

М. А. ИВАНОВСКИЙ, И. А. ГЛАЗКОВА

УПРАВЛЕНИЕ ИТ-ПРОЕКТАМИ



Тамбов
Издательский центр ФГБОУ ВО «ТГТУ»
2025

Министерство науки и высшего образования Российской Федерации

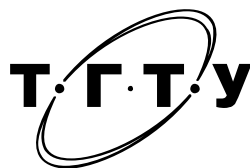
**Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Тамбовский государственный технический университет»**

М. А. ИВАНОВСКИЙ, И. А. ГЛАЗКОВА

УПРАВЛЕНИЕ ИТ-ПРОЕКТАМИ

Утверждено Ученым советом университета
в качестве учебного пособия для студентов направления подготовки
09.03.02 очной, заочной форм обучения,
09.04.02 «Информационные системы и технологии»
очной, заочной форм обучения,
27.04.03 «Системный анализ и управление» очной формы обучения

Учебное электронное издание



Тамбов
Издательский центр ФГБОУ ВО «ТГТУ»
2025

УДК 00.004.078
ББК 39.973.202
И22

Рецензенты:

Доктор технических наук, доцент, заведующий кафедрой
«Мехатроника и технологические измерения» ФГБОУ ВО «ТГТУ»
П. В. Балабанов

Кандидат технических наук, доцент, доцент кафедры математического
моделирования и информационных технологий
ФГБОУ ВО «ТГУ имени Г. Р. Державина»
И. А. Зауголков

Ивановский, М. А.

И22 Управление ИТ-проектами [Электронный ресурс] : учебное пособие /
М. А. Ивановский, И. А. Глазкова. – Тамбов : Издательский центр ФГБОУ
ВО «ТГТУ», 2025. – 1 электрон. опт. диск (CD-ROM). – Системные
требования : ПК не ниже класса Pentium II ; CD-ROM-дисковод ; 3,9 Mb ;
RAM ; Windows 95/98/XP ; мышь. – Загл. с экрана.
ISBN 978-5-8265-2876-1

Содержит вопросы управления ИТ-проектами, требования Министерства
труда и социальной защиты Российской Федерации к квалификации специали-
стов в области управления ИТ-проектами. Рассмотрены процессы управления
проектами, методы оценки трудозатрат на разработку программной системы,
документирование программных средств.

Предназначено для студентов направления подготовки 09.03.02 очной, заоч-
ной форм обучения, 09.04.02 «Информационные системы и технологии» очной,
заочной форм обучения, 27.04.03 «Системный анализ и управление» очной
формы обучения.

УДК 00.004.078
ББК 39.973.202

*Все права на размножение и распространение в любой форме остаются за разработчиком.
Нелегальное копирование и использование данного продукта запрещено.*

ISBN 978-5-8265-2876-1

© Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Тамбовский государственный технический
университет» (ФГБОУ ВО «ТГТУ»), 2025

ВВЕДЕНИЕ

Тенденции развития современных информационных технологий приводят к постоянному возрастанию сложности информационных систем (ИС), создаваемых в различных областях экономики.

Исходя из этого процедурные и аналитические модели для информационных систем диктуют возрастание сложности системотехнических процедур анализа и синтеза моделей сложных объектов информационных систем.

В первой главе рассмотрены термины и определения применительно к информационной системе, ее признаки классификации и архитектура.

Во второй главе рассмотрены особенности создания программных изделий.

В третьей главе – основные стандарты и руководящие документы, регламентирующие проектирование информационных систем.

Четвертая, пятая и шестая главы посвящены вопросам канонического проектирования, недостаточно освещенным в учебной литературе: стадии и этапы создания автоматизированных систем по ГОСТ 34.601–90, по ГОСТ 24.601–86, этапы и содержание работ по ГОСТ 19.102–77.

Седьмая глава посвящена жизненному циклу программного обеспечения и средствам его поддержки. Рассмотрены в частности общая схема самого жизненного цикла, его модели.

Восьмая глава содержит сведения о средствах CASE-технологий.

Девятая глава – содержание этапов проектирования информационной системы.

Десятая глава включает основы объектно ориентированного подхода к анализу и проектированию, моделированию бизнес-процессов.

1. УПРАВЛЕНЧЕСКИЕ ПРОЦЕССЫ

Управленческие процессы планирования, организации, анализа, регулирования, контроля, учета и др. рассматриваются как виды деятельности конкретных работников, технических и информационных систем. Часто все эти процессы считают вспомогательными.

Чтобы любой процесс смог совершаться, необходимо иметь в наличии определенный набор материальных элементов – элементов производственной (производственно-эксплуатационной) системы, т.е. необходима элементная (объектно-субъектная) структура процесса, а именно: машины и оборудование, передаточные устройства и транспортные средства, рабочий инструмент, производственный инвентарь и принадлежности, здания и сооружения, готовый продукт (идеальный образ в голове конструкторов, технологов), сырье и материалы, топливо и энергия, информация, финансы, люди (работники).

При этом должны осуществляться функции управления (стандартный набор функций управления, в том числе и для оперативного управления):

1. Прогнозирование.
2. Планирование (моделирование).
3. Организация.
4. Исполнение (т.е. реализация всех управленческих функций практически, реализация управляющих воздействий).
5. Регулирование (диспетчеризация).
6. Стимулирование.
7. Учет.
8. Анализ.
9. Оценка.
10. Контроль

Неисполнение или некачественное исполнение какой-либо из перечисленных функций может привести к полному срыву производственной деятельности.

2. МОДЕЛЬ ЖИЗНЕННОГО ЦИКЛА ПО СТАНДАРТУ ISO/IEC 12207

Модель жизненного цикла (ЖЦ) определяет последовательность выполнения и взаимосвязи процессов, действий и задач на протяжении ЖЦ. Модель ЖЦ зависит от специфики программных систем (ПС) и специфики условий, в которых система создается и функционирует.

Структура ЖЦ программного обеспечения (ПО) по стандарту ISO/IEC 12207 базируется на трех группах процессов (рис. 2.1):

- основные процессы ЖЦ ПО: приобретение, поставка, разработка, эксплуатация, сопровождение;
- вспомогательные процессы, обеспечивающие выполнение основных процессов: документирование, управление конфигурацией;
- организационные процессы: управление проектами, создание инфраструктуры проекта, определение, оценка и улучшение самого ЖЦ, обучение.

Управление проектом связано с вопросами планирования и организации работ, создания коллективов разработчиков и контроля за сроками и качеством выполняемых работ. Техническое и организационное обеспечение проекта включает выбор методов и инструментальных средств для реализации проекта, определение методов описания промежуточных состояний разработки, разработку методов и средств испытаний ПО, обучение персонала и т.п.

Обеспечение качества проекта связано с проблемами верификации, проверки и тестирования ПО. Верификация – это процесс определения того, отвечает ли текущее состояние разработки, достигнутое на данном этапе, требованиям этого этапа. Проверка позволяет оценить разработку на соответствие ее параметров исходным требованиям. Проверка частично совпадает с тестированием, которое связано с идентификацией различий между действительными и ожидаемыми результатами и оценкой соответствия характеристик ПО исходным требованиям. В процессе реализации проекта важное место занимают вопросы идентификации, описания и контроля конфигурационных требований отдельных компонентов и всей системы в целом.

Основные процессы ЖЦ ПО	Вспомогательные процессы	Организационные процессы
- приобретение - поставка - разработка: - анализ	- документирование - <i>управление конфигурацией</i> - управление версиями - конфигурационный учет	- управление проектами - планирование и организация работ - создание коллективов разработчиков - контроль за сроками и качеством выполняемых работ:
- проектирование - программирование - <i>оформление документации:</i> - проектной - эксплуатационной	- планирование	- верификация - проверка - тестирование - создание инфраструктуры проекта - определение, оценка и улучшение самого ЖЦ
- подготовка тестов - подготовка материалов для обучения		- обучение
- эксплуатация - <i>внедрение компонентов ПО:</i> - конфигурирование - базы данных - рабочих мест пользователей - обеспечение эксплуатационной документацией		
- проведение обучения персонала		
- локализация проблем		
- <i>устранение причин возникновения проблем</i> - <i>модификация ПО в рамках установленного регламента</i> - <i>подготовка предложений по совершенствованию, развитию и модернизации системы</i> - сопровождение		

Рис. 2.1. Группы процессов ЖЦ ПО по стандарту Р ИСО/МЭК 12207

Управление конфигурацией является одним из вспомогательных процессов, поддерживающих основные процессы ЖЦ ПО, прежде всего процессы разработки и сопровождения ПО. При создании проектов сложных ПС, состоящих из многих компонентов, каждый из которых может иметь разновидности или версии, возникает проблема учета их связей и функций, создания унифицированной структуры и обеспечения развития всей системы. Управление конфигу-

рацией позволяет организовать, систематически учитывать и контролировать внесение изменений в ПО на всех стадиях ЖЦ.

Каждый процесс характеризуется определенными задачами и методами их решения, исходными данными, полученными на предыдущем этапе, и результатами. Результатами анализа, в частности, являются функциональные модели, информационные модели и соответствующие им диаграммы. ЖЦ ПО носит итерационный характер: результаты очередного этапа часто вызывают изменения в проектных решениях, выработанных на более ранних стадиях.

3. НАУЧНЫЕ ИССЛЕДОВАНИЯ И ИНФОРМАЦИОННО-ТЕХНОЛОГИЧЕСКИЕ (ИТ) ПРОЕКТЫ

Научные исследования можно разделить на фундаментальные, поисковые и прикладные (табл.3.1).

Фундаментальные и поисковые научно-исследовательские работы (НИР) обычно не входят в комплекс работ по созданию и освоению новой продукции.

Непосредственно к процессам создания новой продукции относятся прикладные НИР. Основные *этапы* НИР:

- 1) разработка технического задания (ТЗ) НИР;
- 2) выбор направления исследования;
- 3) теоретические и экспериментальные исследования;
- 4) обобщение и оценка результатов исследований.

Конкретный состав этапов и работ на них определяется, естественно, спецификой НИР. Примерный перечень работ на этапах НИР приведен в табл. 3.2.

3.1. Научно-исследовательские работы

Вид исследований	Результаты исследований
Фундаментальные	Расширение теоретических знаний. Получение новых научных данных о процессах, явлениях, закономерностях, существующих в исследуемой области; научные основы, методы и принципы исследований
Поисковые	Увеличение объема знаний для более глубокого понимания изучаемого предмета. Разработка прогнозов развития науки и техники; открытие путей применения новых явлений и закономерностей
Прикладные	Разрешение конкретных научных проблем для создания новых изделий. Получение рекомендаций, инструкций, расчетно-технических материалов, методик и т.д.

3.2. Этапы и состав НИР

Этап НИР	Состав НИР
Разработка ТЗ НИР	Научное прогнозирование. Анализ результатов фундаментальных и поисковых исследований. Изучение патентной документации. Учет требований заказчиков
Выбор направления исследования	Сбор и изучение научно-технической информации. Составление аналитического обзора. Проведение патентных исследований. Формулирование возможных направлений решения задач, поставленных в ТЗ НИР, и сравнительная оценка. Выбор и обоснование принятого направления исследований и способов решения задач. Сопоставление ожидаемых показателей новой продукции после внедрения результатов НИР с существующими показателями изделий-аналогов. Оценка ориентировочной экономической эффективности новой продукции. Разработка общей методики проведения исследований (программы работ, планы-графики, сетевые модели). Составление промежуточного отчета
Теоретические и экспериментальные исследования	Разработка рабочих гипотез, построение моделей объекта исследований, обоснование допущений. Выявление необходимости проведения экспериментов для подтверждения отдельных положений теоретических исследований или для получения конкретных значений параметров, необходимых для проведения расчетов. Разработка методики экспериментальных исследований, подготовка моделей (макетов, экспериментальных образцов), а также испытательного оборудования. Проведение экспериментов, обработка полученных данных; сопоставление результатов эксперимента с теоретическими исследованиями. Корректировка теоретических моделей объекта. Проведение при необходимости дополнительных экспериментов. Проведение технико-экономических исследований. Составление промежуточного отчета
Обобщение и оценка результатов исследований	Обобщение результатов предыдущих этапов работ. Оценка полноты решения задач. Разработка рекомендаций по дальнейшим исследованиям и проведению ОКР. Разработка проекта ТЗ на ОКР. Составление итогового отчета. Приемка НИР комиссией

После завершения прикладных НИР при условии получения положительных результатов экономического анализа, удовлетворяющих фирму с точки зрения ее целей, ресурсов и рыночных условий, приступают к выполнению опытно-конструкторских работ (ОКР). ОКР – важнейшее звено материализации результатов предыдущих НИР. На основе полученных результатов исследований создаются и отрабатываются новые товары.

Основные *этапы ОКР*:

- 1) разработка ТЗ на ОКР;
- 2) техническое предложение;
- 3) эскизное проектирование;
- 4) техническое проектирование;
- 5) разработка рабочей документации для изготовления и испытаний опытного образца;
- 6) предварительные испытания опытного образца;
- 7) государственные (ведомственные) испытания опытного образца;
- 8) отработка документации по результатам испытаний.

Примерный перечень работ на этапах ОКР приведен в табл. 3.3.

3.3. Примерный перечень работ на этапах ОКР

Этап ОКР	Основные задачи и состав работ
Разработка ТЗ на ОКР	Составление проекта ТЗ заказчиком. Проработка проекта ТЗ исполнителем. Установление перечня контрагентов и согласование с ними частных ТЗ. Согласование и утверждение ТЗ
Техническое предложение (является основанием для корректировки ТЗ и выполнения эскизного проекта)	Выявление дополнительных или уточненных требований к изделию, его техническим характеристикам и показателям качества, которые не могут быть указаны в ТЗ: – проработка результатов НИР; – проработка результатов прогнозирования; – изучение научно-технической информации; – предварительные расчеты и уточнение требований ТЗ

Этап ОКР	Основные задачи и состав работ
Эскизное проектирование (служит основанием для технического проектирования)	Разработка принципиальных технических решений: – выполнение работ по этапу технического предложения, если этот этап не выполняется; – выбор элементной базы разработки; – выбор основных технических решений; – разработка структурных и функциональных схем изделия; – выбор основных конструктивных элементов; – метрологическая экспертиза проекта; – разработка и испытание макетов
Техническое проектирование	Окончательный выбор технических решений по изделию в целом и по его составным частям: – разработка принципиальных электрических, кинематических, гидравлических и других схем; – уточнение основных параметров изделия; – проведение конструктивной компоновки изделия и выдача данных для его размещения на объекте; – разработка проектов ТУ на поставку и изготовление изделия; – испытание макетов основных приборов изделия в натурных условиях
Разработка рабочей документации для изготовления и испытания опытного образца	Формирование комплекта конструкторских документов: – разработка полного комплекта рабочей документации; – согласование ее с заказчиком и заводом-изготовителем серийной продукции; – проверка конструкторской документации на унификацию и стандартизацию; – изготовление в опытном производстве опытного образца; – настройка и комплексная регулировка опытного образца
Предварительные испытания	Проверка соответствия опытного образца требованиям ТЗ и возможности предъявления его на государственные (ведомственные) испытания: – стендовые испытания; – предварительные испытания на объекте; – испытания на надежность
Государственные испытания	Оценка соответствия ТЗ и возможности организации серийного производства
Отработка документации по результатам испытаний	Внесение необходимых уточнений и изменений в документацию. Присвоение документации литеры " " Передача документации организации-изготовителю

4. ТРЕБОВАНИЯ МИНИСТЕРСТВА ТРУДА И СОЦИАЛЬНОЙ ЗАЩИТЫ РОССИЙСКОЙ ФЕДЕРАЦИИ К КВАЛИФИКАЦИИ СПЕЦИАЛИСТОВ В ОБЛАСТИ УПРАВЛЕНИЯ ИТ-ПРОЕКТАМИ

Приказом Министерства труда и социальной защиты Российской Федерации от 13 октября 2014 г. № 716н определен профессиональный стандарт «Менеджер по информационным технологиям», приказом от 18 ноября 2014 г. № 893н определен профессиональный стандарт «Руководитель проектов в области информационных технологий», приказом от 17 сентября 2014 г. № 645н определен профессиональный стандарт «Руководитель разработки программного обеспечения».

Профессиональное и качественное управление проектами позволяет снизить непроизводительные затраты в ИТ-отрасли и повысить вероятность достижения запланированных параметров проекта (бюджета, сроков и необходимого результата).

Неквалифицированное управление проектами приводит либо к существенному перерасходу бюджета проекта и срыву сроков, либо к невозможности достичь целей проекта, нерациональному использованию денежных средств и полной неудовлетворенности заказчика.

От руководителя проектов в области ИТ требуется большой объем специальных знаний по планированию и контролю именно ИТ-проектов, без которых управление ИТ-проектом не будет эффективным.

Руководитель проекта в области информационных технологий отвечает за достижение целей проекта в рамках утвержденных заказчиком бюджета и сроков. Осуществляет управление проектами (планирование, организацию исполнения, контроль и анализ отклонений) в области создания и/или ввода в эксплуатацию информационных систем:

- обеспечивает выявление требований к информационной системе (ИС);
- осуществляет оценку сроков и стоимости проекта разработки, ввода в эксплуатацию и послепроектного сопровождения ИС;
- организует работы по обеспечению необходимого уровня качества;
- управляет человеческими ресурсами проекта;
- организует и осуществляет коммуникации между всеми заинтересованными лицами проекта, включая заказчика ИС, пользователей ИС, членов команды проекта;
- управляет рисками проекта;
- управляет закупками, включая управление субподрядчиками;
- направляет работу команды проекта на достижение целей проекта;
- обеспечивает проведение сдачи результатов проекта заказчику;
- участвует в совершенствовании системы управления проектами.

Объекты и средства профессиональной деятельности – команда проекта, информационная система, описание содержания проекта, планы и графики выполнения работ, сметы и бюджеты, инструментальные средства управления проектами, стандарты в области управления проектами; стандарты и методы информационного взаимодействия систем, стандарты и методы организации управления, учета и отчетности на предприятиях.

Распределение по видам профессиональной деятельности между менеджером по информационным технологиям, руководителем проектов в области информационных технологий и руководителем разработки программного обеспечения представлено в табл. 4.1, а их обобщенные трудовые функции – в табл. 4.2.

Общий подход к выбору квалификационных уровней в профессии «Руководитель проектов в области информационных технологий» может быть проиллюстрирован следующими принципами:

Первый уровень профессии – «подмастерье».

Второй уровень – «мастер».

Третий уровень – «эксперт».

4.1. Распределение по видам профессиональной деятельности

	06.014 Менеджер по информационным технологиям	06.016 Руководитель проектов в области информационных технологий	06.017 Руководитель разработки программного обеспечения
Вид профессиональной деятельности	Информационные технологии в экономике и государственном управлении	Менеджмент проектов в области информационных технологий (ИТ)	Руководство разработкой программного обеспечения
Цель	Управление предоставлением, использованием и развитием информационных технологий (ИТ)	Менеджмент проектов в области ИТ (планирование, организация исполнения, контроль и анализ отклонений) для эффективного достижения целей проекта в рамках утвержденных заказчиком требований, бюджета и сроков	Руководство процессами разработки, отладки, проверки работоспособности и модификации программного обеспечения, их организация и управление ресурсами
Группа занятий	1120 Руководители и старшие должностные лица государственных органов управления и представительств 1210 Руководители учреждений, организаций и предприятий 1237 Руководители подразделений (служб) научно-технического развития 1229 Руководители специализированных (производственно-эксплуатационных) подразделений (служб), не вошедшие в другие группы 1236 Руководители подразделений (служб) компьютерного обеспечения 1239 Руководители подразделений (служб), не вошедшие в другие группы	1236 Руководители подразделений (служб) компьютерного обеспечения	1236 Руководители подразделений (служб) компьютерного обеспечения 2131 Разработчики и аналитики компьютерных систем

4.2. Обобщенные трудовые функции

	06.014 Менеджер по информационным технологиям	06.016 Руководитель проектов в области информационных технологий	06.017 Руководитель разработки программного обеспечения
Обобщенные трудовые функ- ции	А Управление ресурсами ИТ В Управление сервисами ИТ С Управление информационной средой Д Управление ИТ-инновациями	А Управление проектами в области ИТ на основе полученных планов проектов в условиях, когда проект не выходит за пределы утвержденных параметров В Управление проектами в области ИТ малого и среднего уровня сложности в условиях неопределенностей, порождаемых запросами на изменения, с применением формальных инструментов управления рисками и проблемами проекта С Управление проектами в области ИТ любого масштаба в условиях высокой неопределенности, вызываемой запросами на изменения и рисками, и с учетом влияния организационного окружения проекта; разработка новых инструментов и методов управления проектами в области ИТ	А Непосредственное руководство процессами разработки программного обеспечения В Организация процессов разработки программного обеспечения С Управление программно-техническими, технологическими и человеческими ресурсами
Наименование базовой группы, должности (профессии) или специальности	1236 Руководители подразделений (служб) компьютерного обеспечения	1236 Руководители подразделений (служб) компьютерного обеспечения	1236 Руководители подразделений (служб) компьютерного обеспечения 2131 Разработчики и аналитики компьютерных систем

5. ПРОЦЕССЫ УПРАВЛЕНИЯ ПРОЕКТАМИ

Методология строится как пересечение областей знаний: технологии создания продукта – информационной системы – и технологии управления проектной деятельностью (рис. 5.1).

Основные процессы представлены на рис. 5.2 – 5.8.

На рисунке 5.2 представлены процессы управления проектом по этапам, на рис. 5.3 – процессы управления содержанием проекта, на рис. 5.4 – структура работ по фазам проекта, управление сроками работ, на рис. 5.5 – процессы управления стоимостью проекта, на рис. 5.6 – процессы управления рисками, на рис. 5.7 – процессы управления качеством проекта и на рис. 5.8 – процессы управления командой проекта.

План управления содержанием проекта – это документ, описывающий, как будут определяться, разрабатываться и проверяться работы, которые необходимо выполнить для получения результата с указанными характеристиками, и задающий действия по управлению содержанием проекта.

Процесс создания иерархической структуры работ (ИСР) выполняет разбиение укрупненной структуры работ, представленной в документе «Предварительное описание содержания», на более мелкие, более управляемые элементы (рис. 5.9).

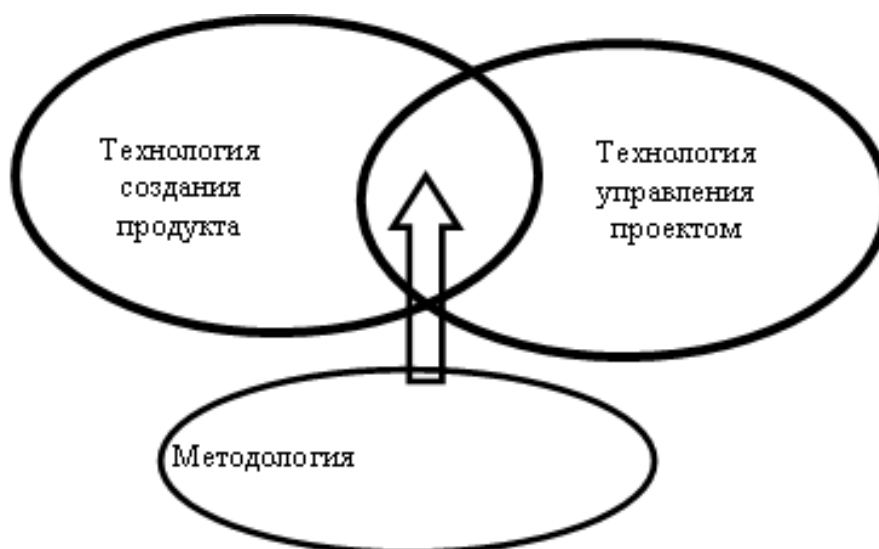


Рис. 5.1. Составляющие методологии



Рис. 5.2. Этапы процессов управления проектом

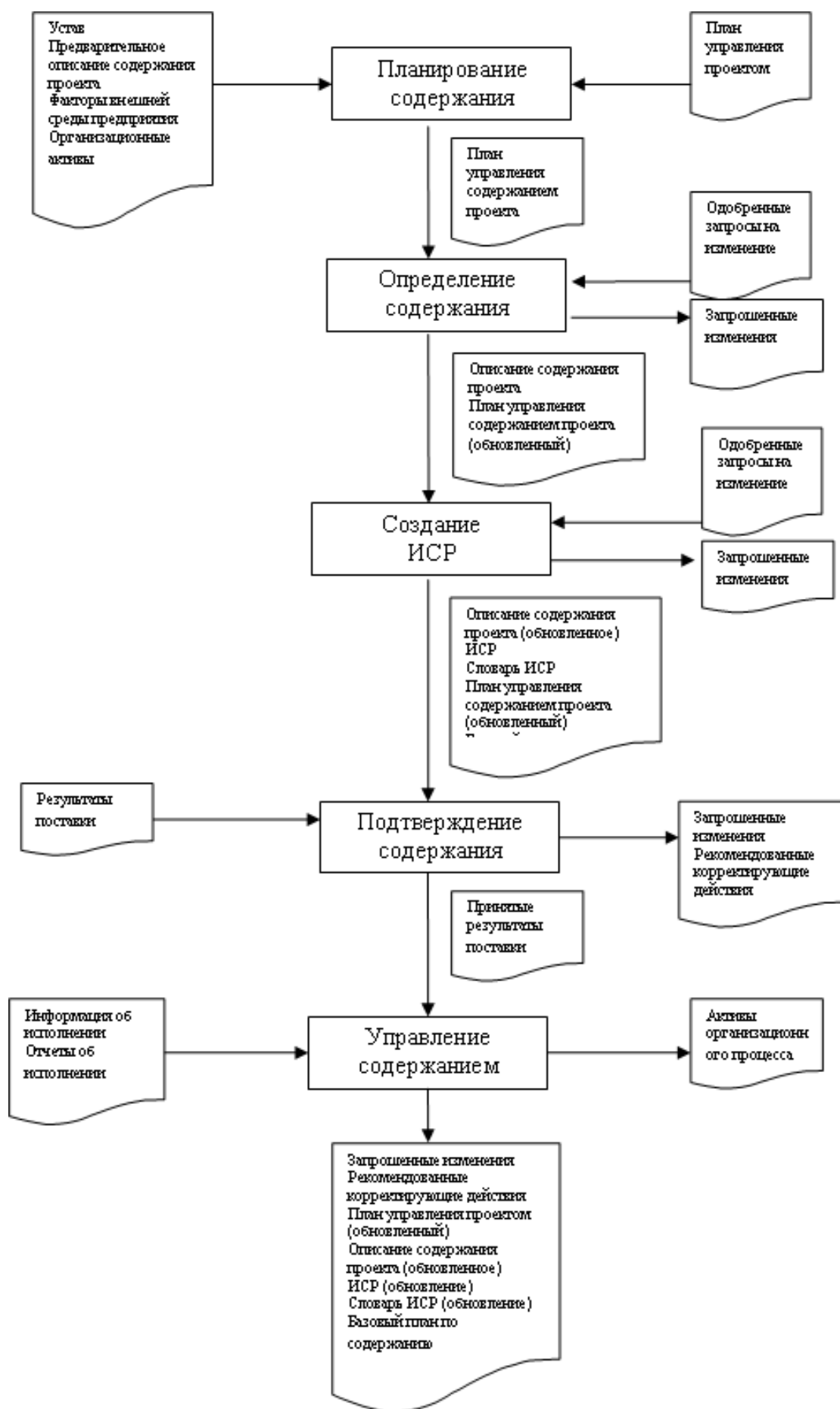


Рис. 5.3. Процессы управления содержанием проекта

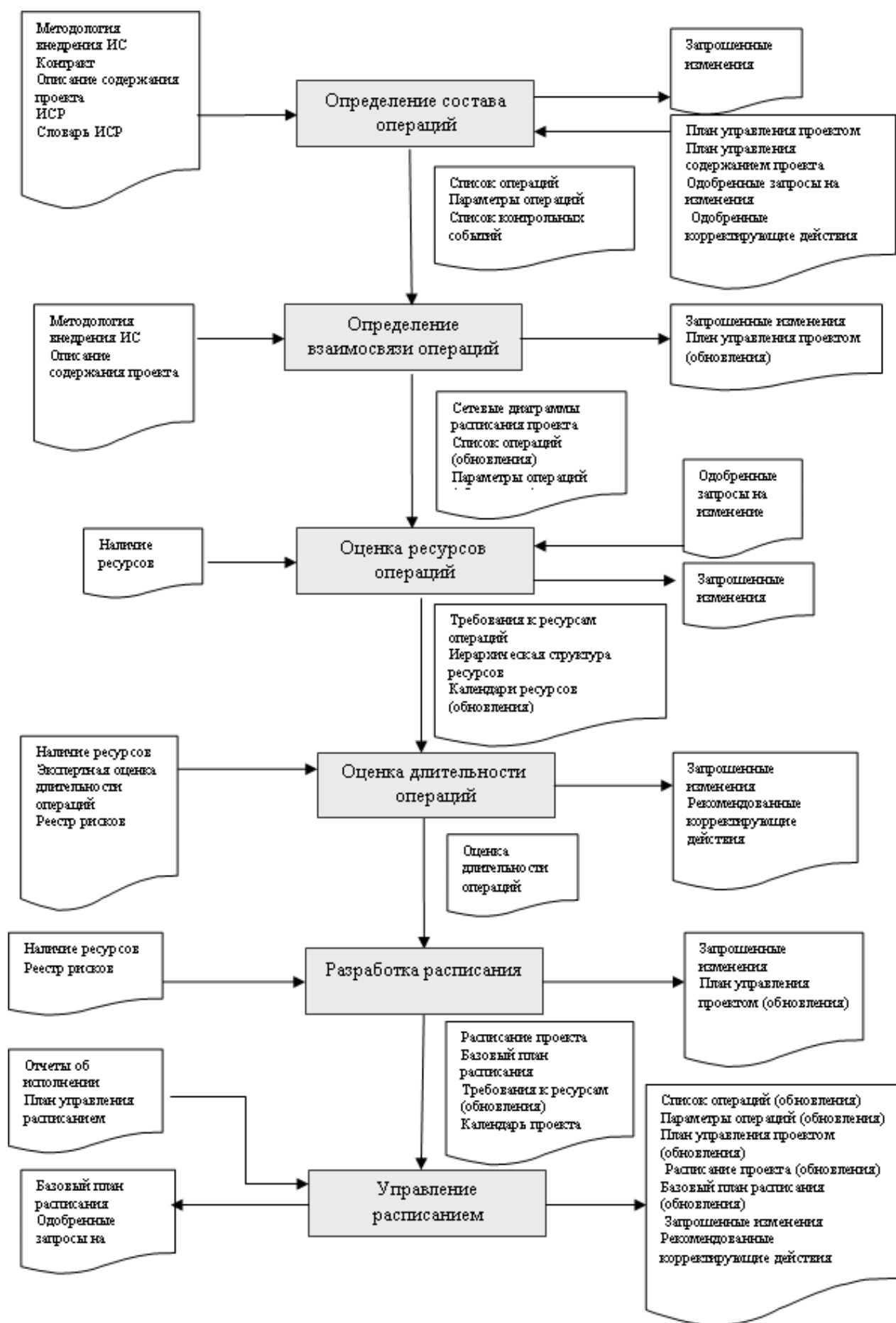


Рис. 5.4. Структура работ по фазам проекта. Управление сроками работ

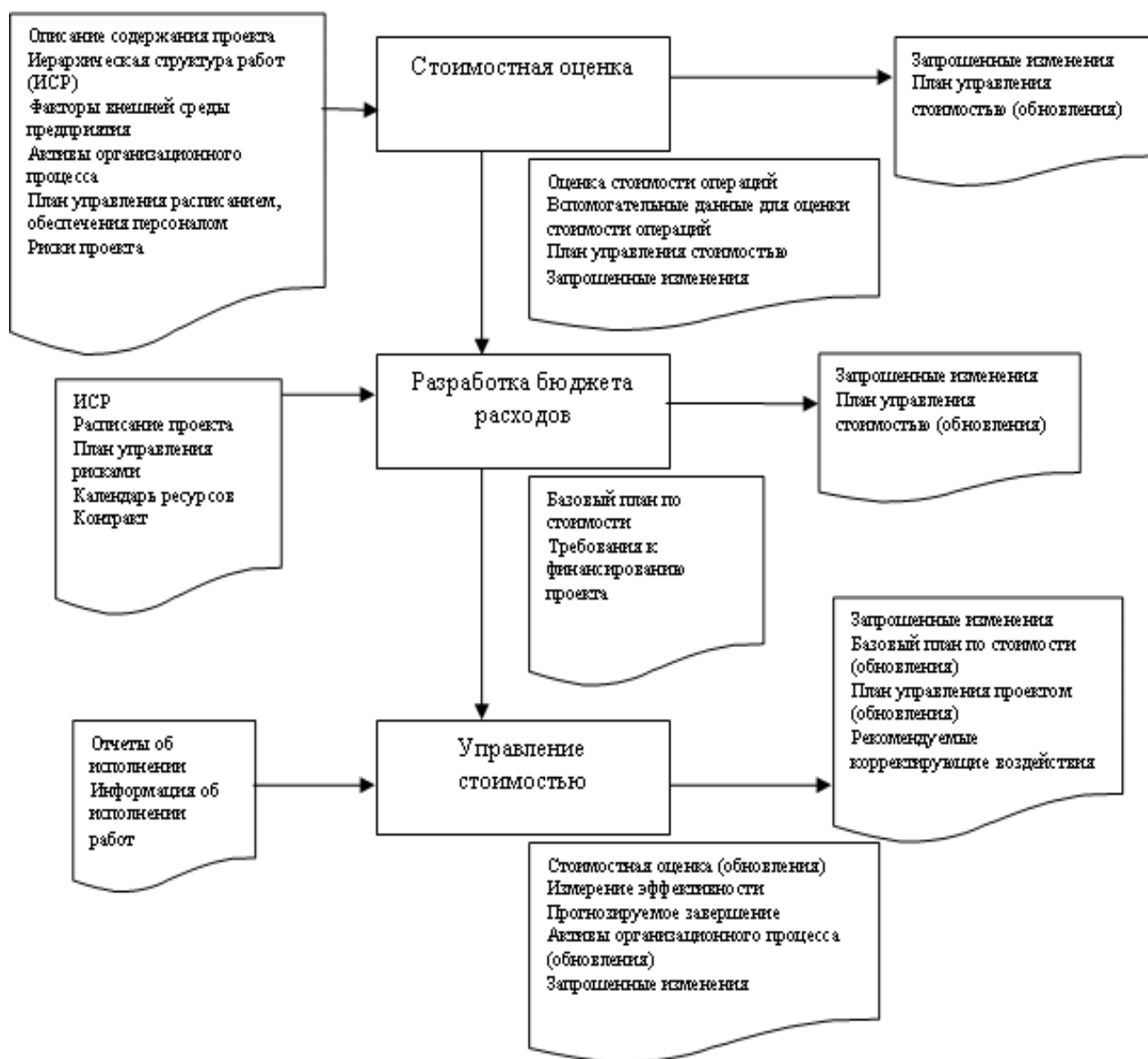


Рис. 5.5. Управление стоимостью проекта

Управление сроками проекта – процесс состоит из шести процессов.

