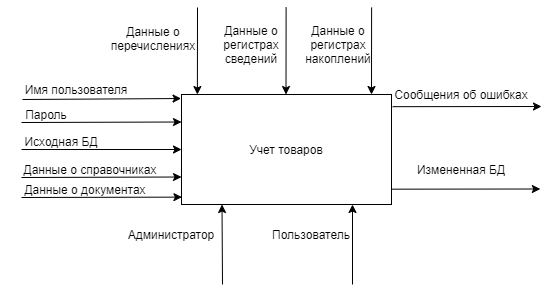


Рисунок 1 - Контекстная диаграмма системы

## **4.2 Диаграмма связей объектов**

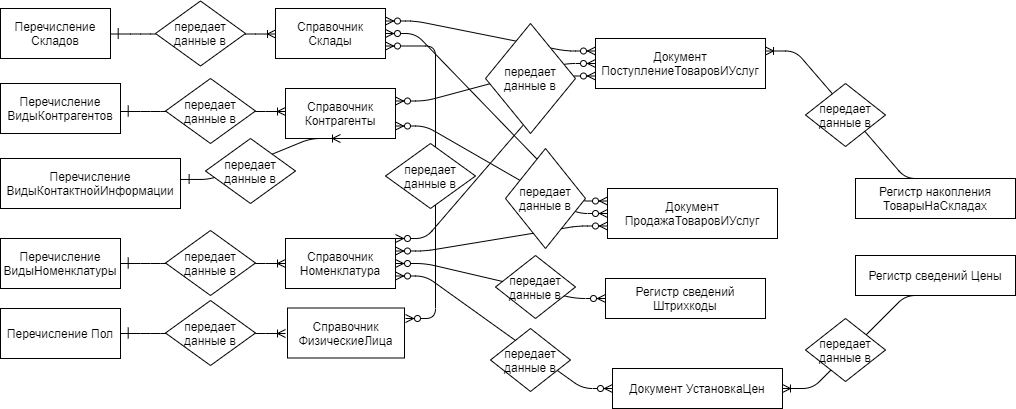
****

Рисунок 2 - Диаграмма связей объектов

# **5 Система измерений проекта**

Система метрик для оценки эффективности процесса разработки:

1. *Phase Containment Effectiveness* (PCE)   
   PCE = [D(I, I) / Sum(D(I, J))]\*100%, где I, J =1,2,..,5, D(I,J) – число ошибок внесённых на фазе I и обнаруженных на фазе J. 1,…,5 – номера фаз проекта (1 – планирование, 2 – разработка требований, 3 – разработка дизайна, 4 – кодирование, 5 – тестирование). Стратегическая цель метрики – повысить качество разрабатываемого ПО, сократить сроки выполнения проектов по разработке ПО. Изучаемый объект метрики – фаза проекта, измеряемый атрибут – эффективность обнаружения ошибок, внесённых на этой фазе. Единица измерения – %.
2. *Problem Resolution Rate* (PRR) PRR = Количество дней на обработку задачи; Стратегическая цель метрики – сократить сроки выполнения проектов по разработке ПО. Изучаемый объект метрики – задача, измеряемый атрибут – время обработки. Единица измерения – день.
3. *On project % time* (OPPT) OPPT = (Рабочее время, затраченное на проект / Общее рабочее время)\*100% Стратегическая цель метрики – сокращение сроков выполнения проектов по разработке ПО. Изучаемый объект метрики – рабочее время, измеряемый атрибут – доля рабочего времени, затраченная на работу по проектам. Единица измерения - %.

Система метрик для оценки качества программного продукта:

1. *In Process Faults* (IPF) IPF = (Число обнаруженных ошибок до выпуска его релиза) / LOC; Стратегическая цель метрики – повысить качество разрабатываемого ПО. Изучаемый объект метрики – продукт, измеряемый атрибут – плотность неполадок. Единица измерения – неполадка / LOC.
2. *Product Fault Density* (PFD) PFD-Req = Число ошибок, допущенных на этапе разработки требований / Размер требований; PFD-Design = Число ошибок, допущенных на этапе дизайна / Размер документов дизайна; PFD-Coding = Число ошибок, допущенных на этапе кодирования / LOC; PFD-Test = Число ошибок, допущенных на этапе тестирования / Количество тестов; Стратегическая цель метрики – повысить качество разрабатываемого ПО. Изучаемый объект метрики – продукт, измеряемый атрибут – плотность неполадок (учитывает все ошибки). Единица измерения – неполадка / единица размера.

