

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**  
**Федеральное государственное бюджетное образовательное учреждение высшего образования**  
**«Кузбасский государственный технический университет имени Т. Ф. Горбачева»**

Кафедра информационных и автоматизированных производственных систем

Составитель  
О. Н. Ванеев

## **ПРОЕКТИРОВАНИЕ И ДИЗАЙН ИНФОРМАЦИОННЫХ СИСТЕМ**

**Методические материалы**

Рекомендовано цикловой методической комиссией специальности  
СПО 09.02.07 Информационные системы и программирование  
в качестве электронного издания  
для использования в образовательном процессе

Кемерово 2018

Рецензенты:

Сыркин И. С. – кандидат технических наук, доцент кафедры информационных и автоматизированных производственных систем ФГБОУ ВО «Кузбасский государственный технический университет имени Т. Ф. Горбачева»

Чичерин И. В. – кандидат технических наук, доцент, зав. кафедрой информационных и автоматизированных производственных систем ФГБОУ ВО «Кузбасский государственный технический университет имени Т. Ф. Горбачева»

**Ванеев Олег Николаевич**

**Проектирование и дизайн информационных систем:** методические материалы [Электронный ресурс] для студентов специальности СПО 09.02.07 Информационные системы и программирование очной формы обучения / сост. О. Н. Ванеев; КузГТУ. – Электрон. издан. – Кемерово, 2018.

Методические материалы для дисциплины «Проектирование и дизайн информационных систем» описывают содержание практических, лабораторных и самостоятельных занятий; перечень вопросов на защиту выполненных работ.

© КузГТУ, 2018

© Ванеев О. Н.,  
составление, 2018

# РАБОТА №1 ПРАКТИЧЕСКАЯ. ПОСТРОЕНИЕ И АНАЛИЗ МОДЕЛИ ПРЕДМЕТНОЙ ОБЛАСТИ НА ОСНОВЕ МЕТОДОЛОГИИ SADT

## 1. ЦЕЛЬ РАБОТЫ

Цель работы – закрепление теоретических знаний студентов по работе в среде ERWin Process Modeller и получение навыков разработки учебных моделей при помощи одной из методологий структурного анализа и моделирования систем – IDEF0.

## 2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ.

Методология SADT (Structured Analysis and Design) – одна из самых известных методологий анализа и проектирования систем, введенная в 1973 года Россом. Главное отличие методологии SADT от других методов структурного анализа заключается в том, что она отводит важную роль механизму. В SADT система описывается в первую очередь с точки зрения ее функций. При полном SADT-моделировании используются взаимодополняющие модели двух типов:

- **функциональные модели**, выделяющие события (функции, бизнес-процессы) в системе;
- **модели данных**, выделяющие объекты (данные) системы.

В обоих случаях используется один и тот же графический язык блоков и дуг, но блоки и дуги меняются ролями.

Функциональный вариант SADT-методологии разработан в виде стандарта IDEF0 (*Integrated computer aided manufacturing Definition*). Наиболее известной программной средой, обеспечивающей разработку моделей на основе стандарта IDEF0 является система ERwin Process modeler.

Основной структурной единицей IDEF0-модели является диаграмма, представляющая собой графическое описание модели предметной области или ее части. Главными компонентами IDEF0-диаграммы являются **блоки**.

Блоки отображают некоторые работы, функции, процессы, задачи, которые происходят или выполняются в течение определенного времени и имеют некоторые результаты. Блоки изображаются

в виде прямоугольников. Каждая сторона функционального блока имеет назначение (рис. 1.1): левая сторона предназначена для входа, правая – для выхода, верхняя – для управления, нижняя – для механизмов.

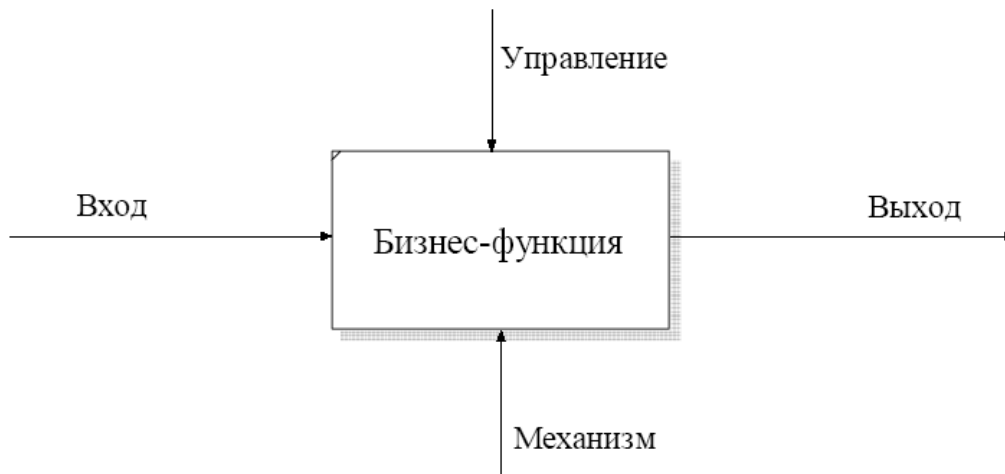


Рис.1.1. Основная конструкция IDEF0-модели

Взаимодействие функций с внешним миром и между собой описывается с помощью дуг (связей), представляемых на диаграммах в виде линий со стрелками (Arrow).

*Стрелки (связи) соответствуют материальным и информационным потокам.*

В IDEF0 различают пять типов дуг:

1. **Вход (Input)** – материал или информация, которые используются или преобразуются блоком для получения результата (выхода). Блок может не иметь ни одной входной дуги. Данный вид дуги поступает на левую сторону блока.

2. **Управление (Control)** – условия, правила, стратегии, стандарты, которые влияют на выполнение функции. Каждый блок должен иметь хотя бы одну дугу управления. Данный вид дуг поступает на верхнюю сторону блока.

3. **Выход (Output)** – результат выполнения функции (материал или информация). Каждая функция должна иметь хотя бы одну выходную дугу. Данный вид дуг выходит из правой стороны блока.

4. **Механизм (Mechanism)** – ресурсы, с помощью которых выполняется работа. Это могут быть, например, денежные средства, персонал предприятия, станки. Данный вид дуг поступает на нижнюю сторону блока.

5. **Вызов (Call)** – специальная дуга, указывающая на другую модель предметной области. Данный вид дуги выходит из нижней стороны блока. Дуга вызова не является компонентом собственно методологии SADT. Она является расширением IDEF0-методологии и предназначена для организации коллективной работы над моделью, разделения модели на независимые модели и объединения различных моделей предметной области в одну модель.

### 3. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

В качестве исходной информации к заданию выступает конкретно определенная преподавателем предметная область, для которой создается модель процесса.

Основные шаги:

1. Изучить методику моделирования SADT с использованием ERwin Process Modeler.
2. Создать новую модель.
3. Описать цель, точку зрения предметной области и границы.
4. Создать контекстную диаграмму.
5. Создать диаграмму декомпозиции (не менее 3 блоков).
6. Создать диаграмму декомпозиции для одной из работ декомпозиции первого уровня.
7. Создать диаграмму дерева узлов.
8. Создать FEO-диаграмму.
9. Создать отчёты отображающие:
  - 9.1. Состав процессов.
  - 9.2. Состав стрелок.
  - 9.3. Связь стрелок с процессами.

### 4. ЗАДАНИЕ ДЛЯ ВЫПОЛНЕНИЯ

С помощью CASE-средства ERwin Process Modeller разработать иерархические функциональные модели следующих предметных областей:

1. Деятельность приёмной комиссии (приём документов).
2. Прохождение медосмотра (с точки зрения поликлиники).
3. Устройство на проживание в общежитии.
4. Устройство на работу (с точки зрения устраивающегося).

5. Постановка автомобиля на учёт (снятие авто с учёта).
  6. Получение прав на вождение автомобиля (с точки зрения задействованных субъектов).
  7. Обеспечение проживания в общежитии (со стороны администрации).
  8. Деятельность магазина города.
  9. Организация и проведение свадьбы.
  10. Деятельность ЖКХ.
  11. Деятельность деканата.
  12. Деятельность театра.
  13. Деятельность библиотеки.
  14. Деятельность ИИТМА.
  15. Деятельность студенческой столовой.
  16. Деятельность кинотеатра.
  17. Деятельность проходной.
  18. Деятельность студсовета университета.
- Все процессы должны рассматриваться с точки зрения стороны выполняющей процесс.

## 5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Понятие Case-средств и их назначение.
2. Назначение и сущность методологии IDEF0.
3. Цель модели в IDEF0. «Точка зрения» модели в IDEF0. Субъект моделирования в IDEF0.
4. Виды диаграмм в IDEF0.
5. Назначение видов модели AS-IS или TO-BE в VPwin.
6. Назначения сторон функциональных блоков на IDEF0-диаграмме.
7. Виды стрелок и типы связей.
8. Контекстная диаграмма модели.
9. Назначение кнопок палитры инструментов для IDEF0-методологии в VPwin.
10. Диаграмм декомпозиции.
11. Диаграммы дерева узлов, FEO-диаграммы.
12. Какие существуют типы отчетов в пакете VPwin, для чего каждый из них предназначен? Виды синтаксических ошибок?

## **РАБОТА №2 ПРАКТИЧЕСКАЯ. ПОСТРОЕНИЕ И АНАЛИЗ МОДЕЛИ ПРЕДМЕТНОЙ ОБЛАСТИ НА ОСНОВЕ ООП. ПОСТРОЕНИЕ МОДЕЛИ СОСТАВА ПРОЦЕССОВ И МОДЕЛИ ОРГАНИЗАЦИОННОЙ СТРУКТУРЫ**

### **1. ЦЕЛЬ И ЗАДАЧИ РАБОТЫ**

Целью работы является получение практических навыков разработки модели состава процессов и организационной структуры для предметной области на основе объектно-ориентированного подхода (ООП).

Задачами работы, обеспечивающими выполнение цели, являются:

- изучение принципов и получение практических навыков;
- выявления процессов деятельности предприятия;
- отображение процессов на основе нотации UML;
- выявления организационной структуры предприятия и отображения её средствами языка UML.

### **2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ**

Основная задача построения модели предметной области состоит в выявлении условий, в которых будет работать разрабатываемая система.

Для решения данной задачи необходимо построить совокупность составляющих подмоделей, отображающих различные стороны представления деятельности организации и их уровни детализации. В данной работе предлагается построить следующие составляющие подмодели модели предметной области:

- Модель целей бизнес-процессов – отображает иерархию целей деятельности предприятия.
- Модель организационной структуры – отображает состав организационных подразделений предприятия их взаимосвязь и состав сотрудников.
- Модель состава бизнес-процессов – отображает состав бизнес-процессов их взаимосвязи, бизнес-актёров, связанных с выполнением процессов и цели, поддерживаемые бизнес-процессами.

- Модель содержания бизнес-процессов – содержит описание состава и последовательности выполнения шагов бизнес-процессов.
- Модель бизнес-правил – содержит описание правил выполнения бизнес-процессов.
- Модель объектов бизнес-процессов содержит описание классов объектов бизнес-процессов, их атрибутов.
- Модель используемых типов данных.
- Модель автоматизируемых элементов БП.

Для создания моделей, может быть использовать инструмент [2] Enterprise Architect (EA), который поддерживает универсальный язык моделирования Unified Modeling Language (UML) версии 2.0. Для построения модели создаётся отдельный проект в среде Enterprise Architect.

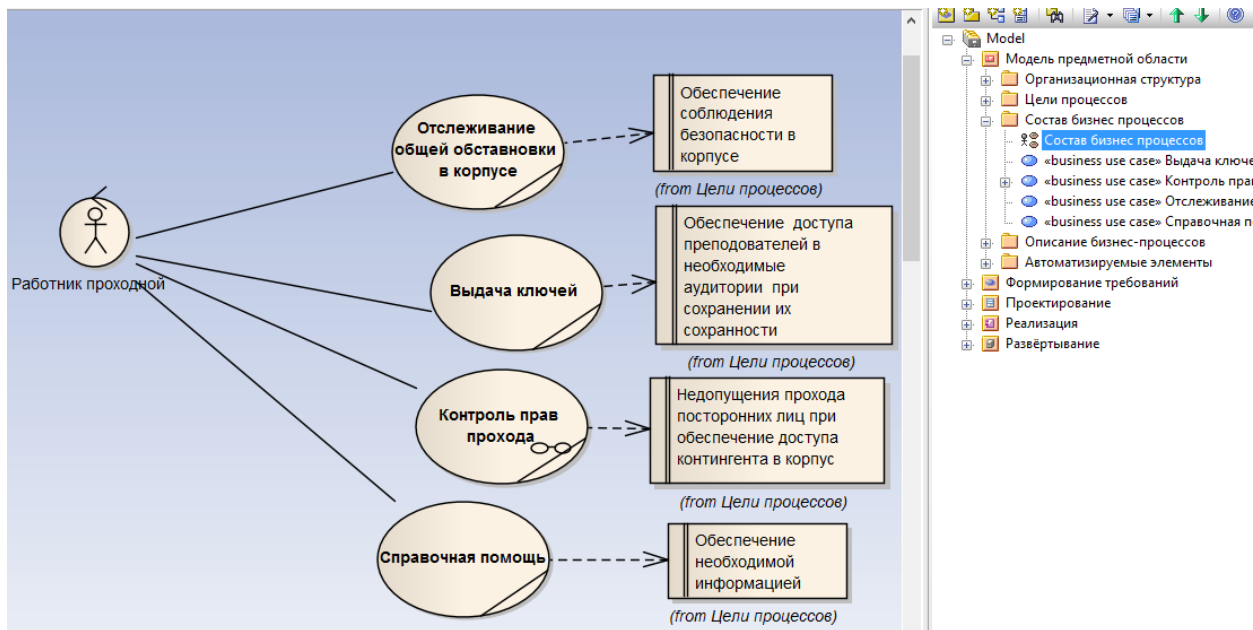


Рис.2.1 Рабочий экран Enterprise Architect  
(панель инструментов, рабочая область, браузер проекта)

Среда EA позволяет разрабатывать модели на основе языка UML. Рабочий экран, как и другие среды моделирования, содержит панель инструментов (обычно слева), браузер проекта, и рабочую область, на которой отображаются диаграммы.

Модель, разработанная в среде EA, включает совокупность вложенных пакетов, в которых содержатся элементы модели. Пакеты верхнего уровня называются областями просмотра модели View. Элементами модели могут быть как структурные элементы UML,



так и диаграммы. Отдельные элементы могут входить в несколько диаграмм. Новые элементы можно создать как через контекстное меню в браузере проекта, тогда они изначально не принадлежат никакой диаграмме, или включая в диаграмму с помощью панели инструментов.

При создании проекта не нужно включать предполагаемые шаблоны составляющих моделей.

Описание предметной области целесообразно локализовать в отдельной области просмотра (View).

В области просмотра «Модель предметной области» следует создать пакеты с наименованиями:

- 01. Цели бизнес-процессов;
- 02. Организационная структура;
- 03. Состав бизнес-процессов;
- 04. Содержание бизнес-процессов;
- 05. Описание объектов бизнес-процессов;
- 06. Описание бизнес-правил;
- 07. Типы данных;
- 08. Автоматизируемые элементы БП.

На рис. 2.2 представлено дерево моделей в браузере ЕА.

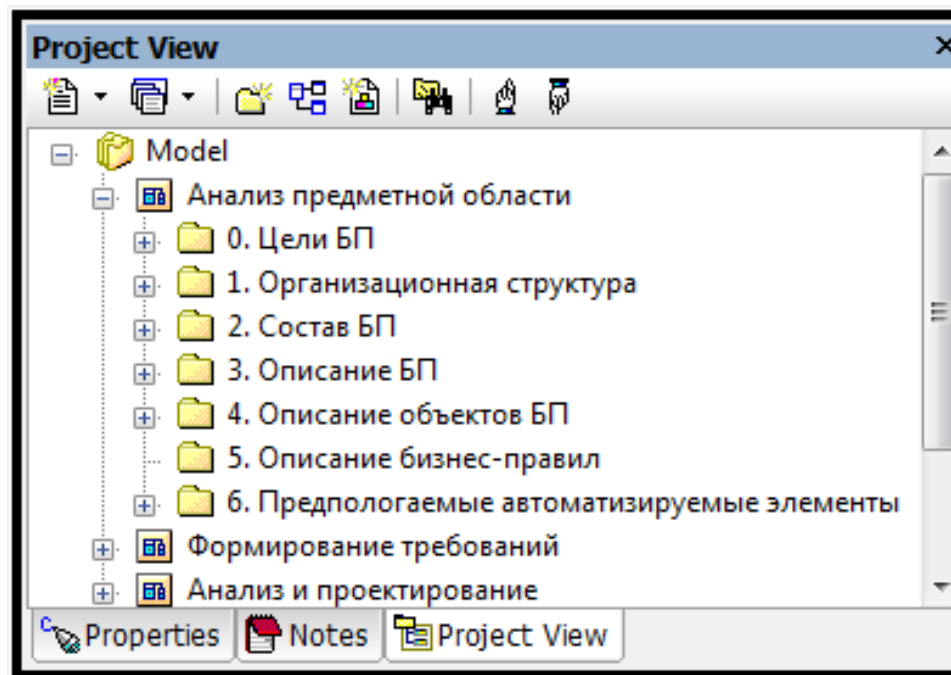


Рис.2.2 Дерево моделей в ЕА

### **Организационная структура**

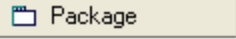
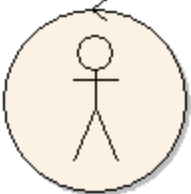
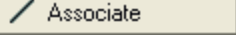
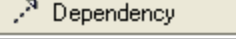
Целью разработки модели «Организационная структура» является отображение бизнес-актёров, задействованных в анализируемом процессе или связанных с ним, состава подразделений, связей между подразделениями и отдельными должностными лицами, и их подчиненность.

Для разработки модели «Организационная структура» может быть использована пакетная диаграмма (UML structural/package) может быть так же использована пользовательская диаграмма (Extended/custom).

Элементы, настраиваемой пользователем диаграммы (custom diagram), используемые для разработки модели, представлены в табл. 2.1.

Элементы настраиваемой пользователем диаграммы (custom diagram), используемые для разработки модели «Организационная структура»

Таблица 2.1

Изображение элемента	Название и назначение элемента
	Пакет. Используется для отображения подразделения предприятия, например департамента, службы, отдела, группы и т. д.
 Работник	Класс со стереотипом «worker» Используется для отображения должностей и профессий сотрудников предприятия
	Связь ассоциация. Используется для отображения Связей между элементами
	Связь зависимость. Используется для отображения связей между элементами. Связь проводится от зависимого элемента к независимому элементу. Связь может использоваться со стереотипом, например «подчиняется»

Модель организационной структуры строиться как иерархия диаграмм.

На первом уровне помещаются пакеты с изображением вышестоящих по иерархии подразделений, их связи и работники, кото-

рым они подчиняются, на последующем уровне нижестоящие подразделения, их связи и работники, которым они подчиняются и т. д. На самом последнем уровне работники и их связи между собой.

На рис. 2.3, 2.4 представлен пример моделей организационной структуры университета.