Определение состава операций – процесс определения конкретных плановых операций, которые необходимо выполнить для получения результатов проекта – внедрения ИС.

Определение взаимосвязей операций – процесс выявления и документирования последовательности выполнения плановых операций.

Определение ресурсов операции – процесс определения необходимых для выполнения каждой плановой операции ресурсов и их количества.

Определение длительности операций – процесс определения продолжительности выполнения каждой плановой операции.

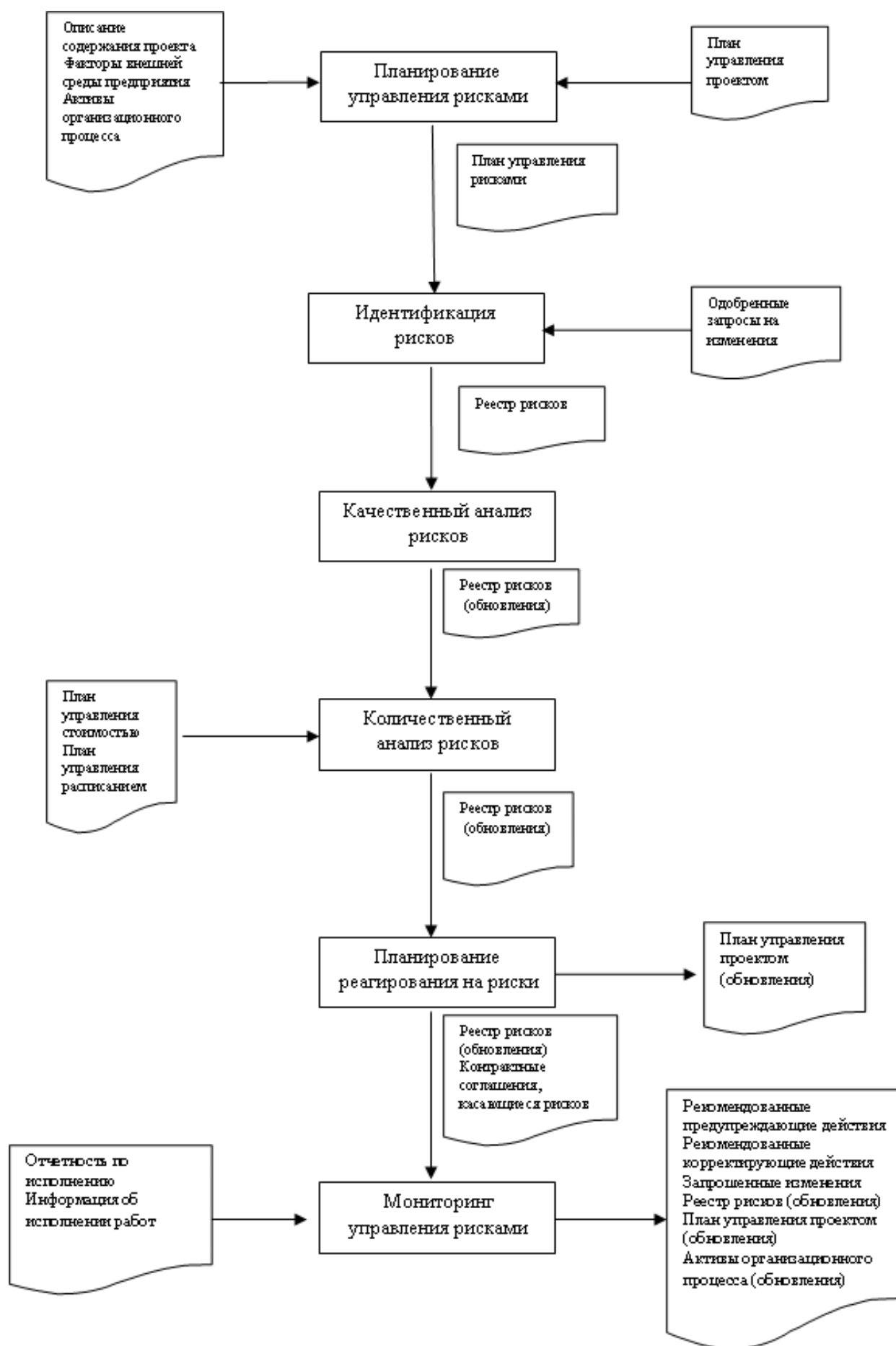


Рис. 5.6. Процессы управления рисками

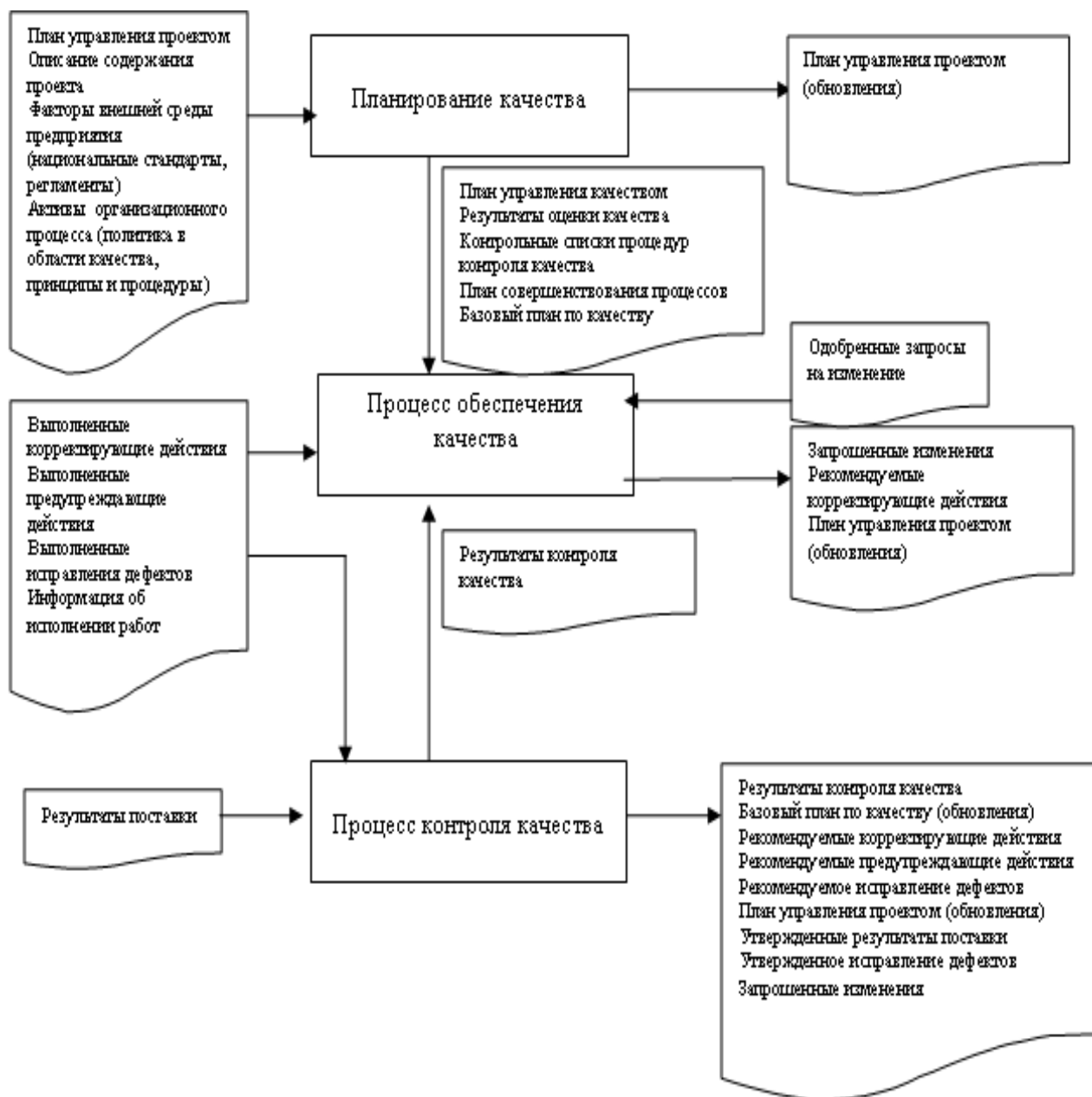


Рис. 5.7. Процессы управления качеством проекта

Разработка расписания – процесс составления расписания проекта с учетом последовательностей операций, их длительности, требований к ресурсам и ограничений на сроки выполнения проекта в целом.

Управление расписанием – процесс управления изменениями расписания проекта.

Первые пять процессов относятся к группе процессов планирования, шестой – к группе процессов мониторинга и управления. Процессы взаимодействуют как между собой, так и с процессами из других областей знаний.

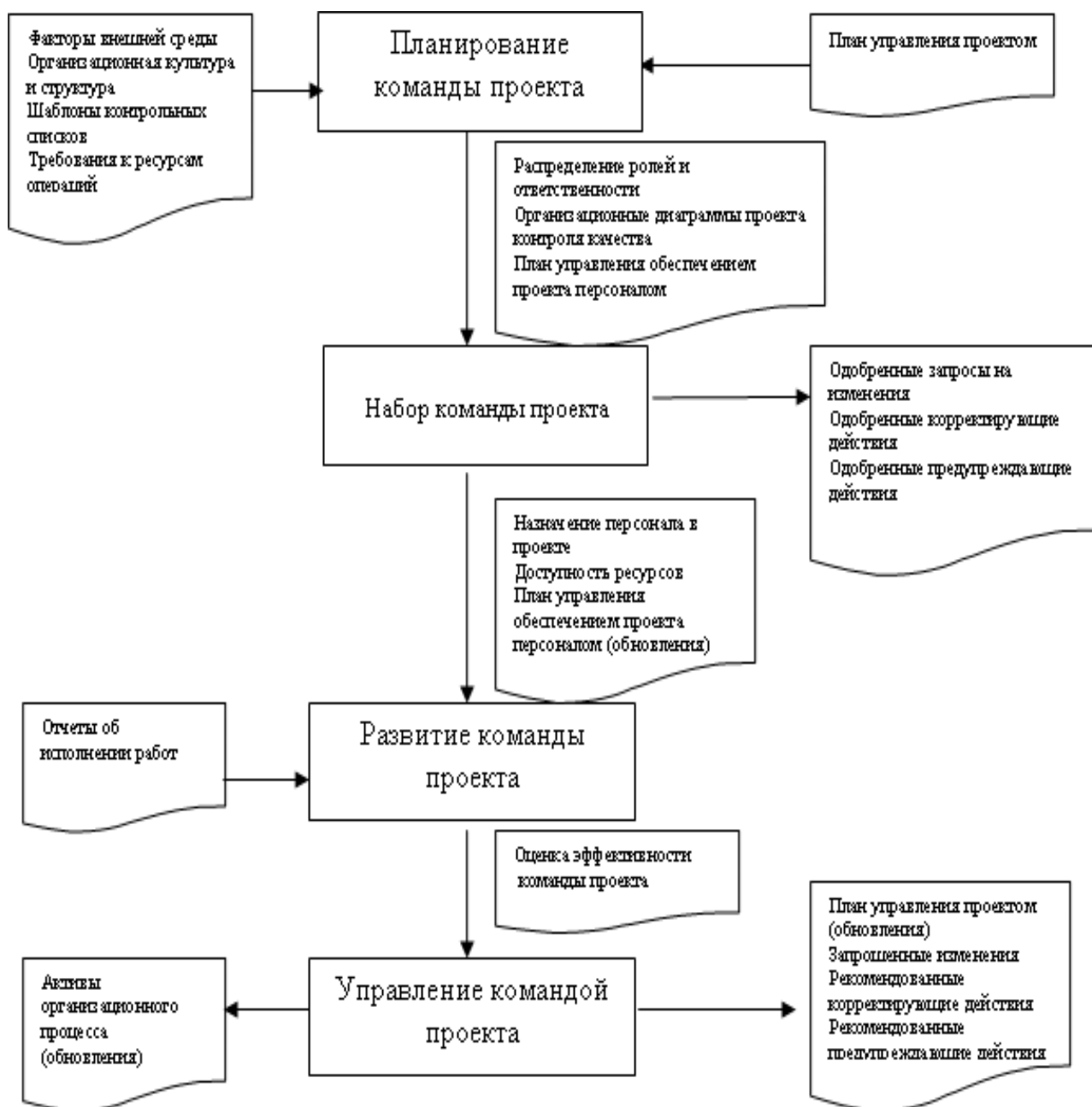


Рис. 5.8. Процессы управления командой проекта

Процессам управления сроками проекта предшествует процесс планирования, определяющий формат и критерии разработки и контроля расписания проекта, управления проектом, в ходе которого разрабатывается план управления расписанием. План управления расписанием входит в план управления проектом либо является его вспомогательным планом.

Планирование сроков проекта может быть выполнено с помощью специализированных программных средств. Пример планирования сроков работ в специализированной системе MS Project приведен на рис. 5.10.

Разработка расписания производится непрерывно по мере выполнения работ проекта.

Управление стоимостью заключается в обеспечении выполнения тройного ограничения на управление проектами – по стоимости, срокам и содержанию.

Управление качеством (в рамках управления проектом) – это система методов, средств и видов деятельности, направленных на выполнение требований участников проекта к качеству самого проекта и его продукции.

Основными процессами обеспечения качества проекта являются планирование качества, его обеспечение и контроль.

Управление человеческими ресурсами проекта – это процесс обеспечения эффективного использования человеческих ресурсов проекта, к которым относятся все участники проекта (спонсоры, заказчики, команда проекта, субподрядчики, подразделения компании и другие участники проекта).

Роль в проекте (проектная роль) – определенный набор функций и полномочий в проекте, созданный в целях распределения обязанностей между членами команды проекта. Проектную роль можно рассматривать как временную должность в организации (компании).



Рис. 5.9. Процесс создания иерархической структуры работ

12.01.09				
	Название задачи	Длительность	Начало	Окончание
1	ВНЕДРЕНИЕ ИС	286 дней	17.12.07	30.01.09
2	Входящие вехи проекта	1 день	17.12.07	18.12.07
8	Обследование	32 дней	18.12.07	12.02.08
9	Формирование и согласование плана проведения интервью	1 день	18.12.07	18.12.07
10	Подготовка и рассылка опросных листов для интервью	1 день	19.12.07	19.12.07
11	Проведение интервью для описания бизнес-процессов	10 дней	20.12.07	15.01.08
12	Веха: Завершен сбор информации для описания бизнес-процессов	0 дней	15.01.08	15.01.08
13	Описание бизнес-процессов	20 дней	16.01.08	12.02.08
14	Описание бизнес-процессов по функциональной области Финансы	20 дней	16.01.08	12.02.08
15	Описание бизнес-процессов по функциональной области Логистике	15 дней	16.01.08	05.02.08
16	Описание бизнес-процессов по функциональной области Персонал	12 дней	16.01.08	31.01.08
17	Веха: Обследование завершено	0 дней	31.01.08	31.01.08
18	Разработка системы	231 дней	01.02.08	19.12.08
19	Разработка решений по функциональной архитектуре	21 дней	01.02.08	29.02.08
20	Подготовка функционального дизайна расширений	35 дней	03.03.08	18.04.08
21	Настройка системы	40 дней	21.04.08	13.06.08
22	Техническое проектирование расширений	24 дней	16.06.08	17.07.08
23	Разработка расширений	44 дней	18.07.08	17.09.08
24	Техническое проектирование программ конвертации данных	25 дней	18.09.08	22.10.08
25	Разработка программ конвертации данных	36 дней	23.10.08	11.12.08
26	Планирование тестирования приложения и интеграционного тестирования	6 дней	12.12.08	19.12.08
27	Тестирование системы	30 дней	22.12.08	30.01.09
28	Разработка сценариев тестирования	9 дней	22.12.08	01.01.09
29	Подготовка тестовых данных	6 дней	02.01.09	09.01.09

Рис. 5.10. Планирование работ в MS Project

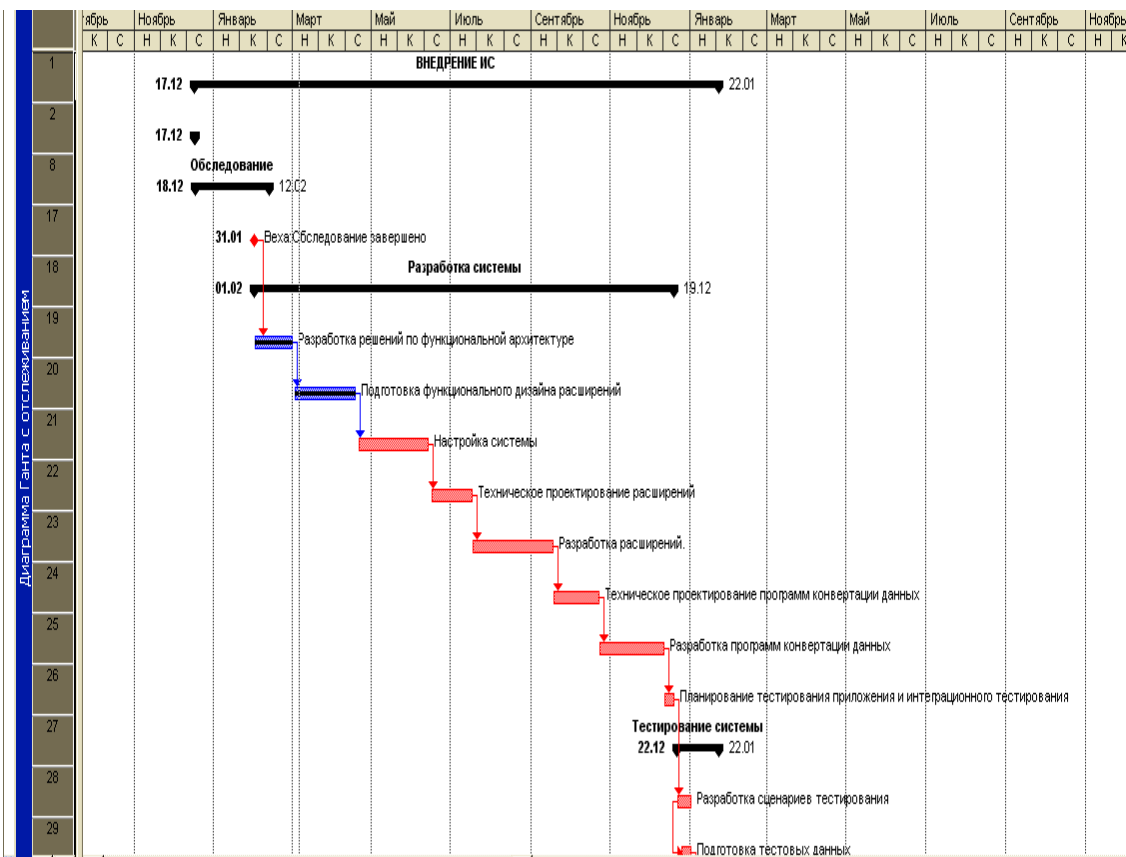


Рис. 5.11. Фрагмент расписания проекта в виде диаграммы Ганта MS Project

Команда проекта – временная рабочая группа, выполняющая работы по проекту и ответственная перед Руководителем проекта за их выполнение. Команда проекта состоит из команды управления, участников проекта, выполняющих работы в рамках проекта – исполнителей проекта.

Для обеспечения всех необходимых функций управления проектом команда управления проектом должна включать в свой состав участников со следующими ролями: руководитель проекта, куратор проекта, архитектор системы, администратор проекта. Подчиненность членов команды управления представлена на рис. 5.14.

Архитектор системы – проектная роль должностного лица, отвечающего за предметную область проекта. Архитектор системы подчиняется непосредственно Руководителю проекта.

Архитектор системы непосредственно отвечает за разработку информационной системы в соответствии с плановыми сроками проекта и с заданным уровнем качества.

Основные функции:

- определение состава, продолжительности и технологии выполнения работ по разработке и внедрению информационной системы;
- определение ресурсов, которые необходимы для разработки и внедрения ИС в рамках, заданных условиями проекта;
- определение квалификационных требований и состава рабочих групп специалистов по направлениям деятельности, распределение их по задачам, организация работ и верификация результатов в процессе реализации проекта;
- обеспечение целостности функциональной архитектуры внедряемой информационной системы;
- организация подготовки, согласования и утверждения всей технической документации, необходимой для создания ИС в рамках проекта;
- планирование и согласование фактических трудозатрат специалистов при исполнении проекта;
- формирование и предоставление руководителю проекта необходимой отчетности;
- анализ хода выполнения и промежуточных результатов создания ИС;
- организация, проведение и документирование процедур передачи Заказчику разработанной ИС.

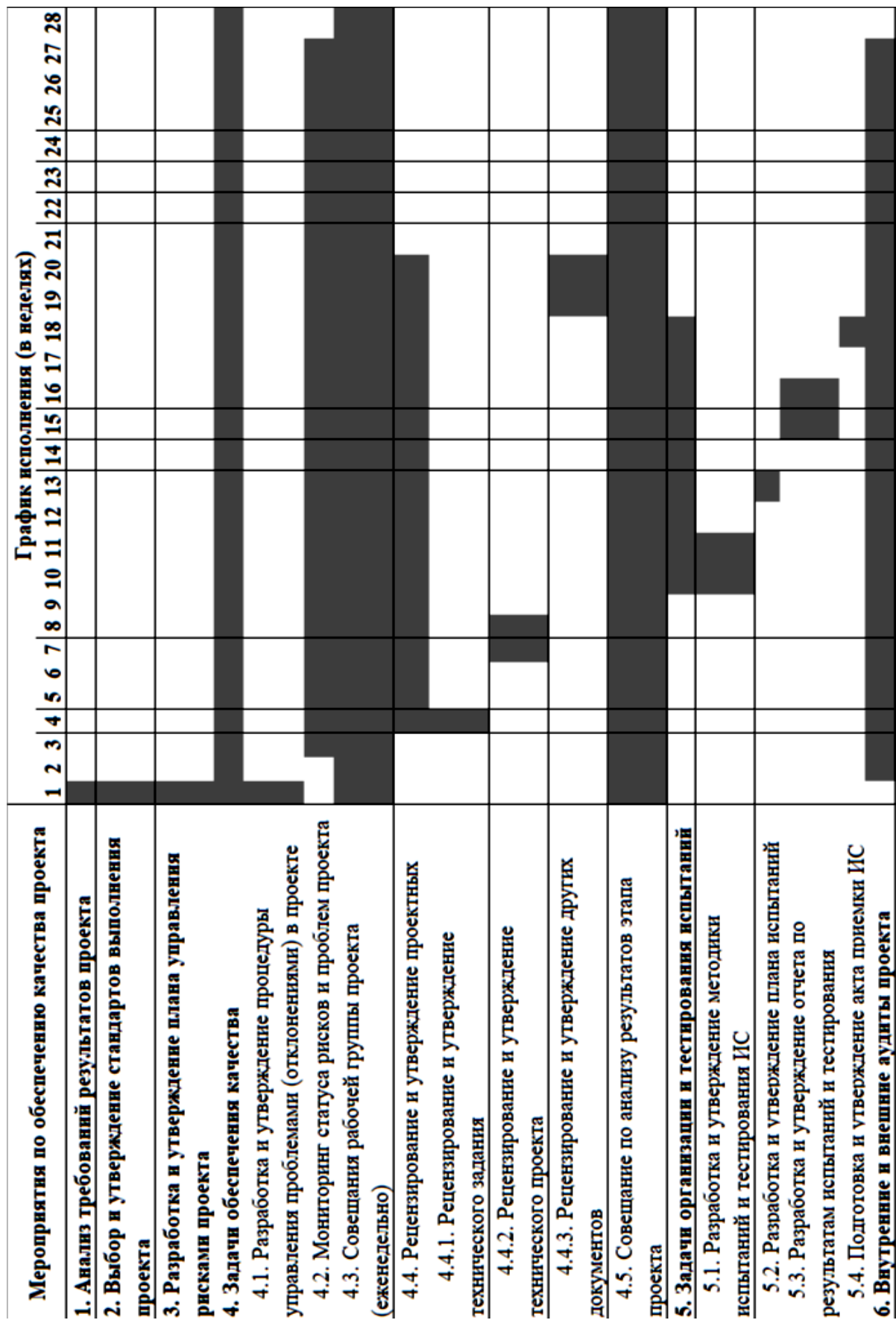


Рис. 5.13. Фрагмент плана обеспечения качества проекта

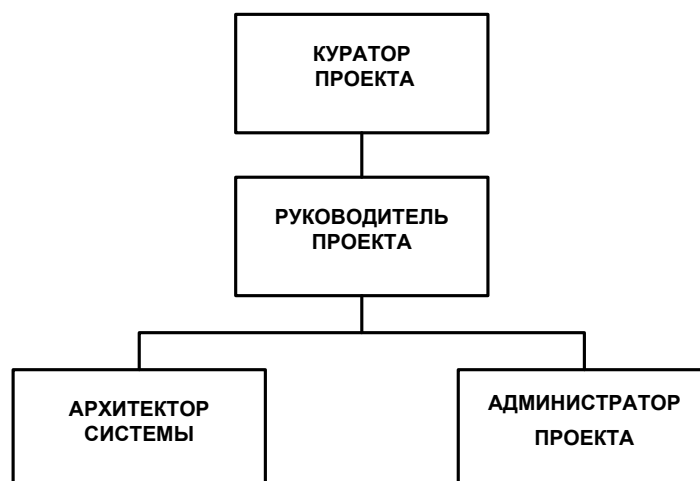


Рис. 5.14. Подчиненность членов команды управления проектом

Администратор проекта – проектная роль должностного лица, отвечающего за информационное обеспечение руководителя проекта, организацию и ведение документооборота по проекту. Администратор проекта функционально закрепляется за конкретным проектом и подчиняется непосредственно Руководителю проекта.

Основные функции:

- обеспечение Руководителя проекта структурированной информацией, обеспечивающей возможность контроля за проектом, планами, ресурсами и приоритетами;
- ведение протоколов совещаний;
- обеспечение своевременной подготовки, движения и архивации документов по проекту.

Распределение основных функций между членами команды управления проекта представлено на рис. 5.15.

Управление командой проекта включает в себя следующие процессы по организации команды проекта и управления ею.

Планирование человеческих ресурсов – процесс определения и документального оформления ролей, ответственности и подотчетности, а также создание плана управления обеспечением проекта персоналом.

Набор команды проекта – процесс привлечения человеческих ресурсов, необходимых для выполнения проекта.

На рисунке 5.16 представлен процесс документирования распределения ролей и ответственности членов команды проекта, выполненного в виде организационной структуры.

Функциональные обязанности	Куратор проекта	Руководитель проекта	Архитектор системы	Администратор проекта
Планирование				
Разработка и периодическая актуализация плана				
Утверждение плана				
Управление командой проекта				
Назначение сотрудника на роль Руководителя проекта				
Формирование команды проекта				
Определение квалификационных требований и состава рабочих групп специалистов по функциональности ИС				
Обеспечение выделения необходимых ресурсов для выполнения проекта				
Непосредственное руководство Командой проекта		+		
Формирование предложений по стимулированию Команды проекта				
Обеспечение стимулирования Команды проекта				
Организация выполнения работ				
Организация взаимодействия с Заказчиком и обеспечение всех необходимых коммуникационных связей с другими участниками проекта				
Организация подготовки, согласования и утверждения всей технической документации, необходимой для создания ИС в рамках проекта				
Организация, проведение и документирование процедур передачи Заказчику разработанной ИС				
Рассмотрение и утверждение регламентирующих документов, необходимых для организации и выполнения проекта				
Ведение организационно-распорядительной и отчетной документации. Поддержание в актуальном состоянии списка команды проекта				
Обеспечение команды проекта необходимыми информационными материалами				
Материально-техническое и хозяйственное обеспечение команды проекта				
Контроль хода выполнения проекта				
Организация и проведение совещаний по обсуждению хода работ проекта				
Подготовка и предоставление Куратору отчетов о ходе работ проекта				
Получение и анализ сводной отчетности о ходе реализации проекта				
Контроль соответствия результатов проекта Техническому заданию на разработку ИС				
Согласование фактических трудозатрат специалистов при исполнении проекта				

**Рис. 5.15. Распределение основных функциональных
обязанностей команды управления проектом**



Рис. 5.16. Организационная структура проекта

Далее на рисунках представлены модели управления качеством.

Диаграмма процесса «Построение СМК»

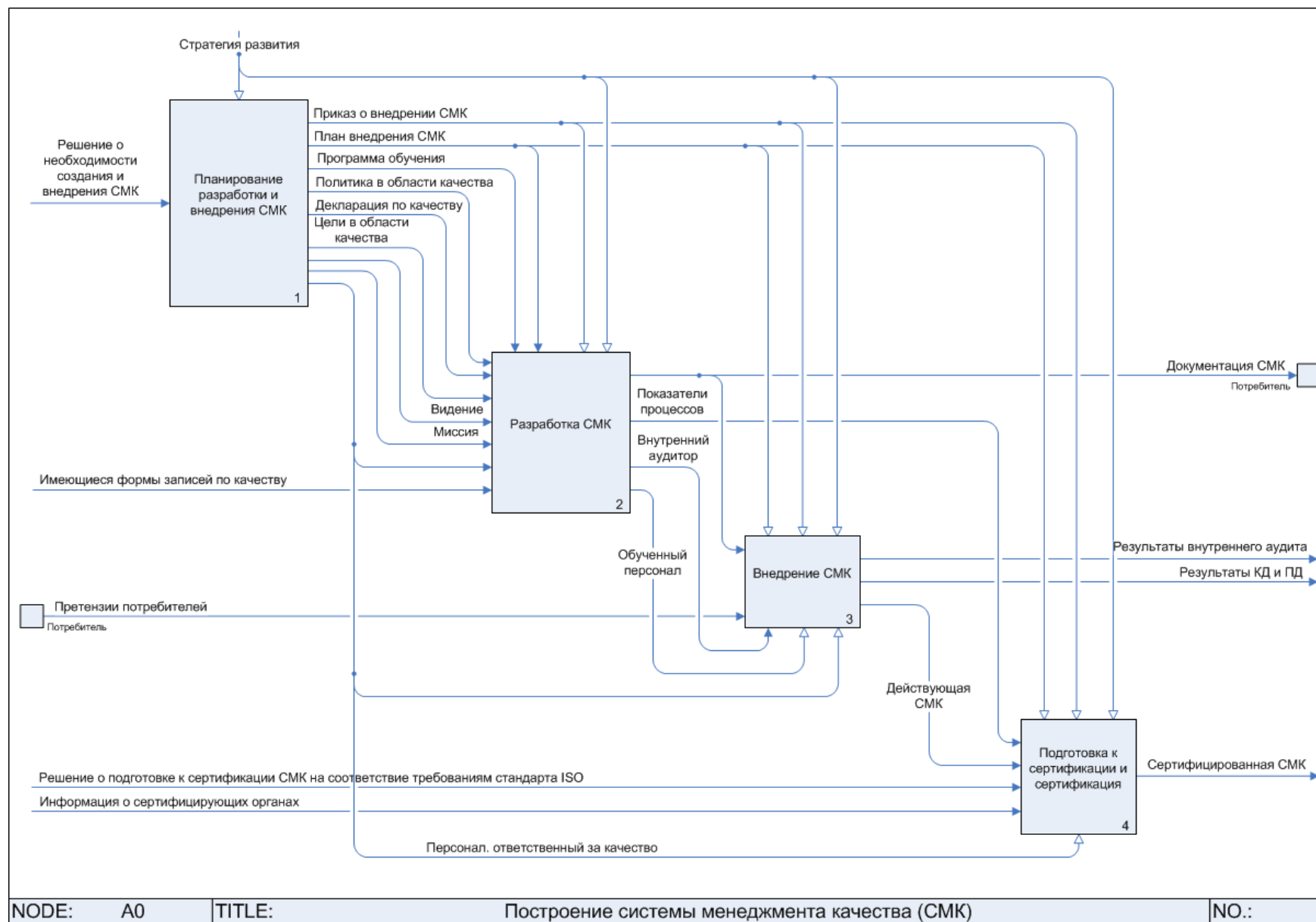


Диаграмма процесса «Планирование разработки и внедрения СМК»

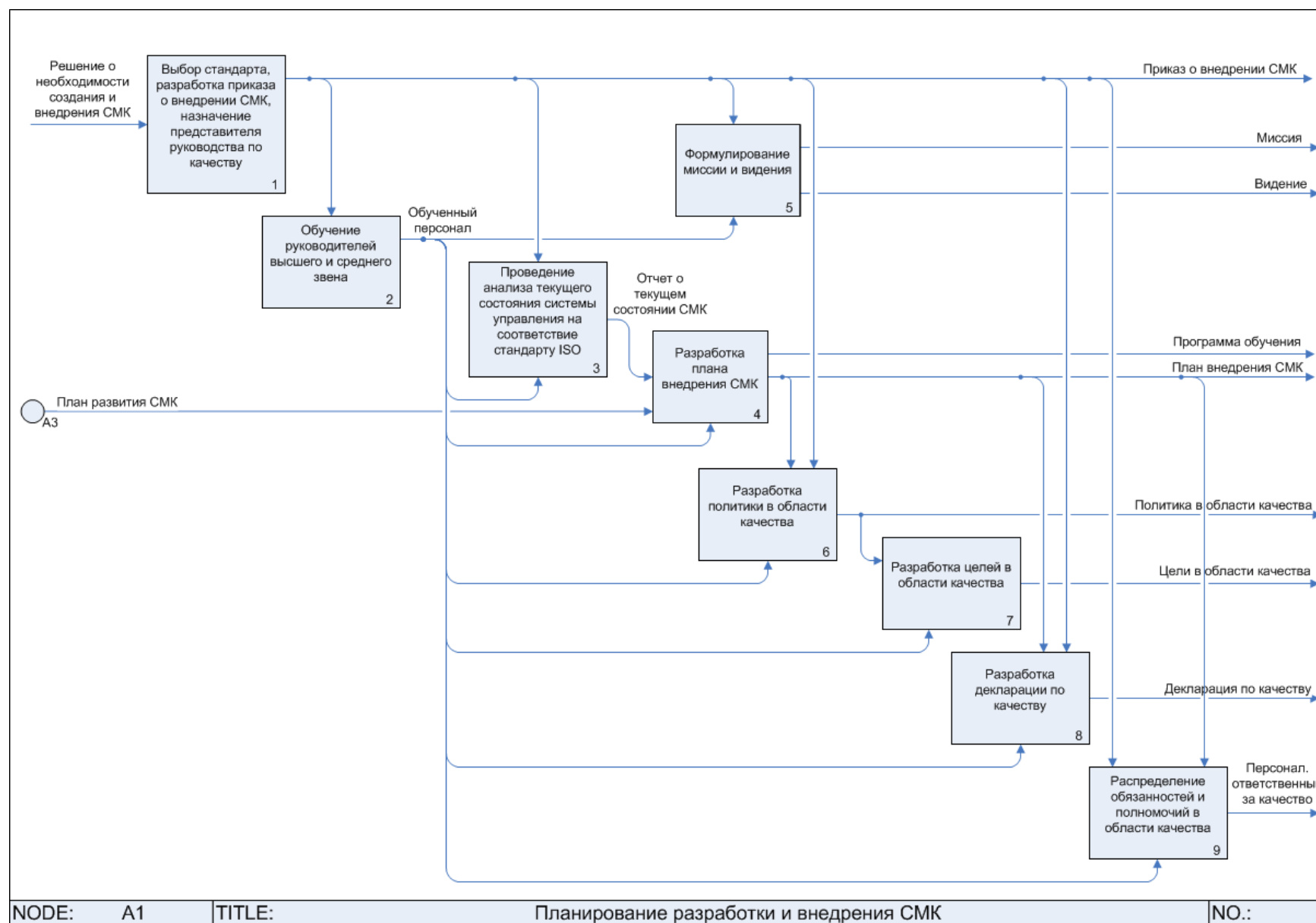


Диаграмма процесса «Разработка СМК»