# **6 Задачи проекта**

Программный продукт состоит из следующих подсистем:

1. Перечисления
2. Справочники
3. Документы
4. Регистры накопления
5. Регистры сведений

## **6.1 Задачи для подсистемы Перечисления**

Задача TASK\_ENUM\_001

Создать объекты перечислений

Создать объекты ВидыНоменклатуры, Пол, ВидыКонтрагентов, ВидыСкладов и ВидыКонтактнойИнформации. Заполнить перечисления значениями согласно требованиям REQ\_ENUM.

## **6.2 Задачи для подсистемы Справочники**

Задача TASK\_CTLG\_001

Создать справочник Номенклатура

Создать объект справочник Номенклатура согласно требованиям REQ\_CTLG\_N. В модуле объекта реализовать процедуру ПередЗаписью, в которой выполняется проверка, описанная в требовании REQ\_CTLG\_N\_006. Создать форму элемента справочника.

Задача TASK\_CTLG\_002

Создать справочник ФизическиеЛица

Создать объект справочник Номенклатура согласно требованиям REQ\_CTLG\_FL. В модуле объекта реализовать процедуру ПередЗаписью, в которой введенный пользователем реквизит Наименование разбивается на реквизиты Фамилия, Имя, Отчество. Создать форму элемента справочника.

Задача TASK\_CTLG\_003

Создать справочник Контрагенты

Создать объект справочник Контрагенты согласно требованиям REQ\_CTLG\_K. В модуле объекта реализовать процедуру ПередЗаписью, в которой выполняются проверки на ограничения, описанные в требованиях REQ\_CTLG\_K\_007 и REQ\_CTLG\_K\_008. Создать форму элемента справочника.

Задача TASK\_CTLG\_004

Создать справочник Склады

Создать объект справочник Склады согласно требованиям REQ\_CTLG\_S. В модуле объекта реализовать процедуру ПередЗаписью, в которой выполняется проверка, описанная в требовании REQ\_CTLG\_S\_007. В модуле объекта реализовать процедуру ПриКопировании, в которой очищается реквизит ОтветственноеЛицо (см. REQ\_CTLG\_S\_006). Создать форму элемента справочника.

## **6.3 Задачи для подсистемы Документы**

Задача TASK\_DOC\_001

Создать документ ПоступлениеТоваровИУслуг

Создать объект документ ПоступлениеТоваровИУслуг согласно требованиям REQ\_DOC\_POST. В модуле объекта реализовать процедуру ОбработкаЗаполнения, в которой выполняется заполнение поля формы документа, описанное в требовании REQ\_DOC\_POST\_007. В модуле объекта реализовать процедуру ОбработкаПроверкиЗаполнения, в которой выполняются проверки, описанные в требованиях REQ\_DOC\_POST\_008 и REQ\_DOC\_POST\_009.

Задача TASK\_DOC\_002

Создать документ ПродажаТоваровИУслуг

Создать объект документ ПродажаТоваровИУслуг согласно требованиям REQ\_DOC\_PROD. В модуле объекта реализовать процедуру ОбработкаЗаполнения, в которой выполняется заполнение поля формы документа, описанное в требовании REQ\_DOC\_PROD\_007. В модуле объекта реализовать процедуру ОбработкаПроверкиЗаполнения, в которой выполняются проверки, описанные в требованиях REQ\_DOC\_PROD\_008 и REQ\_DOC\_POST\_009. В модуле объекта реализовать процедуру ОбработкаПроведения, в которой выполняется проверка, описанная в требовании REQ\_DOC\_PROD\_010.

## **6.4 Задачи для подсистемы Регистры накопления**

Задача TASK\_REGN\_001

Создать регистр ТоварыНаСкладах

Создать регистр накопления ТоварыНаСкладах согласно требованиям REQ\_REG\_OST.

## **6.5 Задачи для подсистемы Регистры сведений**

Задача TASK\_REGS\_001

Создать регистр Цены

Создать регистр накопления Цены согласно требованиям REQ\_REG\_C.