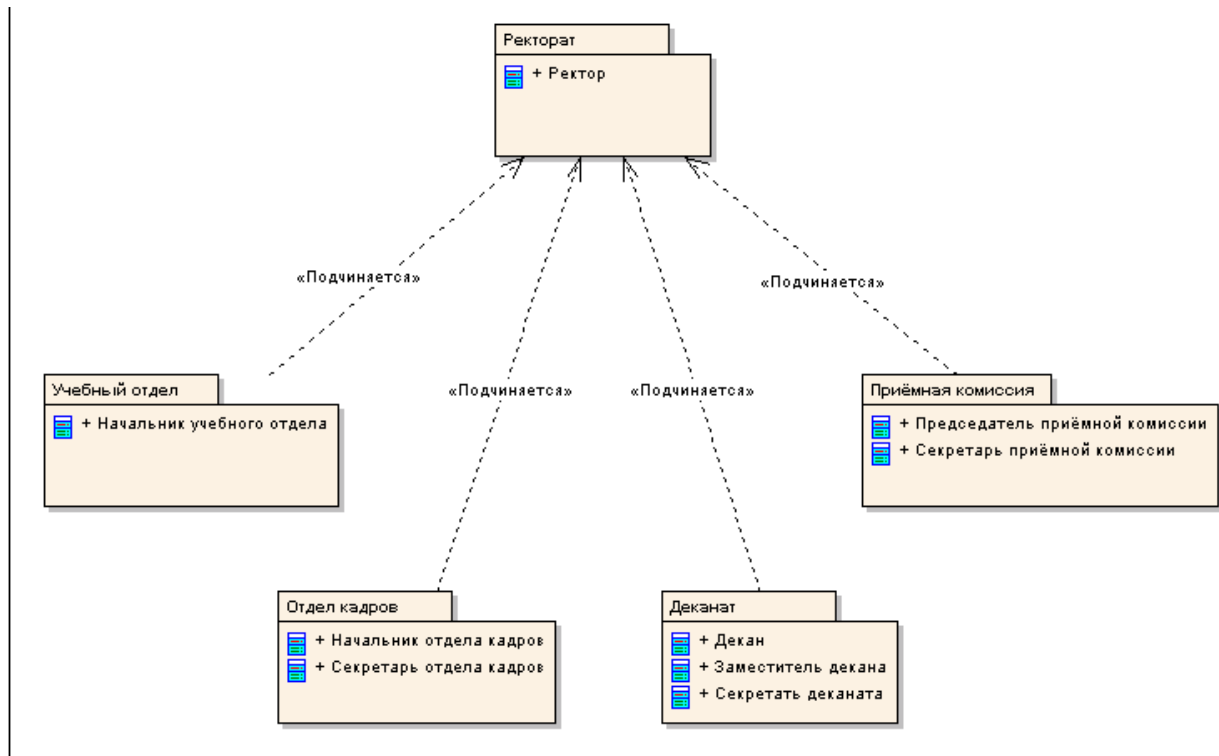


Рис.2.3 Пример пакетной диаграммы, отображающей организационную структуру университета

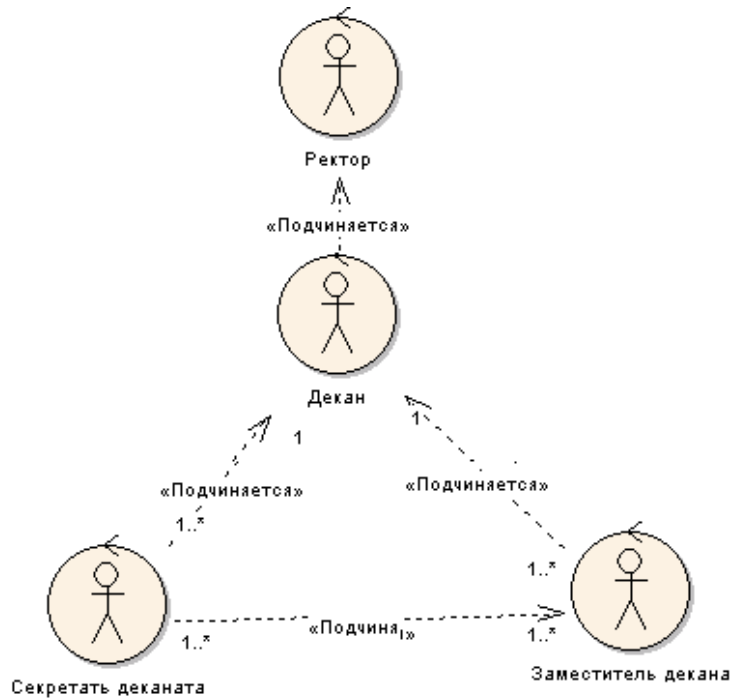


Рис.2.4 Пример диаграммы, отображающей состав сотрудников подразделений и их связи

Пример расположения модели «Организационная структура» в браузере ЕА представлен на рис. 2.5.

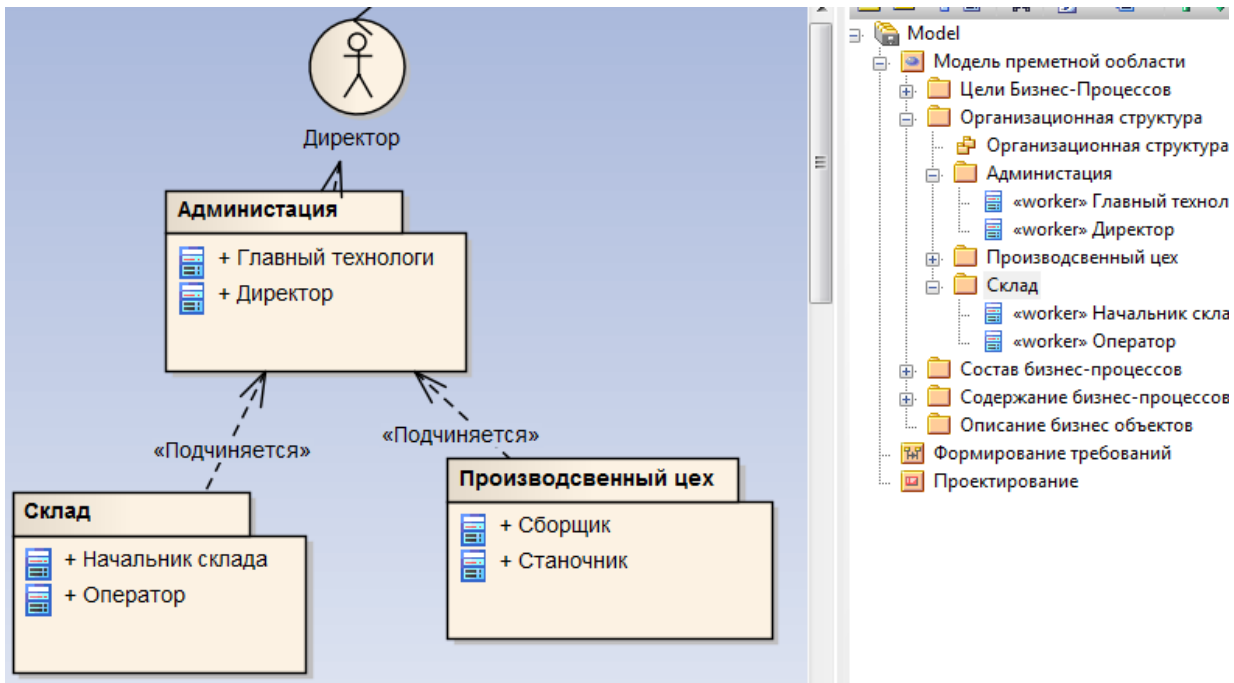


Рис.2.5 Пример расположения модели «Организационная структура» в браузере ЕА

**Цели бизнес-процессов.** Целью модели «Цели бизнес-процессов» является отображение требуемых значений технических, технологических, производственно-экономических или других показателей бизнес-процессов, которые должны быть достигнуты в результате их выполнения.

Для разработки модели «Цели бизнес-процессов» должна использоваться диаграмма классов (class diagram).

Элементы диаграммы классов (class diagram), используемые для разработки модели целей, представлены в табл. 2.2.

Элементы диаграммы классов (class diagram), используемые для разработки модели «Цели бизнес-процессов»

Таблица 2.2

Изображение элемента	Название и назначение элемента
	Класс со стереотипом <<target>>. Используется для отображения целей бизнес-процессов
	Используется для группировки целей в разбивке по различным критериям
	Связь зависимость. Используется для отображения связей между элементами. Связь проводится от зависимого элемента к независимому элементу

Пример модели «Цели системы», представлен на рис. 2.6.

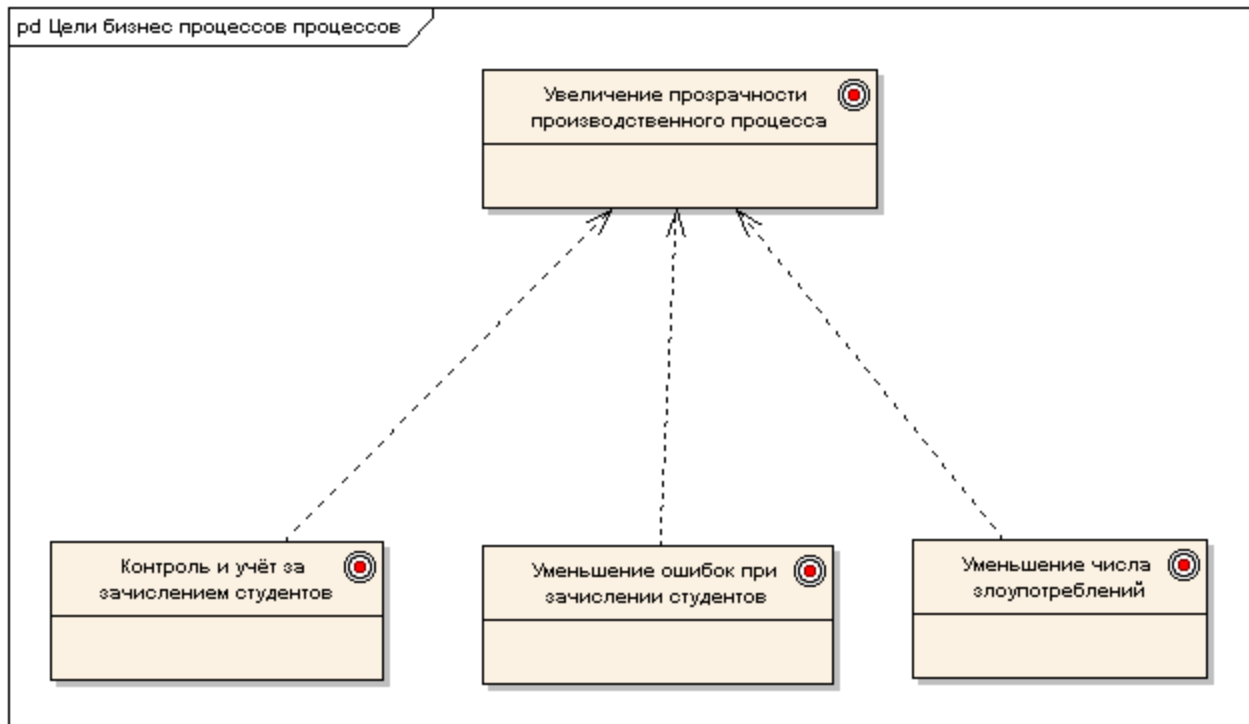


Рис.2.6 Пример модели «Цели бизнес-процессов»

Пример расположения модели «Цели бизнес-процессов» в браузере ЕА представлен на рис. 2.7.

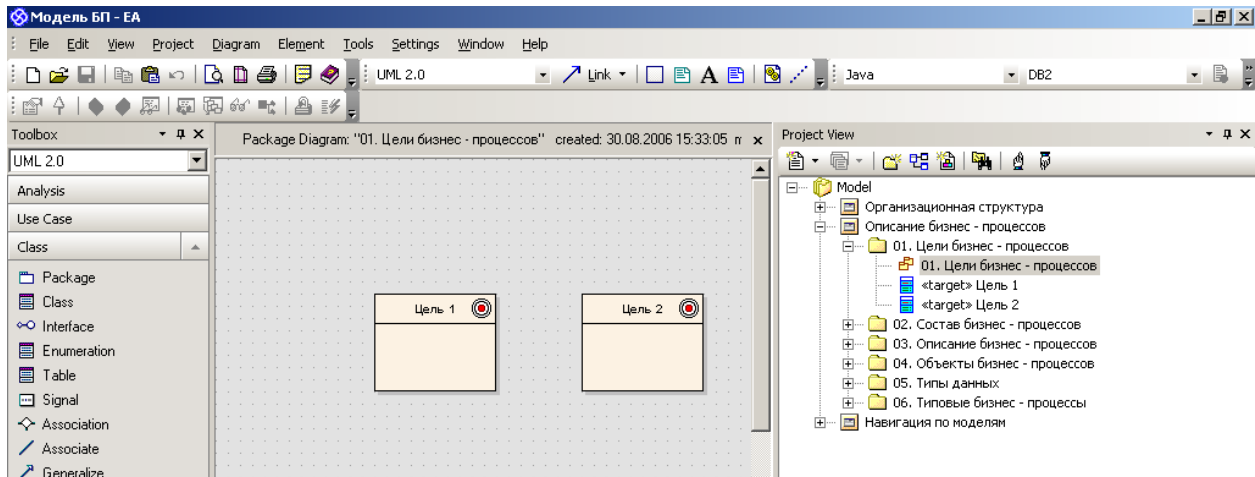


Рис.2.7 Пример расположения модели «Цели бизнес процессов» в браузере ЕА


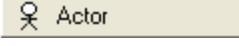
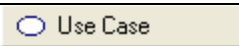

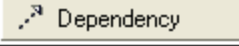

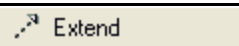
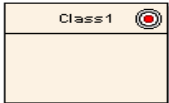
**Состав бизнес-процессов.** Целью разработки модели «Состав бизнес-процессов» является отображение собственно состава обследуемых бизнес-процессов, их связей и целей, которые они поддерживают.

Для разработки модели «Состав бизнес-процессов» должна использоваться use case diagramm (use case diagram).

Элементы use case diagram, используемые для разработки модели, представлены в табл. 2.3.

Элементы диаграммы вариантов использования (use case diagram), используемые для разработки модели «Состав бизнес-процессов»

Таблица 2.3

Изображение элемента	Название и назначение элемента
 Package	Пакет. Используется для отображения групп бизнес-процессов
 Actor	Элемент действующее лицо «Actor». Используется для отображения инициаторов бизнес-процессов, получателей результатов бизнес-процессов
 Use Case	Вариант использования (Use case). При отображении бизнес-процессов может использоваться со стереотипов «бизнес – процесс»
 Associate	Связь ассоциация. Используется для отображения связей между актёрами и вариантами использования
 Dependency	Связь зависимость. Используется для отображения связей между элементами. Связь проводится от зависимого элемента к независимому элементу. Связь может использоваться со стереотипом, например «поддерживает»
 Include	Связь «включает». Специализированная связь, используется для отображения включения функциональности одного бизнес-процесса в другой
 Extend	связь «расширяет». Специализированная связь, используется для расширения функциональности одного бизнес-процесса функциональностью другого при наступлении определенных условий
 Class1	Класс со стереотипом <<target>>. Используется для отображения целей бизнес-процессов

В зависимости от уровня сложности организации модель бизнес-процессов может строиться как иерархия диаграмм.

На первом уровне помещаются пакеты с изображением групп бизнес-процессов, их связи, на последующем уровне также группы бизнес-процессов, их связи и т. д. На самом последнем уровне биз-

нес-процессы, инициаторы бизнес-процессов, получатели результатов бизнес-процессов, цели бизнес-процессов и их связи между собой.

Пример расположения модели «Состав бизнес-процессов» в браузере EA представлен на рис. 2.8.

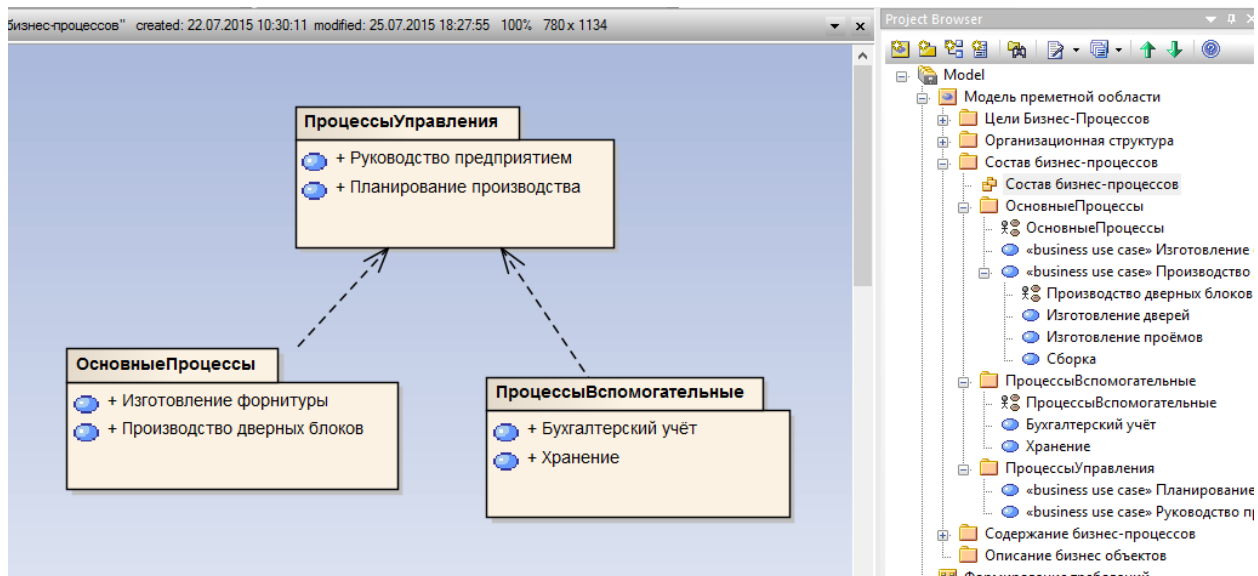


Рис.2.8 Пример расположения модели «Состав бизнес-процессов» в браузере EA

При построении диаграмм элемент «Цель» берётся из пакета «Цели бизнес-процессов», актёров (актантов) необходимо использовать из пакета «Организационная структура». Как уже отмечалось, один элемент модели может быть использован на нескольких диаграммах.

Связь ассоциация используется для отображения связи между бизнес-процессом и актёром (актантом), инициатором бизнес-процесса, получателем результатов бизнес-процессов (рис. 2.9).

Связь – зависимость со стереотипами «include»(включает) и «extend» используется между бизнес-процессами (рис. 2.9).



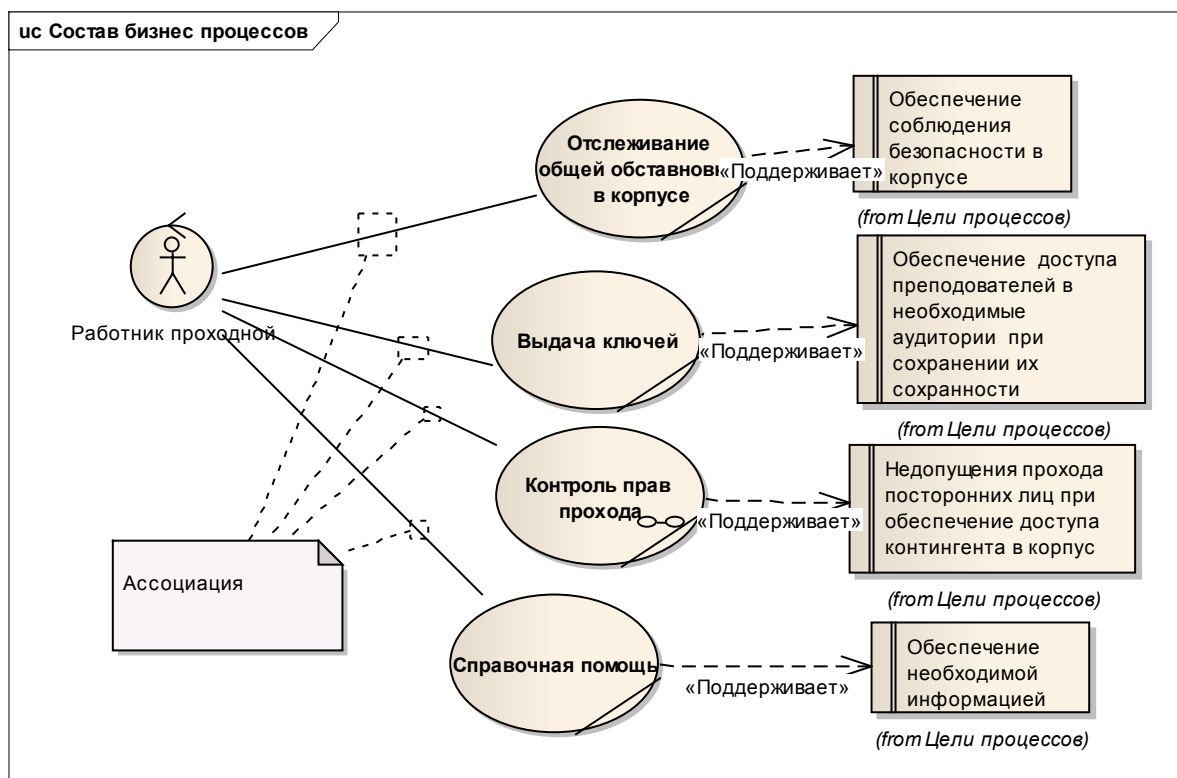


Рис.2.9 Пример использования связей на диаграмме бизнес-вариантов использования

Во многих случаях для отображения некоторой совокупности процессов, объединённой общей целью, удобнее использовать не пакет, в котором группируются процессы, а декомпозицию процесса в подчинённую диаграмму бизнес-вариантов использования. Такая декомпозиция обоснована, когда исходный процесс нельзя непосредственно представить в виде последовательных или параллельных действий, а можно выделить лишь независимо выполняемые подпроцессы.

Декомпозиция любого элемента на подчинённую ему диаграмму может производиться через контекстное меню элемента (Edvanced/composite), кроме того, возможно присоединение подчинённой диаграммы в браузере проекта перетаскиванием значка готовой диаграммы на соответствующий элемент, с последующим указанием для данного элемента свойства декомпозируемости (PropertyEdvanced/composite). Элемент, к которому присоединена декомпозирующая его диаграмма помечается значком  $\infty$ .

### 3. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Создать новую модель в среде Enterprise Architect.
2. В обозревателе модели создать область просмотра (view) «Модель предметной области».
3. Создать пакеты для составляющих подмоделей:
  - a. Организационная структура.
  - b. Цели бизнес-процессов.
  - c. Состав бизнес-процессов.
  - d. Содержание бизнес-процессов.
  - e. Бизнес-правила, выполнения БП.
  - f. Классы объектов предметной области.
  - g. Типы данных предметной области.
  - h. Описание бизнес-правил.
  - i. Автоматизируемые элементы БП.
4. Построить модель организационной структуры предприятия.
5. Сформировать цели предприятия.
6. Определить состав бизнес-процессов, отобразить их в виде диаграммы «бизнес-вариантов использования». Выявить положение бизнес-процесса, автоматизация которого требуется согласно заданию (базового бизнес-процесса)

### 4. ЗАДАНИЯ

Заданием для работы является тип предприятия, деятельность которого необходимо проанализировать для последующей автоматизации, или некоторая деятельность без указания предприятия. Для подробного анализа должен быть взят процесс, указанный в скобках. Студент может сам предложить задание, по аналогии с приведёнными примерами.

#### **Примеры заданий**

1. Деятельность профилирующей кафедры. Подготовка документов для проведения аттестационной комиссии.
2. Деятельность кафедры. Подготовка приказа о прохождении практики.