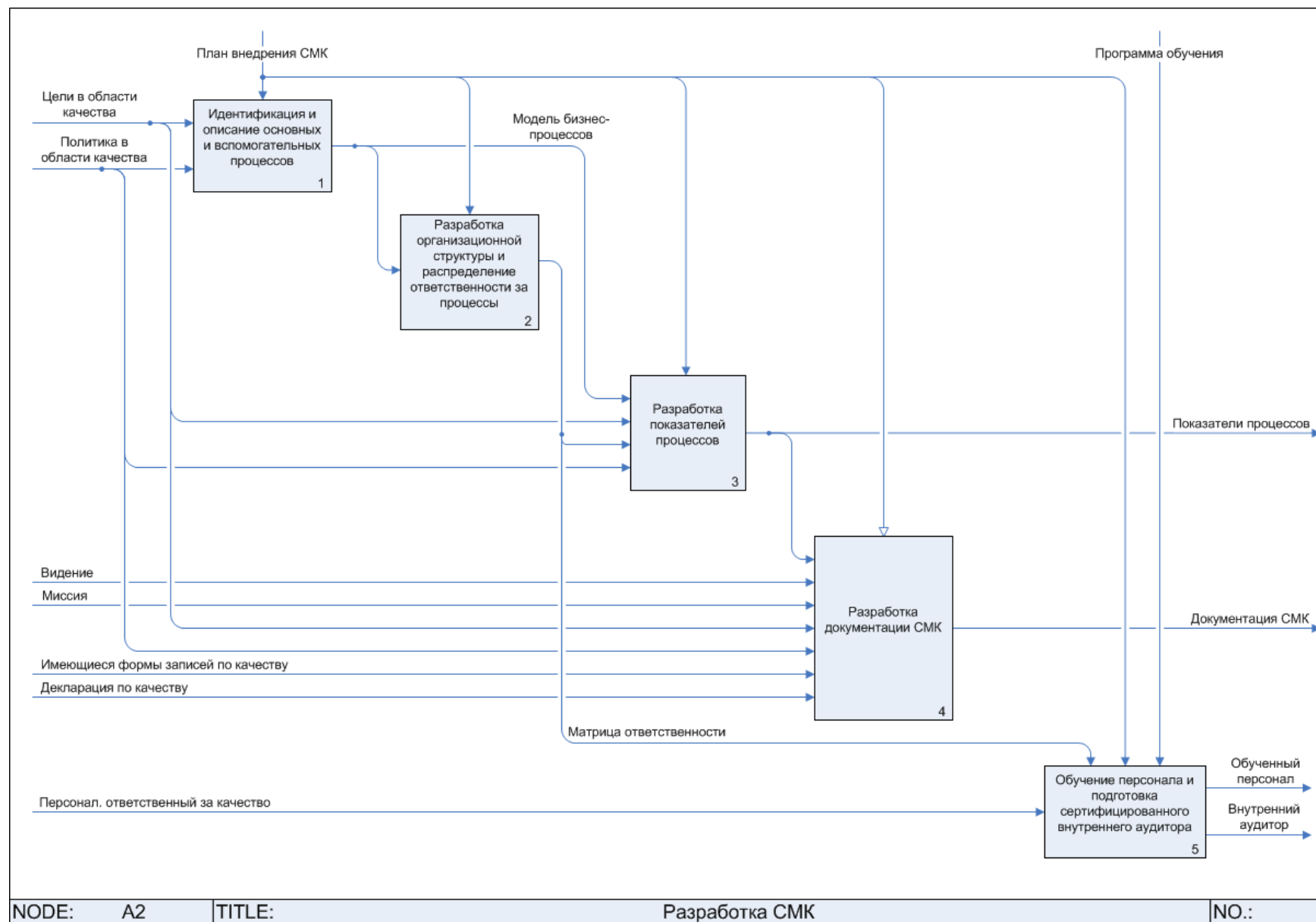


Диаграмма процесса «Разработка документации СМК»

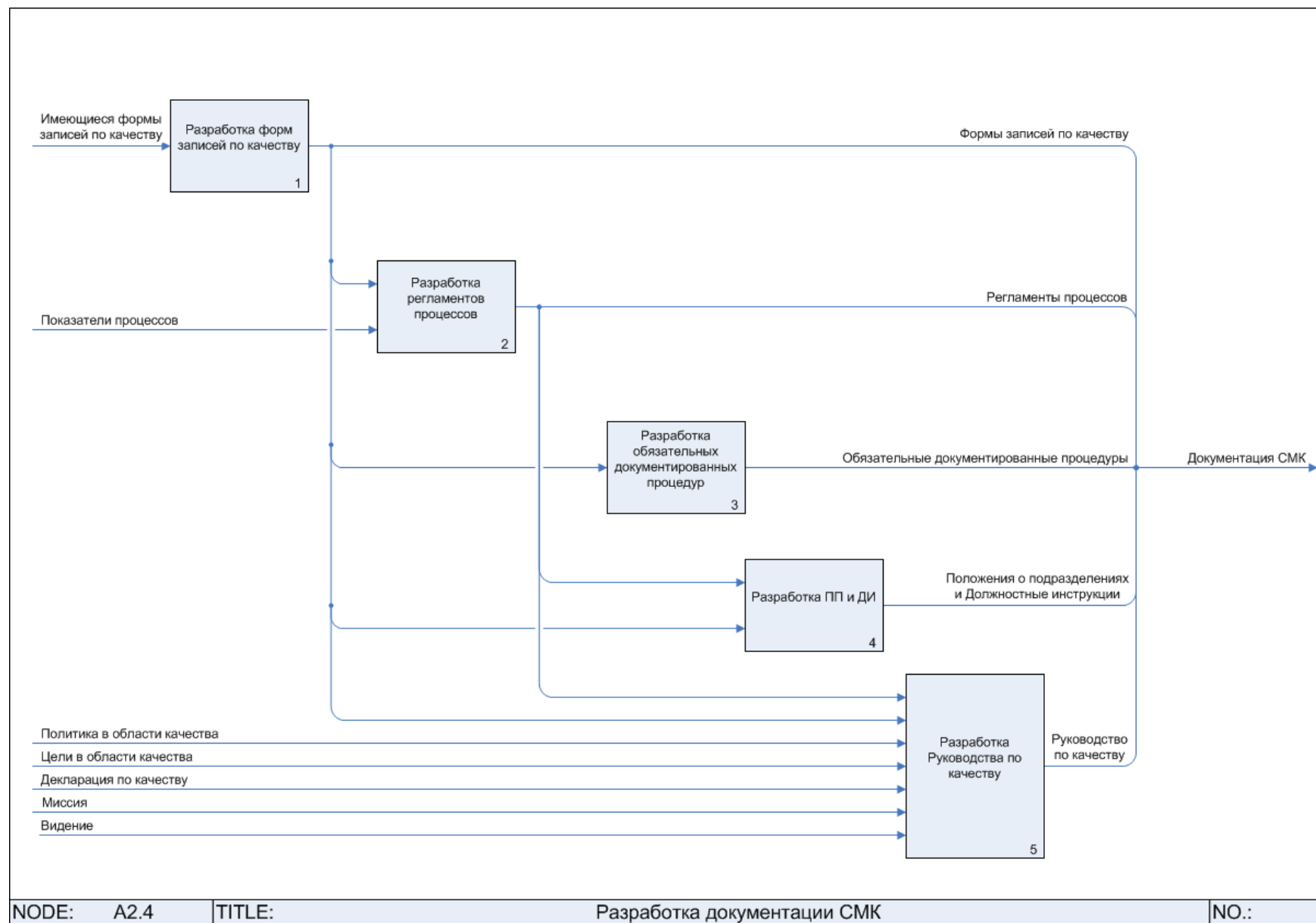


Диаграмма процесса «Внедрение СМК»

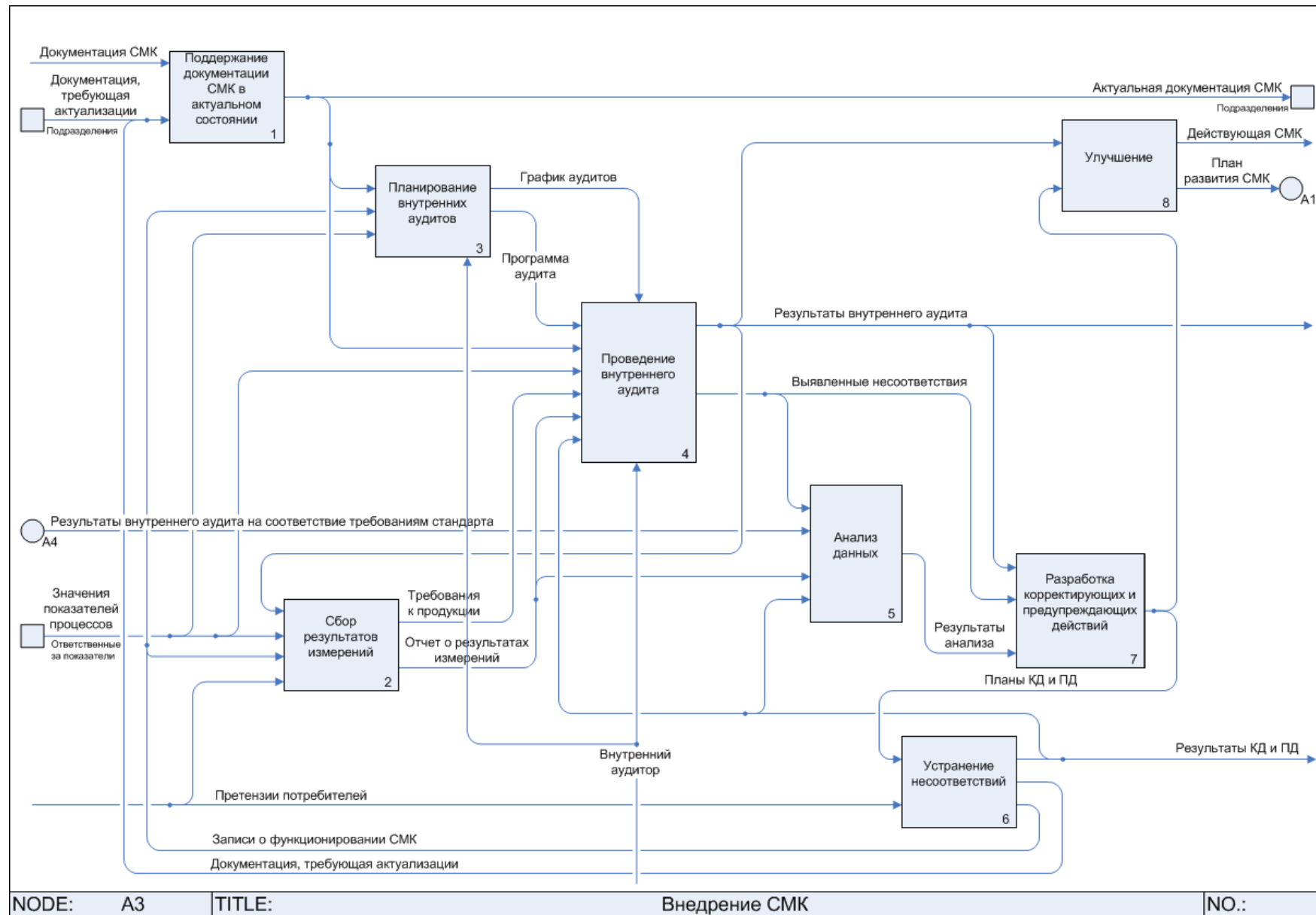
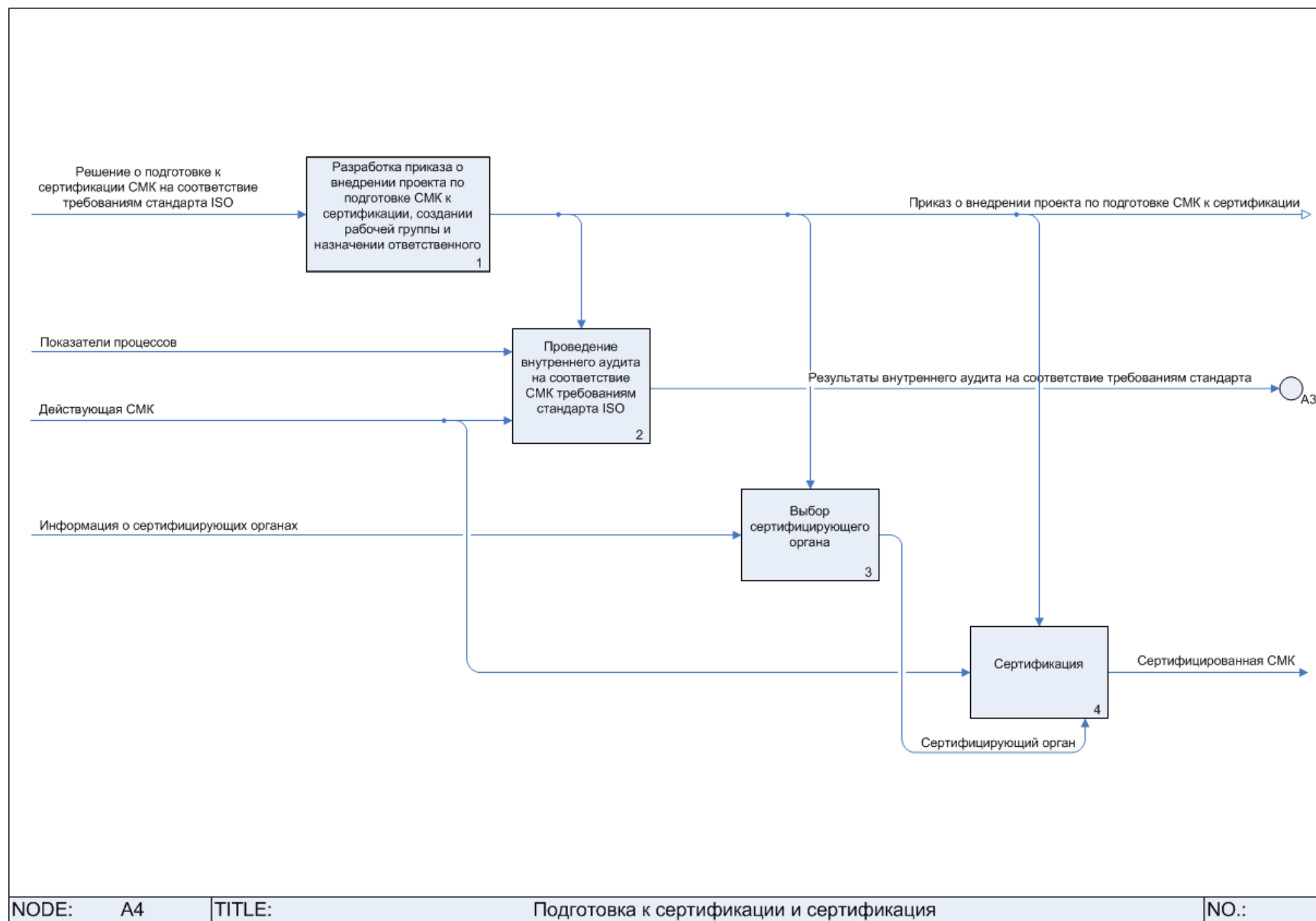


Диаграмма процесса «Подготовка к сертификации и сертификация»



6. ТЕХНИКО-ЭКОНОМИЧЕСКИЙ АНАЛИЗ РАЗРАБОТКИ ПРОЕКТОВ ПРОГРАММНЫХ СРЕДСТВ

Технико-экономический анализ разработки проектов программных средств – это выбор и прогнозирование наиболее адекватных экономических и функциональных критериев для обобщенного описания эффективности, стоимости создания и использования проектов программных средств в зависимости от их назначения, области применения и прочих факторов.

Различаются следующие типы текста программы:

- новый код текстов программ, разработанный полностью для нового приложения, который не включает фрагменты и процедуры ранее написанного и испытанного кода;
- модифицируемый код, разработанный для предыдущих проектов программных средств, который становится пригодным для использования в новом проекте после внесения умеренного объема изменений;
- повторно используемый код, разработанный для предыдущих программных средств, который будет полностью пригодным для новых приложений без внесения каких-либо изменений;
- наследственный код, разработанный для предыдущих проектов, использование которого ожидается новым комплексом программ.

К основным затратам, связанным с проектом, можно отнести:

- допустимые трудозатраты на разработку программного средства с требуемым качеством»;
- время (длительность полного цикла создания программного продукта);
- необходимое и доступное число специалистов соответствующей квалификации.

Трудоемкость разработки программных средств наиболее сильно зависит от масштаба-размера программ, выраженного числом операторов на ассемблере или строк на языке программирования высокого уровня.

Общая трудоемкость разработки программного средства рассчитывается по формуле

$$T_{\text{общ}} = K_{\text{сл}} T_p,$$

где T_p – значение трудоемкости, определенной по объему разрабатываемого программного средства, человеко-дни; $K_{\text{сл}}$ – дополнительный коэффициент сложности.

Трудоемкость определяется для каждого этапа разработки, затем уточненная общая трудоемкость разработки рассчитывается по следующей формуле:

$$T_{\text{общ}} = \sum_{j=1}^k T_j,$$

где T_j – трудоемкость на стадиях технического задания, эскизного проекта, технического проекта, рабочего проекта и внедрения.

Количество исполнителей, необходимое для выполнения работ по созданию программного обеспечения, определяется по формуле

$$\text{Ч} = \frac{T_{\text{общ}}}{\Phi_{\text{д}} D},$$

где Ч – численность исполнителей, чел.; $\Phi_{\text{д}}$ – действительный (полезный) фонд времени одного работающего в месяц, дн; D – директивный срок выполнения разработки, мес.

По уровню сложности все множество программных систем (ПС) можно разбить на три типа.

К первому типу относятся:

- 1) комплексные программные системы (КПС) и технологии, отдельные части которых реализованы на различных платформах;
- 2) территориально-распределенные программные системы и технологии;
- 3) системы автоматизированного либо автоматического управления, функционирующие в режиме реального времени.

Второй тип составляют программные информационно-справочные системы (ИПС), обеспечивающие информационную поддержку основных функций (бизнес-процессов) организации с большим количеством типов исходной информации.

К третьему типу относятся инженерные и научно-технические пакеты программ (ППП) и технологий, характеризующихся четко заданным алгоритмом обработки и малыми объемами исходных данных.

7. МЕТОДЫ ОЦЕНКИ ТРУДОЗАТРАТ НА РАЗРАБОТКУ ПРОГРАММНОЙ СИСТЕМЫ

7.1. ПРЯМОЙ МЕТОД ОЦЕНКИ ТРУДОЗАТРАТ

Прямой метод определения технико-экономических параметров программной системы основан на использовании метода экспертных оценок.

Программная система декомпозируется до уровня элементарных компонент. Оценка размеров каждого из компонент производится либо внешним экспертом, имеющим опыт разработки подобных систем и готовые прототипы, либо специалистами разработчика и заказчика [5].

При декомпозиции следует использовать градации, термины и определения, соответствующие ГОСТ 34.003–90 [6]. Основные из них следующие:

- интегрированная программная система – совокупность двух и более программных систем, в которых функционирование одной из них зависит от результатов функционирования другой;
- программная система – совокупность программных комплексов, реализующих множество функций организации;
- программный комплекс – совокупность программных компонент, реализующих конкретную функцию (бизнес-процесс);
- сложная программная компонента – совокупность программных кодов, реализующих сложную функцию бизнес-процесса;
- программная компонента – совокупность программных кодов, реализующих элементарную функцию бизнес-процесса.

Размеры программной системы определяются в виде количества строк исходного кода в терминах «Lines of Code-LOC» [3].

При оценке количества строк исходного кода следует учитывать следующие положения:

- строка исходного кода содержит только один оператор;
- определение (описание) исходных данных учитывается один раз;

– не учитываются строки, содержащие комментарии и отладочные операторы;

– учитывается каждая инициализация, вызов, либо включение макроса в качестве исходного кода.

В качестве базового показателя количества строк исходного кода следует использовать число операторов языка ассемблер. Преобразование размеров программы по этому измерителю в размеры программы кода, написанного на других языках программирования и наоборот, представлены в табл. 7.1, 7.2.

7.1. Показатель LOC для языков и систем программирования

№ п/п	Язык программирования	Ассемблер LOC	Показатель LOC на функциональную точку
1	Basic Assembler	1	320
2	Macro Assembler	1,5	213
3	Basic	3	107
4	Pascal	3,5	91
5	C++	6	53
6	Java	6	53
7	Oracle, Sybase	8	40
8	Access	8,5	38
9	Delphi	11	29
10	Oracle Developer/2000	14	23
11	Smalltalk	15	21
12	Cobra	16	20
13	HTML	22	15
14	SQL (ANSI)	25	13
15	Exel	50	6

7.2. Бланк экспертной оценки размерности программной системы

Состав программной системы	Оценки		
	оптимистическая	реалистическая	пессимистическая
1. Программная система			
1.1. Программный комплекс			
1.1.1. Программная компонента 1			
1.1.2. Программная компонента 2			
.....			

Средняя оценка по компоненте программной системы (e_{ij}^k) бета-распределение будет определяться как

$$e_{ij}^k = \frac{o_{ij}^k + 4r_{ij}^k + p_{ij}^k}{6}, \quad (7.1)$$

где k – индекс (номер) эксперта; i – индекс (номер) программного комплекса; j – индекс (номер) программного компонента.

Результирующая оценка размера программ определяется путем суммирования экспертных оценок:

$$R = \frac{\sum_{k=1}^L \sum_{i=1}^N \sum_{j=1}^{M_i} e_{ij}^k}{L}, \quad (7.2)$$

где L – количество экспертов; N – число программных комплексов в программной системе; M_i – число программных компонентов в i -м программном комплексе.

Эффективность оценивания может быть существенно повышена при наличии прототипов будущей программной системы. В этом случае эксперту предлагается оценить необходимость степени модернизации имеющегося прототипа.

Оценка трудозатрат, длительности и средней численности разработчиков при реализации проекта основывается на согласовании между разработчиком и заказчиком производительности труда программиста – P .

Среднестатистические оценки производительности труда программиста, полученные на основании многолетних наблюдений, следующие:

- при разработке программных систем первого класса сложности преимущественно на языке ассемблер – 60 – 80 строк/чел.-месяц;
- при разработке программных систем второго класса сложности (ИПС) на языках высокого уровня – 250 – 260 строк/чел.-месяц.

В таблице 7.3 представлены статистические показатели производительности P , рекомендуемые в базовой модели Constructive Cost Model (COCOMO).

7.3. Нормативы трудоемкости разработки программ Р

Класс сложности ПС	Размеры ПС	
	до 30 тыс. строк	до 500 тыс. строк
Первый тип (КПС)	До 140 тыс. строк/чел.-мес	До 80 тыс. строк/чел.-мес
Второй тип (ИПС)	До 220 тыс. строк/чел.-мес	До 160 тыс. строк/чел.-мес
Третий тип (ППП)	До 250 тыс. строк/чел.-мес	До 250 тыс. строк/чел.-мес

Приведенные нормативы отражают не только трудоемкость непосредственного написания текстов программ, но и процессы комплексирования и испытания всего программного комплекса.

Прямой метод оценки технико-экономических показателей целесообразно использовать на ранних стадиях проектирования при разработке концепции и технического задания на будущую программную систему. Это позволит разработчику и заказчику определить трудоемкость реализации каждой функции, проранжировать их в соответствии с пожеланиями заказчика, соизмерить финансовые возможности заказчика и сроки реализации проекта.

7.2. ОЦЕНКА ТРУДОЗАТРАТ МЕТОДОМ ФУНКЦИОНАЛЬНЫХ ТОЧЕК

Оценка трудозатрат методом функциональных точек (Function point – FP) основывается на определении размера программной системы в терминах количества и сложности функций, реализуемых в данном программном коде.

Будущая система описывается в виде многоуровневой графической модели. Каждый из функций включает в себя входные и выходные данные, преобразования, внешние интерфейсы.

Процедура оценивания размеров программной системы состоит из следующей последовательности этапов:

- выделение множества функций (бизнес-процессов);
- оценка количества функциональных точек бизнес-процесса каждой категории;
- определение весовых коэффициентов сложности каждой функции;

- учет факторов и требований среды разработки программной системы;
- вычисление интегральных показателей сложности;
- вычисление итогового количества функциональных точек;
- определение размеров программного комплекса в показателях LOC (согласно табл. 7.1);

- определение размеров программной системы в целом.

При определении количества функций каждого бизнес-процесса следует руководствоваться следующими требованиями:

- учитываются только сложные функции, перечисленные в техническом задании (в соглашении о требованиях);

- при декомпозиции сложной функции учитываются все логические преобразования с данными.

Расчет количества функциональных точек по каждому бизнес-процессу сводится в табл. 7.4.

7.4. Таблица функциональных точек

№ п/п	Категория функций	Простые	Средние	Сложные	Количество функциональных точек
1	Количество выводов	$\alpha_{11}x_{11}$	$\alpha_{12}x_{12}$	$\alpha_{13}x_{13}$	$X_i = \sum_{i=1}^3 \alpha_{1i}x_i$
2	Количество вводов	$\alpha_{21}x_{21}$	$\alpha_{22}x_{22}$	$\alpha_{23}x_{23}$	$X_i = \sum_{i=1}^3 \alpha_{2i}x_i$
3	Количество опросов вывода	$\alpha_{31}x_{31}$	$\alpha_{32}x_{32}$	$\alpha_{33}x_{33}$	$X_i = \sum_{i=1}^3 \alpha_{3i}x_i$
4	Количество опросов ввода	$\alpha_{41}x_{41}$	$\alpha_{42}x_{42}$	$\alpha_{43}x_{43}$	$X_i = \sum_{i=1}^3 \alpha_{4i}x_i$
5	Количество файлов	$\alpha_{51}x_{51}$	$\alpha_{52}x_{52}$	$\alpha_{53}x_{53}$	$X_i = \sum_{i=1}^3 \alpha_{5i}x_i$
6	Количество интерфейсов	$\alpha_{61}x_{61}$	$\alpha_{62}x_{62}$	$\alpha_{63}x_{63}$	$X_i = \sum_{i=1}^3 \alpha_{6i}x_i$
Общее количество функциональных точек					$F = \sum_{i=1}^6 x_i$

Примечание. В формулах α_{ij} – весовой коэффициент сложности i -й функции j -й категории;
 x_{ij} – количество элементов данных i -й функции j -й категории.

Под выводами понимаются единицы информации, получаемые на выходе рассматриваемой функции:

- файлы, создаваемые в данной функции для передачи другим бизнес-процессам либо за пределы программной системы;
- единицы деловой информации, предназначенные для конечных пользователей, оформленные в виде экранных форм либо бумажных документов.

Каждый из выводов, в зависимости от количества файлов, используемых при формировании выходов, следует отнести к одной из категорий сложности: простой, средний, сложный. В таблице 7.5 приводятся весовые коэффициенты сложности выводов.

Под вводами будем понимать единицы информации, поступающие на вход рассматриваемой функции:

- входные файлы, полученные из других бизнес-процессов либо других программных систем;
- уникальная единица деловой информации, вводимая конечным пользователем.

По аналогии с выводом все вводы также рекомендуется разделять на простые, средние и сложные (табл. 7.6).

7.5. Весовые коэффициенты сложности вывода

Количество файлов	Количество элементов данных		
	1 – 5	6 – 19	≥ 20
1 (простой)	$\alpha_{11} = 4$	$\alpha_{12} = 4$	$\alpha_{13} = 5$
2 (средний)	$\alpha_{11} = 4$	$\alpha_{12} = 5$	$\alpha_{13} = 7$
≥ 4 (сложный)	$\alpha_{11} = 5$	$\alpha_{12} = 7$	$\alpha_{13} = 7$

7.6. Весовые коэффициенты сложности ввода

Количество файлов	Количество элементов данных		
	1 – 5	6 – 19	≥ 20
1 (простой)	$\alpha_{21} = 4$	$\alpha_{22} = 4$	$\alpha_{23} = 5$
2 (средний)	$\alpha_{21} = 4$	$\alpha_{22} = 5$	$\alpha_{23} = 7$
≥ 4 (сложный)	$\alpha_{21} = 5$	$\alpha_{22} = 7$	$\alpha_{23} = 7$

Под опросами понимаются действия, исполняемые программной системой в рассматриваемой функции:

- обращение к внешним процедурам, оформленным в виде команд или запросов, генерируемых извне и выполняемых программной системой;
- выполнение процедур, обеспечивающих непосредственный доступ к базе данных и выполняющих выборку с помощью простых ключей в режиме реального времени (но не выполняющих функции обновления).

Рекомендуется учитывать каждую уникальную единицу опроса, если:

- формат опроса отличается от формата ввода, вывода;
- формат опроса совпадает с форматом ввода, вывода, но требует дополнительной логики обработки.

При определении количества опросов не следует учитывать запросы к базам данных, использующие несколько ключей и выполняющие определенные операции либо вычисления с последующим оформлением выводов.

Все опросы также рекомендуется разделять на простые, средние и сложные. В таблицах 7.7 и 7.8 приведены рекомендации по выбору весовых коэффициентов.

7.7. Весовые коэффициенты сложности опросов вывода

Количество файлов	Количество элементов данных		
	1 – 5	6 – 19	≥ 20
1 (простой)	$\alpha_{31} = 4$	$\alpha_{32} = 4$	$\alpha_{33} = 5$
2 (средний)	$\alpha_{31} = 4$	$\alpha_{32} = 5$	$\alpha_{33} = 7$
≥ 4 (сложный)	$\alpha_{31} = 5$	$\alpha_{32} = 7$	$\alpha_{33} = 7$

7.8. Весовые коэффициенты сложности опросов ввода

Количество файлов	Количество элементов данных		
	1 – 5	6 – 19	≥ 20
1 (простой)	$\alpha_{41} = 4$	$\alpha_{42} = 4$	$\alpha_{43} = 5$
2 (средний)	$\alpha_{41} = 4$	$\alpha_{42} = 5$	$\alpha_{43} = 7$
≥ 4 (сложный)	$\alpha_{41} = 5$	$\alpha_{42} = 7$	$\alpha_{43} = 7$

Под файлами будем понимать единицы информации, используемые программной системой в рассматриваемой функции:

- внутренние логические файлы программной системы;
- структуры данных, которые постоянно находятся внутри границ программной системы;
- внешние файлы, доступные пользователям с помощью ввода, вывода, опросов, либо интерфейсов.

Весовые коэффициенты оценки сложности файлов, в зависимости от количества взаимосвязей между таблицами, представлены в табл. 7.9.

Под интерфейсами, используемыми рассматриваемой функцией, будем понимать:

- файлы, сгенерированные другими программными системами и используемые в данной ПС;
- потоки данных, хранящиеся за пределами программной системы, но используемые при управлении вычислительным процессом в любом направлении пересылки;
- структуры данных, используемые в нескольких программных системах.

Весовые коэффициенты оценки сложности интерфейсов представлены в табл. 7.10.

7.9. Весовые коэффициенты сложности структур данных (файлов)

Количество файлов	Количество элементов данных		
	1 – 19	20 – 50	≥ 51
1 (простой)	$\alpha_{51} = 5$	$\alpha_{52} = 5$	$\alpha_{53} = 7$
2 (средний)	$\alpha_{51} = 7$	$\alpha_{52} = 7$	$\alpha_{53} = 10$
≥ 4 (сложный)	$\alpha_{51} = 7$	$\alpha_{52} = 10$	$\alpha_{53} = 10$

7.10. Весовые коэффициенты сложности интерфейсов

Количество файлов	Количество элементов данных		
	1 – 19	20 – 50	≥ 51
1 (простой)	$\alpha_{61} = 5$	$\alpha_{62} = 5$	$\alpha_{63} = 7$
2 (средний)	$\alpha_{61} = 7$	$\alpha_{62} = 7$	$\alpha_{63} = 10$
≥ 6 (сложный)	$\alpha_{61} = 7$	$\alpha_{62} = 10$	$\alpha_{63} = 10$

Общее количество функциональных точек определяется по следующей формуле:

$$F = \sum_{i=1}^6 \sum_{j=1}^3 \alpha_{ij} x_{ji}.$$

Сложность предметной области и качества создаваемого программного обеспечения зависит от среды разработки приложений и требований конечных пользователей (факторов и требований среды разработки). Влияние этих факторов на размеры программного обеспечения оценивается по ряду показателей (табл. 7.11). Каждый из показателей в свою очередь оценивается по пяти-балльной шкале. Рекомендуемая шкала измерения показателей приведена в табл. 7.12.

Уровень влияния факторов внешней среды определяется по следующему соотношению:

$$Z = 0,65 + 0,01M, \quad (7.3)$$

где M – суммарное значение весовых коэффициентов факторов внешней среды.

Уточненное количество функциональных точек с учетом факторов внешней среды определяется по следующей формуле:

$$f = F \times Z. \quad (7.4)$$

Размерность программного обеспечения для конкретного языка программирования в LOC определяется с учетом нормативов, представленных в табл. 7.1 по соотношению:

$$R = f \times \text{LOC}. \quad (7.5)$$

Итоговая размерность программной системы определяется путем суммирования величины каждого R бизнес-процесса (функции).