Задача TASK\_REGS\_002

Создать регистр Штрихкоды

Создать регистр накопления Штрихкоды согласно требованиям REQ\_REG\_SH.

# **7 Рекомендации по кодированию**

Правила оформления текстов модулей:

Требование

Имена переменных следует образовывать от терминов предметной области таким образом, чтобы из имени переменной было понятно ее назначени.

Правильно: ПоказанияСчетчика

Неправильно: ПС, Показ, П

Требование

Имена необходимо образовывать путем удаления пробелов между словами. При этом, каждое слово в имени пишется с прописной буквы. Предлоги и местоимения из одной буквы также пишутся прописными буквами.

Например,

Перем ДиалогРаботыСКаталогом; // Диалог работы с каталогом

Запрет

Имена переменных запрещается начинать с подчеркивания.

Требование

Тексты модулей должны быть написаны на русском языке.

Рекомендация

При длине строки более 120 символов следует использовать переносы.

Требование

Имена объектов и их реквизитов должны начинаться с заглавной буквы и продолжаться строчными. Если идентификатор состоит из нескольких слов, то каждое слово пишется так же с большой буквы без каких-либо разделителей.

Правильно: ПоступлениеТоваровИУслуг

Неправильно: поступление\_товаро\_и\_услуг

Требование

Программные модули не должны иметь закомментированных фрагментов кода, а также фрагментов, которые каким-либо образом связаны с процессом разработки (отладочный код, служебные отметки, например, TODO, MRG и т.п.) и с конкретными разработчиками этого кода.

Требование

Тексты модулей оформляются по принципу "один оператор в одной строке".

Правильно:

Отказ = Истина;

Сообщение = Новый СообщениеПользователю();

Неправильно:

Отказ = Истина; Сообщение = Новый СообщениеПользователю();

Рекомендация

Небольшие комментарии пишутся в конце строки, которую комментируют, например:

НайденныеОшибки.Колонки.Добавить("Номер"); // для совместимости

Рекомендация

Большие комментарии или комментарии к фрагменту кода пишутся перед комментируемым кодом в отдельной строке. Текст выравнивается по левой границе комментируемого фрагмента. Между символами комментария "//" и текстом комментария должен быть пробел.

Запрет

В текстах модулях не допускается использовать букву "ё". Исключения составляют интерфейсные тексты, выводимые пользователю в сообщениях, формах и справке, где употребление буквы «ё» в ряде случаев допустимо.

Запрет

В текстах модулей не допускается использовать неразрывные пробелы и знак минус "-" в других кодировках (короткое, длинное тире и т.п.).

Такие символы часто оказываются в тексте модулей при копировании из офисных документов и приводят к ряду сложностей при разработке. Например, не работает поиск фрагментов текста, включающих «неправильные» минусы и пробелы; некорректно выводятся подсказки типов параметров процедур и функций в конфигураторе и расширенная проверка в 1С:EDT; указание «неправильного» минуса в выражениях приведет к синтаксической ошибке.

Рекомендация

Программные модули не должны иметь неиспользуемых процедур и функций.

# **8 История изменений документа**

|  |  |  |
| --- | --- | --- |
| Дата | Автор | Внесенные изменения |
| 02.10.2020 | Скопина, Магалиш, Ковальская | Разработан регламент инспекции рабочих продуктов |
| 07.10.2020 | Скопина, Магалиш, Ковальская | Модель состояний задач |
| 10.11.2020 | Скопина, Магалиш | Исходная версия требований |
| 11.11.2020 | Магалиш | Добавлены требования к подсистеме Справочники |
| 13.11.2020 | Ковальская | Добавлены требования к подсистеме Документы |
| 15.11.2020 | Скопина | Добавлены требования к подсистеме Регистры накопления и Регистры сведений |
| 10.12.2020 | Магалиш | Система измерений проекта |
| 20.12.2020 | Ковальская | Дизайн продукта |
| 6.01.2021 | Скопина | Исходная версия задач проекта |
| 8.01.2021 | Ковальская | Добавлены задачи для подсистемы документы |
| 10.01.2021 | Скопина | Рекомендации по кодированию |