3. Деятельность приёмной комиссии. Приём документов.
4. Деятельность ресторана (принятие заказа и его выполнение).
5. Деятельность кинотеатра (продажа билетов).
6. Деятельность по обеспечения проживания в общежитии (проход посторонних в гости).
7. Деятельность по обеспечения проживания в общежитии. Процесс устройства в общежитие.
8. Деятельность поликлиники. Медосмотр.
9. Деятельность регистратуры поликлиники (запись на приём к врачу).
10. Деятельность библиотеки (выдача книги).
11. Деятельность по получению прав вождения автомобилем.
12. Деятельность деканата по учету текущей и сессионной успеваемости и посещаемости занятий.
13. Деятельность по приобретению недвижимости.
14. Деятельность областного ГАИ. Постановка на учет автомобиля.
15. Деятельность областного ГАИ. Снятие с учёта автомобиля.
16. Областное ГАИ. Процесс по учёту сдачи на право вождения автомобилем.
17. Студенческой столовой (отпуск блюд клиентам).
18. Проходной корпуса (приём выдача ключей (с учётом того что преподаватель может быть не знакомым вахтёру и должен предъявить удостоверение)).
19. Деятельность деканата (выписка материальной помощи).
20. Деятельность хозяйственного магазина. Процесс покупки в магазине с предварительной выборкой товара оплатой и последующим получение.
21. Деятельность преподавателя. Приём экзамена.
22. Деятельность студента. Процесс упорядочивание дел.
23. Деятельность студента. Ведение личного бюджета.
24. Деятельность студента-спортсмена. Процесс учёта тренировочных нагрузок и спортивной формы.
25. Деятельность студента, следящего за правильным питанием. Процесс учёта питания.
26. Деятельность театра. Процесс планирования репертуара.
27. Процесс обеспечения проживания в общежитии (проход посторонних в гости).

28. Деятельность студенческой столовой (отпуск блюд клиентам).

29. Деятельность проходной корпуса (приём выдача ключей (с учётом того что преподаватель может быть не знакомым вахтёру и должен предъявить удостоверение)).

## 5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Содержание модели предметной области.
2. Какие элементы используются для отображения организационной структуры?
3. Какие элементы и диаграммы используются для отображения состава бизнес-процессов?
4. Какова структура модели в среде Enterprise Architect?

## РАБОТА №3 ПРАКТИЧЕСКАЯ. ПОСТРОЕНИЕ И АНАЛИЗ МОДЕЛИ ПРЕДМЕТНОЙ ОБЛАСТИ НА ОСНОВЕ ООП. МОДЕЛЬ СОДЕРЖАНИЯ БИЗНЕС-ПРОЦЕССОВ

### 1. ЦЕЛЬ И ЗАДАЧИ РАБОТЫ

Получение практических навыков построения модели предметной области на основе объектно-ориентированного подхода.

### 2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ






Целью разработки модели «Описание бизнес-процессов» является отображение последовательности выполнения работ, связанной с бизнес-процессом.

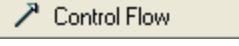
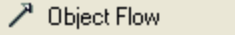

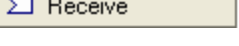
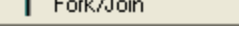
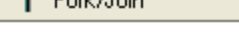
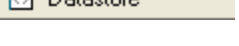
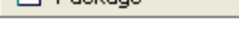
Для разработки модели «Описание бизнес-процессов» должны использоваться диаграммы деятельности (activity diagram). Содержание отдельного бизнес-процесса отображается в виде связанной с ним (декомпозирующей его) диаграммы деятельности.

Элементы диаграммы деятельности (activity diagram), используемые для разработки модели, представлены в табл. 3.1.

Элементы диаграммы деятельности (activity diagram), используемые для разработки модели «Описание бизнес-процессов»

Таблица 3.1

Изображение элемента	Название и назначение элемента
 Initial	Начало. Используется для отображения начала бизнес-процесса
 Final	Конец. Используется для отображения окончания бизнес-процесса
 Activity	Деятельность. Используется для отображения деятельности выполняемой объектами или субъектами системы, шага бизнес-процесса
 Subactivity	«Макро» деятельность. Используется для отображения обобщенной деятельности, которая декомпозируется
 Object	Объект системы или предметной области. Используется для отображения ролевой ответственности при выполнении какой-либо деятельности, для отображения объектов реального мира, связанных

Изображение элемента	Название и назначение элемента
	с бизнес-процессом. Объекты могут быть использованы с различными стереотипами, например, «worker», «документ»
	Поток управления. Используется для отображений связей между деятельностью
	Поток объектов. Используется для отображения связей между деятельностью и объектом
	Решение. Используется для ветвления, слияния, разветвления потока работ в сценарии
	Отправка. Используется для отображения действия, связанного с отправкой сигнала
	Прием. Используется для отображения действия, связанного с приемом запроса
	Горизонтальная вилка. Используется для слияния и расщепления параллельных потоков
	Вертикальная вилка. Используется для слияния и расщепления параллельных потоков
	БД. Используется в бизнес-процессах, связанных с БД
	Пакет. Используется для отображения подразделения в котором выполняется шаг бизнес-процесса

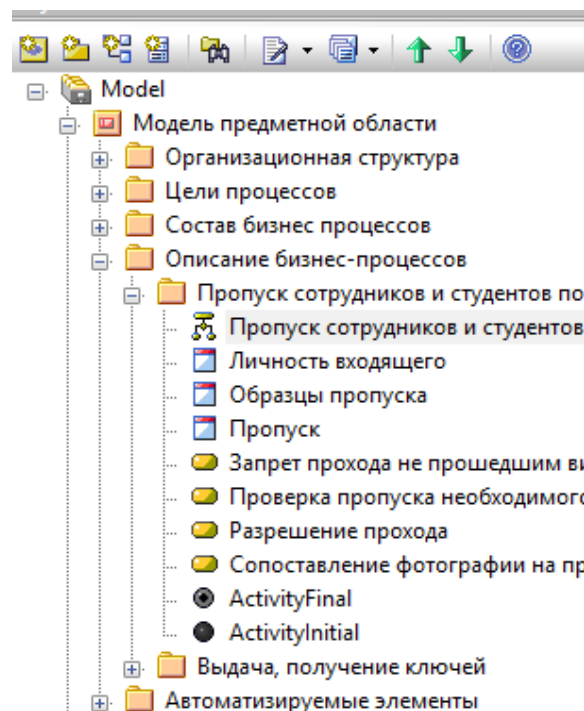


Рис.3.1 Пример расположения модели «Описание бизнес-процессов» в браузере EA

Модель с описанием бизнес-процессов может разрабатываться в два этапа. Сначала разрабатывается модель, отображающая макрошаги бизнес-процесса (рис. 3.2), затем – модели, отображающие детальное описание макро шагов процессов (рис. 3.3). Для процессов с небольшим числом шагов (до 5–9) выделение макрошагов нецелесообразно сразу строится детальная диаграмма.

При разработке диаграмм, отображающих детальное описание процесса, поле диаграммы деятельности можно разбивать на области ответственности с использованием следующих разделительных линий (swimlanes) (рис. 3.4):

- входные/выходные объекты;
- шаг бизнес-процесса;
- участник;
- подразделение;
- бизнес правила.

Если потребуется можно вводить дополнительные разделительные линии, например, инструмент, стоимость, время и т. д.

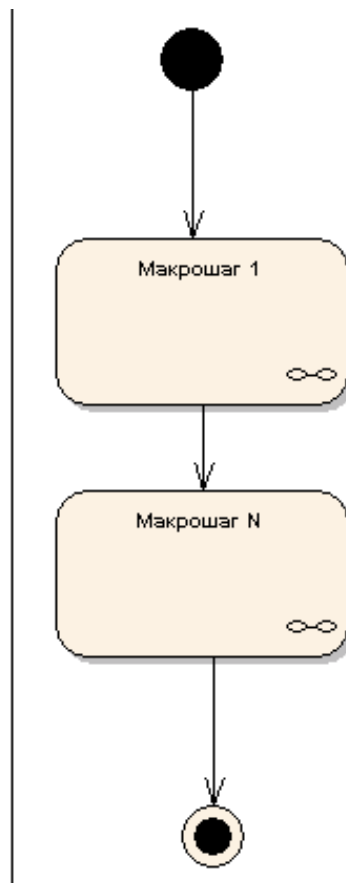


Рис.3.2 Пример описание макрошагов бизнес-процесса

Знак «бесконечность» на элементе деятельности отображает факт нахождения под ней диаграммы ее декомпозирующей.

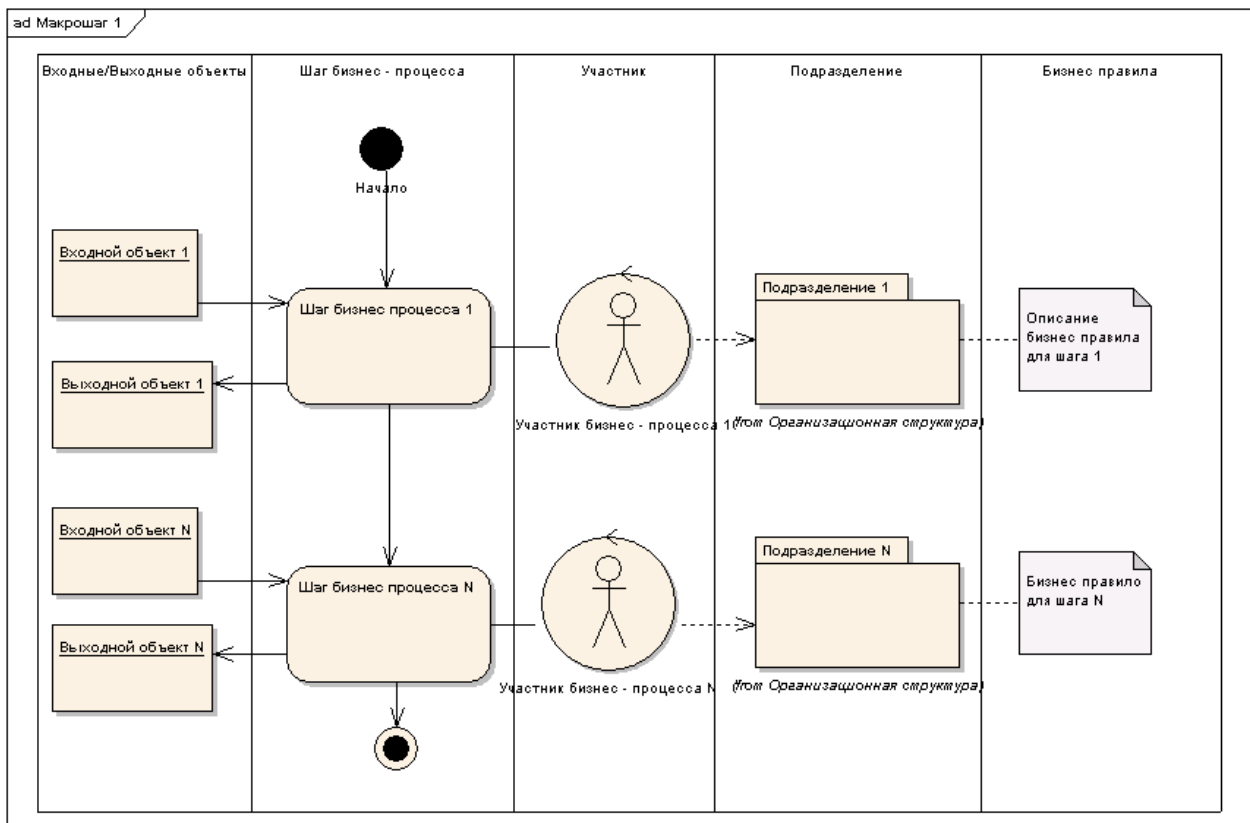


Рис.3.3 Пример описание пошагового описания макрошага бизнес-процесса

При разработке пошагового описания макрошага следует входной объект присоединять к деятельности связью «поток объектов», направленной от объекта к деятельности, выходной объект – связью «поток объектов», направленной от деятельности к объекту.

Объект, изображающий участника бизнес-процесса, следует присоединить к деятельности связью ассоциация. Пакет, отображающий подразделение, в котором выполняется шаг бизнес-процесса, следует переместить на диаграмму из модели «Организационная структура».

Объект, изображающий участника бизнес-процесса, следует присоединить к пакету, отображающему подразделение, связью «зависимость». Связь должна быть направлена от объекта участника к пакету подразделения.



Рекомендуется на одной диаграмме изображать не более девяти элементов «деятельность».

Шаги бизнес-процесса, подлежащие автоматизации, следует отмечать цветом или стереотипом «автоматизируется».

На рис. 3.4 представлен пример описания бизнес-процесса.

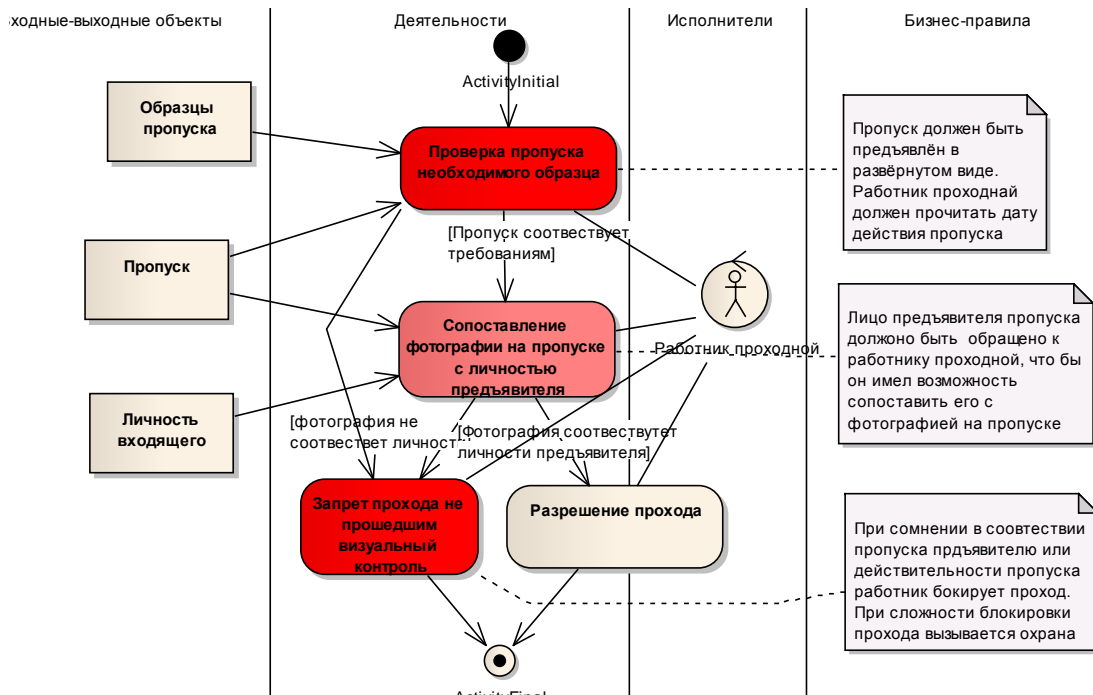


Рис.3.4 Пример описания бизнес-процесса.

Красным цветом отмечены автоматизируемые элементы

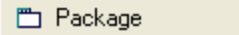
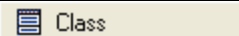
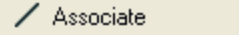
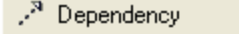
*Объекты бизнес-процессов.* Целью разработки модели «Объекты бизнес-процессов» является описание материальных и информационных объектов реального мира, связанных с бизнес-процессами.

Для разработки модели «Объекты бизнес-процессов» должна использоваться диаграмма классов (class diagram).

Элементы диаграммы классов (class diagram), используемые для разработки модели, представлены в табл. 3.2.

Элементы диаграммы классов (class diagram), используемые для разработки модели «Объекты бизнес-процессов»

Таблица 3.2

Изображение элемента	Название и назначение элемента
 Package	Пакет. Используется для группировки элементов
 Class	Класс. Используется для отображения печатных документов, материальных объектов. При необходимости требуется указание его атрибутов и связи с внешним видом
 Associate	Связь ассоциация. Используется для отображения связей между элементами
 Dependency	Связь зависимость. Используется для отображения связей между элементами. Связь проводится от зависимого элемента к независимому элементу

Модель «Объекты бизнес-процессов» может быть построена как иерархия диаграмм. Сначала – состав объектов в разбивке по бизнес-процессам, далее – конкретные объекты.

На рис. 3.4, 3.5 представлен пример состава объектов бизнес-процессов в виде пакетных диаграмм, на рис. 3.6 – пример конкретного объекта.

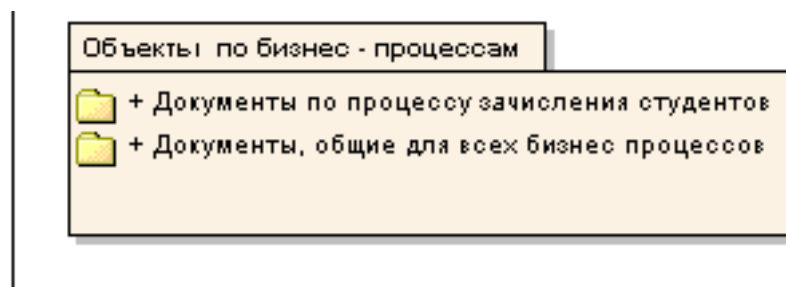


Рис.3.5 Пример состава объектов бизнес-процессов.  
Объекты сгруппированы в пакеты

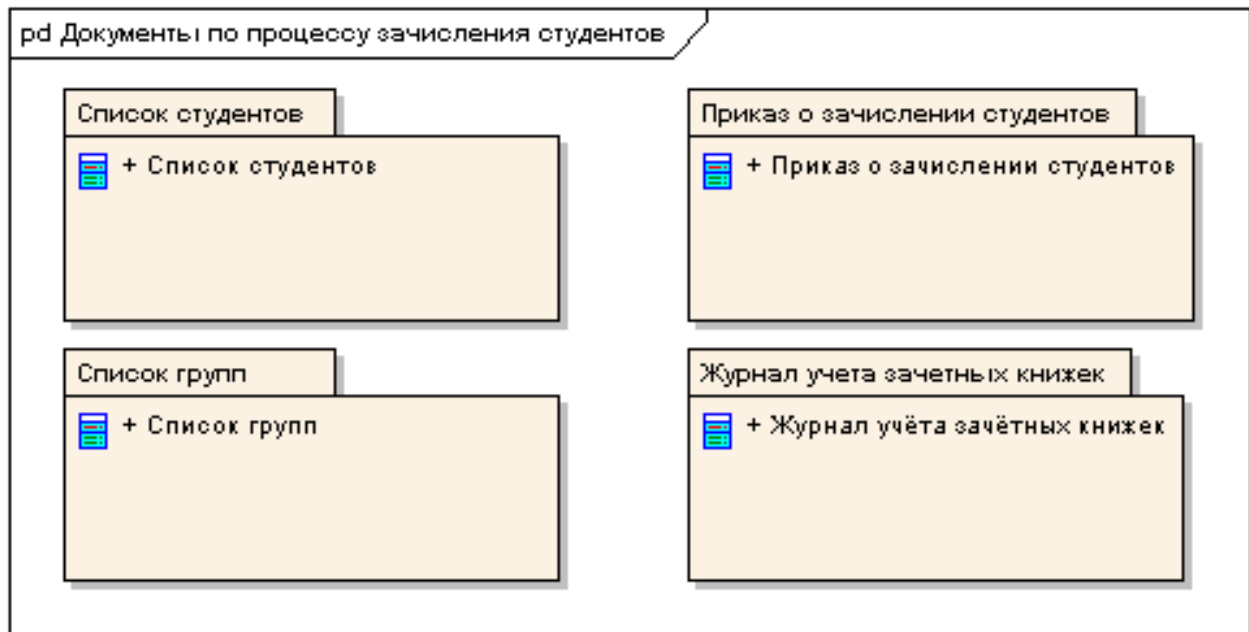


Рис.3.6 Пример отображения состава объектов бизнес-процессов в виде пакетной диаграммы

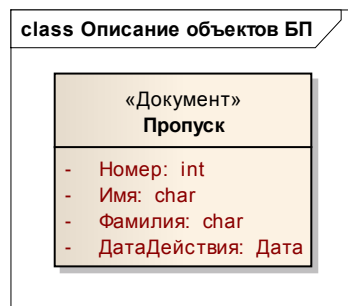


Рис.3.7 Пример объекта бизнес-процесса в виде класса со стереотипом «документ»

Пример расположения модели «Объекты бизнес-процессов» в браузере ЕА представлен на рис. 3.7.