Базовая модель оценки длительность разработки ПС.

По оценке значения трудозатрат R длительность разработки системы T может быть определена следующим образом:

$$T = \frac{R}{P}, \quad (7.6)$$

где P – производительность труда программиста.

7.11. Факторы среды обработки

№ п/п	Фактор среды	Варианты
1	Каналы передачи данных	Входные и выходные данные передаются по локальной сети, магистральным каналам связи, Internet
2	Распределенные вычисления	Пользовательские приложения используют данные, хранящиеся в едином хранилище, получают из других систем
3	Производительность системы	Скорость передачи данных и время отклика системы существенны для данной функции
4	Конфигурирование	Пользовательские приложения выполняются с применением интенсивно используемой, ограниченной либо наполненной конфигурации
5	Частота транзакций	Использование приложений приводит к высокому сетевому трафику, экранные формы динамично изменяются, наблюдается высокая концентрация выходных форм
6	Интерактивная обработка	Частота использования пользовательских приложений, участие пользователя при выполнении запросов
7	Пользовательский интерфейс	Значимость взаимодействия конечных пользователей с программной системой (необходимость дополнительного учета человеческого фактора)
8	Интерактивное обновление базы данных	Степень динамики обновления данных
9	Сложность обработки запросов	Уровень сложности алгоритмов обработки, количество транзакций, требования к безопасности и надежности
10	Сложность инсталляции программной системы	Наличие автоинсталляции, качество технической документации
11	Сложность эксплуатации системы	Наличие процедур запуска, резервирования, копирования, восстановления при ошибках, уровень (сложность) участия пользователей в этих процессах
12	Сложность распределенности системы	Количество и удаленность пользовательских приложений
13	Гибкость изменения функции	Модульная реализация, наличие настроек, уровень поддержки со стороны пользователей, возможность изменения запросов

7.12. Весовые коэффициенты факторов внешней среды

Влияние фактора не существенно	Влияние фактора существенно	Влияние фактора очень существенно
0 – 1	2 – 3	4 – 5

Если срок разработки D задан директивно заказчиком исходя из реальных потребностей и наличия финансовых ресурсов, то средняя численность специалистов N , которые должны быть привлечены к реализации программной системы, определяется по формуле

$$N = \frac{T}{D}. \quad (7.7)$$

8. МОДЕЛЬ СИСТЕМЫ УПРАВЛЕНИЯ ПЕРСОНАЛОМ

Описание основных вариантов использования:

Внести данные о новом сотруднике

Краткое описание:

Данный вариант использования описывает ввод данных о новом сотруднике.

Основной сценарий:

Активизируется, когда пользователь начинает вводить данные.

1. Пользователь вводит необходимые данные.
2. Система проверяет заполнение необходимых полей.
3. Система отправляет данные в БД.

Альтернативные сценарии:

1. Не все необходимые поля заполнены. Пользователь получает уведомление, после чего может исправить данные или выйти.
2. Нет доступа к БД. Пользователь получает уведомление о неисправности.

Предусловия:

1. Пользователь должен войти в систему.
2. Пользователь должен иметь минимальный уровень доступа 2 (отдел работы с персоналом).

Постусловия:

Нет.

Записать данные с проходной

Краткое описание:

Данный вариант использования описывает взаимодействие системы с электронной проходной.

Основной сценарий:

Активизируется, когда сотрудник проходит через проходную.

1. Сотрудник проходит через проходную.
2. Данные поступают на компьютер, связанный с проходной.
3. Данные отправляются в БД.
4. Система может извлекать данные из БД проходной для проверки.

Альтернативные сценарии:

1. Электроника в проходной не работает. Пользователю будет выдано сообщение о неисправности. Требуется вызвать работника, отвечающего за обеспечение работоспособности проходной. После можно повторить попытку.

2. Связанный компьютер не работает. Требуется устранить неполадки, после повторить попытку.

3. Нет доступа к БД. Пользователь получает уведомление о неисправности.

Предусловия:

Нет.

Постусловия:

Нет.

Войти в систему

Краткое описание:

Данный вариант использования описывает вход пользователя в систему.

Основной сценарий:

Активизируется, когда пользователь нажимает кнопку «Войти».

1. Система запрашивает логин и пароль пользователя.
2. Пользователь вводит данные.
3. Система проверяет введенные данные.
4. Система открывает доступ к вкладкам, соответствующим уровню доступа пользователя.

Альтернативные сценарии:

1. Логин и/или пароль введены неверно. Выводится сообщение об ошибке, после чего пользователь может повторить попытку или выйти.

2. Нет доступа к БД. Пользователь получает уведомление о неисправности.

Предусловия:

Нет.

Постусловия:

1. После успешного входа пользователь может выполнять действия в рамках своего уровня доступа.

Поддерживать работоспособность

Краткое описание:

Данный вариант использования описывает возможные действия администратора.

Основной сценарий:

Активизируется, когда администратор входит в систему.

1. Система предоставляет доступ ко всем данным.
2. Администратор может проверить работу системы, устранить неполадки, внести необходимые изменения в данные, обновить БД.
3. По требованию система составляет отчет о своей работе.

Подчиненные сценарии:

Проверить работу:

С помощью ряда проверок администратор получает отчет о нажатии на кнопку.

Устранить неполадки:

Администратор получает уведомление об ошибочной работе того или иного модуля. Он может исправить ошибки с помощью системы либо непосредственно изменяет файлы программы.

Внести изменения:

При необходимости администратор выбирает из предоставленных возможностей изменений (в основном связанные с БД) и применяет изменения либо делает изменения непосредственно в файлах программы.

Обновить БД:

Администратор выбирает из предоставленных возможностей команды импорта/экспорта, выбирает названия БД и таблиц и применяет изменения либо взаимодействует с сервером и локальными БД напрямую.

Альтернативные сценарии:

1. Нет доступа к БД. Администратор получает уведомление о неисправности.

Предусловия:

1. Пользователь должен войти в систему.
2. Пользователь должен иметь уровень доступа 1 (администратор).

Постусловия:

Нет.

Логическое представление.

Модель классов (Class Diagrams) – логическая модель базовой структуры системы, отражает статическую структуру системы и связи между ее элементами. Модели классов соответствуют статическому виду системы. На модели классов подсистемы управления персоналом информационной системы Тамбовского завода «Электроприбор» показаны классы, интерфейсы, объекты и кооперации, а также их отношения. При объектно ориентированном моделировании систем этот тип моделей используют чаще всего. Модель классов разрабатываемой подсистемы управления персоналом представлена на рисунках далее.

Описание классов:

Access – управление доступом – отвечает за распределение доступа по хранящимся в базе данных спецификаторам.

Authentication – аутентификация – проверяет вводимые пользователем данные на правильность и разрешает или запрещает ему доступ к системе.

Checkpoint – электронная проходная – интерфейс между программой и электронной проходной. Позволяет заносить данные в базу, изымать их оттуда и предоставлять пользователю.

Converter – конвертер – позволяет переводить данные в единый числовой формат, принятый на предприятии.

Database – база данных – осуществляет взаимодействие с базами данных. Благодаря реализации с помощью Qt возможностей для подключения разных типов стало больше, чем предполагалось изначально.

Exception – исключения – класс содержит все исключения, которые могут возникнуть во время работы программы.

Login – регистрация – класс, отвечающий за отображения окна ввода логина/пароля, обработку данных и непосредственно вход в систему.

Payment – платежная система – отвечает за расчет заработной платы и стажа. Не содержит в себе сложных финансовых расчетов, при необходимости может быть расширен специалистами в данной области.

Search – поиск – предоставляет возможность поиска по сотрудникам.

Test – проверка – выполняет проверку правильности данных, введенных в табель.

Timecard – основная графическая оболочка – содержит все графические элементы и взаимодействия с другими классами.

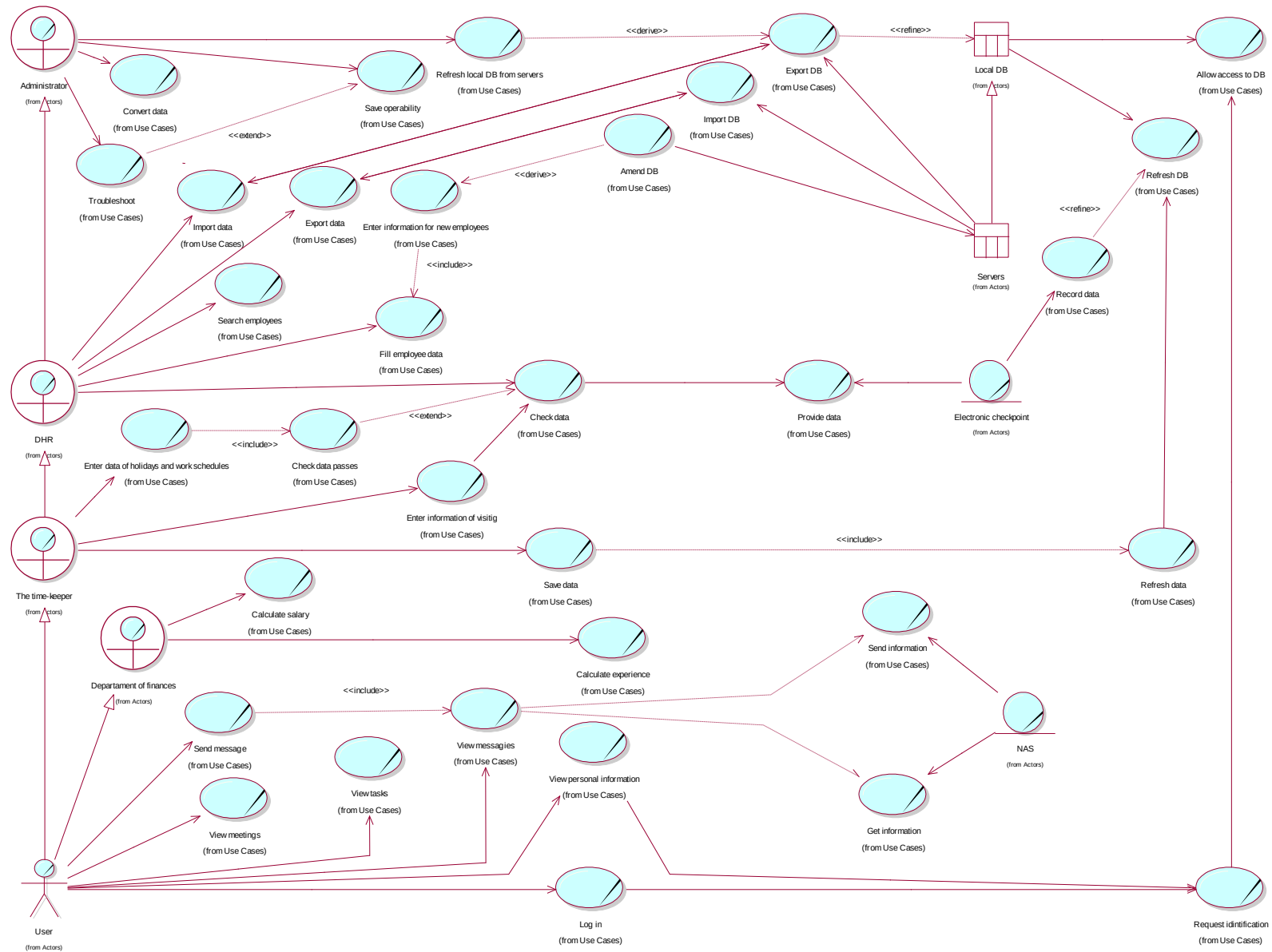
Timetable – табель – отвечает за взаимодействие с таблицей, в которую внесены данные по табельному учету. Выводит данные для пользователей.

Физическое представление.

Модели размещения (Deployment Diagrams) – модель физической архитектуры системы, отображает технологическую конфигурацию системы. На модели размещения подсистемы управления персоналом информационной системы Тамбовского завода «Электроприбор» представлена конфигурация обрабатывающих узлов системы и размещенных в них компонентов. Модель размещения разрабатываемой подсистемы управления персоналом представлена далее.

В данной модели, по сравнению с существующей, изменен алгоритм функционирования баз данных и серверов. Для рабочих мест предусмотрены локальные базы, так как подключение каждого рабочего места к серверу приведет к потере эффективности. Вместо этого данные после изъятия из локальных БД помещаются в общую базу, откуда к ним имеют доступ отдел работы с персоналом и отдел финансов. После завершения расчетного периода данные помещаются в сервер для хранения. Таким образом, основное взаимодействие происходит между менее загруженными компонентами, тем самым достигается большее быстродействие.