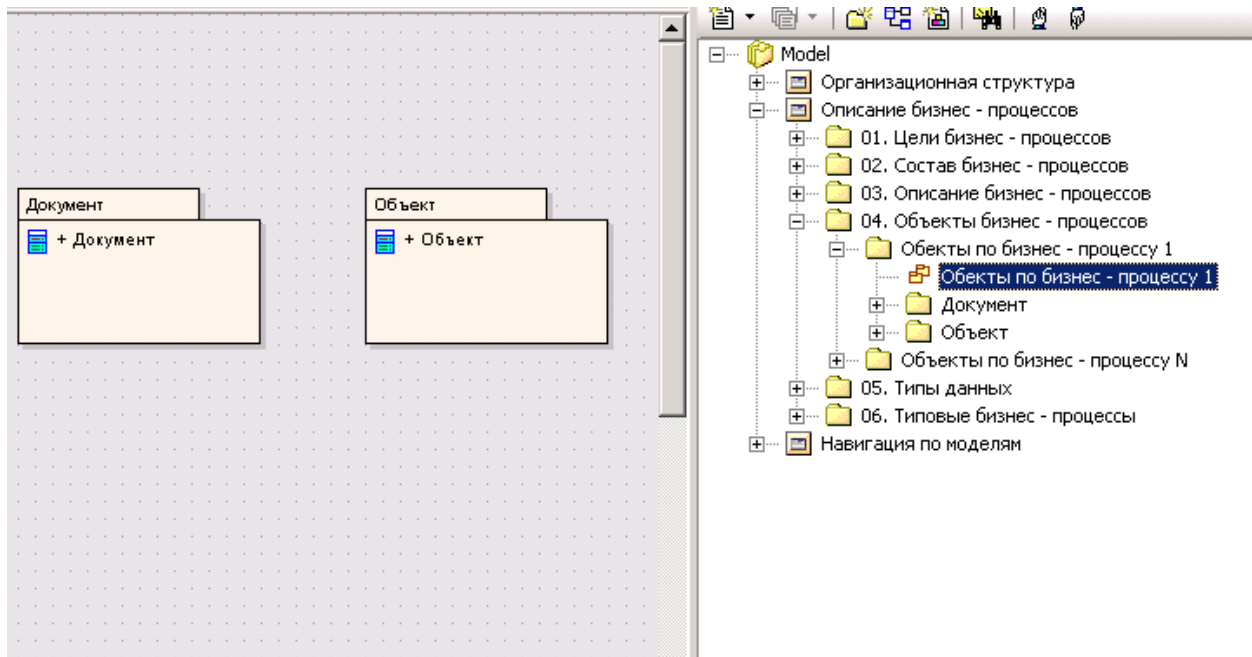


Рис.3.8 Пример расположения модели «Объекты бизнес-процессов» в браузере ЕА



**Типы данных.** Целью разработки модели «Типы данных» является унификация описание атрибутов объектов реального мира, связанных с бизнес-процессами, например, документов, материальных объектов.

Для разработки модели «Типы данных» должна использоваться диаграмма классов (class diagram).

Элементы диаграммы классов (class diagram), используемые для разработки модели, представлены в табл. 3.3.

Элементы диаграммы классов (class diagram), используемые для разработки модели «Типы данных»

Таблица 3.3

Изображение элемента	Название и назначение элемента
 Package	Пакет. Используется для группировки элементов
 Class	Класс. Используется для отображения типов данных печатных документов, материальных объектов. При необходимости требуется указание его атрибутов

На рис. 3.8 представлен пример описания используемых типов данных.

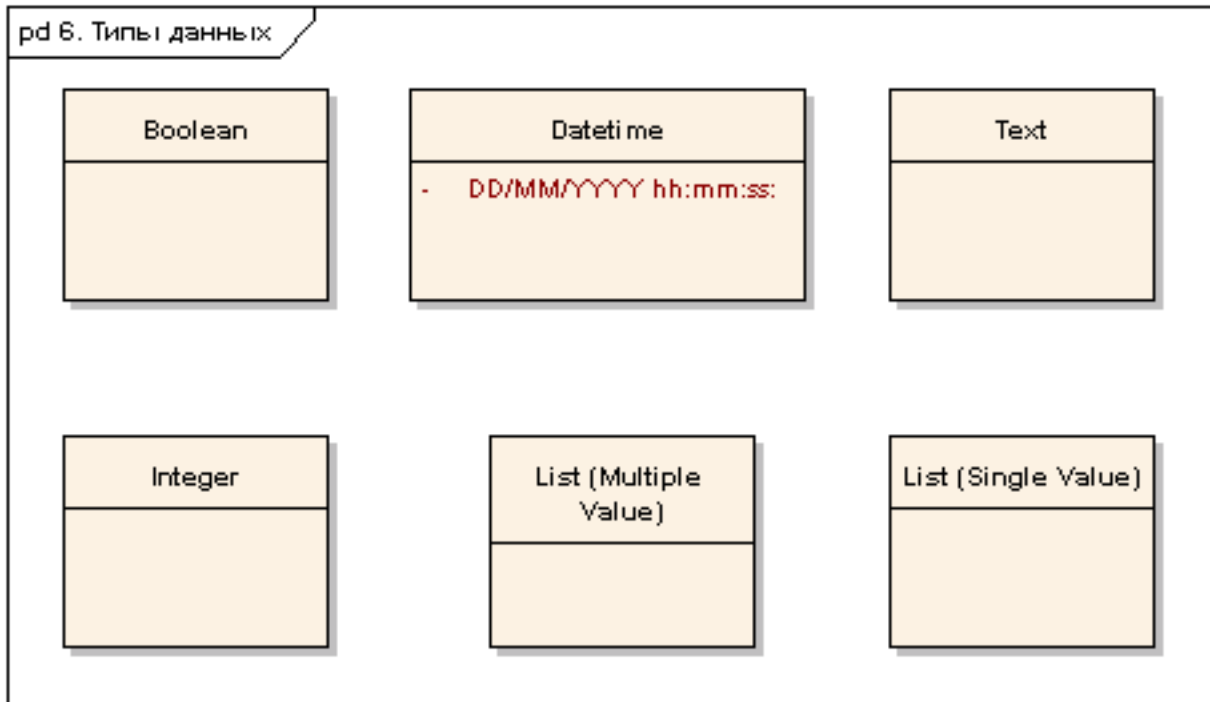


Рис.3.9 Пример отображения используемых типов данных

Пример расположения модели «Типы данных» в браузере ЕА представлен на рис. 3.9.

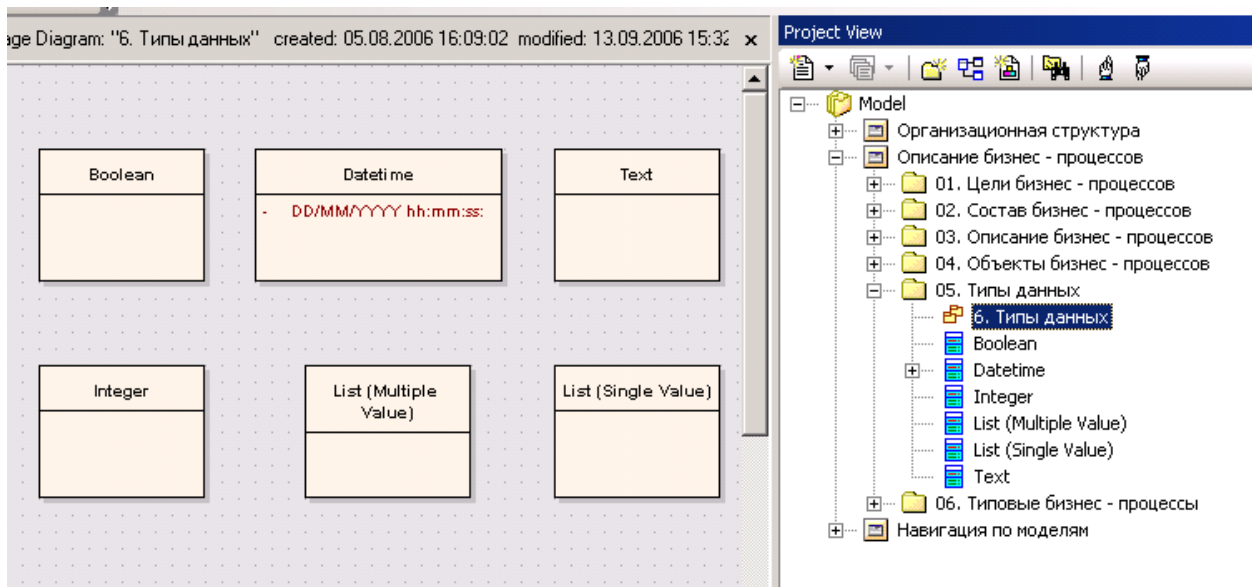


Рис.3.10 Пример расположения модели «Типы данных» в браузере ЕА

### **Выявление автоматизируемых элементов бизнес-процессов.**

Работа по построению и анализу модели предметной области условий заказчика завершается выявлением элементов бизнес-

процессов, требующих автоматизации (автоматизируемые элементы бизнес-процессов АЭ БП). Элементы БП, требующие автоматизации предварительно намечаются при выявлении содержания бизнес-процессов. Они на диаграммах, декомпозирующих БП отмечаются красным цветом или стереотипом «автоматизируется».

Автоматизируемые элементы соответствуют элементам БП, занимающим узловое положение и автоматизация которых приведет к максимальному выигрышу. Такие элементы могут повторяться в нескольких БП.

Автоматизируемые элементы включаются в модель в виде элемента UML «Требование (Requirement)» и отображаются в виде пользовательской («custom») диаграммы.

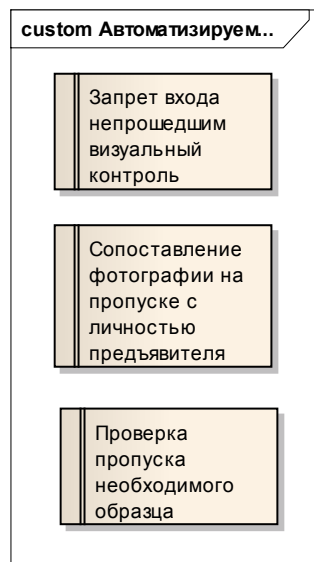


Рис.3.11 Диаграмма состава автоматизируемых элементов бизнес-процессов

Каждый элемент, вынесенный на диаграмму должен содержать описание. Описание должно отображать место, которое занимает автоматизируемый элемент в БП и деятельности предприятия в целом, как повлияет на выполнение БП его автоматизация. На какие показатели БП и деятельности предприятия повлияет автоматизация БП.

Автоматизируемые элементы БП являются прообразами требований к разрабатываемой системе, автоматизирующей деятельность предприятия.

### 3. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Определить содержание базового бизнес-процесса и, если необходимо, связанных с ним БП, отобразить содержание БП в виде диаграмм деятельности:

а. Выявить объекты, участвующие в выполнении БП.  
б. Выявить бизнес-правила, выполнения отдельных элементов БП.

с. Выделить автоматизируемые элементы БП.

2. Для объектов, выявленных в бизнес-процессах сформировать классы:

а. Создать классы объектов предметной в соответствующем пакете обозревателя модели.

б. Задать для объектов предметной области, участвующих в выполнении рассматриваемых бизнес-процессов (соответствующие диаграммы деятельности) классификаторы из состава сформированных классов.

с. Построить диаграмму классов предметной области.

3. Выявить типы данных предметной области.

4. Отобразить выявленные автоматизируемые элементы БП на отдельной диаграмме в соответствующем пакете обозревателя модели.

5. Создать отчёт по выполненной работе.

### 4. ЗАДАНИЕ НА ВЫПОЛНЕНИЯ

Заданием для работы является отображение содержания одного из процессов деятельности предприятия, рассмотренного в прошлой практической работе.

### 5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Какие элементы и диаграммы используются для отображения содержания бизнес-процессов?

2. Каким образом выявляются классы объектов предметной области?

3. Каким образом выявляются элементы, подлежащие автоматизации? Какими элементами и на каких диаграммах они отображаются?

## **РАБОТА №4 ПРАКТИЧЕСКАЯ. ОПРЕДЕЛЕНИЕ СОСТАВА ФУНКЦИОНАЛЬНЫХ ТРЕБОВАНИЙ К СИСТЕМЕ НА ОСНОВЕ АНАЛИЗА ПРЕДМЕТНОЙ ОБЛАСТИ**

### **1. ЦЕЛЬ РАБОТЫ**

Целью работы является освоение процесса формирования требований на основе объектно-ориентированного подхода, в связи с этим задачами работы является:

- определить состав функциональных требований;
- определить состав сценариев, реализующих функциональные требования и их варианты использования, построить диаграмму вариантов использования;
- разработать содержание сценариев, при этом предварительно выявить объекты, участвующие в сценариях;
- определить требования к пользовательскому интерфейсу;
- определить требования к хранилищу данных

### **2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ**

Требования к информационной системе (ИС) представляют один из важнейших артефактов, используемый при разработке. Правильное, полное представление требований определяет саму возможность разработки системы в соответствии с потребностью заказчика.

Результат процесса формирования требований при использовании нотации UML представляется в виде «модели вариантов использования» с сопутствующими описаниями и составляющими диаграммами. Наиболее удобные возможности для отображения результатов процесса формирования требований предоставляют интегрированные среды моделирования Rational Rose, Enterprise Architect и другие.

При использовании среды Enterprise Architect, результаты процесса формирования требований обычно отображаются в виде отдельной составляющей общей модели разработки ИС (пакета верхнего уровня – «view»). В рамках пакета верхнего уровня «Формирование требований» обычно выделяются подпакеты содержащие ре-



зультаты решения составляющих подзадач, связанных с определением требований:

- состав требований;
- состав и структура вариантов использования;
- детализация вариантов использования;
- требования к пользовательскому интерфейсу.

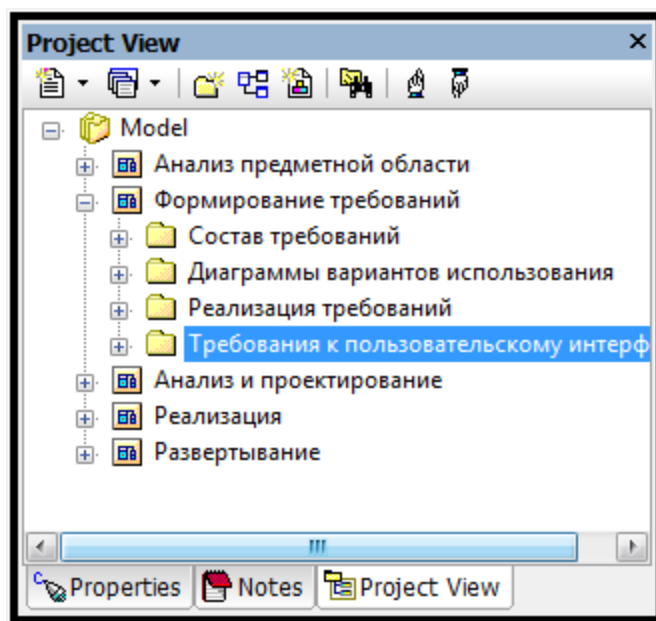


Рис.4.1 Содержание моделей разрабатываемых в процессе формирования требований

Формирование требований производится в несколько итераций, которые могут выполняться, как на стадии технического задания, так и на стадии технической проект. Однако, это не должно нарушать целостность процесса определения требований, результатом которого должна быть единая модель, отображающая структурированное представление требований – «модель вариантов использования». Кроме того, описание требований должны быть представлены в виде подробного документа «спецификация программных требований» или в виде соответствующего раздела документа «техническое задание» (ГОСТ 34.602). Данные документы согласуется с заказчиком и является основой для заключения о соответствии разработанной системы требованиям заказчика.

Требования к системы обычно разделяются на функциональные, не функциональные и дополнительные. Функциональные определяют требуемые функции к системе, нефункциональные харак-

теризуют требуемые параметры выполнения функций системы, дополнительные требования используются для описания параметров работы системы в целом или её подсистем или группы функций. Большое значение имеет правильное определение функциональных требований.

**Определение функциональных требований.** Логически функциональные требования должны вытекать из анализа предметной области, однако, возможен вариант их формироваться и без предварительного построения и анализа модели предметной области. Источником формирования требований может быть, так называемый, «список кандидатов в требования». Данный список создается на основе предложений заказчика, бесед со специалистами в рассматриваемой области и анализа систем, аналогичных разрабатываемой. Однако более обоснованное формирование требований производится на базе построения и анализа бизнес модели предметной области и выявление «элементов, требующих автоматизации» в сценариях выполнения бизнес-процессов. Выявленные элементы, требующие автоматизации и являются кандидатами в требования. Требования формируются, как отображение «элементов, требующих автоматизации» в предметной области в предполагаемые функции системы. Для каждого элемента формируется одно соответствующее требование. Возможно отклонение от данного принципа, например, когда несколько элементов различных бизнес-процессов автоматизируются одной функцией системы. Требования могут быть добавлены не только как отображение автоматизируемого элемента, но и дополнительно, как в примере, показанном на рис. 4.2.

В среде моделирования результат определение состава требований отображается в виде диаграммы дополнительного (пользовательского) типа (custom diagram). Требования и автоматизируемые элементы бизнес-процесса отображаются в виде элемента «requirement». Данный элемент представляет расширение стандартных элементов UML и обычно в средах моделирования на инструментальной панели находится в группе «пользовательских» (custom) элементов.

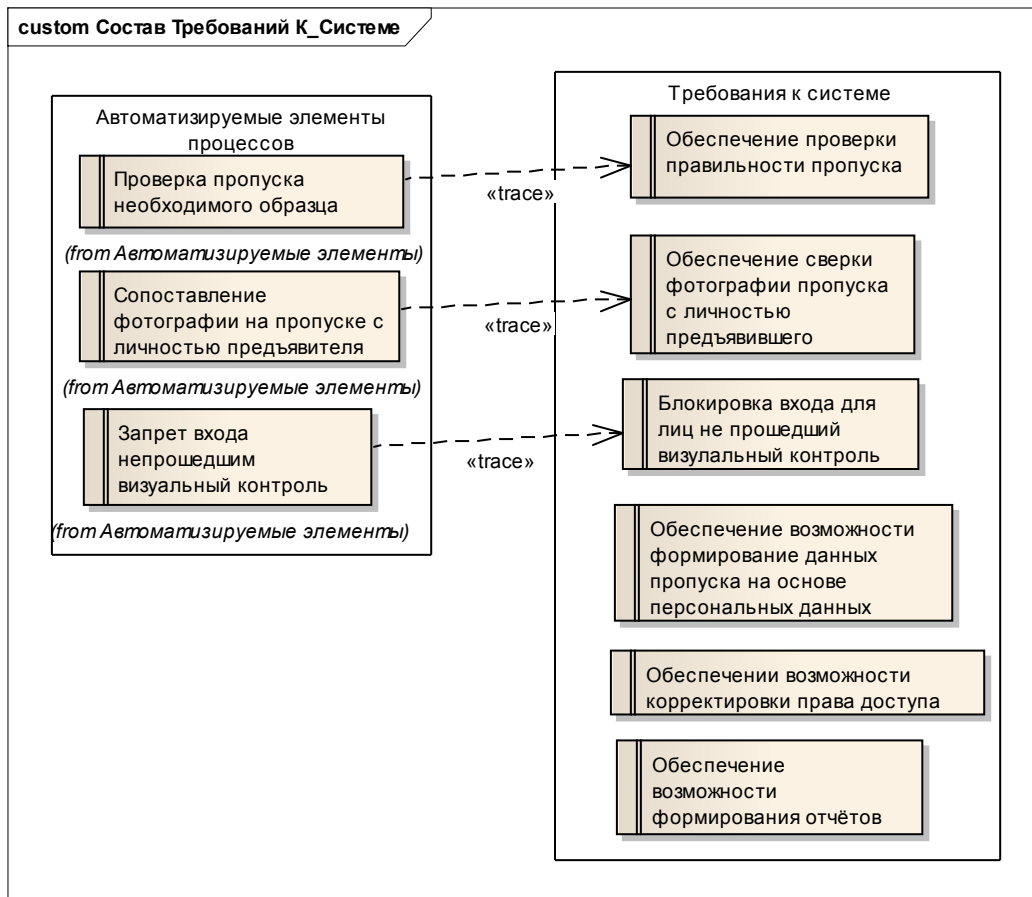


Рис.4.2 Формирование состава требований на основе выявленных элементов бизнес-процессов, требующих автоматизации

В том случае, когда выявлено достаточно много требований к системе, их целесообразно сгруппировать в подсистемы. В одну подсистему объединяются однотипные функциональные требования, автоматизирующие однотипные действия бизнес-процессов (БП) или связанных групп БП.

Для отображения выявленных подсистем необходимо построить соответствие (матрицу трассировки) «Подсистема – Функциональное требование», аналогично соответствию, связывающему автоматизируемые бизнес-решения и требования.



Рис.4.3 Разделение требований по функциональным подсистемам

Для того, что бы требование было правильно реализовано необходимо уточнить следующие его аспекты:

- состав функций (сценариев) которыми реализуется требование;
- какие пользователи или внешние системы могут вызывать функции, реализующее данное требование;
- необходимо детализировать сценарии, реализующие требования, то есть установить необходимую последовательность действий в функциях;
- установить объекты, задействованные в выполнении функций, реализующих требования.

**Определение состава функций системы (сценариев), реализующих требование.** В данном случае под сценарием подразумевается описание действий заложенных в систему. Для реализации одного требования может быть использовано несколько сценариев системы. Отдельный сценарий отображается с помощью элемента UML «use case» или «вариант использования». Состав разработанных сценариев, их связь с требованиями и с внешними сущностями отображается в виде *диаграммы вариантов использования*. В том

случае, если в системе выделено достаточно большое количество требований, диаграммы вариантов использования целесообразно строить отдельно для каждой подсистемы.

Варианты использования могут быть соединены отношениями зависимости со стереотипами «include» и «extend». В том случае, если вариант использования включается в выполнение некоторого другого варианта использования или в некоторых случаях расширяет его функциональность.

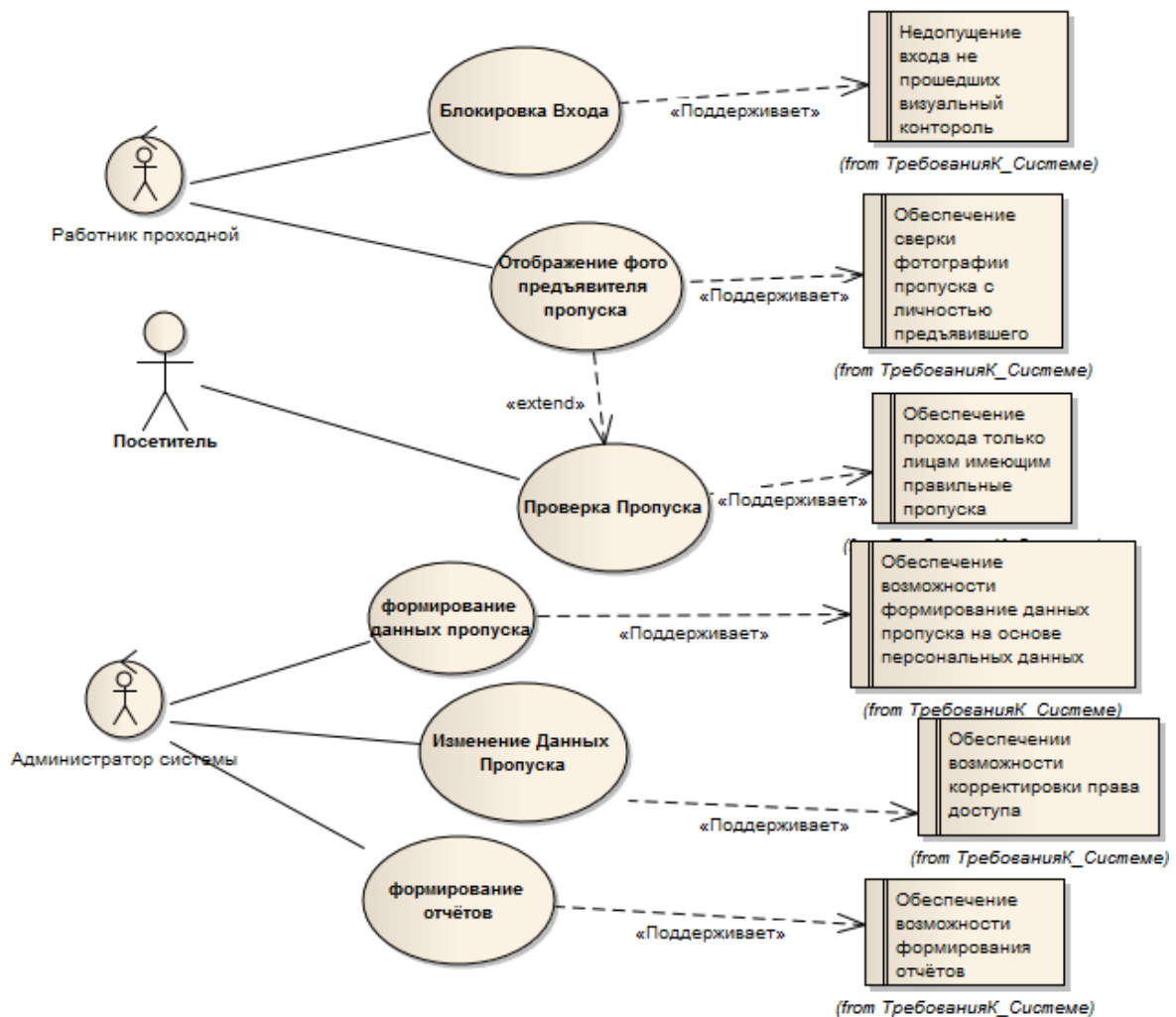


Рис.4.4 Диаграмма вариантов использования

Внешние сущности, использующие функции системы, или инициирующие их выполнение, отображаются в виде соответствующих элементов UML «Actor», обычно обозначаемых в отечественной литературе как «актер» или «актант».

Каждый сценарий, соответствующий отдельной функции системы, отображается в виде элемента UML «use case», или «вариант

использования». Каждый вариант использования подразумевает какую-то деятельность системы, вследствие которой «актант» получает ценный для него результат. Вариант использования связывается отношением зависимости со стереотипом «реализует (realize)» или «поддерживает» с соответствующим требованием.

Для каждого сценария необходимо сформировать его текстовое описание в виде возможных потоков событий, то есть действий, включаемы в него, обычно выделяется основной поток событий, альтернативный (используемый в штатном режиме, но при определенных условиях и поток событий исключительный, обычно используемый при обработке ошибочной ситуации).

Сумма всех вариантов использования со связанными с ними актантами и поддерживаемыми функциональными требованиями, и составляет *модель вариантов использования*, которая описывает полную функциональность системы.

**Определение содержимого сценариев работы** системы производится в виде декомпозиции соответствующих им «вариантов использования». Для каждого «варианта использования» формируется декомпозирующая диаграмма деятельности. Декомпозицию каждого варианта использования целесообразно выполнять в отдельном пакете. На ней отображаются деятельности проектируемой системы по выполнению данной функции. Отдельная деятельность (activity) обычно включает следующие действия:

- ввод данных в интерфейсные формы системы;
- сохранение данных;
- выполнение преобразования данных;
- другие действия, которые должна выполнять система, для выполнения функционального требования, которое реализует сценарий.

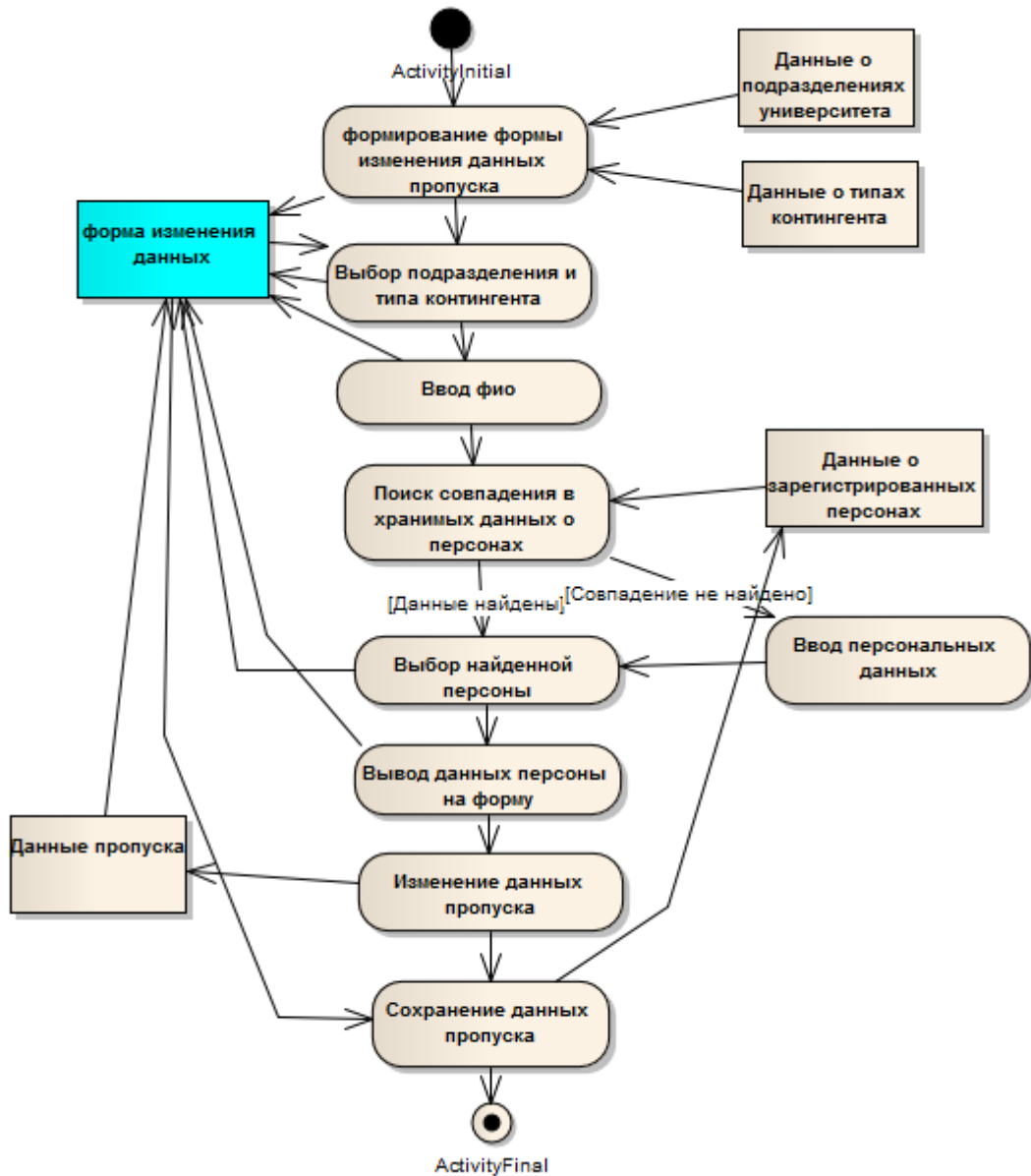


Рис.4.5 Пример диаграммы деятельности, декомпозирующий сценарий работы системы. В данном случае декомпозируется сценарий «Изменений данных пропуска»

На диаграмме деятельности необходимо отобразить для каждого элемента-деятельности (activity) потоки входных и выходных объектов (object flow). Необходимо определить:

- какие данные вводятся в деятельность (activity) – отображается как объекты-сущности (стереотип объекта entity);
- с помощью какого объекта (формы) вводятся/выводятся данные – отображается как объекты-сущности (стереотип объекта boundary).

В некоторых случаях требуется введение специального контрольного объекта, выполняющего вспомогательные контрольные функции. Контрольный объект вводится, если рассматриваемая деятельность не связана непосредственно с выполнением бизнес-функции, а соответствует некоторым вспомогательным действиям, например по контролю доступа, проверке прав, проверке правильности заполнения и др. Контрольный объект отображается с соответствующим стереотипом – *control*.

На основании сформированных сценариев и выявленных объектов сформировать *требования к пользовательскому интерфейсу*.

**Формирование требований к пользовательскому интерфейсу.** Требования к пользовательскому интерфейсу включают:

- его стиль;
- предпочтительную цветовую гамму;
- тип используемых управляющих элементов;
- состав экранных форм;
- состав управляющих элементов на экранных формах.

Состав экранных форм определяется на основе граничных объектов, выявленных в сценариях выполнения функций системы. Каждый граничный объект может соответствовать или форме или закладке на форме.

В данном пункте необходимо разработать предварительный эскиз форм, то есть определить их вид, состав управляющих элементов, их расположение (рис. 4.6). Для разработки эскиза форм можно воспользоваться средствами среды моделирования.



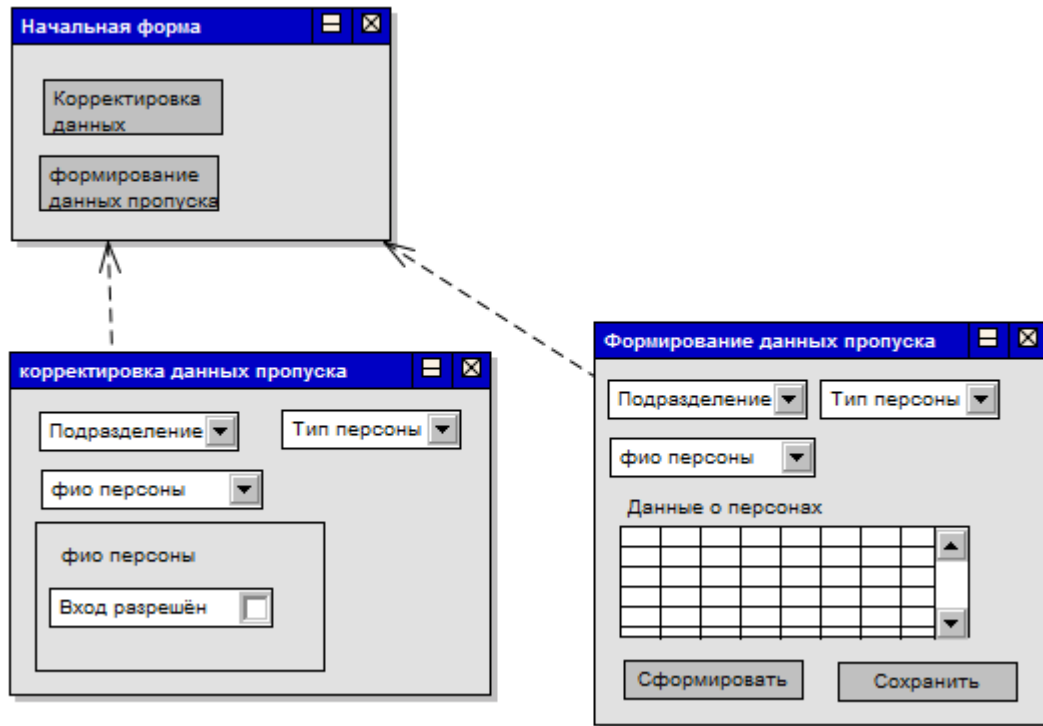


Рис.4.6 Пример проекта пользовательского интерфейса

При выполнении данной работы необходимо стремиться, чтобы пользовательский интерфейс реализовывал выявленные функциональные требования. Целесообразно, чтобы каждому требованию соответствовала отдельная экранная форма.

Процесс формирования требований включает так же предварительное определение требований к хранилищу данных.

**Определение требований к хранилищу данных.** В связи со специфическими особенностями данной задачи, она выделяется в отдельный подраздел.

Для проектирования базы данных необходимо, чтобы на уровне требований были определены особенности процесса хранения.

В качестве особенностей процесса хранения может быть задан тип используемой СУБД, ее архитектура, возможно предварительное определение состава информационных объектов БД. Предварительный состав хранимых объектов определяется на основе объектов-сущностей, выявленных в сценариях выполнения требований. Вопросы, связанные с проектированием хранилища данных, можно рассматривать в соответствии с [3].

В рассматриваемом примере в качестве объектов сущностей, данные о которых должны храниться в базе данных, предваритель-

но можно выделить – подразделения института, типы персон, журнал проходов.

Результатом процесса формирования требований является структурированная модель вариантов использования и описание, представленное в виде раздела отчёта к практической работе.

### 3. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Уточнить постановку задачи с преподавателем. Предварительно определить требования к системе.

2. В общей модели разработки создать пакет верхнего уровня (view) «формирование требований», в котором будут отображаться работы по формированию требований к системе. В пакете верхнего уровня создать подпакеты:

- состав требований;
- диаграммы вариантов использования;
- реализация требований;
- требования к пользовательскому интерфейсу.

3. Создать в подпакете «состав требований» пользовательскую диаграмму «требования к системе». На диаграмме с помощью элементов «requirement» (группа «custom» на панели инструментов) отобразить функциональные требования к системе. В свойствах каждого элемента описать, что подразумевается под данным функциональным требованием. При необходимости, разделить функциональные требования на подсистемы.

4. Разработка диаграмм вариантов использования.

4.1. Создать в подпакете «диаграммы вариантов использования» диаграммы вариантов использования (use case diagram) для каждой выделенной подсистемы.

4.2. Вынести на созданную диаграмму вариантов использования из обозревателя модели требования определённые в пункте 2.

4.3. Определить для каждого требования сценарии его реализующие. Отобразить на диаграмме сценарии с помощью элемента «use case». Соединить элементы «use case» соответствующими им требованиями отношениями зависимости со стереотипом «поддерживает». При необходимости структурировать варианты использования, задав между ними отношения включения и расширения.

4.4. Определить внешние сущности (пользователи и внешние системы), участвующие в функционировании системы. Отобразить внешние сущности с помощью элемента «actor».

4.5. Соединить внешние сущности с элементами «use case» отношением ассоциации. Отношение должно отображать возможность актанта запускать на выполнение соответствующий сценарий или получать от него значимый результат.

5. Определение содержание сценариев выполнения функций.

5.1. Декомпозировать сценарии выполнения функций в виде диаграмм деятельности.

5.1.1. Выявить объекты сущности и граничные объекты, участвующие в выполнении сценариев.

5.1.2. Описать объекты сценариев.

6. На основе выявленных в сценариях граничных объектов сформировать создать проект экранных форм и описать требования к пользовательскому интерфейсу.

7. Сформулировать предварительные требования к хранилищу данных.

#### 4. ЗАДАНИЯ ДЛЯ РАБОТЫ

Задаaniem для работы является формулировка требований к системе, автоматизирующие процессы, выявленные в прошлой работе.

#### 5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Содержание модели отображающей требования к системе.
2. Каким образом определяются функциональные требования к системе?
3. Что отображает диаграмма «вариантов использования», каковы её элементы, какие между ними могут быть связи?
4. Как описывается содержание сценария выполнения функционального требования?
5. Как определяется состав и содержание экранных форм, необходимых для реализации сценария?

## РАБОТА №5 ПРАКТИЧЕСКАЯ. РАЗРАБОТКА ПРОЕКТНОЙ АРХИТЕКТУРЫ СИСТЕМЫ

### 1. ЦЕЛЬ РАБОТЫ

Целью работы является освоение содержание процесса проектирования информационной системы. В связи с этим задачами работы является:

- определение классов объектов, реализующих функциональность системы;
- определение взаимодействий объектов, реализующих функциональность системы;
- определение методов классов объектов;
- выявление объектов, требующих длительное хранение.

### 2. ТЕОРЕТИЧЕСКИЕ ПОЛОЖЕНИЯ

При использовании среды Enterprise Architect, результаты процесса анализа и проектирования системы обычно отображаются в виде отдельной составляющей общей модели разработки ИС (пакета верхнего уровня – «view»). В рамках пакета «Анализ и проектирование» обычно выделяются подпакеты содержащие результаты решения составляющих подзадач:

- классы системы;
- реализация вариантов использования.

Классы являются главными элементами, через которые строится система при объектно-ориентированном подходе.

Выделяются следующие стереотипы классов, на основе которых строится система:

- граничные классы – классы объектов, реализующих формы, программные и аппаратные интерфейсы;
- классы объектов сущностей – классы объектов, реализующих объекты-сущности;
- классы управляющих объектов – классы объектов, реализующих контрольные объекты.

Классы анализа системы определяются на основе объектов, выявленных при разработке сценариев, реализующих требования.

Для каждого объекта, введённого в сценарии описывающие требования, должен быть определён класс. Кроме того, при формировании классов системы необходимо ориентироваться на классы объектов предметной области, выявленные в первой практической работе.

Результатом данной работы является систематизированная информация, включающая описание состава классов, их атрибутов, с указанием объектов, относящихся к данному классу.

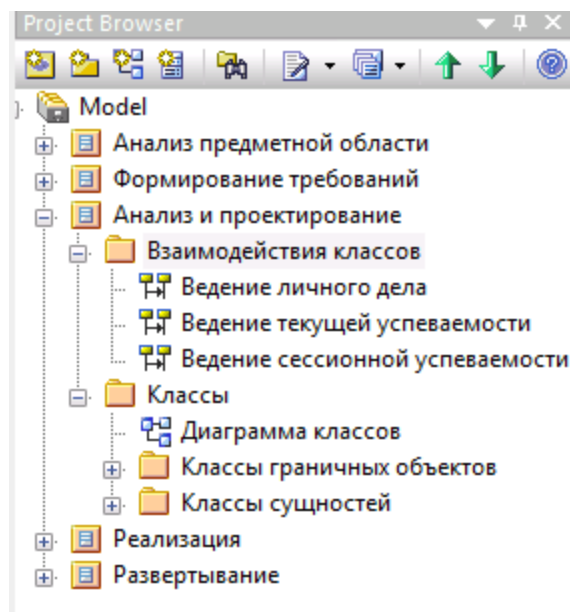


Рис.5.1 Содержание моделей разрабатываемых в процессе анализа и проектирования

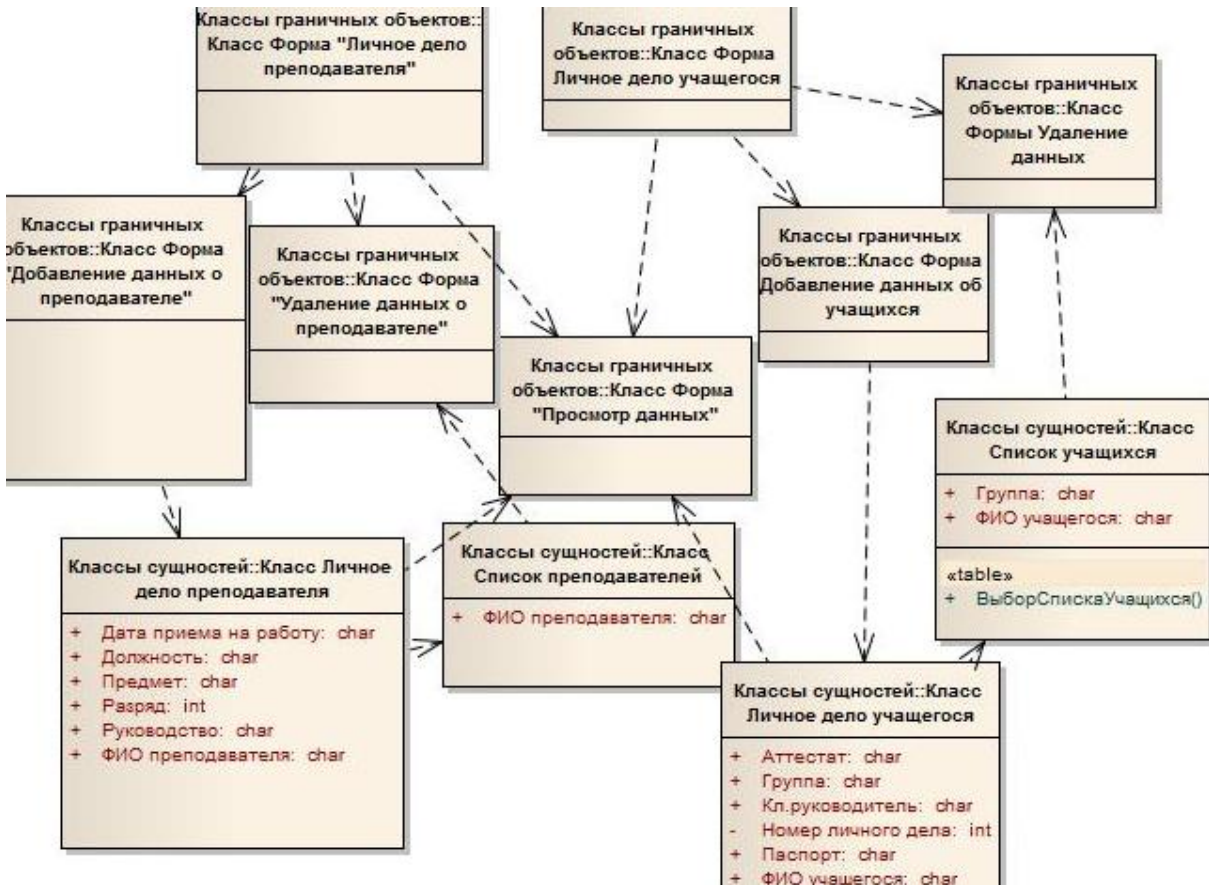


Рис.5.2 Диаграмма классов

Методы объектов определяются на основе построения диаграмм последовательностей (рис. 5.3.) для сценариев требований.