Диаграмма прецедентов



Общая диаграмма классов

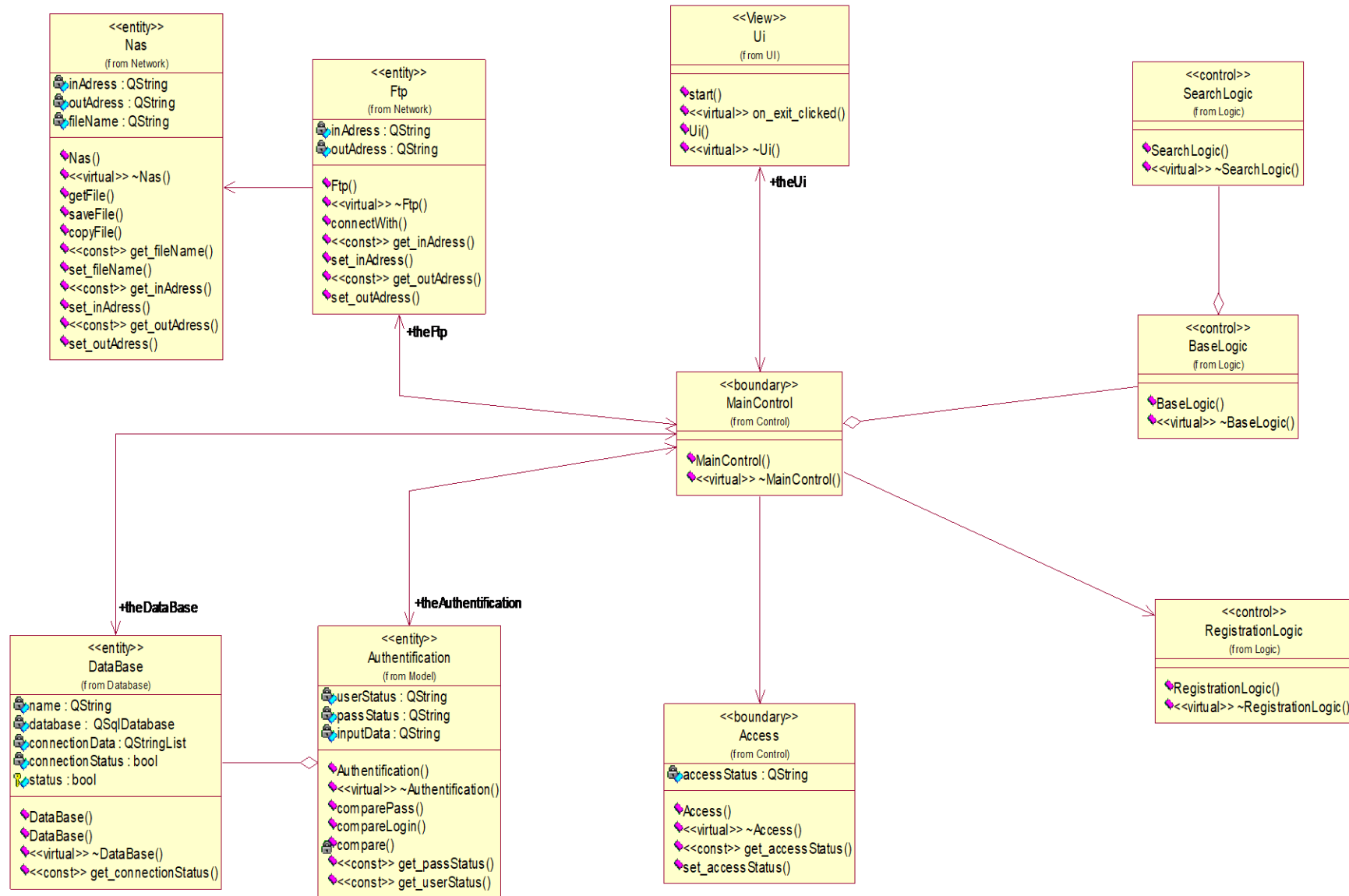


Диаграмма классов пакета Logic

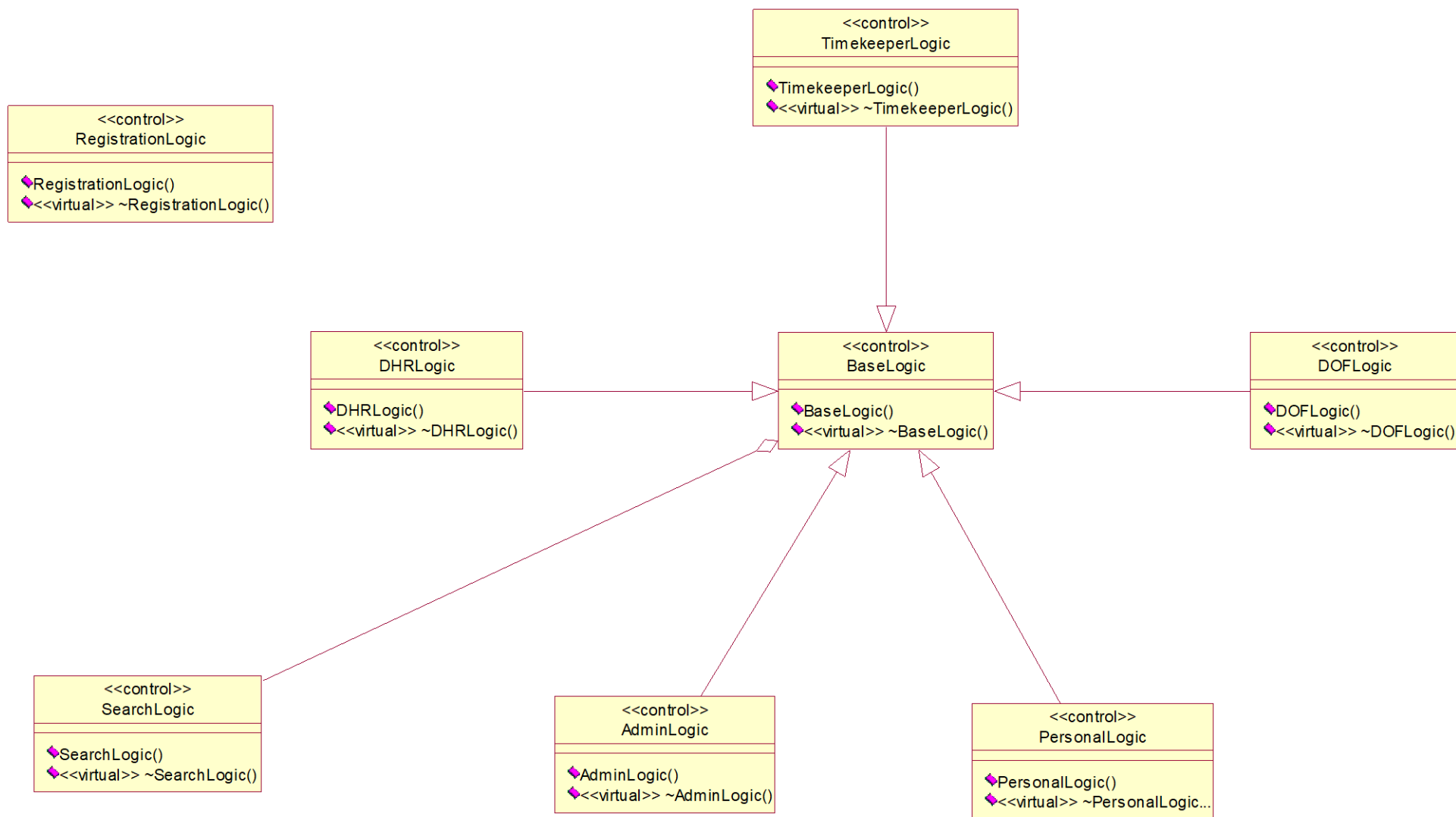


Диаграмма классов пакета controller

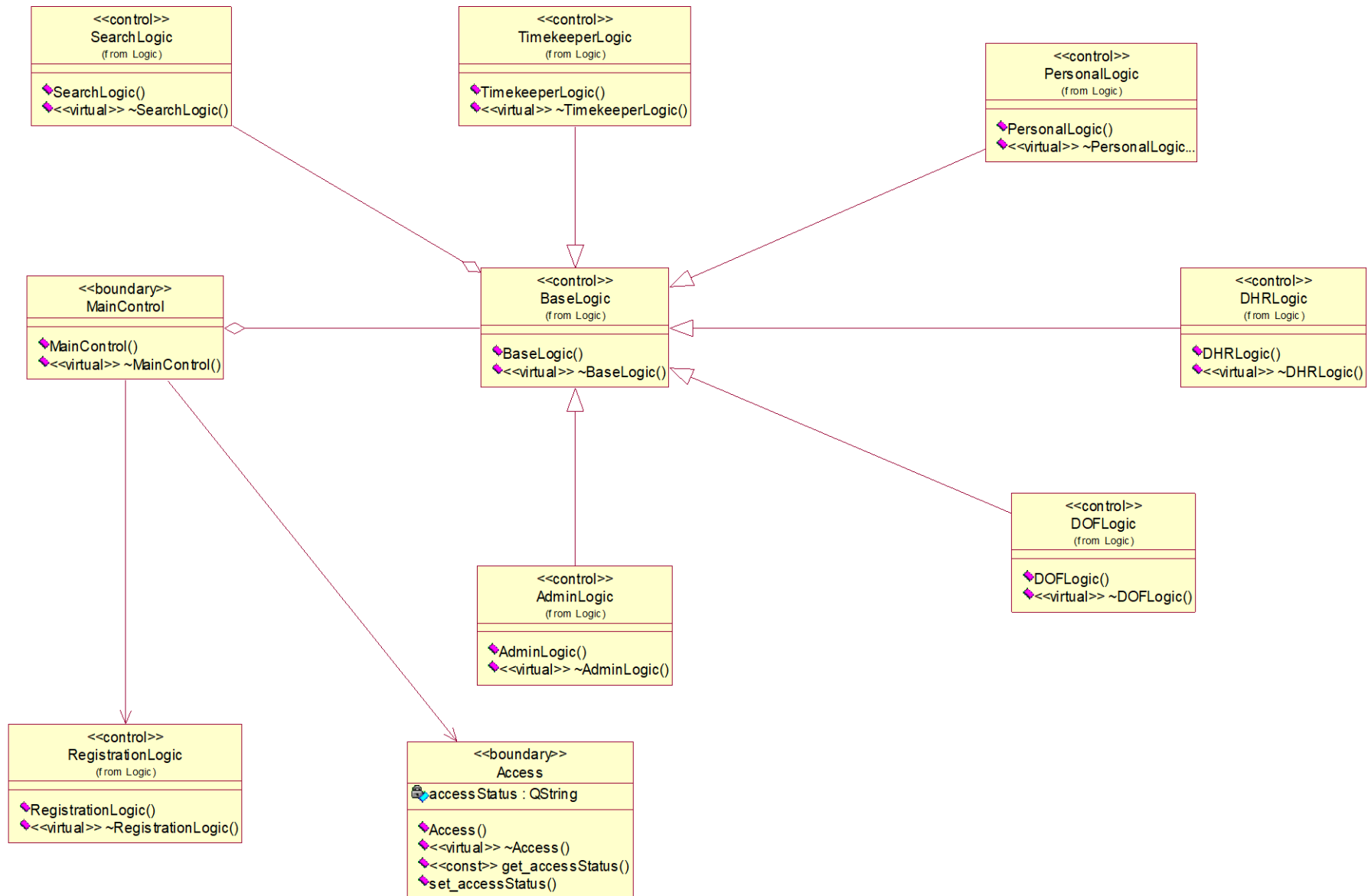


Диаграмма классов пакета database

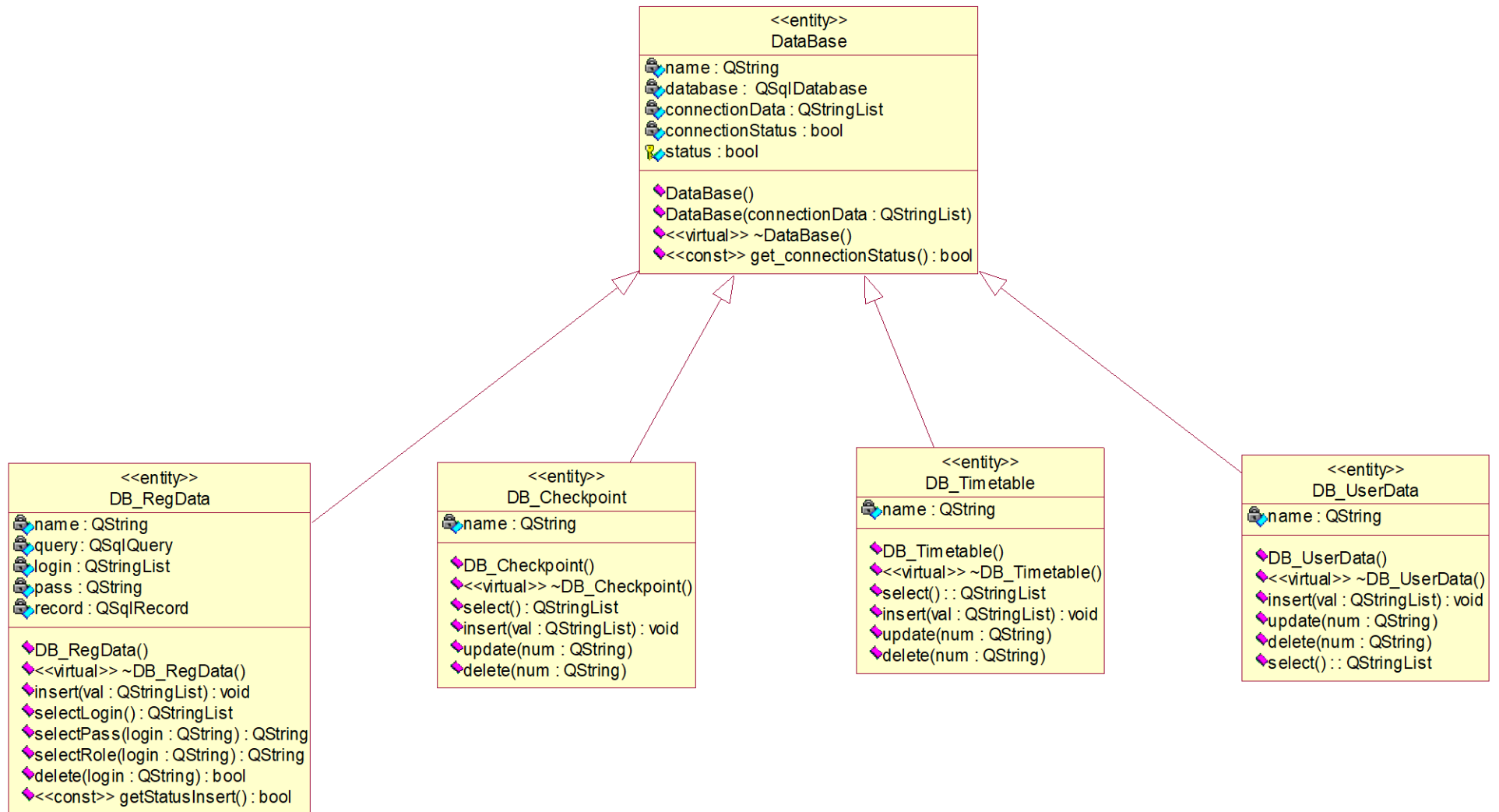


Диаграмма классов пакета Network

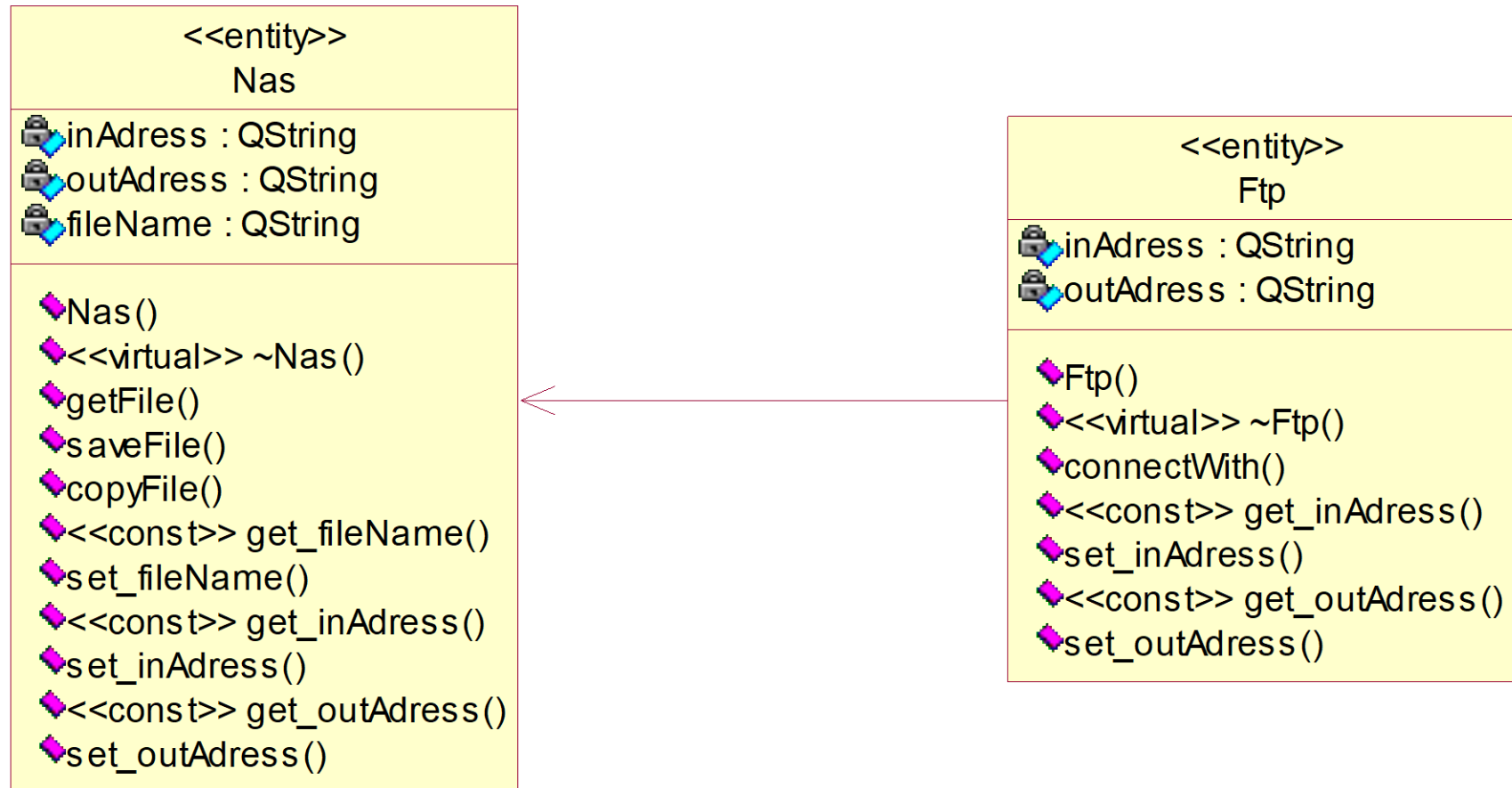


Диаграмма классов пакета model

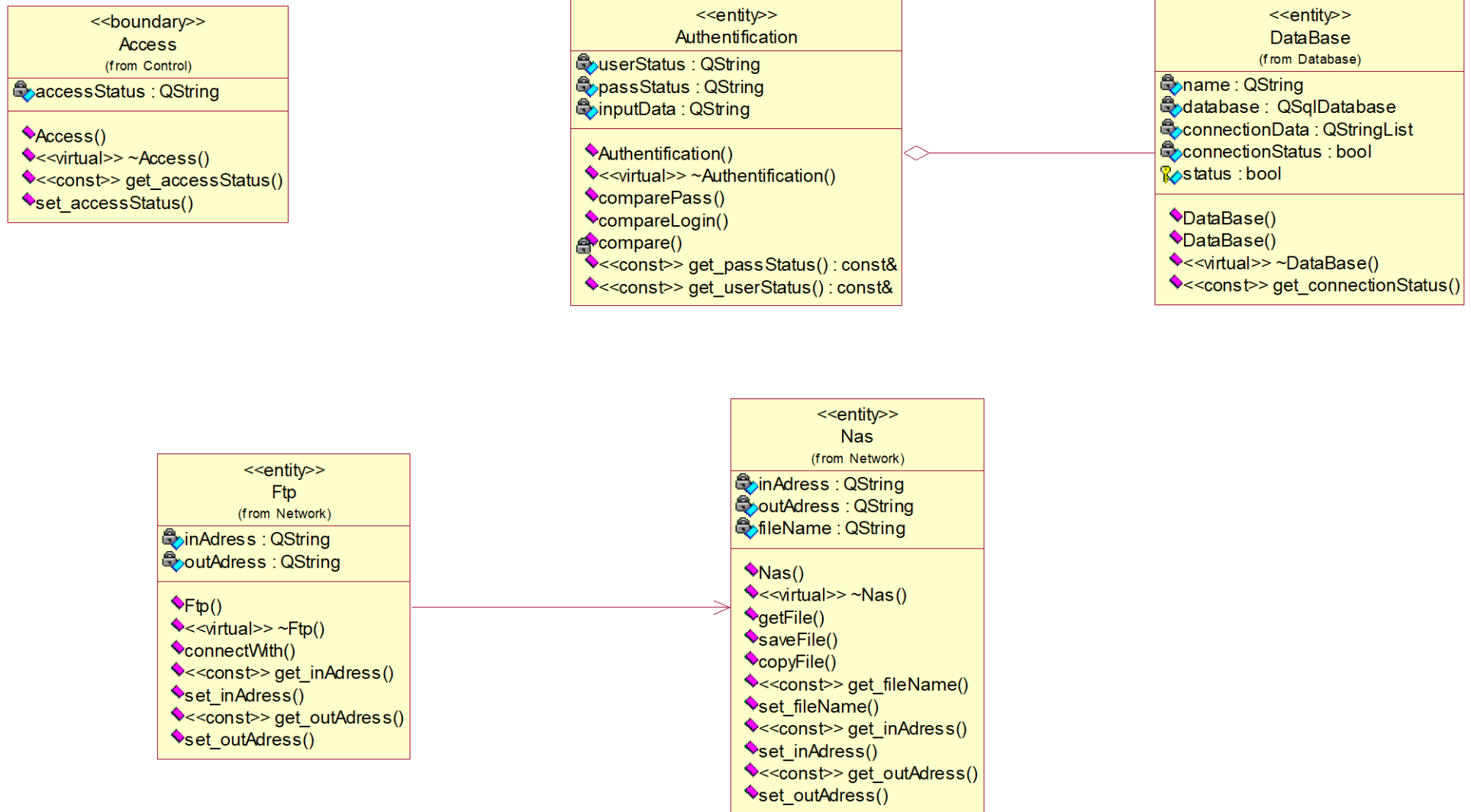


Диаграмма классов пакета Ui

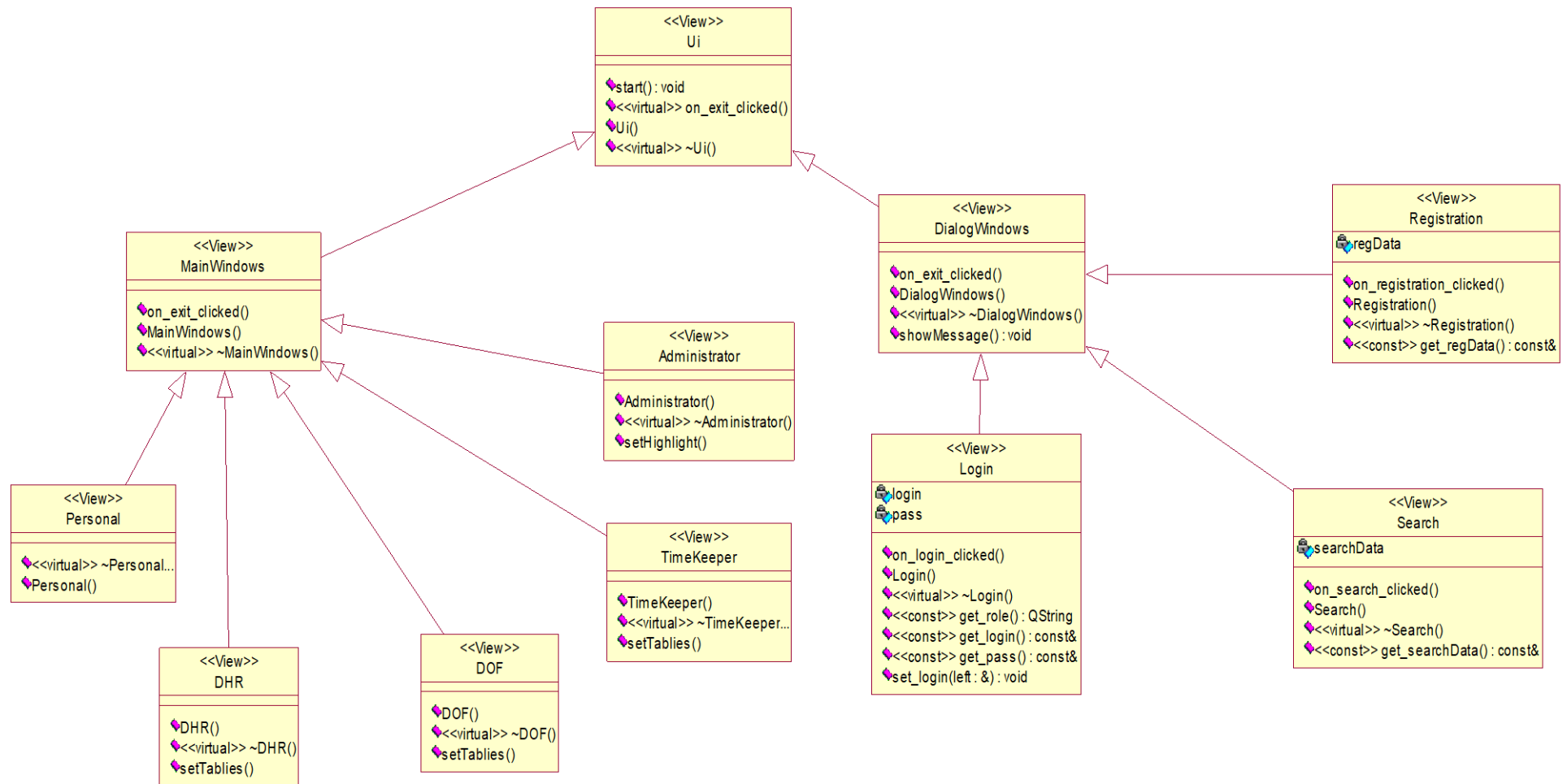
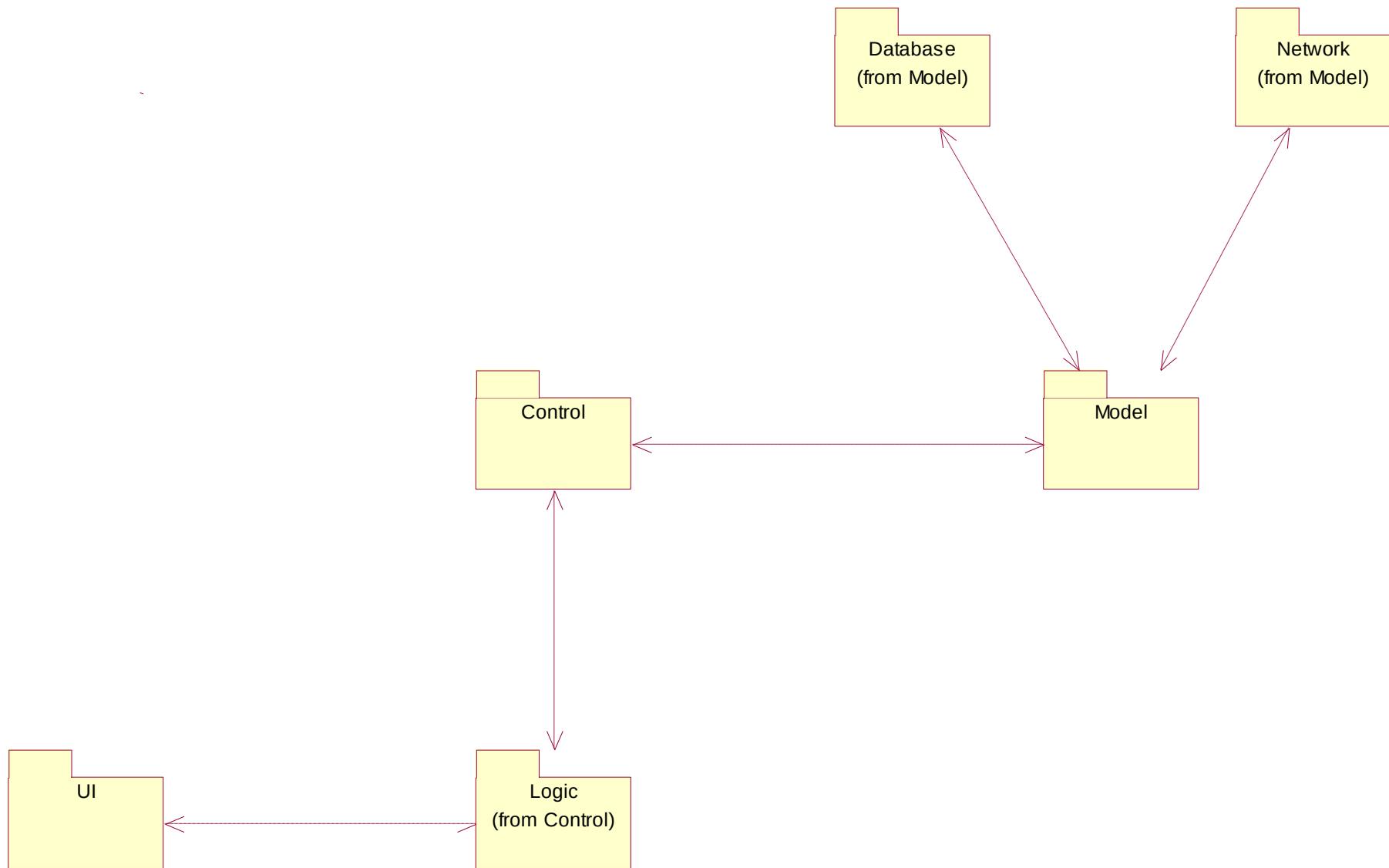