При построении диаграмм последовательностей каждое сообщение, направленное на объект, необходимо рассмотреть как операцию соответствующего класса.

**Результатом данной работы является систематизированная информация, включающая:**

- диаграммы последовательностей с описанием выявленных методов классов. Для каждого метода необходимо указать особенности его реализации;
- итоговая диаграмма классов, включая отношения между выявленными классами.

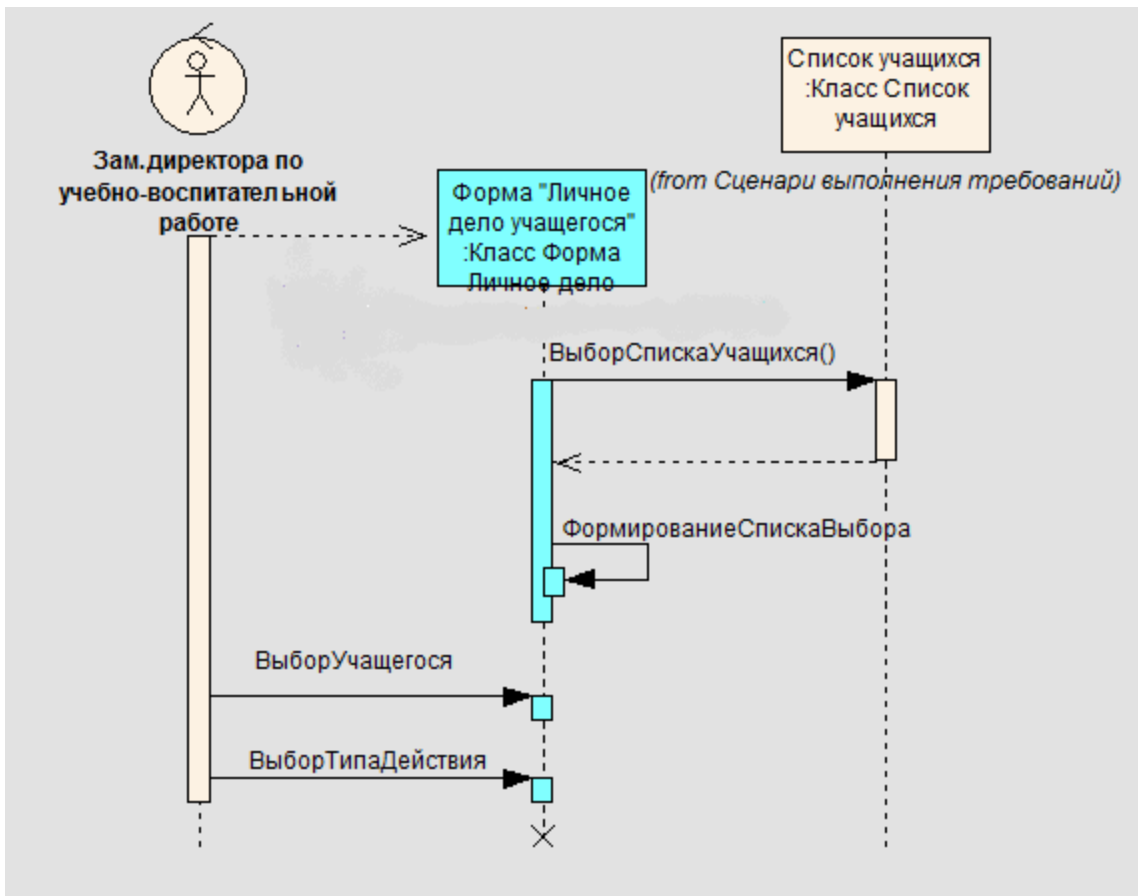


Рис.5.3 Диаграмма последовательностей для сценария «Выбор учащегося и типа действия»

### ***Выбор технологий реализации***

К технологическим вопросам относятся:

- языки программирования;
- методы доступа к данным;
- стандарты взаимодействия компонентов;
- системные сервисы;
- операционные системы (в рамках которых возможна реализация требований, предъявляемых к ИС);
- параметры среды развёртывания.

При выборе технологий проводится повторный анализ требований к системе. Выделяются те требования, связанные с технологиями реализации. Эти требования могут быть связаны:

- с принципами взаимодействия пользователя с системой;
- с требуемым принципом доступа к данным;
- с принципом обмена информацией;
- с физическими ограничениями инфраструктуры;
- с физическими требованиями к решению.

Рассматриваются риски, связанные с реализацией данных требований в рамках существующей инфраструктуры.

*При выборе операционной системы* необходимо указать, в рамках какой операционной системы возможна реализация требований. Выбор операционной системы должен обеспечивать предполагаемое развитие системы, ее расширение и развертывание.

*При выборе типа взаимодействия* пользователя с системой, технологии пользовательского интерфейса определяется тип интерфейса. Обычно выбор состоит из двух альтернатив:

- Web-интерфейс;
- интерфейс в стиле операционной системы или родной интерфейс.

Если интерфейс в стиле операционной системы обеспечивает большие возможности для отображения пользовательских функций, лучшие характеристики по оперативности обмена данными с другими компонентами, то Web-интерфейс дает лучшие возможности по обновляемости пользовательской компоненты, обеспечивает удаленное взаимодействие компонент.

*Выбор технологии взаимодействия пользовательских компонент с данными* определяется выбранной операционной системой, методами и средствами хранения данных доступностью и квалификацией пользователей. В среде Windows основной технологией доступа к данным является технология ADO.NET, реализованная через соответствующие компоненты сред проектирования Visual Studio и Delphi Studio, однако возможно применение и других технологий при соответствующем обосновании.

*Выбор языка программирования и среды* разработки должен обосновываться наличием необходимой квалификации, опытом работы и доступностью лицензионных версий соответствующих продуктов.

*При определении параметров среды развёртывания* необходимо установить какого типа физические узлы будут использоваться для развёртывания приложения, характеристики этих узлов. Характеристики узлов должны согласовываться с требованиями (ограничениями заказчика). Среда развёртывания должна быть представлена в виде начального варианта диаграммы развёртывания.



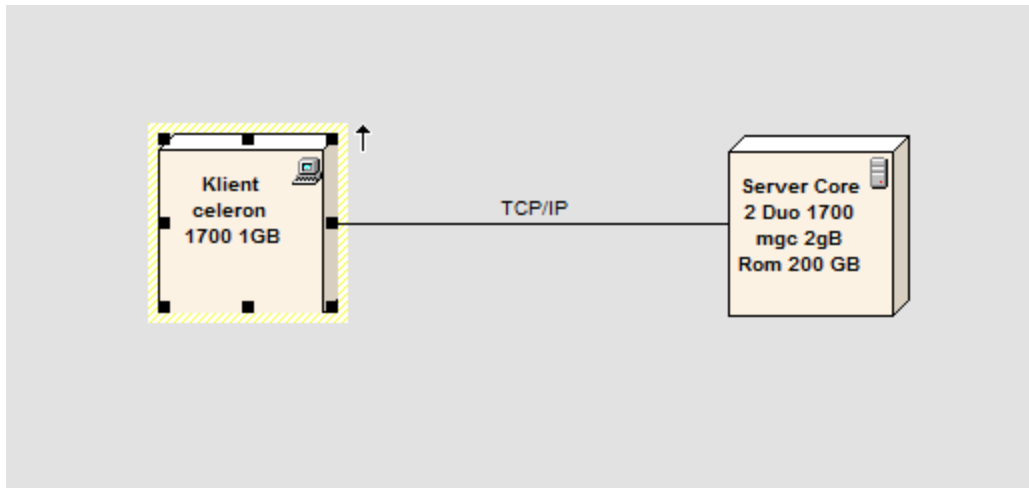


Рис.5.4 Начальный вариант среды развёртывания

### 3. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Уточнить сценарии, требующие реализации.
2. В общей модели разработки создать пакет верхнего уровня (view) «Анализ и проектирование», в котором будут отображаться результаты работы. В пакете верхнего уровня создать подпакеты:
  - классы;
  - кооперации классов.
3. Создать в подпакете «Кооперации классов» диаграмму классов «Классы системы». Для объектов, выявленных в системе создать соответствующие классы. Для имен классов использовать идентификаторы без служебных символов. На диаграмме с помощью элементов «class».
4. Отнести объекты, введённые в сценариях выполнения требований к введённым классам. (Контекстное меню объекта /advanced/instant classifier)
5. Разработка коопераций классов.
  - 5.1. Для каждого варианта использования создать диаграмму последовательностей.
  - 5.2. Вынести на диаграмму последовательностей объекты из соответствующей из диаграммы деятельности, детализирующей данный вариант использования. Объекты должны быть предварительно отнесены к классам (п. 4).
  - 5.3. При необходимости добавить объекты отсутствующие в диаграмме деятельности, но необходимые в соответствии с требуемой логикой выполнения сценария.

5.4. Выявить объект инициатор сценария, обычно это актант-внешняя сущность.

5.5. Отобразить логику работы сценария в виде обмена сообщениями между объектами.

5.6. Уточнить свойства сообщений (new, return и т. д.)

5.7. На основании сообщений в диаграмме последовательностей определить операции классов.

6. Описать выявленные классы для класса указать его тип (граничный, сущность), является ли постоянным или оперативным.

7. Описать сценарии, указать какие операции выявлены для каких классов.

## 4. ЗАДАНИЯ

Задаaniem для практической работы является разработка модели проектирования для системы, выданной в качестве задания к предшествующей практической работе.

## 5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Содержание модели проектирования.

2. Какие задачи решаются в процессе проектирования информационной системы?

3. Каким образом определяются классы информационной системы?

4. Каким образом в модели проектирования отображается реализация варианта использования?

5. Как строится диаграмма последовательностей?

6. Как определить состав операций классов информационной системы на основе диаграммы последовательностей, в которой участвуют объекты класса?

## РАБОТА №6 ПРАКТИЧЕСКАЯ. РАСЧЁТ ПОКАЗАТЕЛЕЙ ВЫПОЛНЕНИЯ БИЗНЕС-ПРОЦЕССА НА ОСНОВЕ МОДЕЛИ IDEFO

### 1. ЦЕЛЬ РАБОТЫ

**Цель работы** – приобретение студентами практических навыков проведения стоимостного анализа и создания свойств, определяемых пользователем.

### 2. ТЕОРЕТИЧЕСКИЕ ПОЛОЖЕНИЯ

#### *ABC анализ*

Как правило, в процессе моделирования строится несколько моделей ТО-ВЕ, из которых по какому-либо критерию выбирается лучшая. Для того чтобы определить качество созданной модели с точки зрения эффективности бизнес-процессов, необходима система метрики.

Аналитику предоставляются два инструмента для оценки модели:

- ABC анализ (Activity Based Costing);
- свойства, определяемые пользователем (User Defined Properties).

ABC включает следующие основные понятия:

- **объект затрат** – причина, по которой работа выполняется; обычно, основной выход работы, стоимость работ есть суммарная стоимость объектов затрат;
- **движитель затрат** – характеристика входов и управлений работы, которые влияют на то, как выполняется и как долго длится работа;
- **центры затрат**, которые можно трактовать как статьи расхода.

Результаты стоимостного анализа могут существенно повлиять на очередность выполнения работ. ABC позволяет оценить стоимостные и временные характеристики системы.

### ***Свойства, определяемые пользователем (UDP)***

Если стоимостных показателей недостаточно, имеется возможность внесения собственных метрик – свойств, определенных пользователем. UDP позволяет произвести дополнительный анализ.

Свойства, определяемые пользователем, предназначены для оценки работ модели без суммирующих подсчетов по критериям, вытекающим из особенностей моделируемой предметной области.

Каждой работе можно поставить в соответствие несколько свойств. Список доступных свойств хранится в словаре (UDP Dictionary). На свойства можно ссылаться по ключевому слову (словарь UDP Keyword List). Одно ключевое слово может соответствовать нескольким свойствам и одно свойство может иметь несколько ключевых слов.

Имеется возможность задания 18 различных типов свойств, в том числе управляющих команд и массивов.

#### Типы UDP и их использование

Таблица 6.1

Тип	Использование
Text	При задании свойства стрелки или работы просто вносятся текст, например это может быть просто дополнительное пояснение
Paragraph Text	Текст в несколько строк
Integer	Целое число, например значение свойства «Количество баллов»
Command	Командная строка.
Character	Один символ
Date mm/dd/yy (yy)	Дата
Real Number	Действительное число, например значение свойства «Потребление электроэнергии, кВт-ч»
Text List (Single selection)	Массив строк. Значения свойства этого типа должны быть определены в диалоге UDP Dictionary (поле Value). Объекту модели можно присваивать только одно значение из предварительно заданного списка
Integer List (Single selection)	Массив целых чисел. Значения свойства этого типа должны быть определены в диалоге UDP Dictionary (поле Value). Объекту модели можно присваивать только одно значение из предварительно заданного списка
Command List	Массив команд. Значения свойства этого типа должны быть определены в диалоге UDP Dictionary (поле

Тип	Использование
	Value). Объекту модели можно присваивать только одно значение' из предварительно заданного списка
Date List mm/dd/yy (yy)	Массив дат. Значения свойства этого типа должны быть определены в диалоге UDP Dictionary (поле Value). Объекту модели можно присваивать только одно значение из предварительно заданного списка
Real Number List (Single selection)	Массив действительных чисел. Значения свойства этого типа должны быть определены в диалоге UDP Dictionary (поле Value). Объекту модели можно присваивать только одно значение из предварительно заданного списка
Character List (Single selection)	Массив символов. Значения свойства этого типа должны быть определены в диалоге UDP Dictionary (поле Value). Объекту модели можно присваивать только одно значение из предварительно заданного списка
Text List (Multiple selections)	Массив строк (множественный выбор). Значения свойства этого типа должны быть определены в диалоге UDP Dictionary (поле Value). Объекту модели можно присваивать одновременно несколько значений из предварительно заданного списка
Integer List (Multiple selections)	Массив целых чисел (множественный выбор). Значения свойства этого типа должны быть определены в диалоге UDP Dictionary (поле Value). Объекту модели можно присваивать одновременно несколько значений из предварительно заданного списка
Date List (Multiple selections)	Массив дат (множественный выбор). Значения свойства этого типа должны быть определены в диалоге UDP Dictionary (поле Value). Объекту модели можно присваивать одновременно несколько значений из предварительно заданного списка
Real Number List (Multiple selections)	Массив действительных чисел (множественный выбор). Значения свойства этого типа должны быть определены в диалоге UDP Dictionary (поле Value). Объекту модели можно присваивать одновременно несколько значений из предварительно заданного списка
Character List (Multiple selections)	Массив символов (множественный выбор). Значения свойства этого типа должны быть определены в диалоге UDP Dictionary (поле Value). Объекту модели можно присваивать одновременно несколько значений из предварительно заданного списка

### ***Задание стоимости работ***

При проведении стоимостного анализа сначала задаются единицы измерения времени и денег.

Для задания единиц измерения следует вызвать диалог Model Properties (меню Model/Model Properties) вкладка ABC Units (рис. 6.1).

Каждому центру затрат необходимо дать описание в Cost Center (пункт Dictionary/Cost Center).

Для задания стоимости работы (для каждой работы на диаграмме) следует щелкнуть правой кнопкой мыши по работе и на всплывающем меню выбрать Costs. Во вкладке Costs (рис. 6.2) указывается частота проведения данной работы в рамках общего процесса (Frequency) и продолжительность (Duration). Затем выбирается в списке один из центров затрат и в окне Costs задать его стоимость. Аналогично назначаются суммы по каждому центру затрат.

Если в процессе назначения стоимости возникает необходимость внесения дополнительных центров затрат, диалог Cost Center Editor вызывается прямо из диалога Activity Cost соответствующей кнопкой.

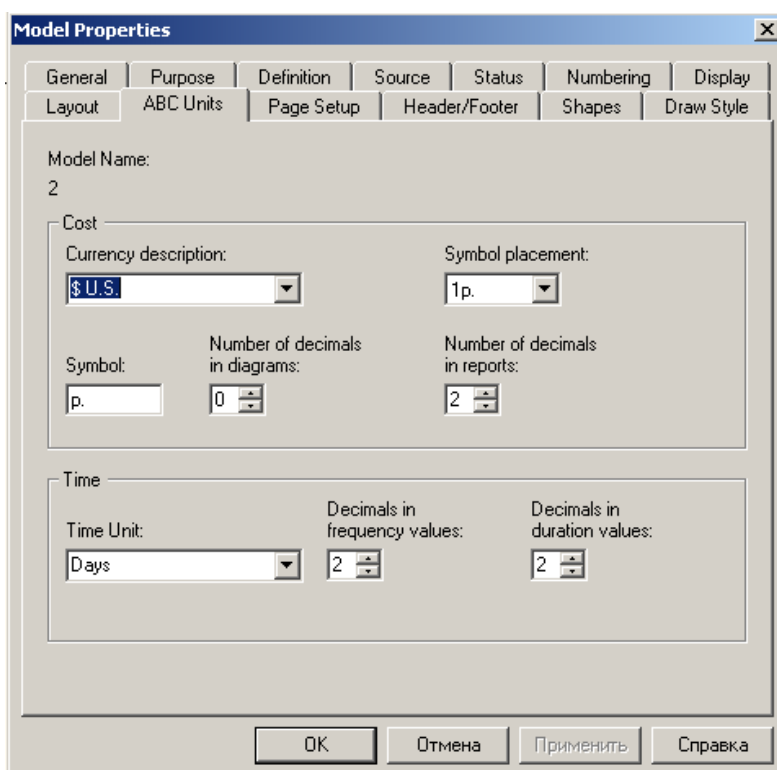


Рис.6.1 Настройка единиц измерения валюты и времени

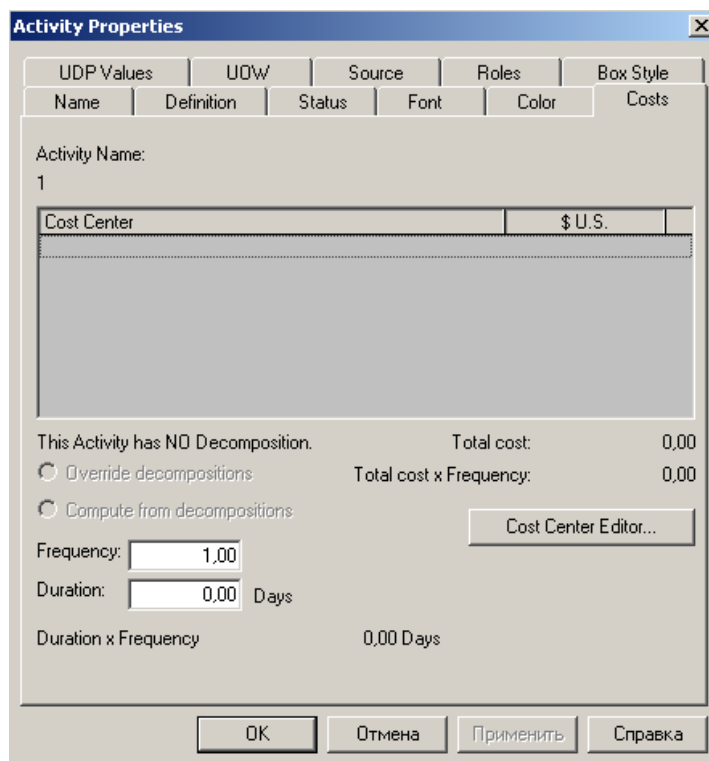


Рис.6.2 Задание стоимости работ в диалоге Activity Cost

Общие затраты по работе рассчитываются как сумма по всем центрам затрат. При вычислении затрат вышестоящей (родительской) работы сначала вычисляется произведение затрат дочерней работы на частоту работы (число раз, которое работа выполняется в рамках проведения родительской работы), затем результаты складываются. Если во всех работах модели включен режим *Compute from Decompositions*, подобные вычисления автоматически проводятся по всей иерархии работ снизу вверх.

Такой принцип подсчета справедлив, если работы выполняются последовательно.

Если схема выполнения более сложная (например, работы производятся альтернативно), можно отказаться от подсчета и задать итоговые суммы для каждой работы вручную (*Override Decompositions*). В этом случае результаты расчетов с нижних уровней декомпозиции будут игнорироваться, при расчетах на верхних уровнях будет учитываться сумма, заданная вручную.

Результаты стоимостного анализа наглядно представляются на специальном отчете *Activity Cost Report* (меню *Tools/Report/Activity Cost Report*) (рис. 6.3).



Рис.6.3 Диалог ройки отчета по стоимости работ

Результаты отображаются и непосредственно на диаграммах. В левом нижнем углу прямоугольника работы может показываться либо стоимость, либо продолжительность, либо частота проведения работ

По умолчанию на диаграмме отображается стоимость работы, если требуется изменить настройку, то необходимо воспользоваться диалогом настройки свойств модели (Model/Model Properties/Display – ABC Data и ABC Units);

### ***Задание свойств UDP***

Для описания UDP служит диалог UDP Dictionary Editor (меню Model/UDP Dictionary Editor).

Для задания нового свойства (UDP) следует в словаре UDP Dictionary перейти к нижней строке списка и дважды щёлкнуть по полю Name, ввести имя, указать тип свойства и выбрать необходимые ключевые слова.

В качестве свойств может рассматриваться любая мера, позволяющая качественно или количественно оценить работу или связь, например:



– оценка качества или чего-то другого путём выбора из списка возможных значений (очень высокое, высокое, среднее, низкое, очень низкое) и т. п.;

- потребляемая мощность или подобная количественная оценка;
- сопровождающая документация, спецификации и т. п.

По результатам анализа, основанного на свойствах, определяемых пользователем, можно сформировать отчёт (Tools/Report/Diagram Object Report) задав в поле User-Defined property необходимые свойства для вывода. Свойства для вывода задаются с помощью кнопки UDP Filter.

### 3. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Согласуйте с преподавателем необходимый набор стоимостных центров и свойств пользователя.
2. Задайте для процессов стоимость их выполнения по стоимостным центрам.
3. Сформируйте отчёт по стоимости работ.
4. Задайте свойства пользователя для процессов декомпозиции.
5. Оформите отчёт и представьте его преподавателю.

### 4. ЗАДАНИЕ

Заданием для работы является процесс, рассмотренный в первой практической работе.

### 5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Какие виды инструментов для оценки модели вы знаете?
2. Что такое объект затрат?
3. Что такое центры затрат? (стоимостные центры)?
4. Какие характеристики для процесса задаются при ABC анализе?
5. Для чего предназначены свойства, определяемые пользователем?
6. Кратко описать типы свойств.
7. Как рассчитываются общие затраты по работе?
8. Что рассматривается в качестве свойств UDP?

## РАБОТА №7 ПРАКТИЧЕСКАЯ. РЕИНЖИНИРИНГ БИЗНЕС-ПРОЦЕССА МЕТОДОМ ИНТЕГРАЦИИ

### 1. ЦЕЛЬ РАБОТЫ

Целью работы является получение практических навыков реинжиниринга бизнес-процессов.

### 2. ТЕОРЕТИЧЕСКИЕ ПОЛОЖЕНИЯ

Реинжиниринг (reengineering) – перепроектирование, перестройка, реконструкция, реорганизация.

Реинжиниринг используется периодически для радикальных преобразований, обеспечивающих существенное повышение эффективности (улучшение показателей деятельности компании в десятки раз). А в период между этими «скачками» применяются методы постепенного улучшения для «настройки» усовершенствованных процессов (технологии управления качеством)).

Причины когда возникает необходимость реинжиниринга:

- диверсификация товаров и услуг (ориентация на различные сегменты рынка), вызывающая многообразие бизнес-процессов;
- изменение сектора рынка (переход от массового производства к работе по индивидуальным заказам, требующая высокую степень адаптации базового бизнес-процесса к потребностям клиента);
- внедрение новых технологий (инновационных проектов), затрагивающих все основные бизнес-процессы предприятия;
- расширение кооперативных связей с партнерами предприятия и поставщиками материалов, обуславливающих альтернативность построения бизнес-процесса.

Принципы БИ.

Важнейшими принципами инжиниринга (реинжиниринга) бизнес-процессов являются:

- интеграция – соединение отдельных бизнес-процессов в единый бизнес-процесс, с точки зрения управления, представления.
- горизонтальное сжатие процесса – несколько рабочих процедур объединяются в одну;
- вертикальное сжатие процесса – сплющивание вертикали;

- распараллеленность процесса – шаги процесса выполняются в естественном порядке, работа выполняется в том месте, где это целесообразно.

Важным акцентом в принципах реинжиниринга является использование новых информационных технологий. Информационные технологии (ИТ) играют критически важную роль в ВРР. Они рассматриваются как важная составная часть производственных процессов. Но реинжиниринг – это не то же самое что автоматизация, при которой с помощью ИТ автоматизируются существующие бизнес-процессы со всеми их недостатками. Реинжиниринг использует ИТ для автоматизации новых, реконструированных процессов. Более того, сама реконструкция бизнес-процессов, как правило, становится возможной только при использовании информационных технологий, так, как ИТ зачастую меняют сущность процессов.

### 3. ПОРЯДОК ВЫПОЛНЕНИЯ И ЗАДАНИЯ ДЛЯ РАБОТЫ

1. Построить в одной из нотаций диаграмму «as-is» отображающую исходное содержание заданного бизнес-процесса.
2. Выделить деятельности (процессы второго уровня) интеграция которых возможна.
3. Сформировать интегрированные деятельности, на основе выявленных групп интегрируемых процессов. Описать данные процессы.
4. Построить диаграмму «to-be» отображающую полученное содержания рассматриваемого бизнес-процесса.

### 5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Определение бизнес-процесса.
2. Что такое реинжиниринг, когда он выполняется?
3. Какие принципы реинжиниринга выделяются?
4. В чём состоит принцип интеграции?

## **РАБОТА №8 ПРАКТИЧЕСКАЯ. РЕИНЖИНИРИНГ БИЗНЕС-ПРОЦЕССОВ МЕТОДОМ ГОРИЗОНТАЛЬНОГО И/ИЛИ ВЕРТИКАЛЬНОГО СЖАТИЯ**

### **1. ЦЕЛЬ И ЗАДАЧИ РАБОТЫ**

Целью работы является получение практических навыков выполнения реинжиниринга бизнес-процессов.

### **2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ**

Принципы БИ.

Важнейшими принципами инжиниринга (реинжиниринга) бизнес-процессов являются:

- горизонтальное сжатие процесса;
- вертикальное сжатие процесса.

При горизонтальном сжатии рабочих процедур объединяются в одну. Вследствие этого рабочие места становятся многофункциональными. То есть на одном рабочем месте выполняется несколько различных операций. Для обеспечения этой возможности должно быть внедрено соответствующее оборудование.

При вертикальном сжатии процесса происходит «сплющивание» вертикали управления, то есть исполнители получают возможность принимать самостоятельные решения при выполнении бизнес-процесса. Следствием «вертикального сжатия» является повышение ответственности, заинтересованности в результатах своего труда работника.

При проведении реинжиниринга выделяются следующие этапы:

1. Разработка модели to-be.
  - 1.1. Разработка образа будущей компании, моделей, схем.
  - 1.2. Спецификация целей компании.
2. Выполнение обратного инжиниринга.
  - 2.1. Создание модели существующего предприятия.
  - 2.2. Идентификация процессов на предприятии.
  - 2.3. Документирование потоков работ.
  - 2.4. Определение стоимости существующих процессов.
3. Выполнение прямого инжиниринга.
  - 3.1. Перепроектирование бизнес-процессов.

3.1.1. Идентификация необходимых изменений в работе персонала и информационных технологий.

3.2. Разработка новой организационной структуры.

3.3. Проектирование работ, разработка системы мотивации работников.

3.4. Организация командной работы.

3.5. Разработка вопросов управления качеством.

4. Разработка (модернизация информационной системы).

4.1. Разработка требований к модернизации или разработке информационной системы.

4.2. Внесение изменений или внедрение новой информационной системы.

5. Внедрение изменений в БП.

5.1. Подготовка персонала.

5.2. Интеграция и тестирование информационной системы.

5.3. Внедрение перепроектированных процессов

### 3. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Построить в одной из нотаций диаграмму «as-is» отображающую исходное содержание заданного бизнес-процесса.

2. Выделить деятельности (процессы второго уровня) для которых возможен реинжиниринг методом горизонтального или вертикального сжатия.

3. Сформировать преобразованные деятельности, на основе выявленных групп. Описать полученные деятельности..

4. Построить диаграмму «to-be» отображающую полученное содержания рассматриваемого бизнес-процесса.

### 4. ЗАДАНИЕ

Задаaniem для практической работы является установка и настройка экземпляра сервера на домашнем компьютере. И подготовка отчёта о настройке.

## 5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Определение бизнес-процесса.
2. Что такое реинжиниринг, когда он выполняется?
3. Какие принципы реинжиниринга выделяются?
4. В чём состоит принцип горизонтального сжатия? Какие последствия его применения с точки зрения организационной структуры? Особенности выполнения отдельных действий? Выполнения всего процесса?
5. В чём состоит принцип вертикального сжатия? Какие последствия его применения с точки зрения организационной структуры? Особенности выполнения отдельных действий? Выполнения всего процесса?

## **РАБОТА №9 ПРАКТИЧЕСКАЯ. РАЗРАБОТКА ТРЕБОВАНИЙ БЕЗОПАСНОСТИ ИНФОРМАЦИОННОЙ СИСТЕМЫ**

### **1. ЦЕЛЬ И ЗАДАЧИ РАБОТЫ**

Целью работы является получение практических навыков разработки требований безопасности к информационной системе.

### **2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ**

Критерии оценки безопасности ИС отображены в стандарте ISO/IEC 15408 «Критерии оценки безопасности информационных технологий» (издан 1 декабря 1999 года) относится к оценочным стандартам.

Стандарт содержит два основных вида требований безопасности:

- функциональные – соответствуют активному аспекту защиты – предъявляемые к функциям безопасности и реализующим их механизмам;
- требования доверия – соответствуют пассивному аспекту – предъявляемые к технологии и процессу разработки и эксплуатации.

Угрозы безопасности в стандарте характеризуются следующими параметрами:

- источник угрозы;
- метод воздействия;
- уязвимые места, которые могут быть использованы;
- ресурсы (активы), которые могут пострадать.

Для структуризации пространства требований, в стандарте введена иерархия класс – семейство – компонент – элемент.

Классы определяют наиболее общую, «предметную» группировку требований (например, функциональные требования подотчетности).

Семейства в пределах класса различаются по строгости и другим тонкостям требований.

Компонент – минимальный набор требований, фигурирующий как целое.

Элемент – неделимое требование.

Между компонентами могут существовать зависимости, которые возникают, когда компонент сам по себе недостаточен для достижения цели безопасности.

Подобный принцип организации защиты напоминает принцип программирования с использованием библиотек, в которых содержатся стандартные (часто используемые) функции, из комбинаций которых формируется алгоритм решения.

Стандарт позволяет с помощью подобных библиотек (компонент) формировать два вида нормативных документов: профиль защиты и задание по безопасности.

Профиль защиты представляет собой типовой набор требований, которым должны удовлетворять продукты и/или системы определенного класса (например, операционные системы на компьютерах в правительственных организациях).

Задание по безопасности содержит совокупность требований к конкретной разработке, выполнение которых обеспечивает достижение поставленных целей безопасности.

Функциональный пакет – это неоднократно используемая совокупность компонентов, объединенных для достижения определенных целей безопасности.

Базовый профиль защиты должен включать требования к основным (обязательным в любом случае) возможностям. Производные профили получаются из базового путем добавления необходимых пакетов расширения, то есть подобно тому, как создаются производные классы в объектно-ориентированных языках программирования.

### ***Функциональные требования***

Все функциональные требования объединены в группы на основе выполняемой ими роли или обслуживаемой цели безопасности. Всего в «Общих критериях» представлено 11 функциональных классов, 66 семейств, 135 компонентов.

«Общие критерии» включают следующие классы функциональных требований:

- Идентификация и аутентификация.
- Защита данных пользователя.
- Защита функций безопасности (требования относятся к целостности и контролю данных сервисов безопасности и реализующих их механизмов).



- Управление безопасностью (требования этого класса относятся к управлению атрибутами и параметрами безопасности).
- Аудит безопасности (выявление, регистрация, хранение, анализ данных, затрагивающих безопасность объекта оценки, реагирование на возможное нарушение безопасности).
- Доступ к объекту оценки.
- Приватность (защита пользователя от раскрытия и несанкционированного использования его идентификационных данных).
- Использование ресурсов (требования к доступности информации).
- Криптографическая поддержка (управление ключами).
- Связь (аутентификация сторон, участвующих в обмене данными).
- Доверенный маршрут/канал (для связи с сервисами безопасности).

Рассмотрим содержание одного из классов.

Класс функциональных требований «Использование ресурсов» включает три семейства.

- Отказоустойчивость. Требования этого семейства направлены на сохранение доступности информационных сервисов даже в случае сбоя или отказа. В стандарте различаются активная и пассивная отказоустойчивость. Активный механизм содержит специальные функции, которые активизируются в случае сбоя. Пассивная отказоустойчивость подразумевает наличие избыточности с возможностью нейтрализации ошибок.

- Обслуживание по приоритетам. Выполнение этих требований позволяет управлять использованием ресурсов так, что низкоприоритетные операции не могут помешать высокоприоритетным.

- Распределение ресурсов. Требования направлены на защиту (путем применения механизма квот) от несанкционированной монополизации ресурсов.

Аналогично и другие классы включают наборы семейств требований, которые используются для формулировки требований к системе безопасности.

Общие критерии – достаточно продуманный и полный документ с точки зрения функциональных требований и именно на этот стандарт безопасности ориентируются соответствующие организации в нашей стране и в первую очередь Гостехкомиссия РФ.

### ***Требования доверия***

Вторая форма требований безопасности в «Общих критериях» – требования доверия безопасности.

Установление доверия безопасности основывается на активном исследовании объекта оценки.

Форма представления требований доверия, та же, что и для функциональных требований (класс – семейство – компонент).

Всего в «Общих критериях» 10 классов, 44 семейства, 93 компонента требований доверия безопасности.

Классы требований доверия безопасности:

- Разработка (требования для поэтапной детализации функций безопасности от краткой спецификации до реализации).
- Поддержка жизненного цикла (требования к модели жизненного цикла, включая порядок устранения недостатков и защиту среды разработки).
- Тестирование.
- Оценка уязвимостей (включая оценку стойкости функций безопасности).
- Поставка и эксплуатация.
- Управление конфигурацией.
- Руководства (требования к эксплуатационной документации).
- Поддержка доверия (для поддержки этапов жизненного цикла после сертификации).
- Оценка профиля защиты.
- Оценка задания по безопасности.

Применительно к требованиям доверия (для функциональных требований не предусмотрены) в «Общих критериях» введены оценочные уровни доверия (их семь), содержащие осмысленные комбинации компонентов.

Степень доверия возрастает от первого к седьмому уровню. Так, оценочный уровень доверия 1 (начальный) применяется, когда угрозы не рассматриваются как серьезные, а оценочный уровень 7 применяется к ситуациям чрезвычайно высокого риска.

### 3. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Проанализировать заданную систему.
2. Сформулировать пять функциональных требований безопасности актуальных для системы рассматриваемого типа.
3. Сформулировать пять требований доверия к безопасности, характерных для системы рассматриваемого типа.
4. Создать отчёт по работе, содержащий перечень сформированных требований.

### 4. ЗАДАНИЯ ДЛЯ ВЫПОЛНЕНИЯ

В качестве задания для работы преподавателем задаётся одна из систем, знакомая студенту, например:

- система контроля входа в корпус;
- система записи на приём в поликлинику;
- система пожарного контроля в корпусе;
- ИС КузГТУ.

### 5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Какие виды требований включает стандарт ISO/IEC 15408?
2. Чем отличаются функциональные требования от требований доверия?
3. В чем заключается иерархический принцип «класс – семейство – компонент – элемент»?
4. Какова цель требований по отказоустойчивости информационных систем?
5. Сколько классов функциональных требований?

# **РАБОТА №10 ПРАКТИЧЕСКАЯ. РАЗРАБОТКА ФУНКЦИОНАЛЬНОЙ СПЕЦИФИКАЦИИ И ТЕХНИЧЕСКОГО ЗАДАНИЯ НА РАЗРАБОТКУ ИС**

## **1. ЦЕЛЬ И ЗАДАЧИ РАБОТЫ**

Целью выполнения работы является получение практических навыков разработки технического задания на создание информационной системы. Для выполнения целей работы должны быть решены следующие задачи:

- Студенты должны ознакомиться с содержанием технического задания.
- Получить навыки формулировки целей разработки информационной системы.
- Формулировки требований к ИС.

## **2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ**

Техническое задание (ТЗ) является основным документом, в котором определяются требования к автоматизированной системе и порядок её создание. В нём закрепляются итоги начальной фазы разработки системы, в результате которой должны быть определены границы системы и принято решение о разработке системы. Техническое задание составляется совместно с заказчиком системы и является главным документом, на основе которого производится приёмка системы. Содержание технического задания определено в ГОСТ 34.602-89.

Согласно ГОСТ ТЗ должно содержать следующие подразделы, которые могут быть на подразделы:

1. Общие сведения.
2. Назначение и цели создания системы.
3. Характеристика объектов автоматизации.
4. Требования к системе.
5. Состав и содержание работ по созданию системы.
6. Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие.
7. Требования к документированию.
8. Источники разработки.

9. В ТЗ могут включаться приложения.

В зависимости от вида, назначения, специфических особенностей объекта автоматизации и условий функционирования системы допускается оформлять отдельные разделы ТЗ в виде приложений, вводить дополнительные разделы, исключать или дополнять.

### **Раздел 1. Общие сведения.**

1. Общие сведения о системе:

1.1. Полное наименование системы и её условное обозначение.

1.2. Указание на заказчика и разработчика системы

1.3. Перечень документов, на основе которых производится разработка.

1.4. Плановые сроки начала и окончания работа по созданию системы.

1.5. Сведения об источниках и порядке финансирования работ.

1.6. Порядок оформления и предъявления заказчику результатов работ по созданию системы.

*Полное наименование системы* должно указывать на объекты автоматизации, то есть на процессы, которые автоматизирует система, то есть связано с темой автоматизации. Например «Система формирования заказов на доставку», однако от данного принципа возможны отклонения. Условное обозначение должно облегчать описание системе, быть легко произносимым, и может нести в себе рекламные функции. Например, система «Быстрый Документ», «Документ доставки». Условное обозначение может быть составлено в виде аббревиатуры полного наименования.

В *перечень документов*, на основании которых создается система, включается указание на договора, приказы, соглашения, на основе которых производится разработки.

*Порядок оформления и предъявления заказчику* результатов работ по созданию системы подразумевает возможность предоставления предварительных версий, компонентов системы до окончания всех работ.

### **Раздел 2. Назначение и цели создания системы.**

Назначение системы подразумевает общее указание на процесс, автоматизация которого обеспечивается системой.

Формулировка цели создания системы должна характеризовать показатели работы автоматизируемого процесса, на которых

повлияет использование системы. Цель должна указывать на конкретный показатель и величину его изменения.

**Раздел 3. Характеристика объекта автоматизации** должна включать описание процесса, который будет автоматизировать разрабатываемая система. Общая характеристика процесса, его входы, выходы, особенности выполнения, влияние на деятельность предприятия в целом.

#### **Раздел 4. Требования к системе.**

Раздел включает описание требований, характеризующих систему как единое целое (п. 4.1) и характеристику отдельных функций (п. 4.2).

**4.1. Требования к системе в целом** содержит следующие подпункты.

*Требования к структуре и функционированию системы.* В данном подразделе должна быть приведена общая характеристика структуры разрабатываемой системы. Тип архитектуры, используемое системное программное обеспечение, приведена общая характеристика основных программных и аппаратных компонент и особенности связей между ними.

В подразделе *«Требования к численности и квалификации персонала системы и режиму его работ»* необходимо привести численность, квалификация и режим работы персонала, который должен обеспечивать работоспособность системы.

В подразделах *«Требования к надежности»* и *«Требования безопасности»* необходимо привести конкретные показатели надежности и требования безопасности, которые должна обеспечивать система.

*«Требования к патентной чистоте»* должны оговариваться лицензии, указываться тип свободного программного обеспечения используемый при разработке.

Пункт *«Требования по стандартизации и унификации»* должен указывать на стандарты, используемые при разработке. И используемые принципы унификации проектных и программных решений.

*Дополнительные требования* включают: требования к оснащению системы устройствами для обучения персонала (тренажерами, другими устройствами аналогичного назначения) и документацией на них; требования к сервисной аппаратуре, стендам для проверки элементов системы; требования к системе, связанные с особыми ус-

ловиями эксплуатации; специальные требования по усмотрению разработчика или заказчика системы.

#### **4.2. Требование к функциям (задачам).**

В подразделе «Требование к функциям (задачам)», выполняемым системой, приводят: по каждой подсистеме перечень функций (в том числе обеспечивающих взаимодействие частей системы), подлежащих автоматизации.