Диаграмма пакетов



Диаграммы последовательности admin

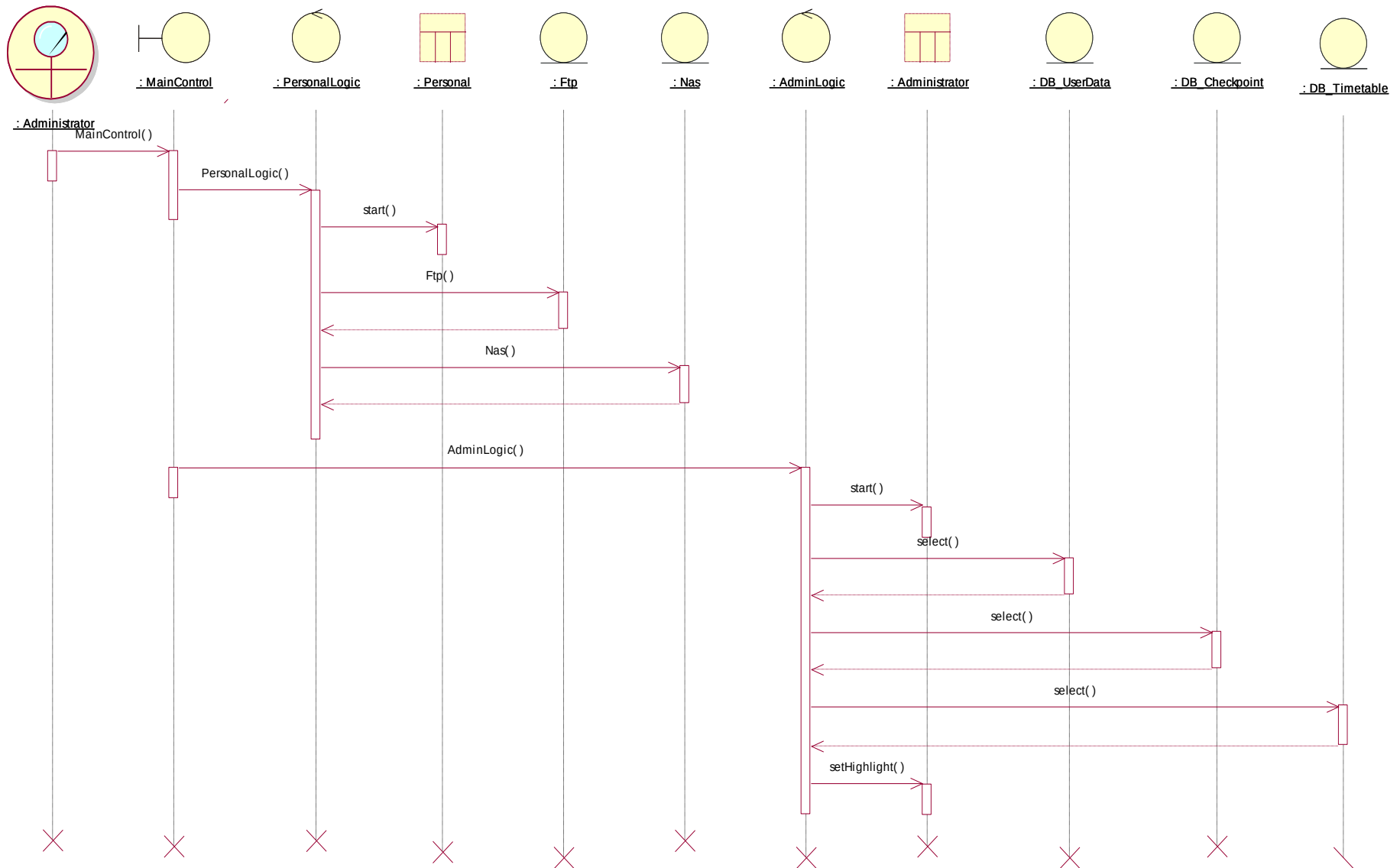


Диаграмма последовательности DHR

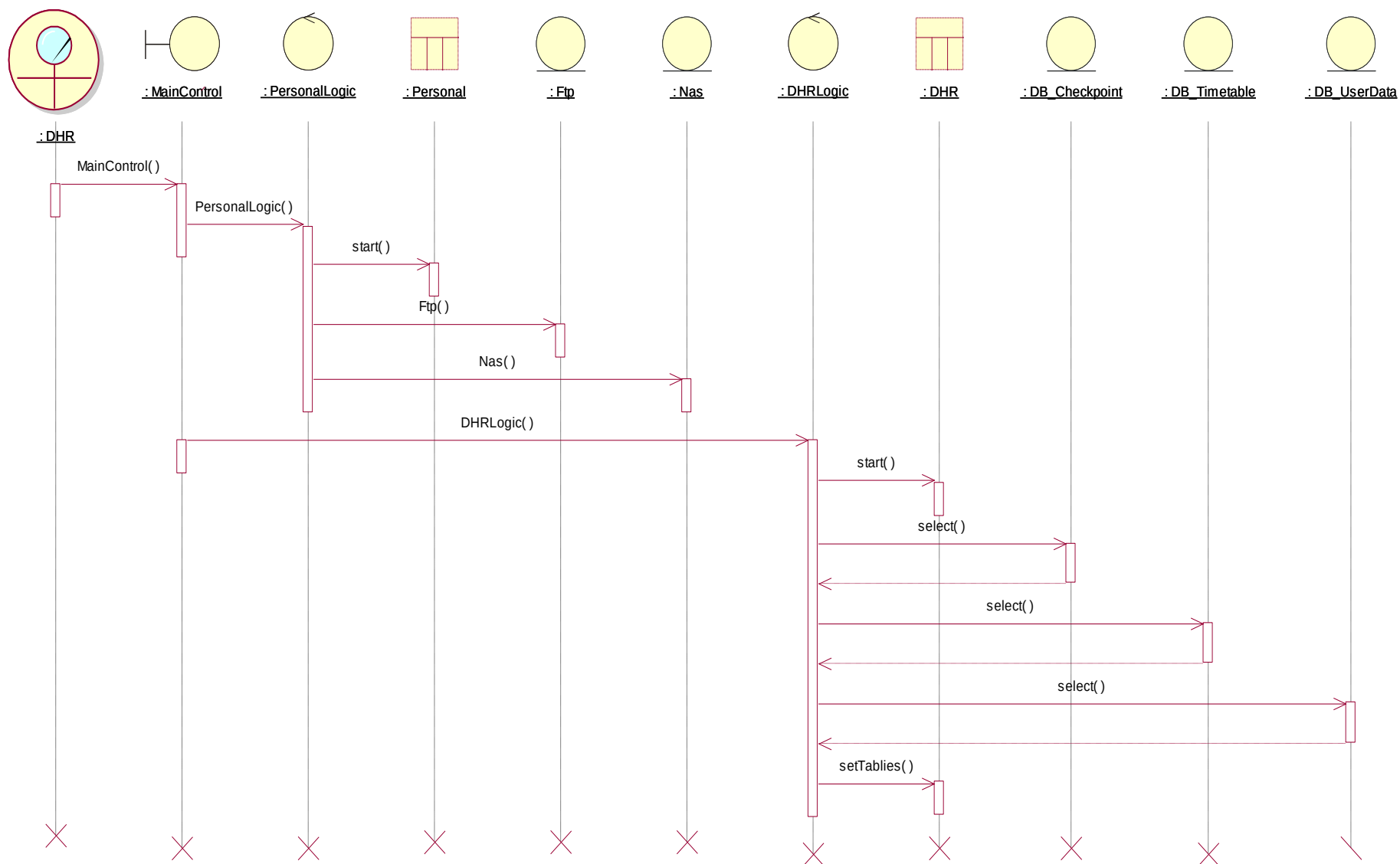


Диаграмма последовательности DOF

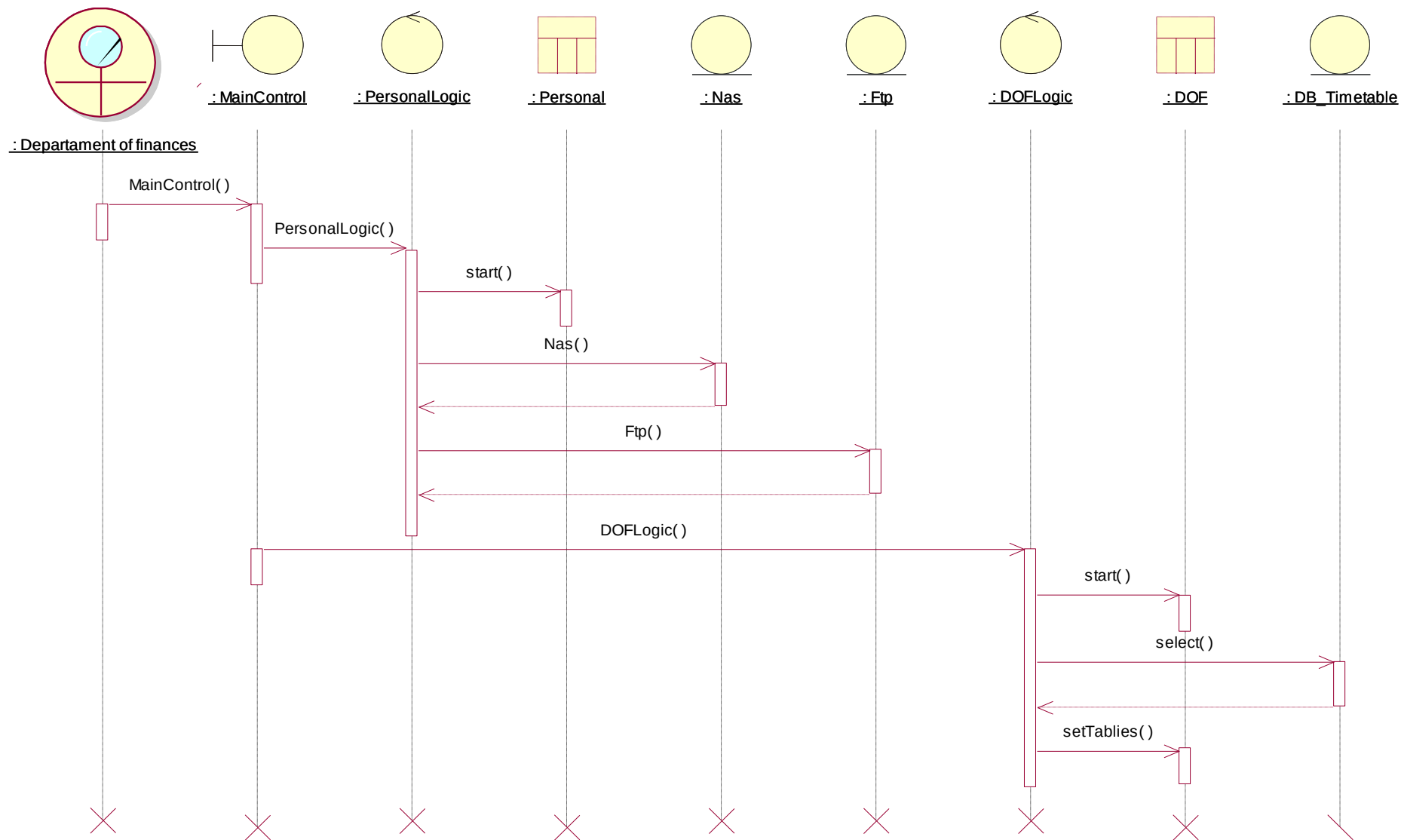


Диаграмма последовательности admin

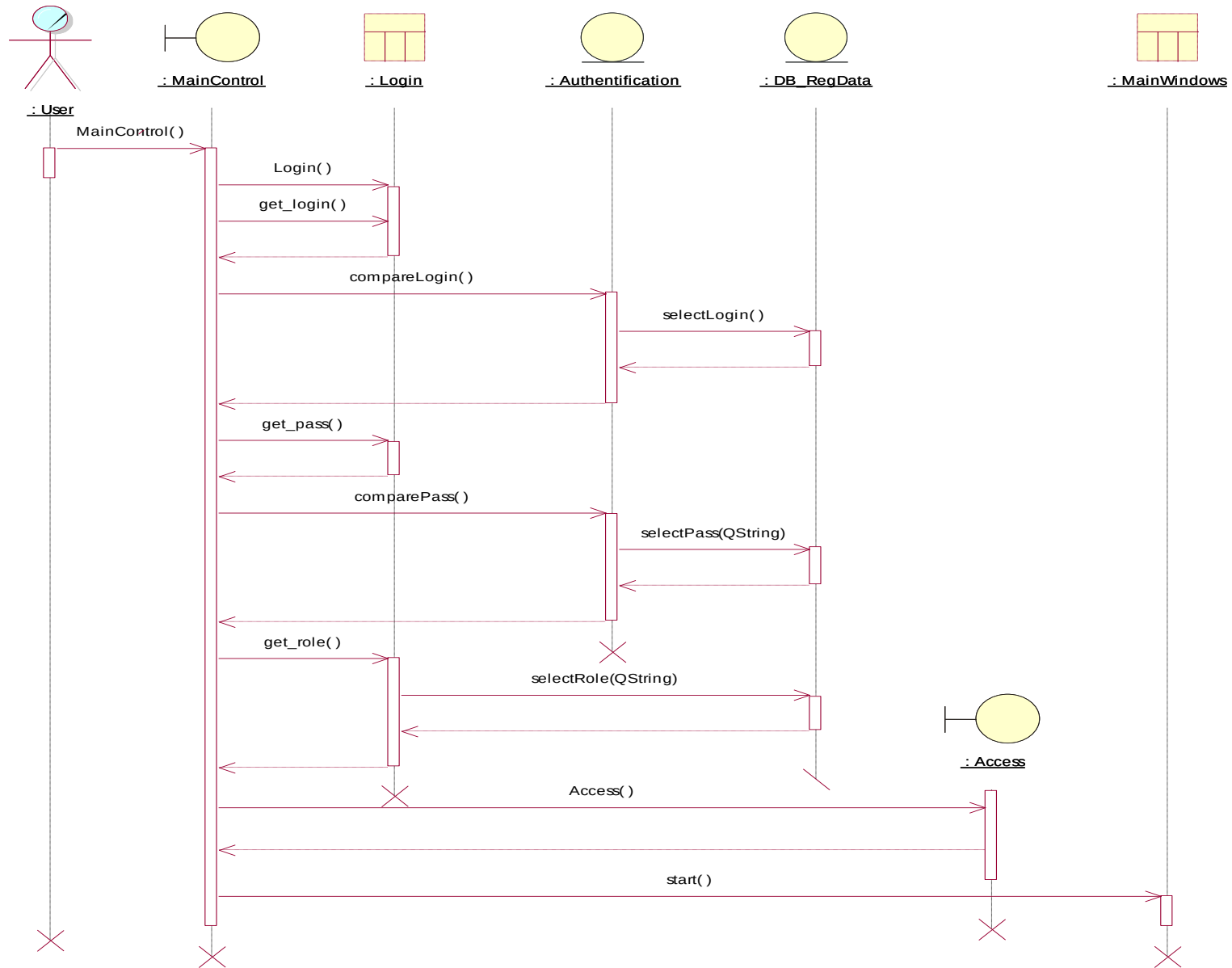


Диаграмма последовательности registration

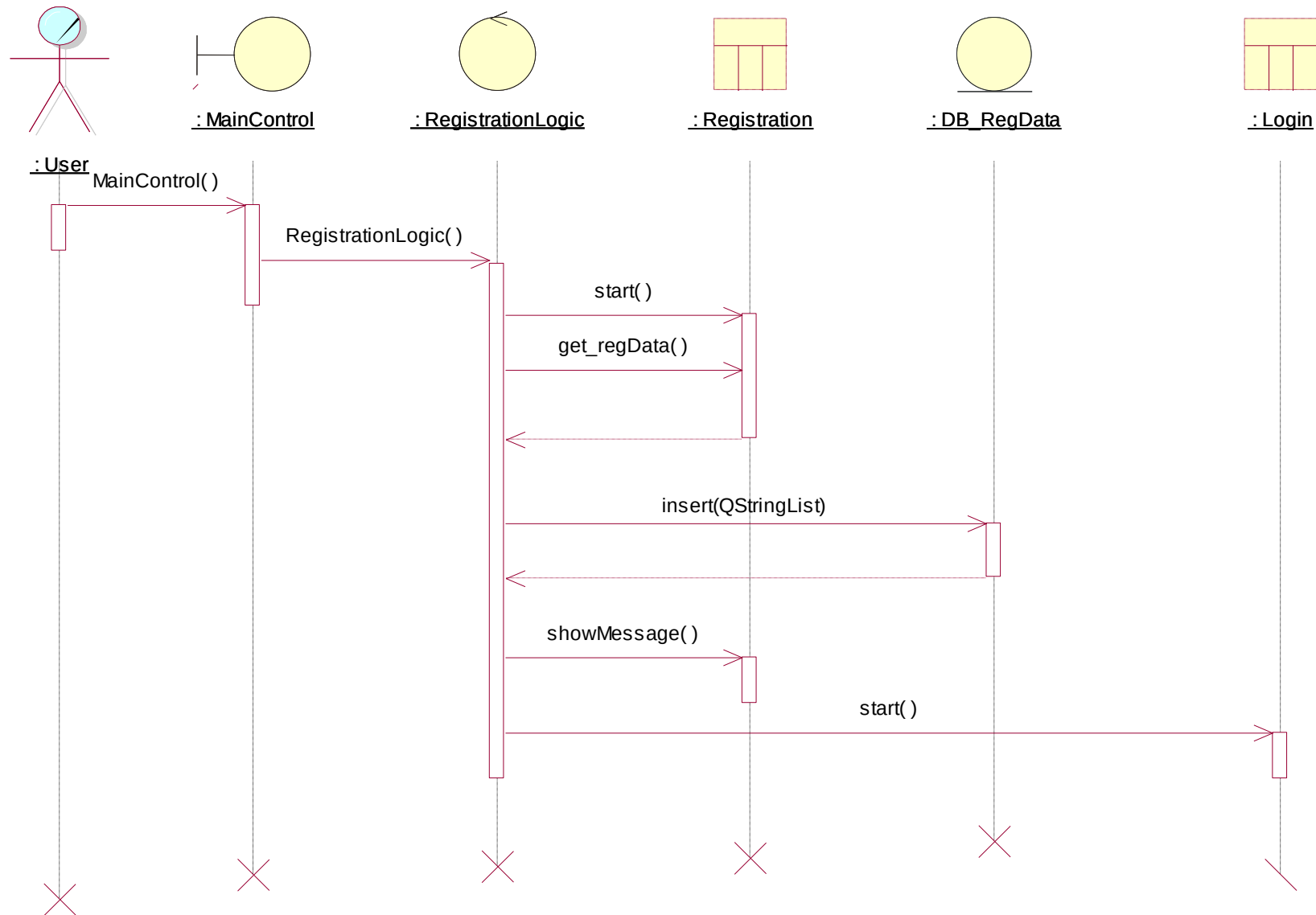


Диаграмма последовательности search

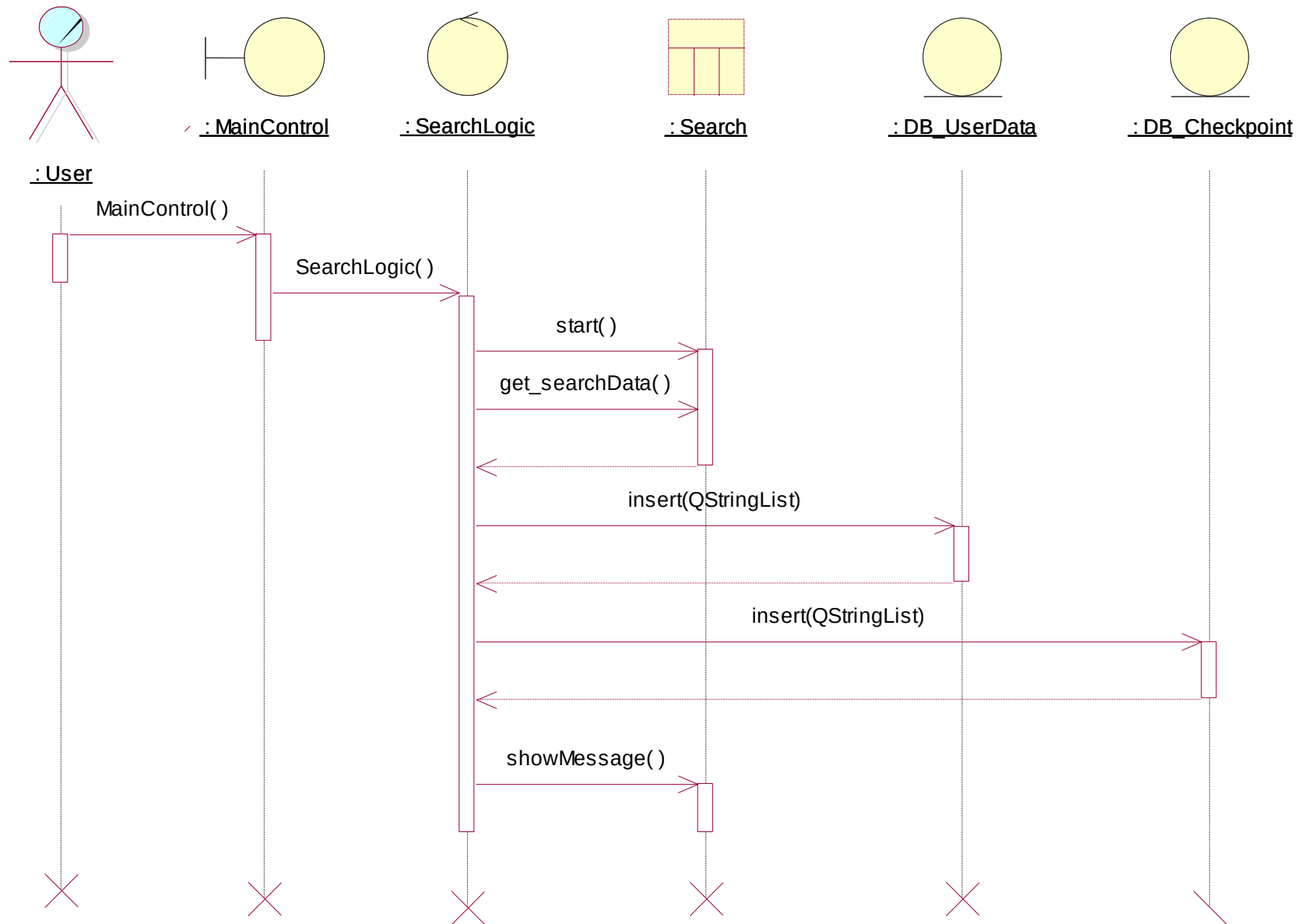


Диаграмма последовательности Timekeeper

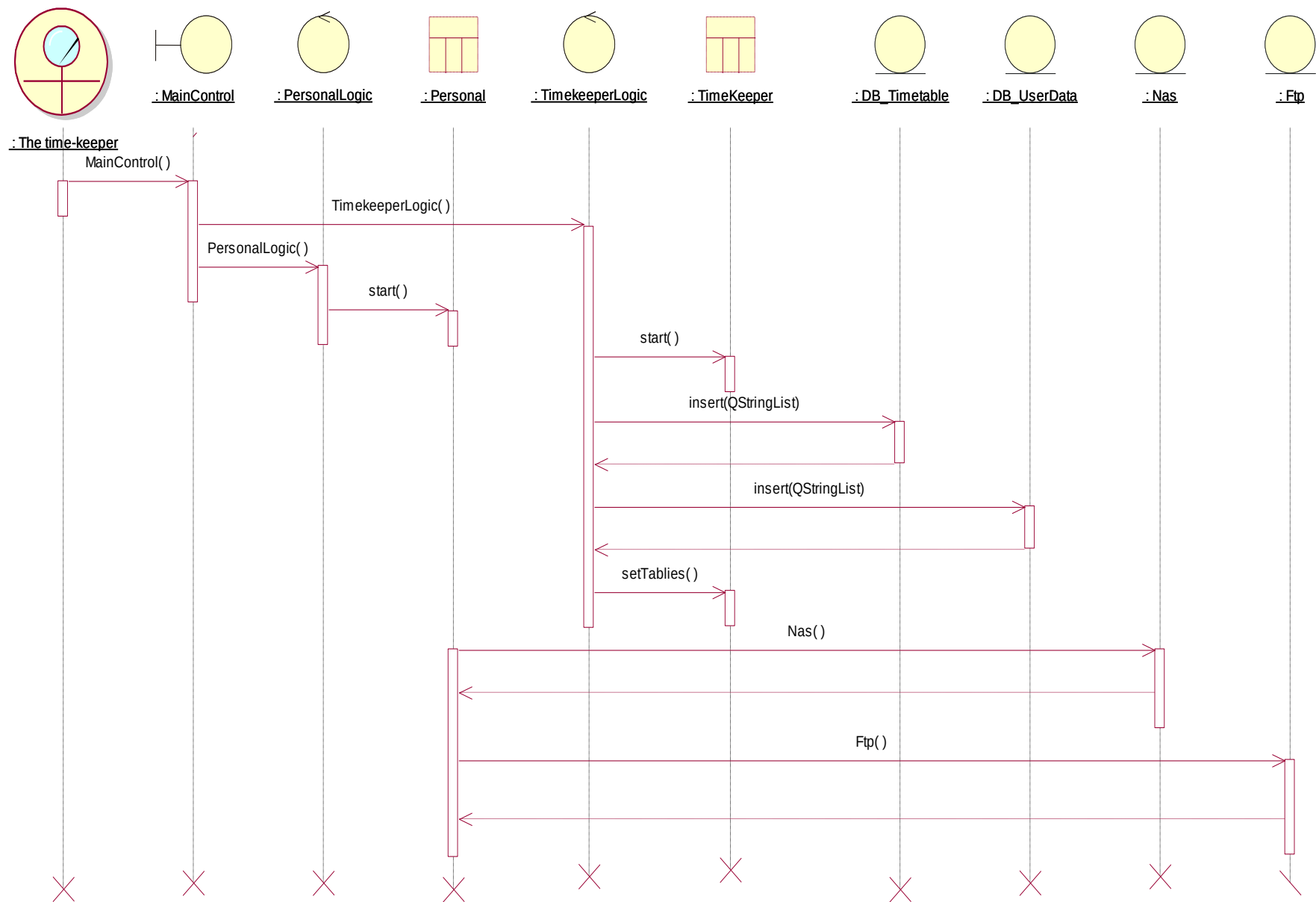


Диаграмма последовательности personal

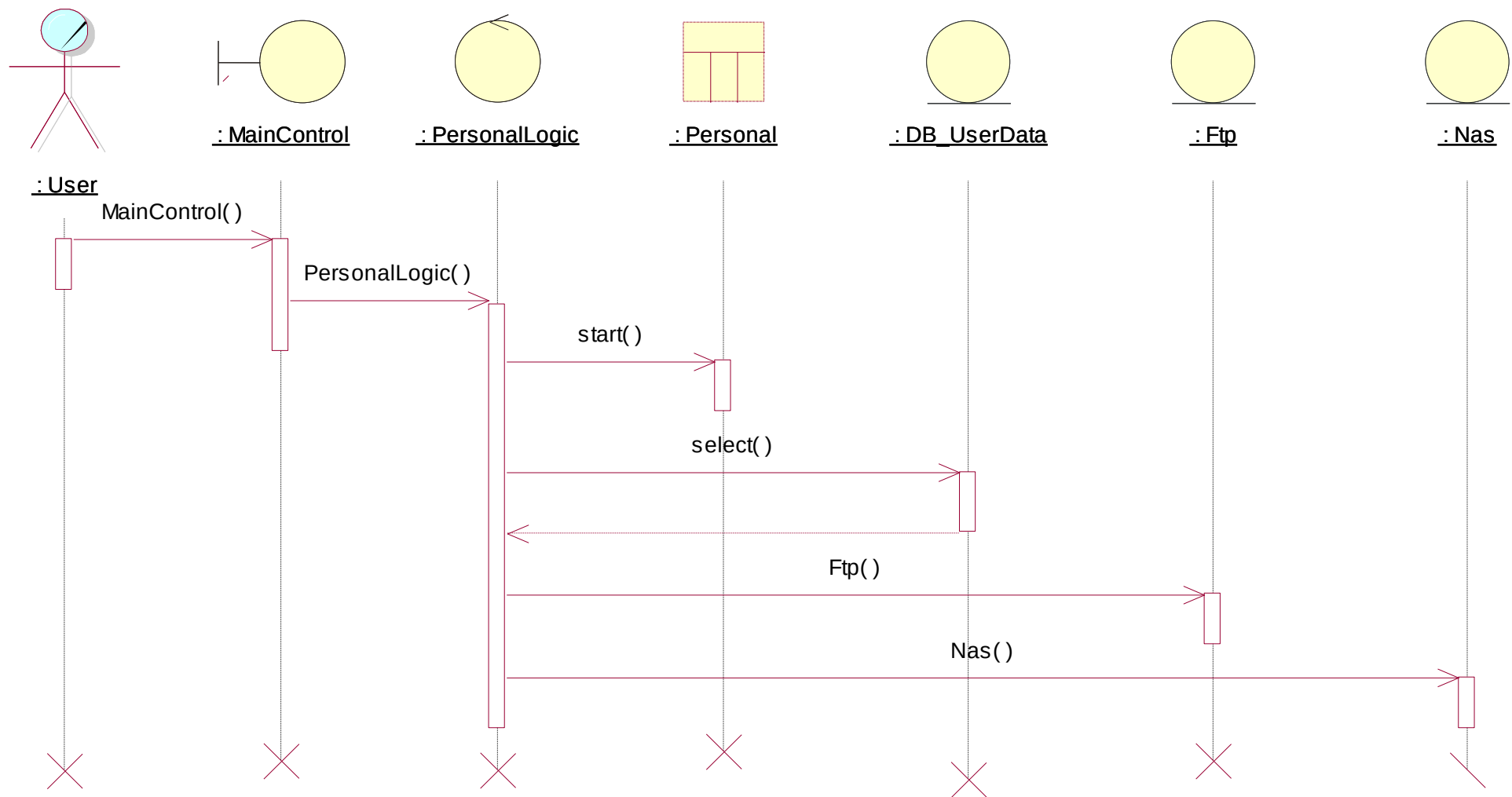


Диаграмма коопераций admin

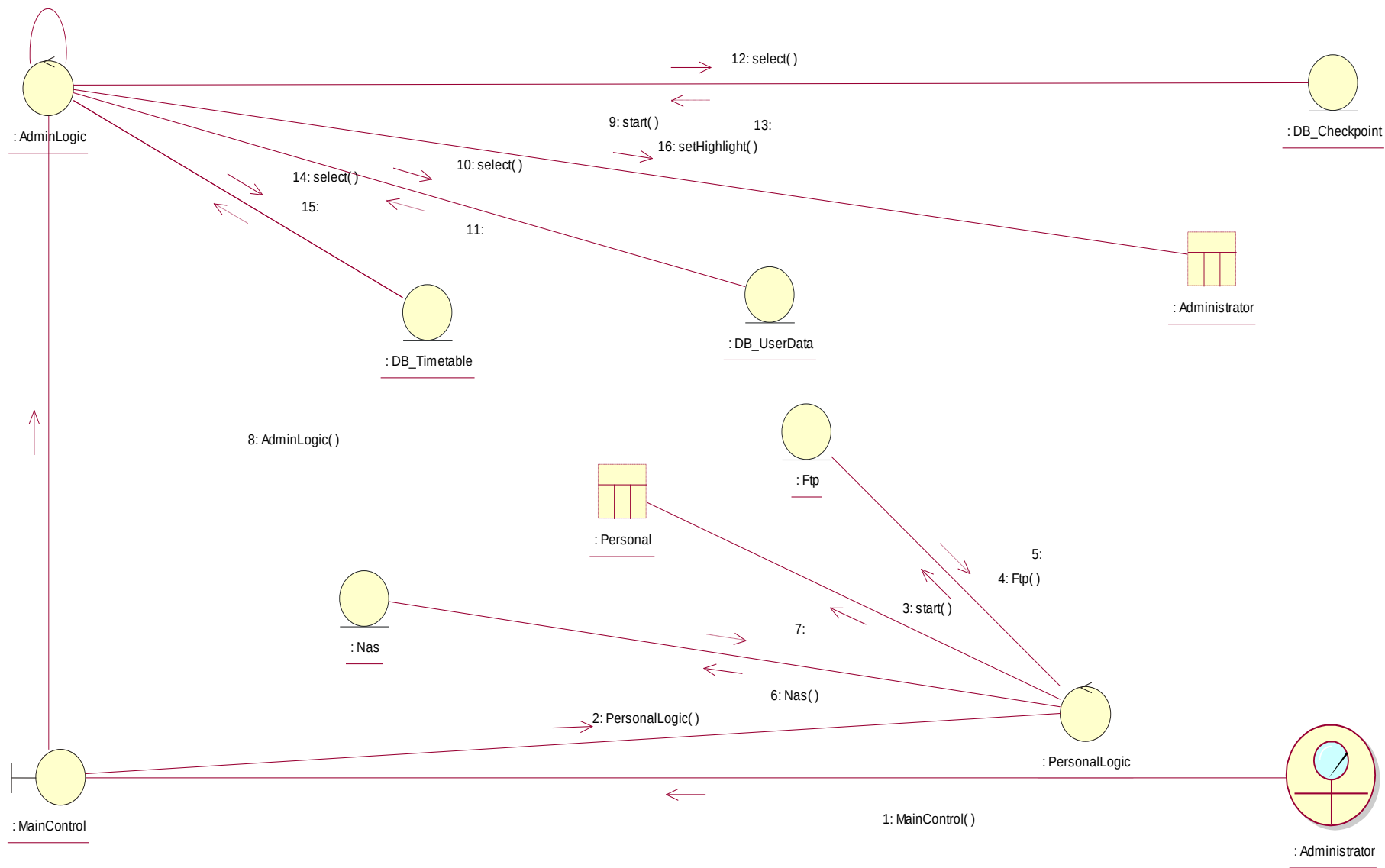


Диаграмма коопераций DHR

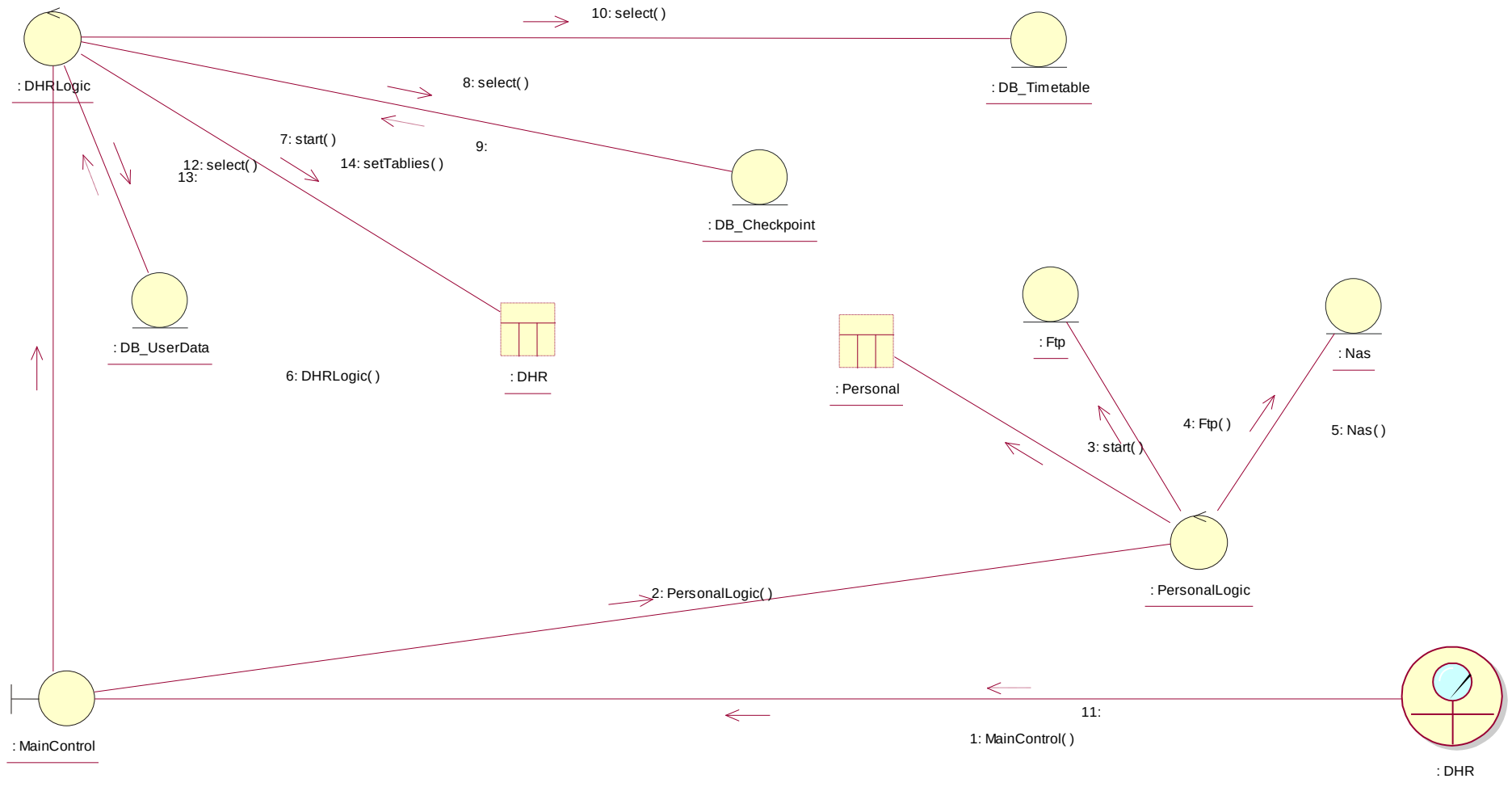


Диаграмма коопераций DOF

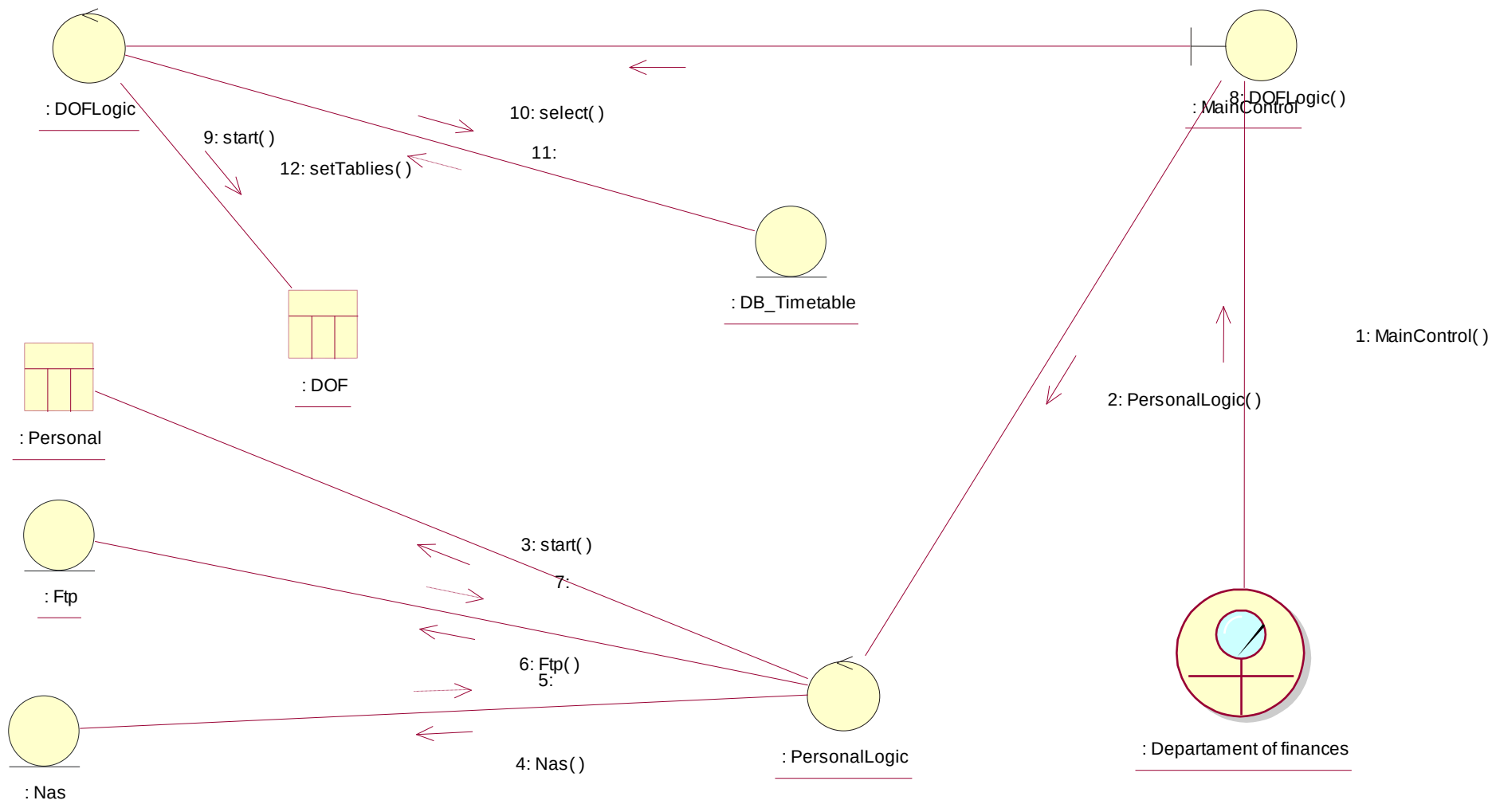


Диаграмма коопераций admin

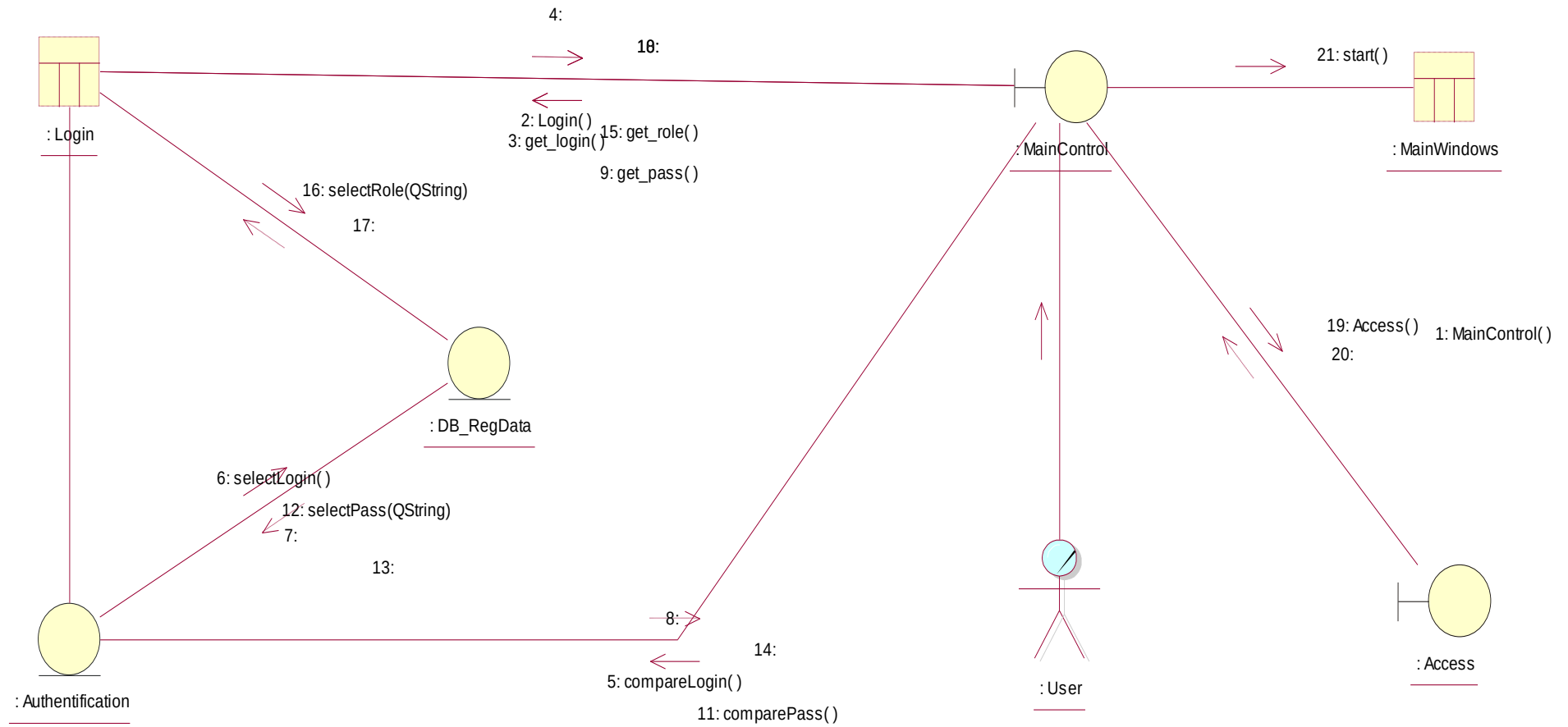


Диаграмма коопераций registration

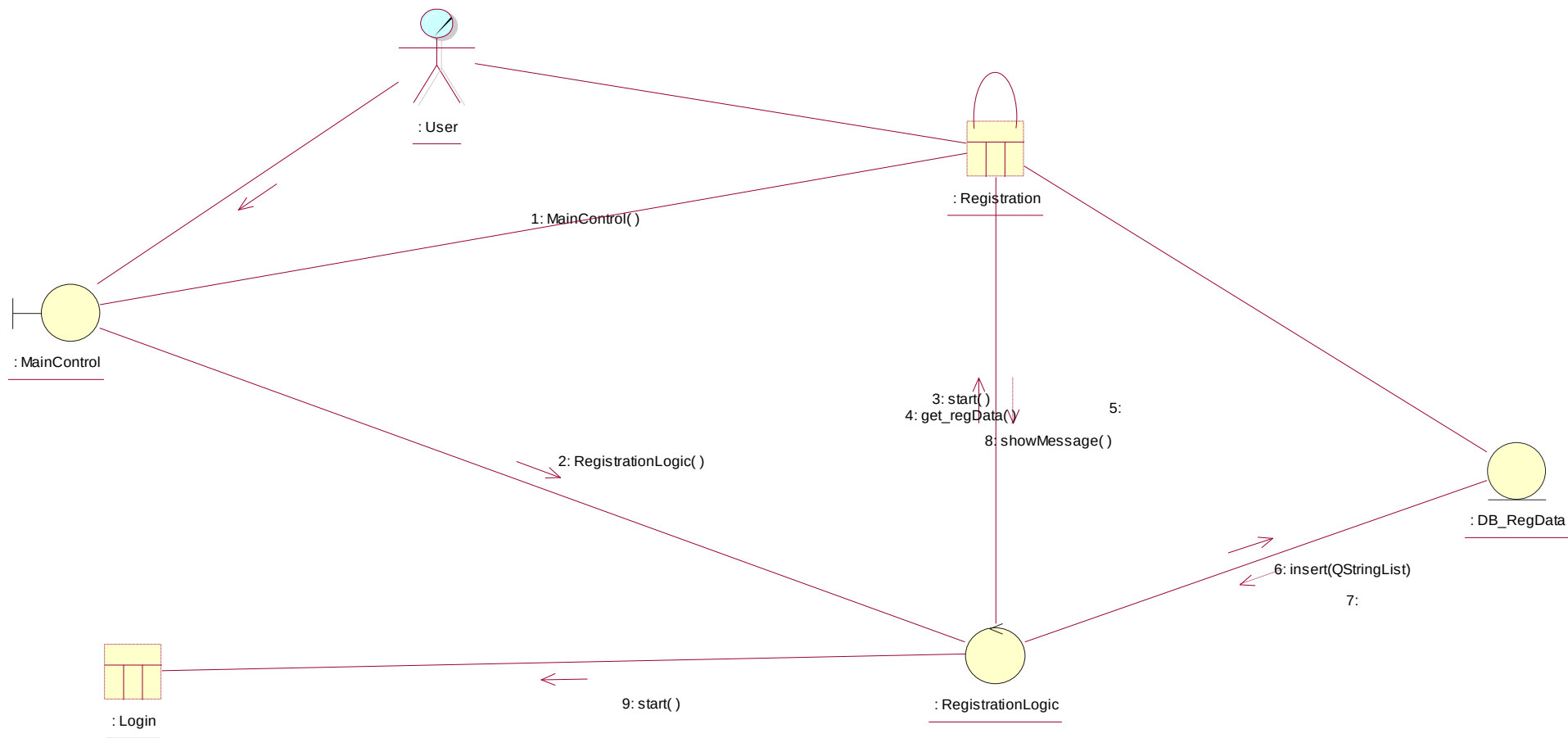


Диаграмма коопераций search

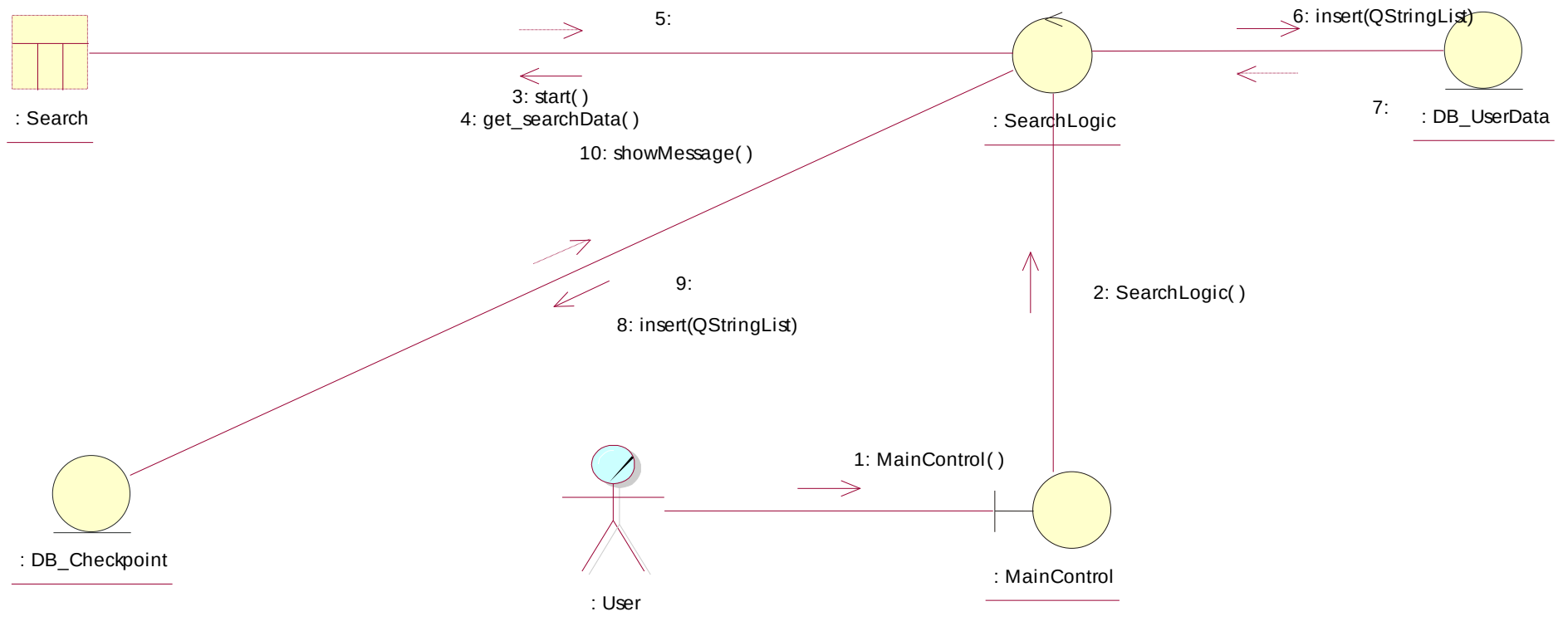


Диаграмма коопераций Timekeeper

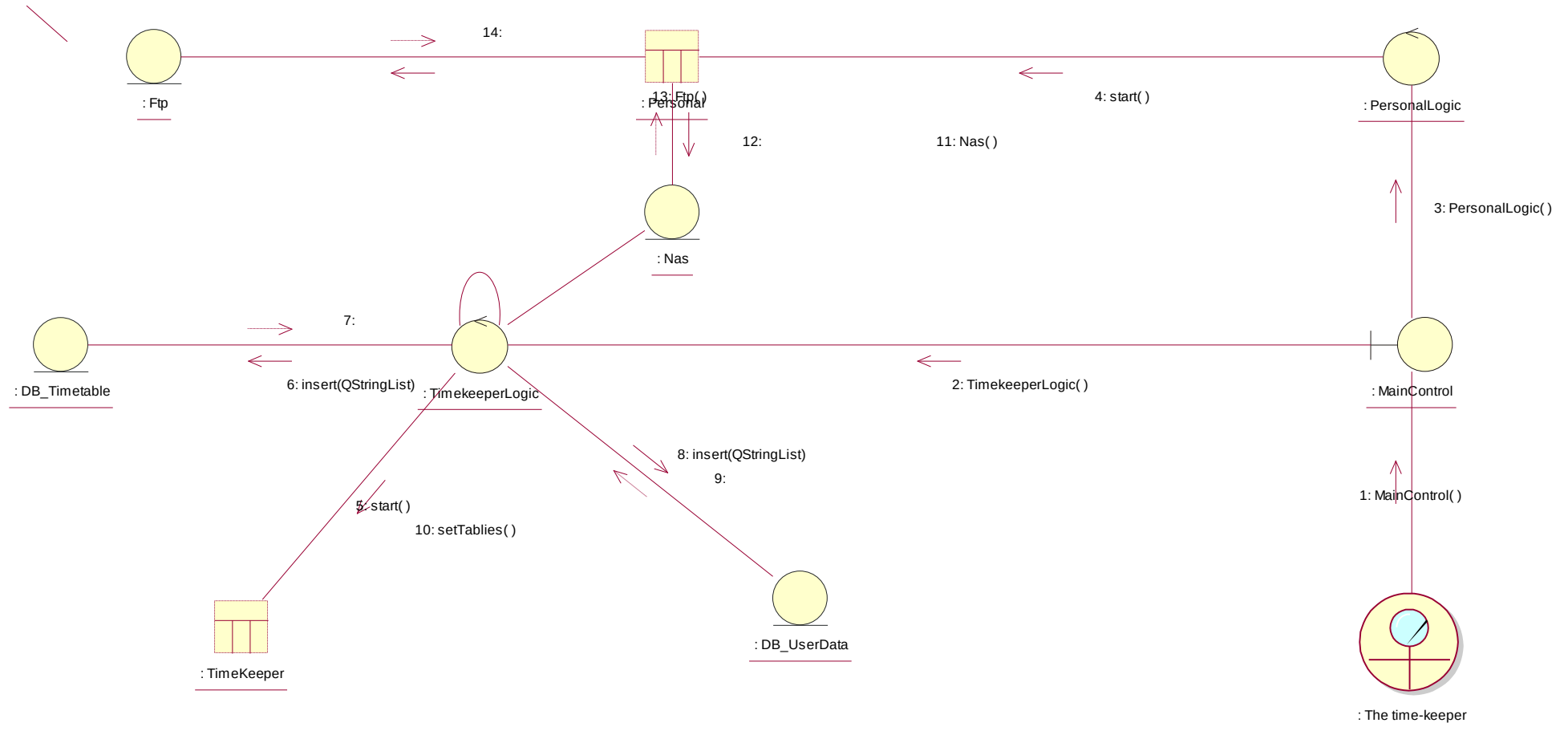


Диаграмма коопераций personal

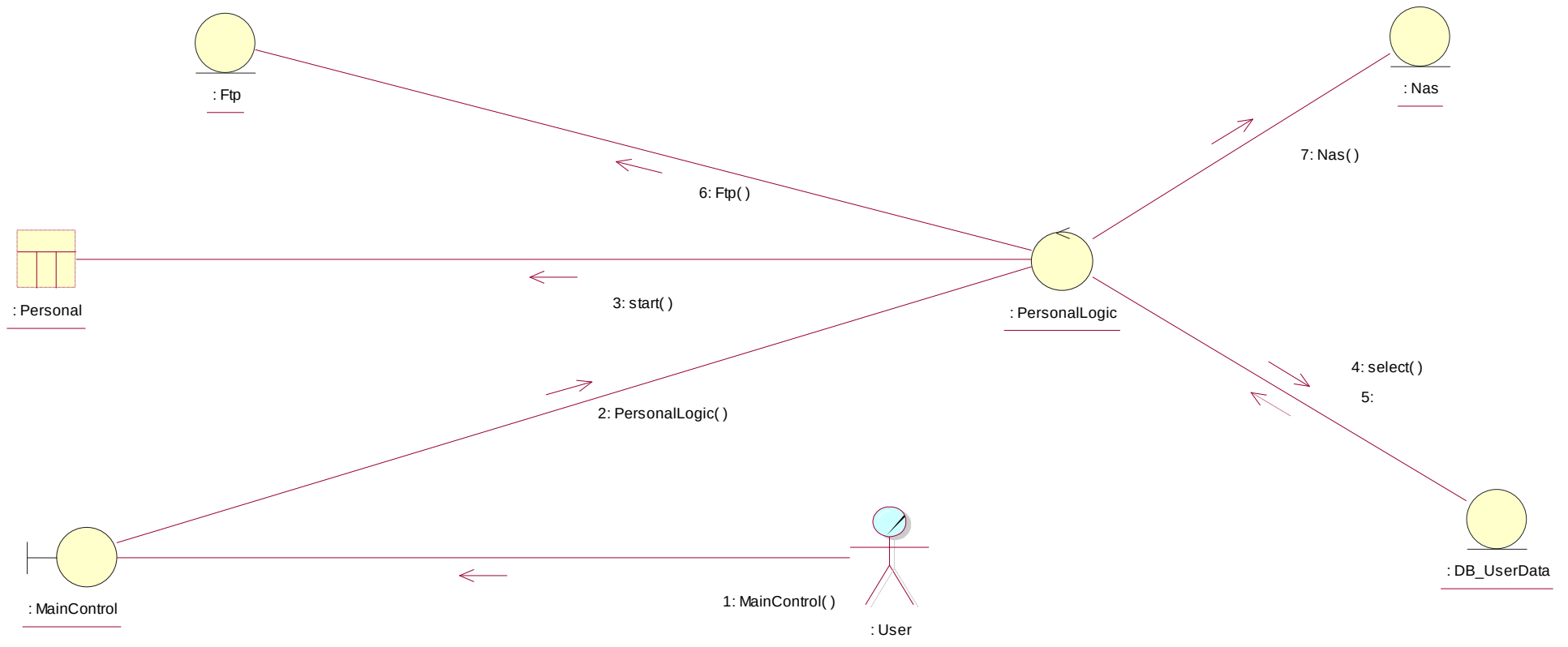


Диаграмма реализаций вариантов использования

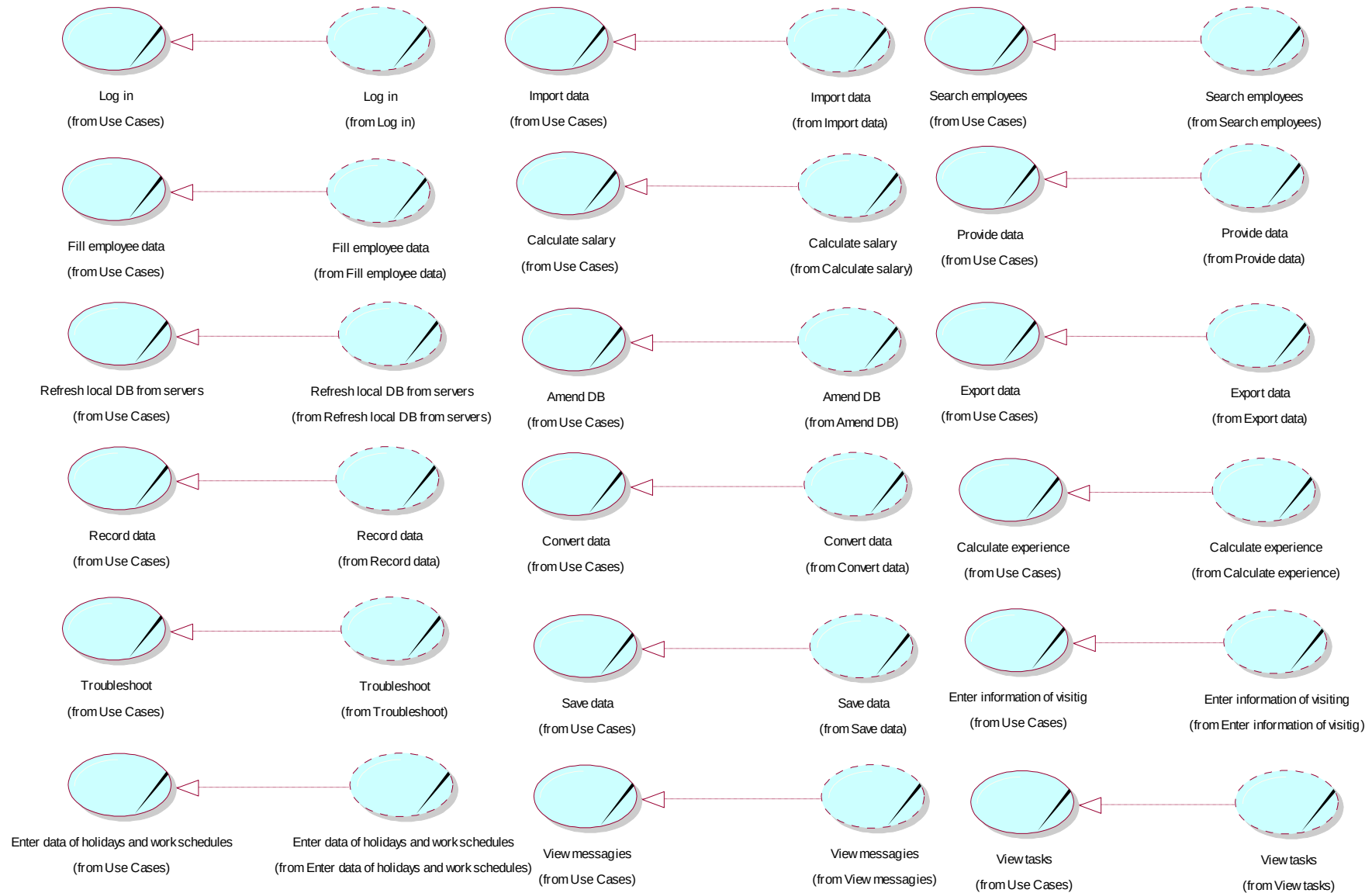
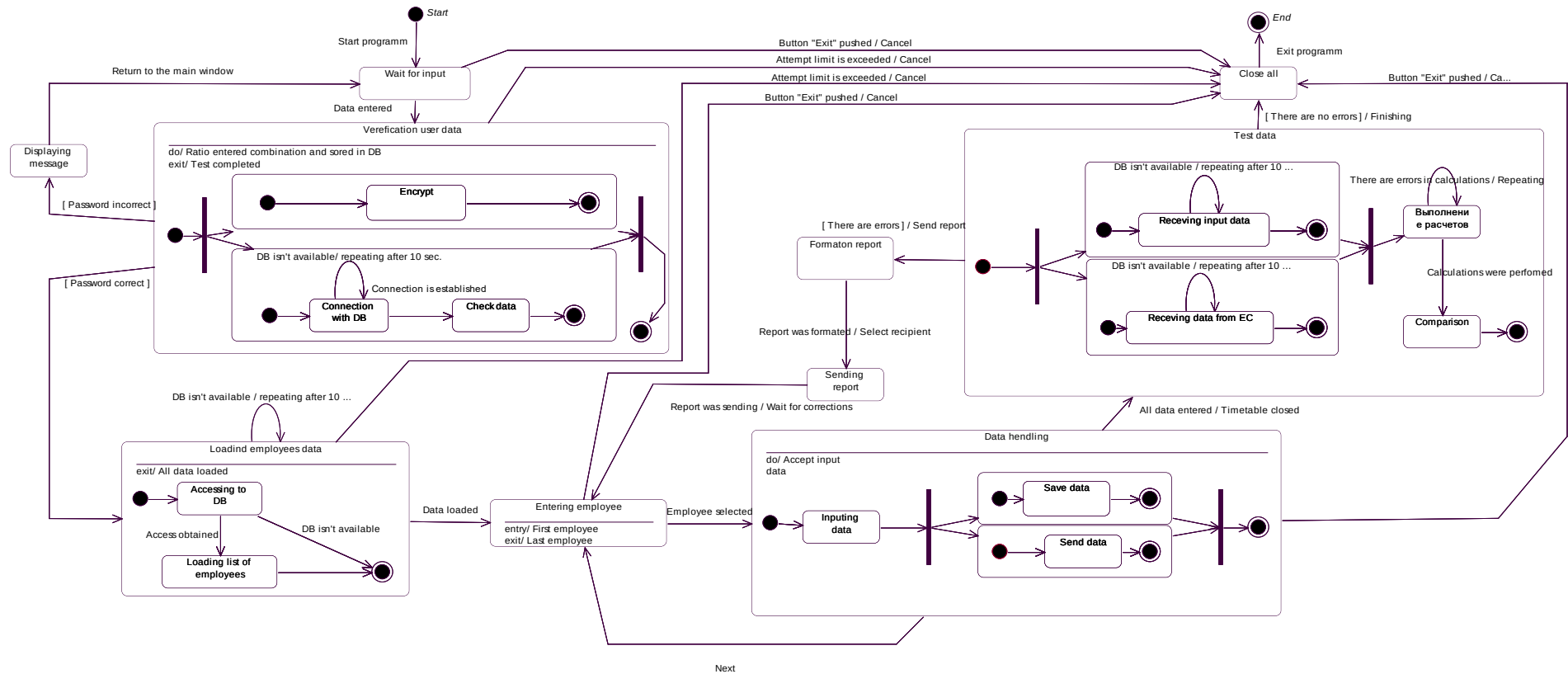