При создании системы в две или более очереди (итерации) – перечень функциональных подсистем, отдельных функций, вводимых в действие в каждой итерации. Для каждой функции приводится её описание, временной регламент реализации, требования к качеству реализации функции, к форме представления выходной информации, характеристики необходимой точности и времени выполнения, достоверности выдачи результатов, перечень и критерии отказов для каждой функции, по которой задаются требования по надежности. *Описание функций при согласовании с заказчиком можно привести в виде моделей UML или другой нотации.*

#### **4.3. Требования к видам обеспечения.**

В подразделе «Требования к видам обеспечения» в зависимости от вида системы приводят требования к математическому, информационному, лингвистическому, программному, техническому, метрологическому, организационному, методическому и другим видам обеспечения системы.

Для математического обеспечения системы приводят требования к составу, области применения (ограничения) и способам, использования в системе математических методов и моделей, типовых алгоритмов и алгоритмов, подлежащих разработке. Например, математические методы и алгоритмы, используемые для шифрования/дешифрования данных.

Для информационного обеспечения системы приводят требования к составу, структуре и способам организации данных в системе, к информационному обмену между компонентами системы, к информационной совместимости со смежными системами, по применению систем управления базами данных.

Для лингвистического обеспечения системы приводят требования к применению в системе языков программирования высокого уровня, языков взаимодействия пользователей и технических средств системы, а также требования к кодированию и декодированию

нию данных, к языкам ввода-вывода данных, языкам манипулирования данными, средствам описания предметной области (объекта автоматизации), к способам организации диалога.

Для программного обеспечения системы приводят перечень покупных программных средств, а также требования, к независимости программных средств от используемых аппаратных средств и операционной среды, к качеству программных средств, а также к способам его обеспечения и контроля.

Для технического обеспечения системы приводят требования к видам технических средств, в том числе к видам комплексов технических средств, программно-технических комплексов и других комплектующих изделий, допустимых к использованию в системе, к функциональным, конструктивным и эксплуатационным характеристикам средств технического обеспечения системы.

Для организационного обеспечения приводят требования к структуре и функциям подразделений, участвующих в функционировании системы или обеспечивающих эксплуатацию к организации функционирования системы и порядку взаимодействия персонала системы и персонала объекта автоматизации к защите от ошибочных действий персонала системы.

**4.4. Требования к информационному обмену между компонентами системы.**

**4.5. Требования информационной совместимости со смежными системами.**

**4.6. Требования по использованию общесоюзных и зарегистрированных республиканских, отраслевых классификаторов, унифицированных документов и классификаторов, действующих на данном предприятии.**

При разработке системы должны быть применены классификаторы, разработанные для кодирования информации (например, документ «Инструкция по кодированию информации»).

**4.7. Требование к программному обеспечению**

Например, «Система должна работать в среде операционной системы Windows XP sp3 и более поздними версиями..», «Для разработки системы не должно использоваться дополнительное покупное программное обеспечение».

**Раздел 5. Состав и содержание работ по созданию (развитию) системы.**



### 3. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Работа выполняется на основе предшествующей работы по формированию требований. В результате выполнения работы должен быть оформлен документ «Техническое задание».

### 4. ЗАДАНИЯ ДЛЯ ВЫПОЛНЕНИЯ

Задаaniem для работы является разработка технического задания для ИС, являющейся темой работы по формулировке требований.

### 5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что такое техническое задание?
2. Какие подразделы выделяются в техническом задании?
3. Что включается в первый и второй подразделы технического задания? Как формулируется назначение и цель системы?
4. Что является объектом автоматизации?
5. Какие виды требований выделяются в техническом задании?
6. Где в техническом задании отображаются функциональные требования к системе?
7. Какие виды обеспечений выделяются у информационной системы в техническом задании?

## РАБОТА №11 ПРАКТИЧЕСКАЯ. РАЗРАБОТКА РУКОВОДСТВА ПОЛЬЗОВАТЕЛЯ ИС

### 1. ЦЕЛЬ И ЗАДАЧИ РАБОТЫ

Целью выполнения работы является получение практических навыков разработки руководства пользователя информационной системы.

### 2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Руководство пользователя формируется IT-специалистом документации на систему и её.

Для руководства пользователя разработано ряд стандартов, в частности ГОСТ РД 50-34.698-90. В соответствии с данными стандартом руководство пользователя содержит следующие разделы.

Введение.

1. Назначение и условия применения.
2. Подготовка к работе.
3. Описание операций.
4. Аварийные ситуации.
5. Рекомендации по освоению.

Содержание разделов будет описано на основе конкретного примера.

#### ***1. Введение***

В разделе «Введение» указывают:

- область применения;
- краткое описание возможностей;
- уровень подготовки пользователя;
- перечень эксплуатационной документации, с которой необходимо ознакомиться пользователю.

#### ***1.1. Область применения***

Требования настоящего документа применяются при:

- предварительных комплексных испытаниях;
- опытной эксплуатации;
- приемочных испытаниях;
- промышленной эксплуатации.

### ***1.2. Краткое описание возможностей***

Информационно-аналитическая система Корпоративное Хранилище Данных (ИАС КХД) предназначена для оптимизации технологии принятия тактических и стратегических управленческих решений конечными бизнес-пользователями на основе информации о всех аспектах финансово-хозяйственной деятельности Компании.

ИАС КХД предоставляет возможность работы с регламентированной и нерегламентированной отчетностью.

При работе с отчетностью используется инструмент пользователя Oracle Discoverer Plus, который предоставляет следующие возможности:

- формирование табличных и кросс-табличных отчетов;
- построение различных диаграмм;
- экспорт и импорт результатов анализа;
- печать результатов анализа;
- распространение результатов анализа.

### ***1.3. Уровень подготовки пользователя***

Пользователь ИАС КХД должен иметь опыт работы с ОС MS Windows (95/98/NT/2000/XP), навык работы с ПО Internet Explorer, Oracle Discoverer, а также обладать следующими знаниями:

- знать соответствующую предметную область;
- знать основы многомерного анализа;
- понимать многомерную модель соответствующей предметной области;
- знать и иметь навыки работы с аналитическими приложениями.

Квалификация пользователя должна позволять:

- формировать отчеты в Oracle Discoverer Plus;
- осуществлять анализ данных.

### ***1.4. Перечень эксплуатационной документации, с которой необходимо ознакомиться пользователю***

- паспорт информационно-аналитической системы «Корпоративное хранилище данных»;
- общее описание информационно-аналитической системы «Корпоративное хранилище данных».

## ***2. Назначение и условия применения Oracle Discoverer Plus***

В разделе «Назначение и условия применения» указывают:

- виды деятельности, функции, для автоматизации которых предназначено данное средство автоматизации;
- условия, при соблюдении (выполнении, наступлении) которых обеспечивается применение средства автоматизации в соответствии с назначением (например, вид ЭВМ и конфигурация технических средств, операционная среда и общесистемные программные средства, входная информация, носители данных, база данных, требования к подготовке специалистов и т. п.).

Oracle Discoverer Plus в составе ИАС КХД предназначен для автоматизации подготовки, настройки отчетных форм по показателям деятельности, а также для углубленного исследования данных на основе корпоративной информации хранилища данных.

Работа с Oracle Discoverer Plus в составе ИАС КХД возможна всегда, когда есть необходимость в получении информации для анализа, контроля, мониторинга и принятия решений на ее основе.

Работа с Oracle Discoverer Plus в составе ИАС КХД доступна всем пользователям с установленными правами доступа.

### ***3. Подготовка к работе***

В разделе «Подготовка к работе» указывают:

- состав и содержание дистрибутивного носителя данных;
- порядок загрузки данных и программ;
- порядок проверки работоспособности.

#### ***3.1. Состав и содержание дистрибутивного носителя данных***

Для работы с ИАС КХД необходимо следующее программное обеспечение:

Internet Explorer (входит в состав операционной системы Windows);

Oracle JInitiator устанавливается автоматически при первом обращении пользователя к ИАС КХД.

#### ***3.2. Порядок загрузки данных и программ***

Перед началом работы с ИАС КХД на рабочем месте пользователя необходимо выполнить следующие действия:

1. Необходимо зайти на сайт ИАС КХД [ias-dwh.ru](http://ias-dwh.ru).
2. Во время загрузки в появившемся окне «Предупреждение о безопасности», которое будет содержать следующее: «Хотите установить и выполнить «Oracle JInitiator» – нажмите на кнопку «Да».

3. После чего запуститься установка Oracle JInitiator на Ваш компьютер.

4. Выбираем кнопку Next и затем ОК.

### **3.3. Порядок проверки работоспособности**

Для проверки доступности ИАС КХД с рабочего места пользователя необходимо выполнить следующие действия:

1. Открыть Internet Explorer, для этого необходимо кликнуть по ярлыку «Internet Explorer» на рабочем столе или вызвать из меню «Пуск».

2. Ввести в адресную строку Internet Explorer адрес: ias-dwh.ru и нажать «Переход».

3. В форме аутентификации ввести пользовательский логин и пароль. Нажать кнопку «Далее».

4. Убедиться, что в окне открылось приложение Oracle Discoverer Plus.

В случае если приложение Oracle Discoverer Plus не запускается, то следует обратиться в службу поддержки.

### **4. Описание операций**

В разделе «Описание операций» указывают:

- описание всех выполняемых функций, задач, комплексов задач, процедур;

- описание операций технологического процесса обработки данных, необходимых для выполнения функций, комплексов задач (задач), процедур.

- Для каждой операции обработки данных указывают:

- наименование;

- условия, при соблюдении которых возможно выполнение операции;

- подготовительные действия;

- основные действия в требуемой последовательности;

- заключительные действия;

- ресурсы, расходуемые на операцию.

В описании действий допускаются ссылки на файлы подсказок, размещенные на магнитных носителях.

#### **4.1. Выполняемые функции и задачи**

Oracle Discoverer Plus в составе ИАС КХД выполняет функции и задачи, приведенные в таблице ниже:

1. Функция – обеспечивает многомерный анализа в табличной и графической формах.

1.1. Задача – «визуализация отчетности».

Описание – в ходе выполнения данной задачи пользователю системы предоставляется возможность работы с выбранным отчетом из состава преднастроенных.

1.2. Задача – «формирование табличных и графических форм отчетности».

Описание – в ходе выполнения данной задачи пользователю системы предоставляется возможность формирования собственного отчета в табличном или графическом виде на базе преднастроенных компонентов.

#### **4.2. Описание операций технологического процесса обработки данных, необходимых для выполнения задач**

Ниже приведено описание пользовательских операций для выполнения каждой из задач.

Задача: «Визуализация отчетности»

Операция 1: Регистрация на портале ИАС КХД

Условия, при соблюдении которых возможно выполнение операции:

- Компьютер пользователя подключен к корпоративной сети.
- Портал ИАС КХД доступен.
- ИАС КХД функционирует в штатном режиме.

Подготовительные действия:

На компьютере пользователя необходимо выполнить дополнительные настройки, приведенные в п. 3.3 настоящего документа.

Основные действия в требуемой последовательности:

1. На иконке «ИАС КХД» рабочего стола произвести двойной щелчок левой кнопкой мышки.

2. В открывшемся окне в поле «Логин» ввести имя пользователя, в поле «Пароль» ввести пароль пользователя. Нажать кнопку «Далее».

Заключительные действия:

- Не требуются.

Ресурсы, расходуемые на операцию:

- 15–30 секунд.

Операция 2: Выбор отчета

Условия, при соблюдении которых возможно выполнение операции:

- Успешная регистрация на Портале ИАС КХД.

Подготовительные действия:

- Не требуются.

Основные действия в требуемой последовательности:

1. В появившемся окне «Мастер создания рабочих книг» поставить точку напротив пункта «Открыть существующую рабочую книгу».

2. Выбрать нужную рабочую книгу и нажать кнопку «Открыть».

Заключительные действия:

1. После завершения работы с отчетом необходимо выбрать пункт меню «Файл».

2. Далее выбрать пункт «Закрыть».

Ресурсы, расходуемые на операцию:

15 секунд.

Задача: «Формирование табличных и графических форм отчетности»

Заполняется по аналогии.

### **5. Аварийные ситуации**

В разделе «Аварийные ситуации» указывают:

1. Действия в случае несоблюдения условий выполнения технологического процесса, в том числе при длительных отказах технических средств.

2. Действия по восстановлению программ и/или данных при отказе магнитных носителей или обнаружении ошибок в данных.

3. Действия в случаях обнаружении несанкционированного вмешательства в данные.

4. Действия в других аварийных ситуациях.

В случае возникновения ошибок при работе ИАС КХД, не описанных ниже в данном разделе, необходимо обращаться к сотруднику подразделения технической поддержки ДИТ (HelpDesk) либо к ответственному Администратору ИАС КХД.

Описание возможных ошибок приведено ниже.

**Класс ошибки** – Портал ИАС КХД.

**Ошибка** – Сервер не найден. Невозможно отобразить страницу.

**Описание ошибки** – Возможны проблемы с сетью или с доступом к portalу ИАС КХД.

**Требуемые действия** – Для устранения проблем с сетью обратиться к сотруднику подразделения технической поддержки (HelpDesk). В других случаях к администратору ИАС КХД.

**Ошибка** – Требуется ввести действительное имя пользователя.

**Описание ошибки** – При регистрации на портале ИАС КХД не введено имя пользователя.

**Требуемые действия** – Ввести имя пользователя.

**Ошибка** – Требуется ввести пароль для регистрации.

**Описание ошибки** – При регистрации на портале ИАС КХД не введен пароль.

**Требуемые действия** – Ввести пароль.

**Класс ошибки** – Сбой в электропитании рабочей станции.

**Ошибка** – Нет электропитания рабочей станции или произошел сбой в электропитании.

**Описание ошибки** – Рабочая станция выключилась или перезагрузилась.

**Требуемые действия** – Перезагрузить рабочую станцию.

Проверить доступность сервера ИАС КХД по порту 80, выполнив следующие команды:

- нажать кнопку «Пуск»;
- выбрать пункт «Выполнить»;
- в строке ввода набрать команду telnet ias\_dwh.ru 80;
- если открылось окно Telnet, значит соединение возможно;
- повторить попытку подключения (входа) в ИАС КХД.

## **6. Рекомендации по освоению**

В разделе «Рекомендации по освоению» указывают рекомендации по освоению и эксплуатации, включая описание контрольного примера, правила его запуска и выполнения.

Рекомендуемая литература:

- Oracle® Business Intelligence Discoverer Viewer User's Guide
- Oracle® Business Intelligence Discoverer Plus User's Guide

Рекомендуемые курсы обучения:

- Discoverer 10g: Создание запросов и отчетов



### 3. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Ознакомится с теоретическими сведениями. Изучить состав типового руководства пользователя.
2. Создать руководство пользователя для заданной системы.
3. Оформить руководство пользователя в виде отчёта по работе.

### 4. ЗАДАНИЕ ДЛЯ РАБОТЫ

Заданием для работы является разработка руководства пользователя для системы, рассмотренной в предшествующей работе.

### 5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Для чего разрабатывается руководство пользователя?
2. Кто разрабатывает руководство пользователя?
3. Перечислите основные разделы руководства пользователя.
4. Что включается в описание операции?

## **РАБОТА №12 ЛАБОРАТОРНАЯ. ИЗУЧЕНИЕ СРЕДСТВ АВТОМАТИЗИРОВАННОГО ДОКУМЕНТИРОВАНИЯ**

### **1. ЦЕЛЬ И ЗАДАЧИ РАБОТЫ**

Целью работы является получение теоретических навыков по работе со средствами автоматического создания документации.

### **2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ**

Для создания документации в процессе разработки ИС используются разнообразные средства формирования отчетов, а также компоненты издательских систем. Обычно средства документирования встроены в конкретные CASE-средства. Исключением являются некоторые пакеты, предоставляющие дополнительный сервис при документировании. Из них наиболее активно используется SoDA (Software Document Automation).

Продукт SoDA предназначен для автоматизации разработки проектной документации на всех фазах ЖЦ ПО. Он позволяет автоматически извлекать разнообразную информацию, получаемую на разных стадиях разработки проекта, и включать ее в выходные документы. При этом контролируется соответствие документации проекту, взаимосвязь документов, обеспечивается их своевременное обновление. Результирующая документация автоматически формируется из множества источников, число которых не ограничено.

SoDA не зависит от применяемых инструментальных средств. Связь с приложениями осуществляется через стандартный программный интерфейс API. Переход на новые инструментальные средства не влечет за собой дополнительных затрат по документированию проекта.

SoDA содержит набор шаблонов документов, определяемых стандартом на программное обеспечение DOD 2167A. На их основе можно без специального программирования создавать новые формы документов, определяемые пользователями.

Пакет включает в себя графический редактор для подготовки шаблонов документов. Он позволяет задавать необходимый стиль, фон, шрифт, определять расположение заголовков, резервировать места, где будет размещаться извлекаемая из разнообразных источ-

ников информация. Изменения автоматически вносятся только в те части документации, на которые они повлияли в программе. Это сокращает время подготовки документации за счет отказа от регенерации всей документации.

SoDA реализована на базе издательской системы FrameBuilder и предоставляет полный набор средств по редактированию и верстке выпускаемой документации. Разные версии документации могут быть для наглядности отмечены своими отличительными признаками. В системе создаются таблицы требований к проекту, по которым можно проследить, как реализуются эти требования. Разные виды документации, сопровождающие различные этапы ЖЦ, связаны между собой, и можно проследить состояние проекта от первоначальных требований до анализа, проектирования, кодирования и тестирования программного продукта.

Итоговым результатом работы системы SoDA является готовый документ (или книга). Документ может храниться в файле формата SoDA (Frame Builder), который получается в результате генерации документа. Вывод на печать этого документа (или его части) возможен из системы SoDA.

Среда функционирования SoDA – ОС типа UNIX на рабочих станциях Sun SPARCstation, IBM RISC System/6000 или Hewlett Packard HP 9000 700/800.

SoDA требует по крайней мере 32 МВ оперативной памяти, 100-300 МВ для установки и 64 МВ рабочего пространства

### 3. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Ознакомится с теоретическими положениями.
2. Запустить систему RatinalSoda.
3. Создать пример документации.

### 4. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Для чего нужны средства автоматизации документирования?
2. Особенности системы Ratinal SoDA.

## СОДЕРЖАНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

Цель самостоятельной работы обучающихся – получить новые знания по дисциплине «Проектирование и дизайн информационных систем».

Самостоятельная работа необходима для формирования у обучающихся способности самостоятельно решать задачи профессиональной деятельности, формирования умения и навыков планирования времени, формирования стремления развиваться и совершенствоваться.

Виды самостоятельной работы обучающихся указаны в табл. 1.

Таблица 1

Виды самостоятельной работы

№ п/п	Вид СРС
1	Исследование средств управления проектами
2	Построение модели управления качеством процесса изучения модуля «Проектирование и разработка информационных систем»
3	Разработка руководства по инсталляции программного продукта для ИС
4	Разработка сетевого графика проекта реализации ИС

## УЧЕБНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ

### Основная литература

1. Федорова, Г. Н. Разработка, внедрение и адаптация программного обеспечения отраслевой направленности. – Москва : НИЦ ИНФРА-М, 2019. – 336 с. – Режим доступа: <http://znanium.com/go.php?id=989682>. – Загл. с экрана.

### Дополнительная литература

1. Рудаков, А. В. Технология разработки программных продуктов [Электронный ресурс] : учебник для студентов учреждений среднего профессионального образования, обучающихся по специальности «Программное обеспечение вычислительной техники и автоматизированных систем» : [профессиональный модуль ПМ.03 «Участие в интеграции программных модулей» (МДК.03.01)] / А. В. Рудаков. – Москва : Академия, 2017. – 208 с. – Режим доступа: <http://www.academia-moscow.ru/catalogue/4831/362819/>. – Загл. с экрана.

2. Казанский, А. А. Программирование на visual c# 2013.[электронный ресурс]. – Москва : Юрайт, 2018. – 191 с. – Режим доступа: <https://biblio-online.ru/book/programmirovanie-na-visual-c-2013-414752>. – Загл. с экрана.

### Программное обеспечение и интернет-ресурсы

1. <https://www.intuit.ru/studies/courses/497/353/info> – Учебный курс Введение в программную инженерию; <https://www.intuit.ru/studies/courses/532/388/info> Учебный курс Управление развитием ИС

2. <https://iiba.ru/category/overview-of-all-methods-of-work-with-requirements/> обзор методик работы с требованиями <http://www.enter-agile.com/2010/01/user-story-primer-intro.html> Основы пользовательских историй

## СОДЕРЖАНИЕ

Работа №1 Практическая. Построение и анализ модели предметной области на основе методологии SADT .....	2
Работа №2 Практическая. Построение и анализ модели предметной области на основе ООП. Построение модели состава процессов и модели организационной структуры .....	6
Работа №3 Практическая. Построение и анализ модели предметной области на основе ООП. модель содержания бизнес-процессов .....	20
Работа №4 Практическая. Определение состава функциональных требований к системе на основе анализа предметной области .....	31
Работа №5 Практическая. Разработка проектной архитектуры системы.....	43
Работа №6 Практическая. Расчёт показателей выполнения бизнес-процесса на основе модели IDEF0 .....	50
Работа №7 Практическая. Реинжиниринг бизнес-процесса методом интеграции .....	57
Работа №8 Практическая. Реинжиниринг бизнес-процессов методом горизонтального и/или вертикального сжатия .....	59
Работа №9 Практическая. Разработка требований безопасности информационной системы.....	62
Работа №10 Практическая. Разработка функциональной спецификации и технического задания на разработку ИС .....	67
Работа №11 Практическая. Разработка руководства пользователя ИС .....	73
Работа №12 Лабораторная. Изучение средств автоматизированного документирования .....	81
СОДЕРЖАНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ.....	83
Учебно-методические материалы по дисциплине .....	84
Содержание .....	85