Диаграмма состояний



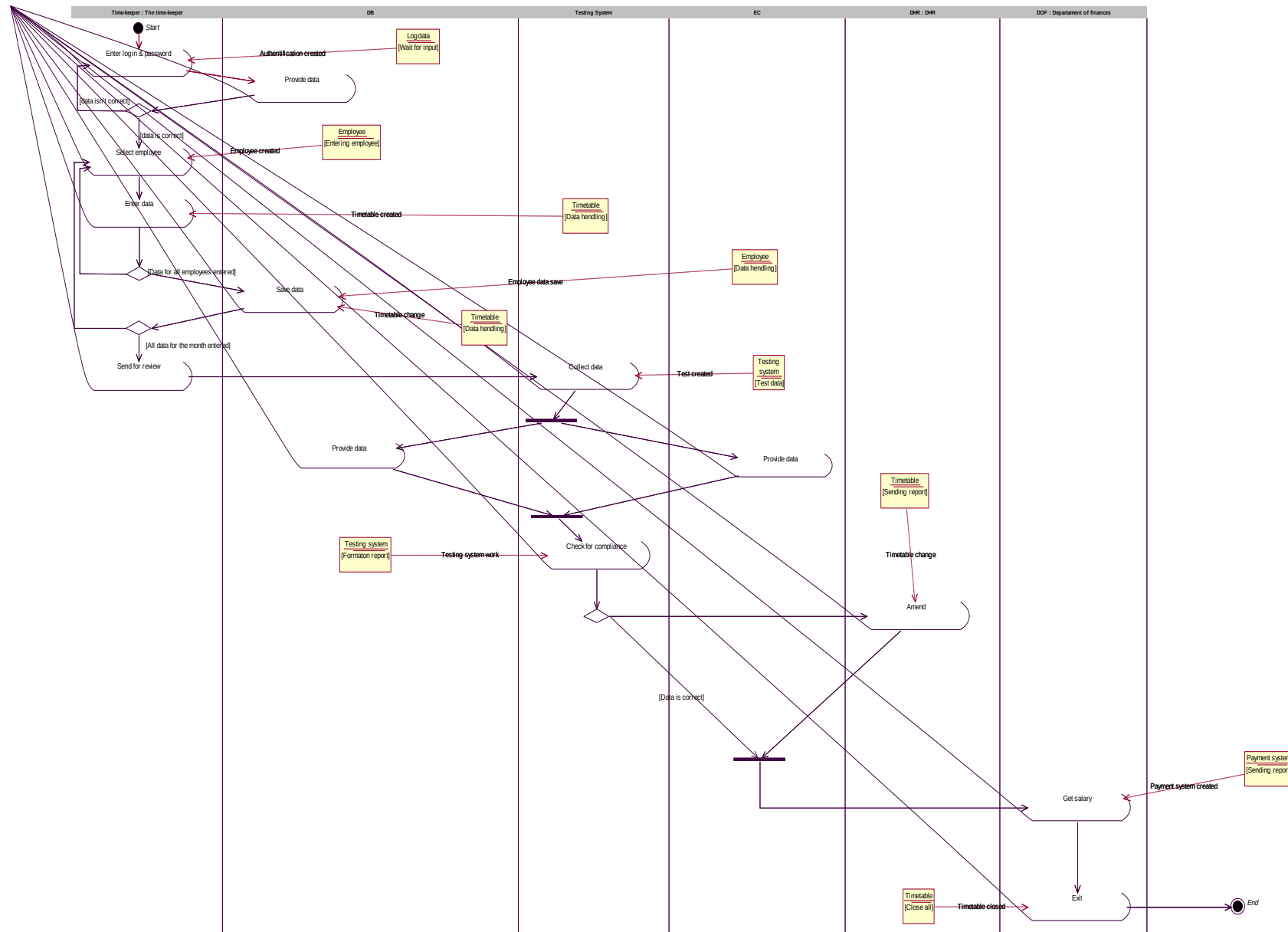


Диаграмма компонентов

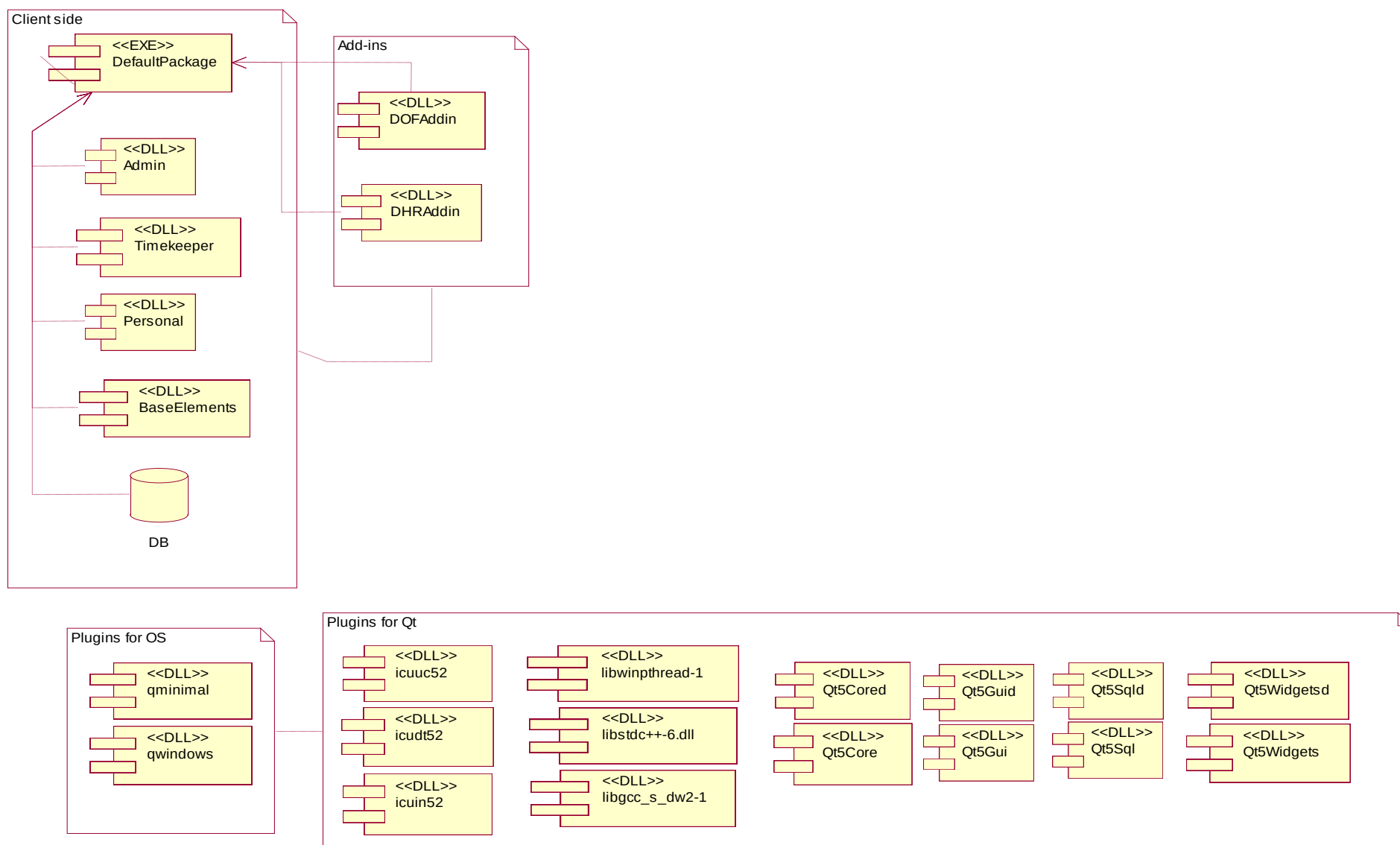


Диаграмма реализаций компонентов

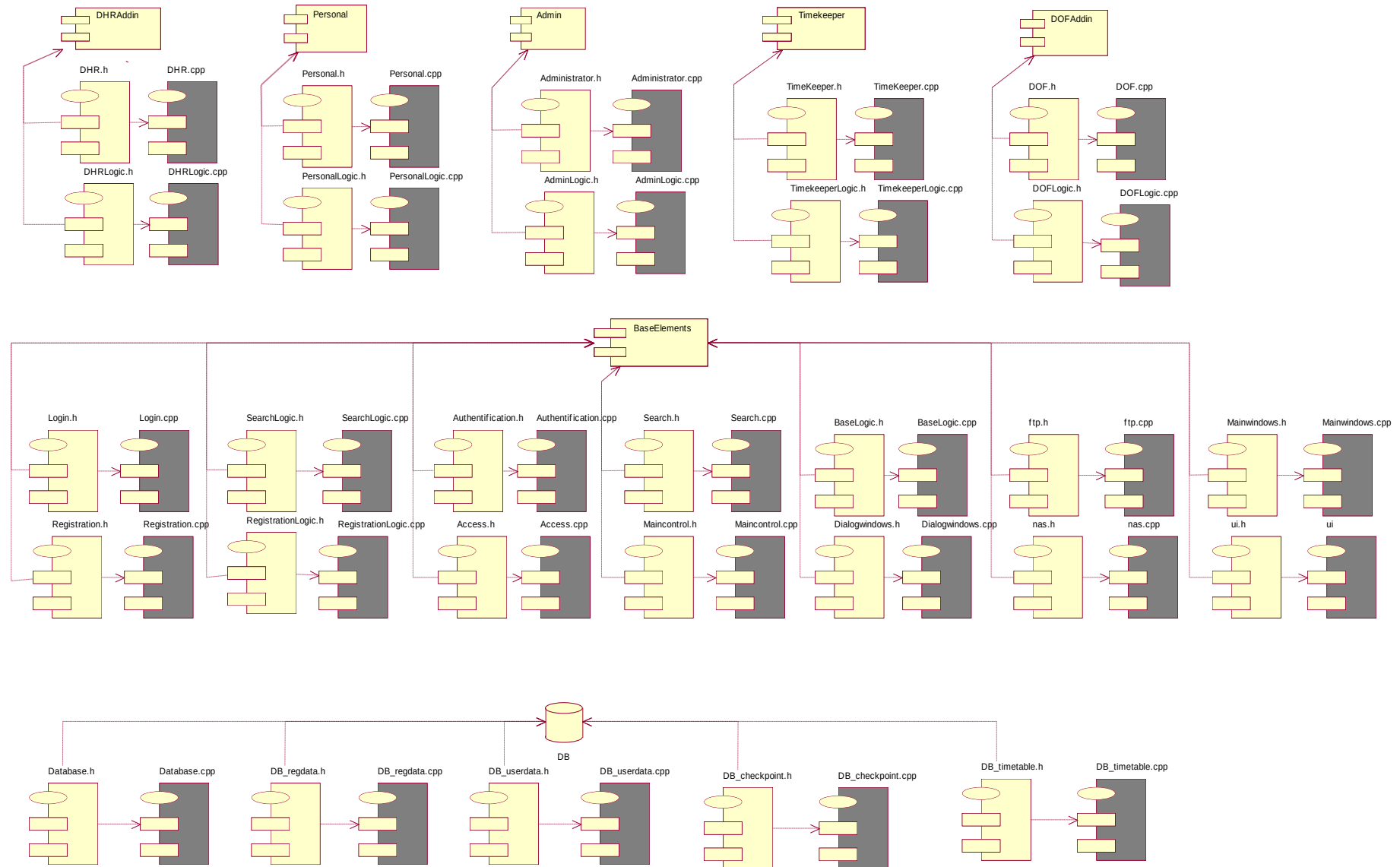
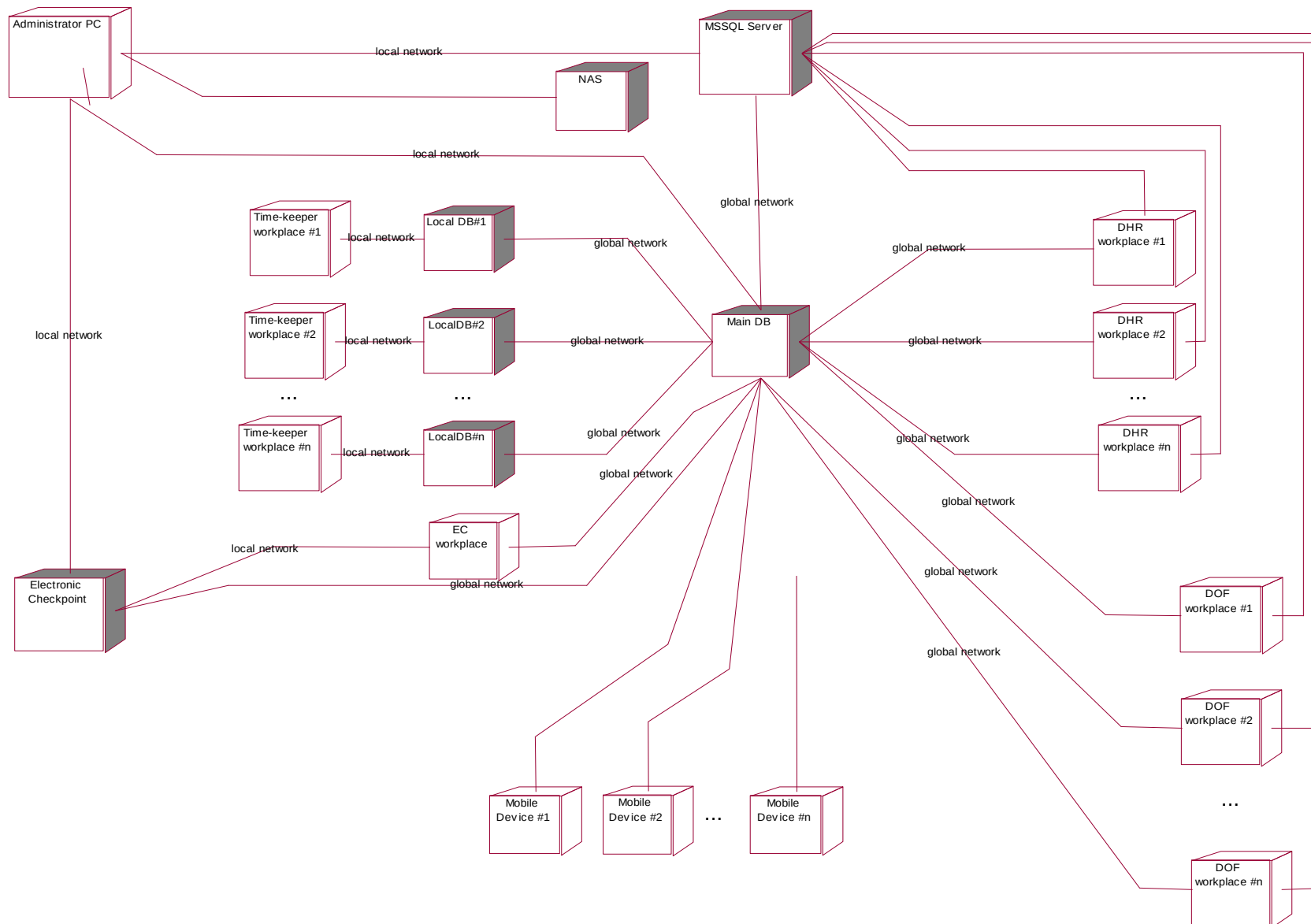


Диаграмма размещения



9. МОДЕЛЬ СИСТЕМЫ ОБЕСПЕЧЕНИЯ УЧАСТНИКОВ ИТ-ПРОЕКТА ИНФОРМАЦИЕЙ

Система обеспечивает доступ к электронной библиотеке предприятия. Работа осуществляется только с зарегистрированными пользователями. Источники, предложенные пользователям, открыты для ознакомления. При внесении новинок все пользователи оповещаются по электронной почте, новинки выносятся на главную страницу и имеют пометку «Новинка». Данная пометка снимается, как только каждый пользователь ознакомился с источником.

Пользователь может проводить поиск по источникам, предлагать источники на внесение в библиотеку.

Решение о добавлении источника принимает администратор.

Администратор может вносить источники и редактировать уже существующие.

Разделение доступа. В системе присутствует два уровня доступа: пользователь и администратор.

Пользователь имеет возможность просмотра, поиска и предложения источников.

Администратор может обновлять и редактировать базу, добавлять новых пользователей.

Регистрация пользователя возможна только при наличии договора предприятия.

После внесения регистрационных данных заявка передается администратору, который сверяет номер договора с бумажной версией и решает судьбу пользователя.

Пользователь заходит на сайт, выполняет вход в систему, далее он может просмотреть все источники по интересующему его критерию, при выборе конкретного открывается окно просмотра, в котором можно просмотреть весь текст источника, поставить отметку об ознакомлении либо узнать более подробную информацию по источнику.

При выборе поиска выводится окно с критериями, заполнение всех полей необязательно, после поиска все найденные данные выводятся в виде списка, в котором действуют все правила, что и в обычном списке источников.

При выборе пункта внесение источников перед пользователем появляется окно с полями для названия, автора, года и ссылки на источник.

Система проверяет базу на наличие аналогичных записей, при нахождении соответствий пользователь получает сообщение о том, что такой источник есть в базе, иначе заявка передается администратору.

Программа для администратора (по начальным расчетам) будет представлять СУБД с возможностью просмотра заявок пользователей.

Реализация. В основе разрабатываемой системы лежит концепция *Model-View-Controller*. Эта модель представляет собой конструктивный шаблон, определяющий способ построения архитектуры информационной системы, при котором логическая часть, пользовательский интерфейс и взаимодействие с пользователем выделены в независимые части.

Модель представлена несколькими модулями:

- 1) ядро;
- 2) адаптеры для ядра;
- 3) модуль работы с базой данных;
- 4) необходимые для системы валидаторы.

В ядре находятся наиболее общие для любого сайта классы и интерфейсы, предназначенные для работы с базами данных, электронной почтой, URL, модулями графического интерфейса и т.д.

Адаптеры предназначены для настройки ядра под конкретную систему. Они реализуют некоторые интерфейсы, находящиеся в ядре в соответствии с конкретной предметной областью.

База данных реализует взаимодействие с базой данных. Для каждого объекта базы (таблица) создается экземпляр класса, унаследованный от базового класса, определенного в ядре.

Валидаторы выполняют проверку данных. По большей части проверке подвергаются данные, вносимые в базу. Наследуются от базового класса, определенного в ядре.

Контроллер состоит из маршрутизатора и контроллеров.

Маршрутизатор определяет нужный контроллер в соответствии с действием пользователя.

Контроллеры определены для каждой страницы.

Вид представлен модулями и *view* для каждого окна и формы.

Модуль – графический элемент (кнопка, меню и т.д.).

Взаимодействует с *tpl*-файлами, подставляет необходимые данные в *html*.

View – шаблонизатор, отрисовывающий формы в соответствии с выбранным набором модулей.

В качестве стандартной графической нотации, с помощью которой можно визуализировать ER-модель, была предложена диаграмма сущность-связь (ER-диаграмма) (*Entity-Relationship Diagram*), представленная на рис. 9.1, описание классов дано в табл. 9.1.

В модели связи между таблицами имеют тип «один-ко-многим» и «один-к-одному». Отношение «один-ко-многим» имеет место, когда одной записи родительской таблицы может соответствовать несколько записей дочерней. Связь «один-ко-многим» иногда называют связью «многие-к-одному». И в том, и в другом случае сущность связи между таблицами остается неизменной. Связь «один-ко-многим» является самой распространенной для реляционных баз данных. Она позволяет моделировать также иерархические структуры данных.

Связь «один-к-одному» образуется, когда ключевой столбец (идентификатор) присутствует в другой таблице, в которой тоже является ключом либо свойствами столбца задана его уникальность (одно и то же значение не может повторяться в разных строках), она может возникнуть, когда требуется разделить данные одной таблицы на несколько отдельных таблиц с целью безопасности.

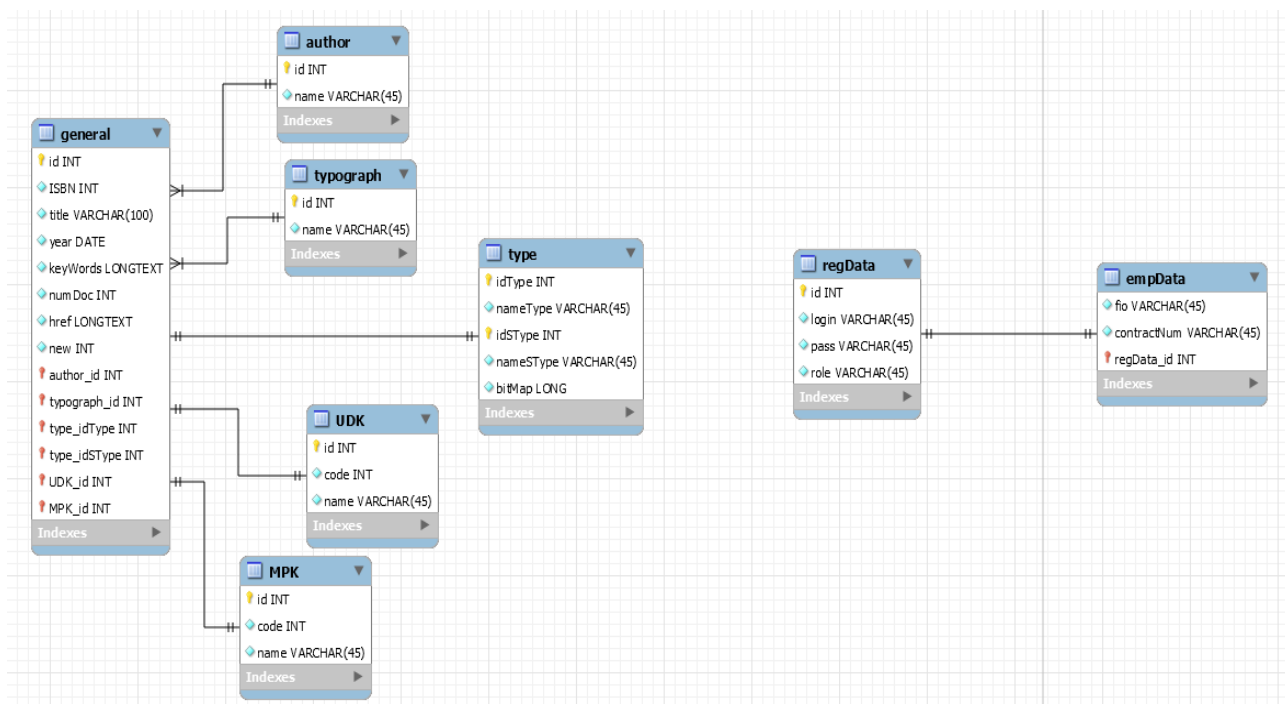


Рис. 9.1. ER-диаграмма

9.1. Описание классов

general		
id	INT	Уникальный номер каждого источника
ISBN	INT	Международный стандартный книжный номер
title	VARCHAR	Название
year	DATE	Год выпуска
keyWords	LONGTEXT	Ключевые слова
numDoc	INT	Номер документа
href	LONGTEXT	Ссылка
new	INT	Маркер новинки
author_id	INT	Автор
typograph_id	INT	Типография
type_idType	INT	Тип
type_idSType	INT	Подтип
UDK_id	INT	УДК
MPK_id	INT	МПК

author		
id	INT	Уникальный номер автора
name	VARCHAR	ФИО
typograph		
id	INT	Уникальный номер типографии
name	VARCHAR	Название
type		
idType	INT	Уникальный номер типа
nameType	VARCHAR	Название типа
isSType	INT	Уникальный номер подтипа
nameSType	VARCHAR	Название подтипа
bitMap	LONG	Список необходимых полей
UDK		
id	INT	Уникальный номер
code	INT	Универсальная десятичная классификация
name	VARCHAR	Значение
MPK		
id	INT	Уникальный номер
code	INT	Международная патентная классификация
name	VARCHAR	Значение
regData		
id	INT	Уникальный номер
login	VARCHAR	Логин
pass	VARCHAR	Пароль
role	VARCHAR	Уровень доступа
empData		
fio	VARCHAR	ФИО
contractNum	VARCHAR	Номер договора
regData_id	INT	Уникальный номер

Диаграмма вариантов использования приведена на рис. 9.2, а описание сценариев – в табл. 9.2.

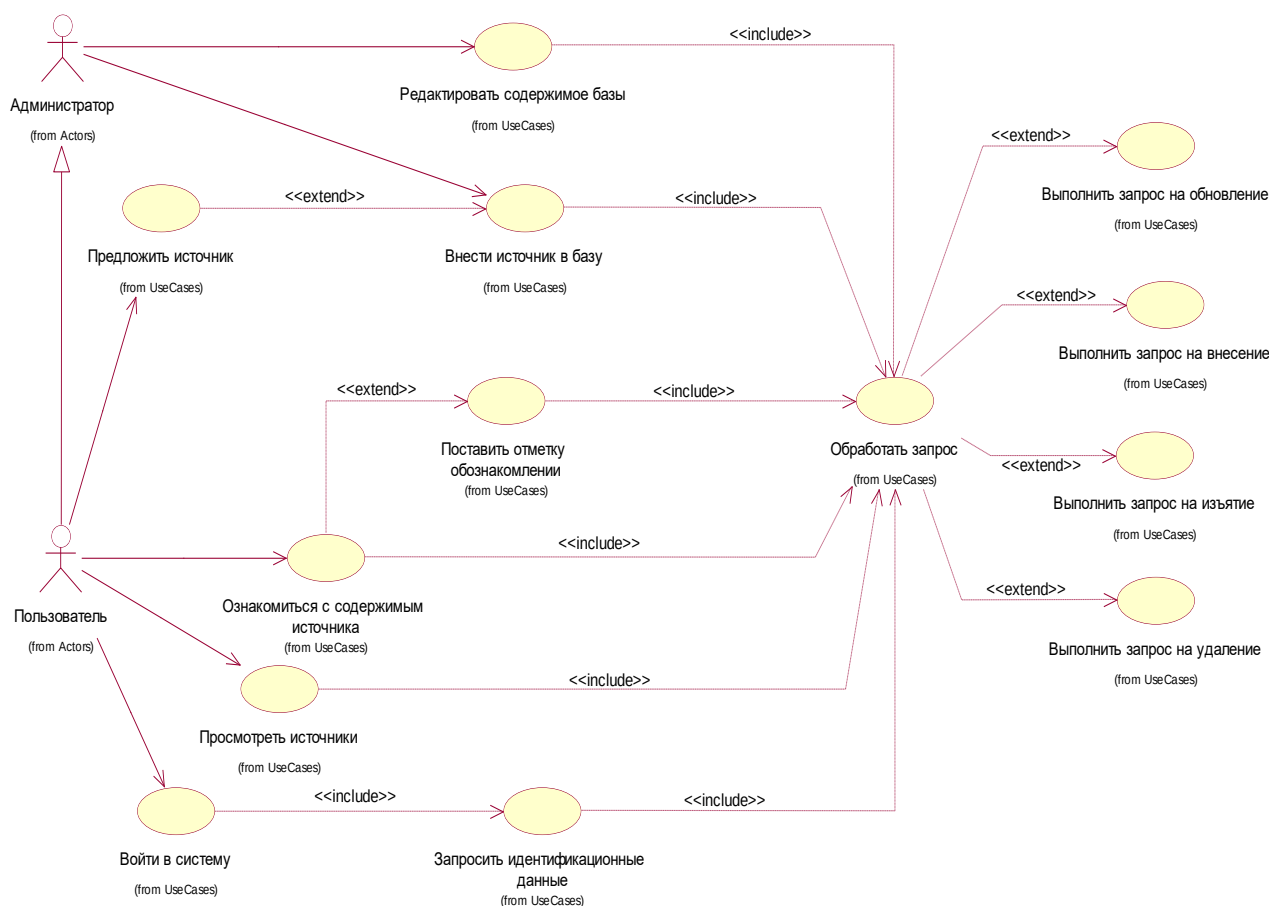


Рис. 9.2. Диаграмма вариантов использования

9.2. Описание сценариев

Краткое описание	Действия
<i>Вариант использования – войти в систему</i>	
Основной сценарий	Активизируется, когда пользователь заходит в форму авторизации. 1. Пользователь вводит необходимые данные. 2. Система проверяет заполнение необходимых полей. 3. Система отправляет данные на проверку. 4. Система обеспечивает присвоение прав, соответствующих уровню доступа
Альтернативные сценарии	Не все необходимые поля заполнены. Пользователь получает уведомление, после чего может исправить данные или выйти
	Нет доступа к БД. Пользователь получает уведомление о неисправности
Предусловия	Нет
Постусловия	Система должна выполнить проверку введенных данных

Краткое описание	Действия
<i>Вариант использования – просмотреть источники</i>	
Основной сценарий	Активизируется, когда пользователь заходит в форму просмотра списка источников. 1. Пользователь вводит критерии для отбора источников. 2. Система осуществляет выборку данных. 3. Система отображает результат
Альтернативные сценарии	Нет доступа к БД. Пользователь получает уведомление о неисправности
Предусловия	Пользователь должен быть зарегистрирован в системе
Постусловия	Система должна выполнить проверку введенных данных
<i>Вариант использования – ознакомиться с содержимым источника</i>	
Основной сценарий	Активизируется, когда пользователь выбирает конкретный источник из списка. 1. Система осуществляет выборку данных. 2. Система отображает данные. 3. Пользователь помечает источник как прочитанный
Альтернативные сценарии	Нет доступа к БД. Пользователь получает уведомление о неисправности
Предусловия	Пользователь должен быть зарегистрирован в системе
Постусловия	Система должна проверить установку метки пользователем
<i>Вариант использования – предложить источник</i>	
Основной сценарий	Активизируется, когда пользователь заходит в форму внесения источников. 1. Пользователь вводит необходимые данные. 2. Система проверяет заполнение необходимых полей. 3. Система отправляет данные на проверку
Альтернативные сценарии	Не все необходимые поля заполнены. Пользователь получает уведомление, после чего может исправить данные или выйти
	Нет доступа к БД. Пользователь получает уведомление о неисправности
Предусловия	Пользователь должен быть зарегистрирован в системе
Постусловия	Система должна выполнить проверку введенных данных

Краткое описание	Действия
<i>Вариант использования – внести источник в базу</i>	
Основной сценарий	Активизируется, когда администратор вносит источник в базу. 1. Администратор вводит необходимые данные. 2. Система проверяет заполнение необходимых полей. 3. Система отправляет данные на проверку
Альтернативные сценарии	Не все необходимые поля заполнены. Пользователь получает уведомление, после чего может исправить данные или выйти
	Нет доступа к БД. Пользователь получает уведомление о неисправности
Предусловия	Пользователь должен иметь уровень доступа – Администратор
Постусловия	Система должна выполнить проверку введенных данных
<i>Вариант использования – редактировать содержимое базы</i>	
Основной сценарий	Активизируется, когда администратор производит редактирование базы данных. 1. Администратор вводит необходимые правки. 2. Система проверяет заполнение полей. 3. Система отправляет данные на проверку. 4. Система выполняет обновление
Альтернативные сценарии	Не все необходимые поля заполнены. Пользователь получает уведомление, после чего может исправить данные или выйти
	Нет доступа к БД. Пользователь получает уведомление о неисправности
Предусловия	Пользователь должен иметь уровень доступа – Администратор
Постусловия	Система должна выполнить проверку введенных данных

На рисунке 9.3 приведена диаграмма классов логической части системы, а на рис. 9.4 – диаграмма классов пользовательского интерфейса.

Учетная Запись отвечает за хранение текущей учетной записи пользователя, списка возможных его действий и обеспечивает необходимые операции над этими данными.

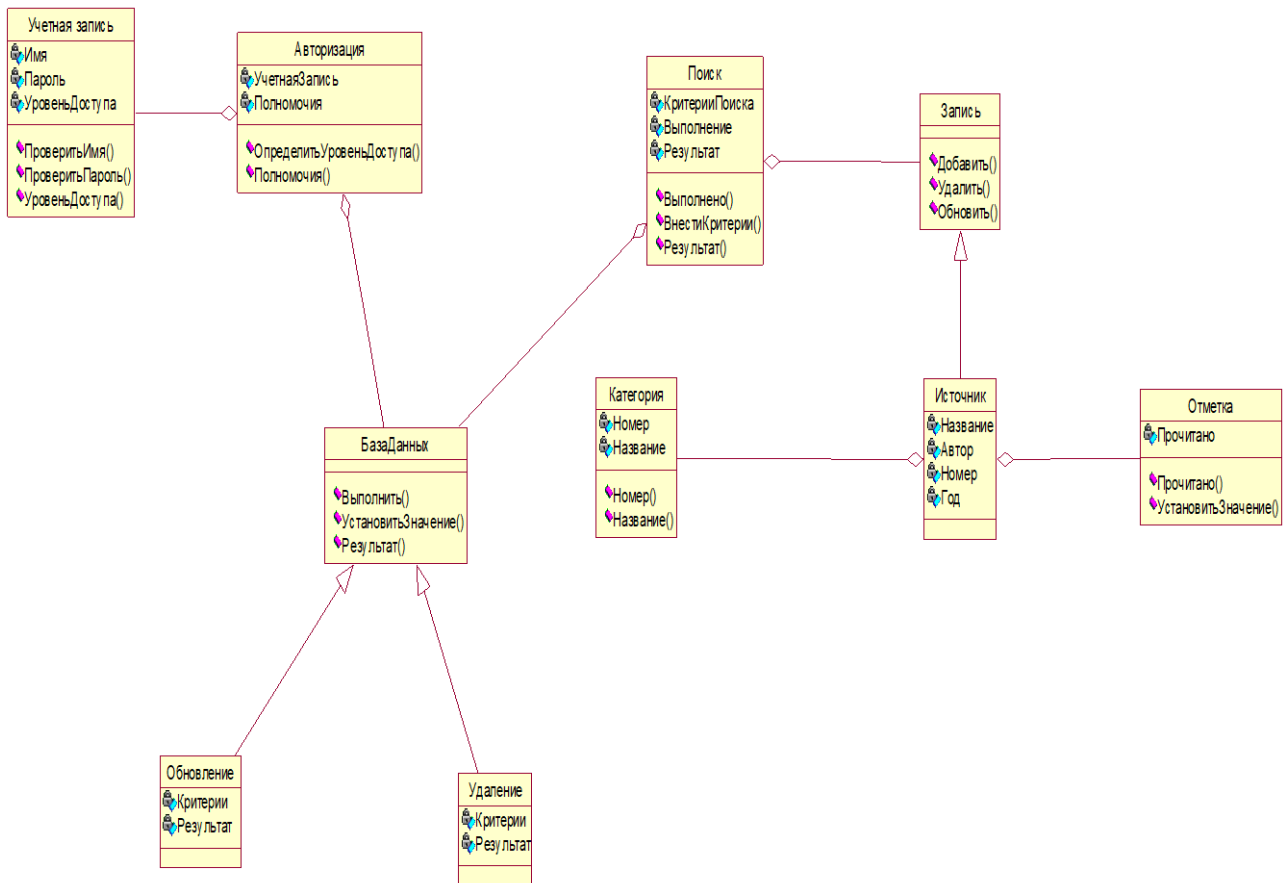


Рис. 9.3. Диаграмма классов логической части системы

Авторизация выполняет включение/выключение возможностей системы для различных уровней доступа

База Данных – отвечает за работу с базой.

Обновление – специализированный класс для запросов на обновление в базу данных.

Удаление – специализированный класс для запросов на удаление в базу данных.

Поиск хранит список записей из базы, критерии для выбора и методы для внесения и изъятия этих данных, запуска поиска и определения статуса

Запись – хранит конкретную запись из базы.

Источник расширяет класс запись для работы с литературными источниками.

Категория – номер и название категории, с которой осуществляется работа.

Отметка – установка отметки о прочтении и ее хранение.

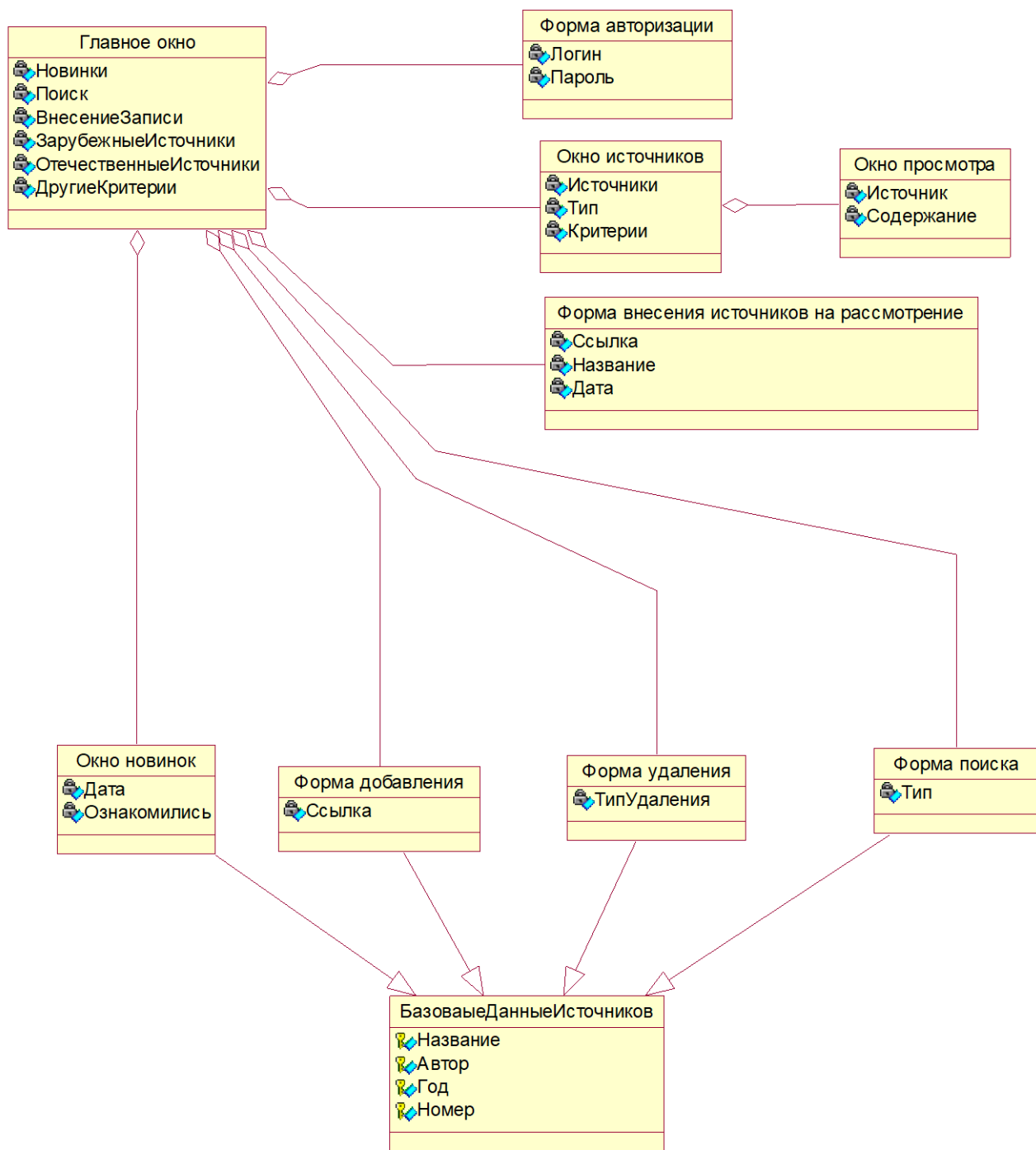


Рис. 9.4. Диаграмма классов пользовательского интерфейса

Каждый из классов отвечает за соответствующую форму или вкладку, где атрибуты являются графическими элементами, вроде кнопки, поля для заполнения и т.д.

Диаграмма последовательностей (*Sequence Diagrams*) отображает взаимодействие объектов в динамике.

Диаграмма последовательности авторизации дана на рис. 9.5.

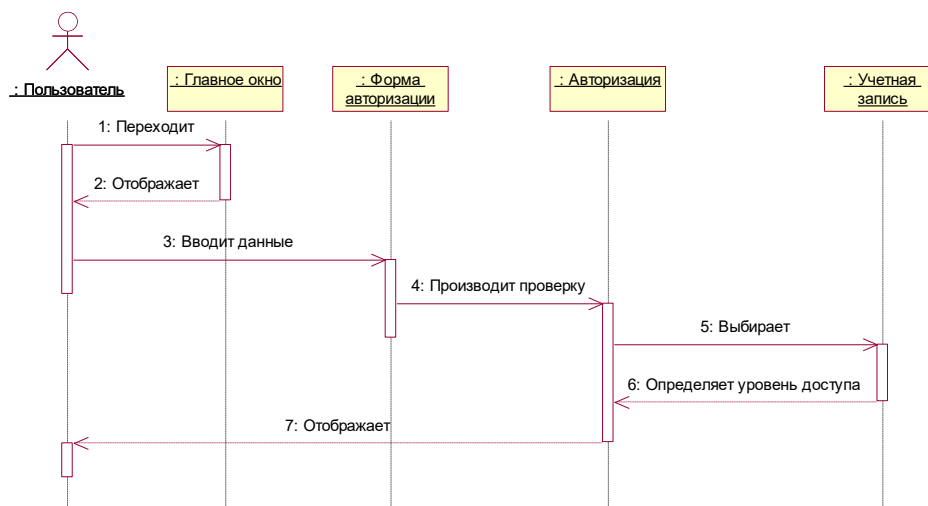


Рис. 9.5. Диаграмма последовательности авторизации

Пользователь переходит на главную страницу сайта:

- система отображает начальную страницу, содержащую гиперссылки на страницы поиска, авторизации и просмотра данных по некоторым критериям;
- пользователь вводит идентификационные данные;
- система проводит проверку на соответствие, выбирает соответствующую учетную запись, определяет уровень доступа;
- система отображает сообщение о статусе входа в систему.

Диаграмма кооперации авторизации приведена на рис. 9.6.

Диаграмма последовательности обновления базы приведена на рис. 9.7.

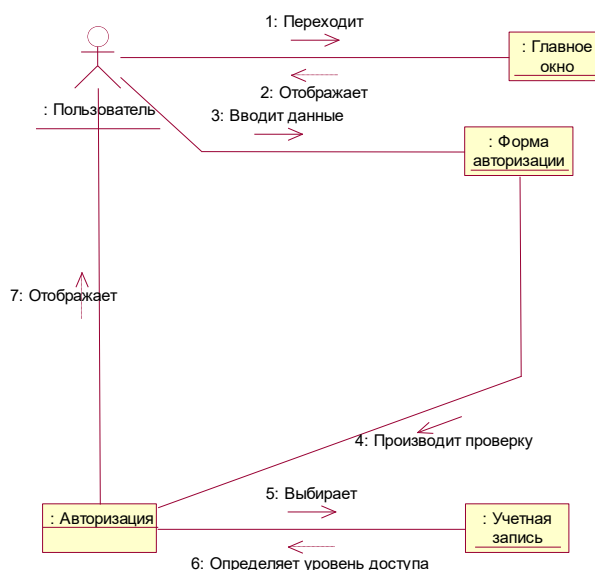


Рис. 9.6. Диаграмма кооперации авторизации

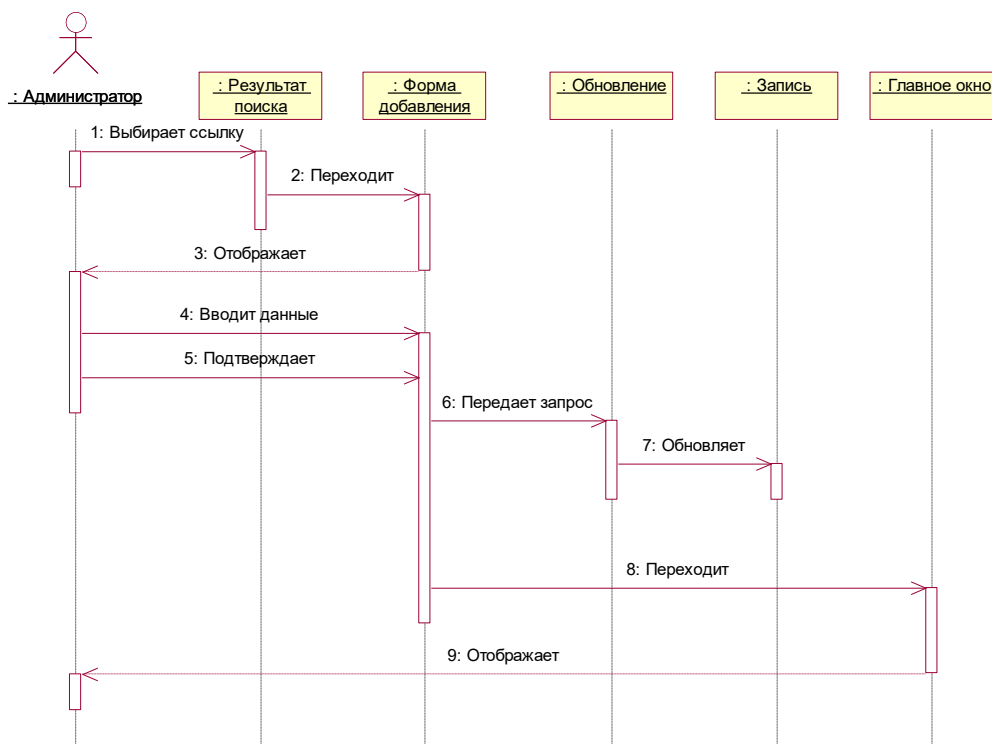


Рис. 9.7. Диаграмма последовательности обновления базы

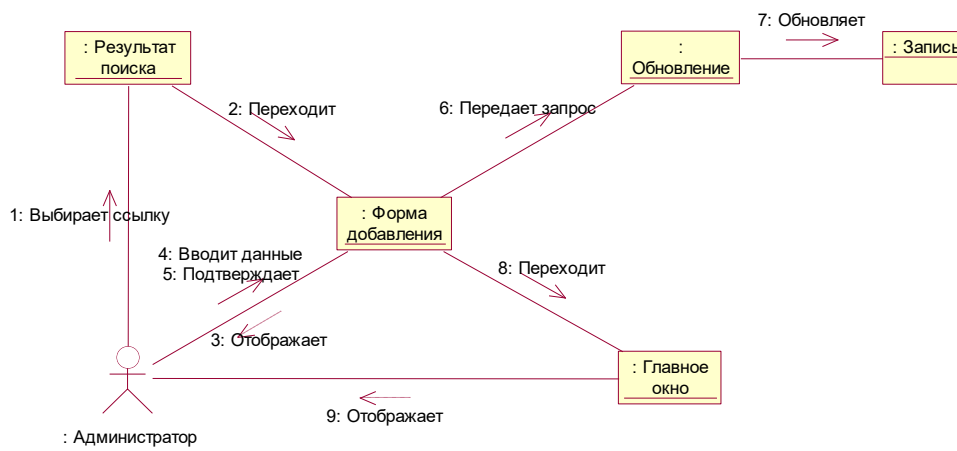


Рис. 9.8. Диаграмма кооперации обновления базы

Диаграмма кооперации обновления базы приведена на рис. 9.8.

Администратор после проведения поиска выбирает определенную ссылку, нажимает кнопку обновить, система отображает окно добавления источников.

Администратор вводит данные, подтверждает проведение операции, система передает запрос на обновление.

Выбранная запись обновляется, после успешного выполнения осуществляется переход на главную страницу.

Диаграмма последовательности ознакомления с источником приведена на рис. 9.9, а диаграмма кооперации – на рис. 9.10.

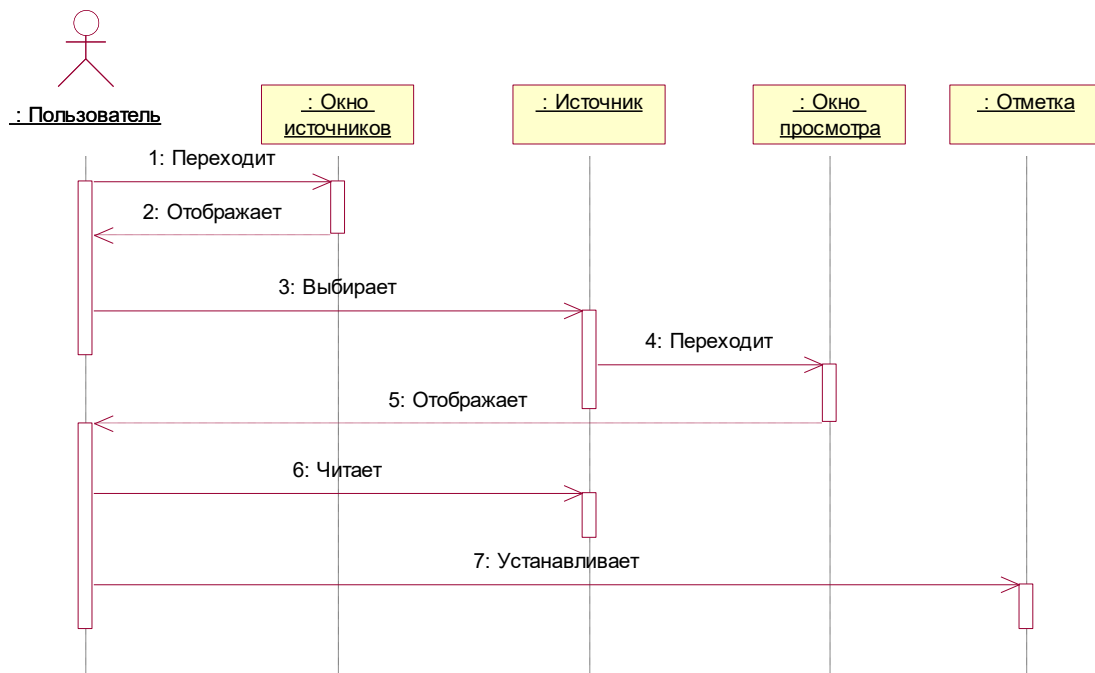


Рис. 9.9. Диаграмма последовательности ознакомления с источником

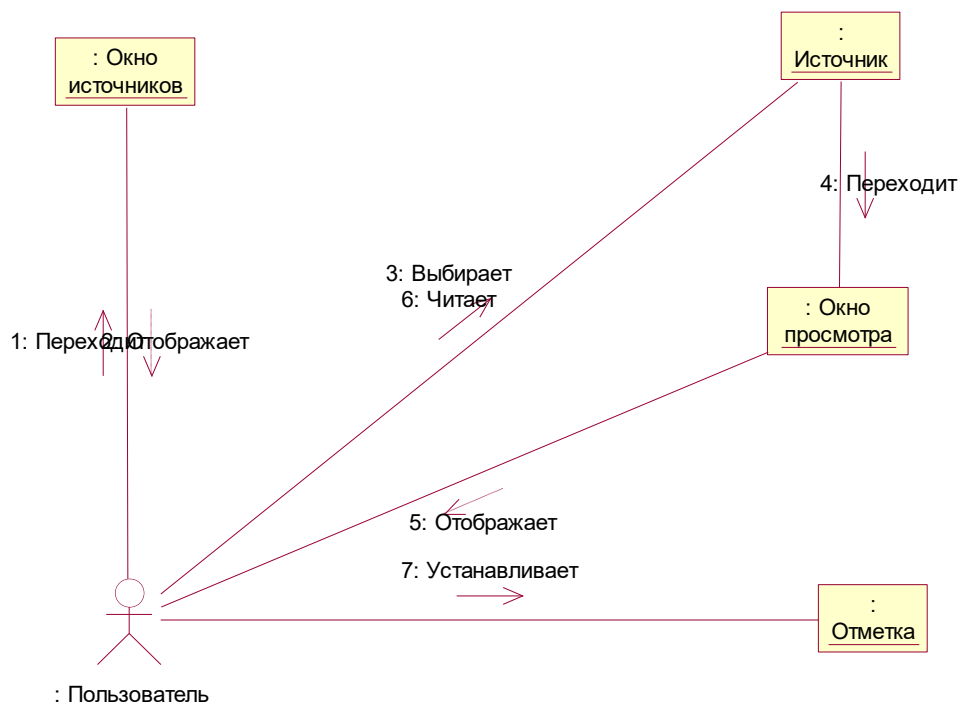


Рис. 9.10. Диаграмма кооперации ознакомления с источником

Пользователь переходит в окно источников, система отображает окно источников, содержащее ссылки на все источники, хранящиеся в базе, и критерии отбора источников.

Пользователь выбирает интересующий его источник, система отображает окно просмотра, в котором можно ознакомиться с содержимым и поставить отметку о прочтении. Пользователь просматривает содержимое, ставит отметку об ознакомлении.

Диаграмма последовательности поиска приведена на рис. 9.11, а диаграмма кооперации поиска – на рис. 9.12.

Пользователь переходит на главную страницу сайта, система отображает начальную страницу, содержащую гиперссылки на страницы поиска, авторизации и просмотра данных по некоторым критериям.

Пользователь открывает страницу поиска, вводит критерии, система осуществляет поиск по базе. Соответствующие критериям записи заносятся в список, сформированный список предоставляется пользователю.

Диаграмма последовательности внесения источника на рассмотрение приведена на рис. 9.13, а диаграмма кооперации – на рис. 9.14.

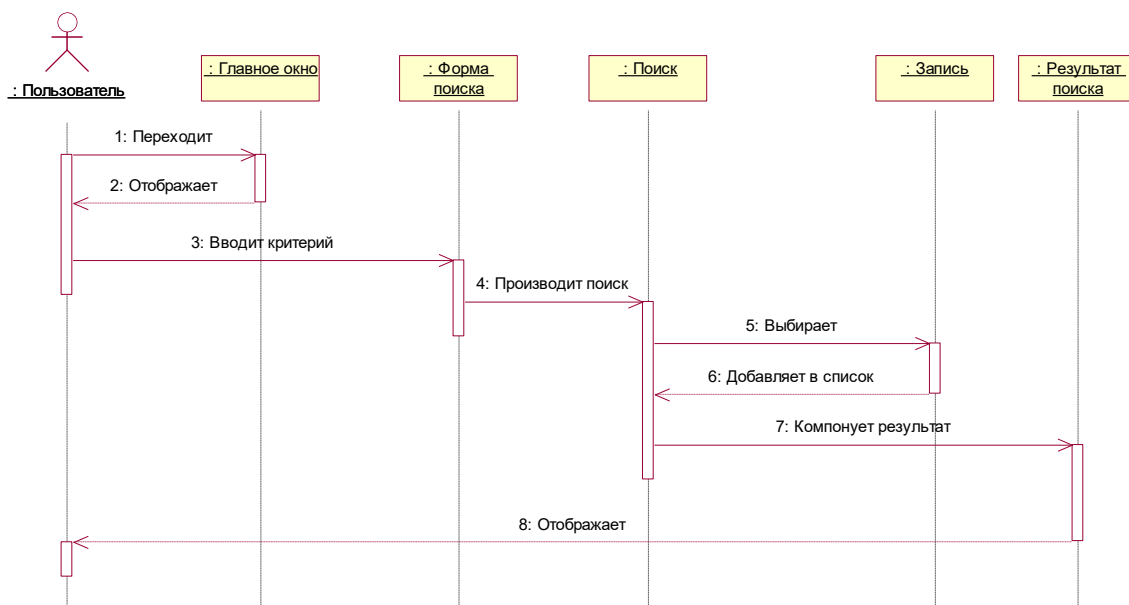


Рис. 9.11. Диаграмма последовательности поиска

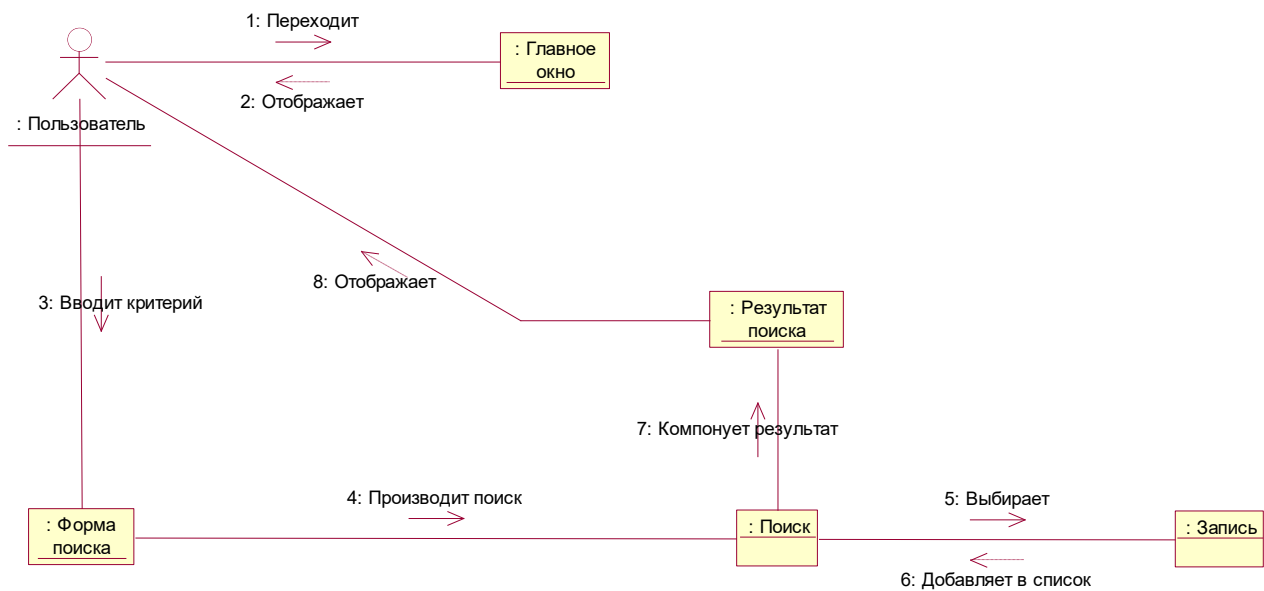


Рис. 9.12. Диаграмма кооперации поиска

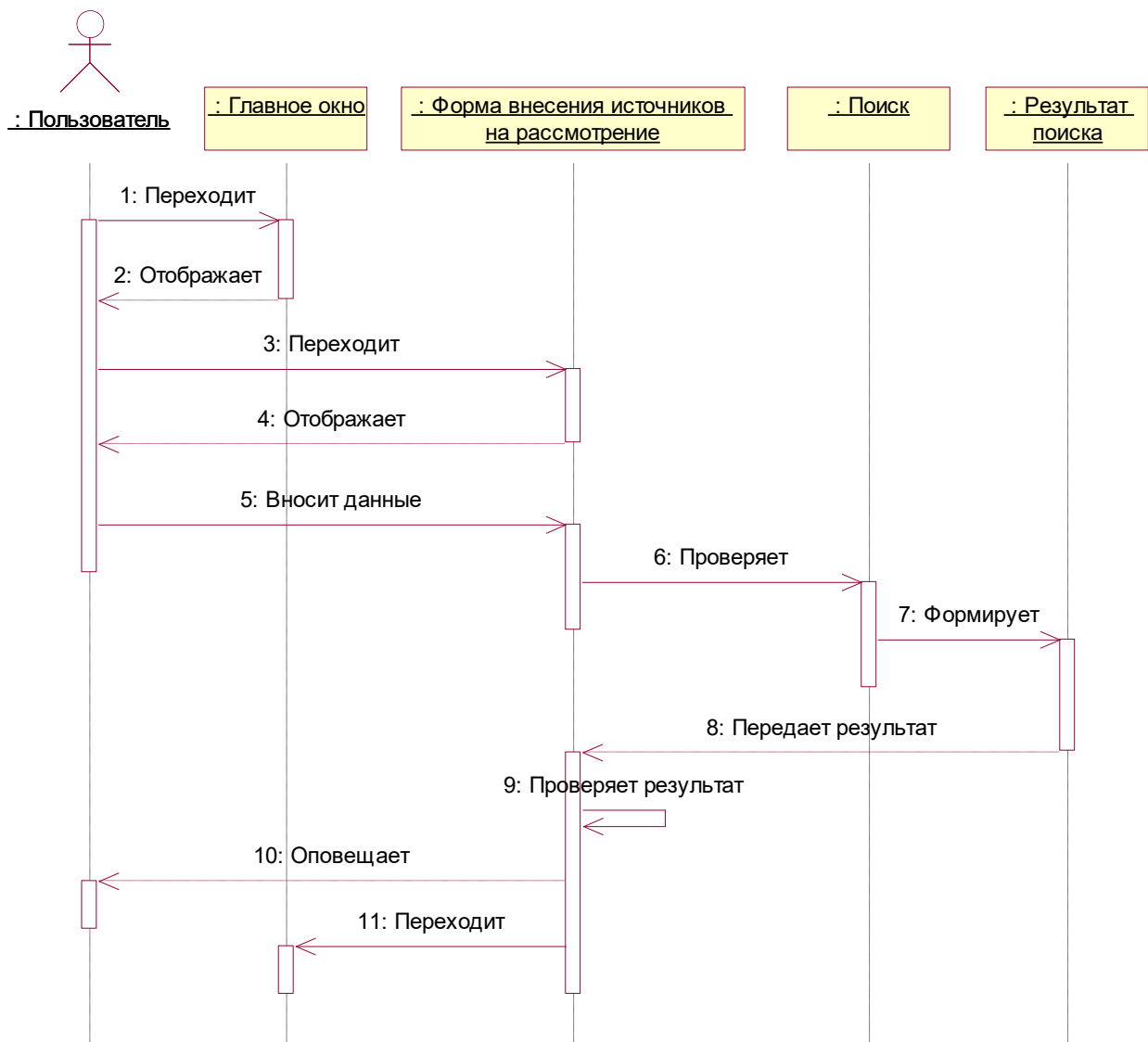


Рис. 9.13. Диаграмма последовательности внесения источника на рассмотрение

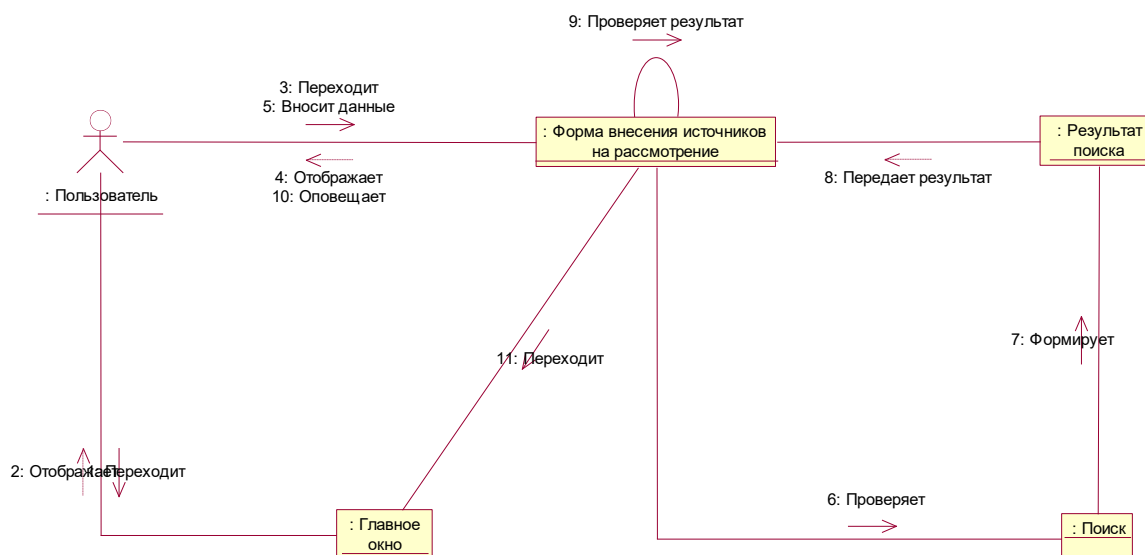


Рис. 9.14. Диаграмма кооперации внесения источника на рассмотрение

Пользователь переходит на главную страницу сайта, система отображает начальную страницу, содержащую гиперссылки на страницы поиска, авторизации и просмотра данных по некоторым критериям.

Пользователь открывает страницу внесения источников, заносит необходимые данные, система проверяет на наличие аналогичных записей.

Результат поиска анализируется, система оповещает пользователя о результатах, осуществляется переход на главную страницу.

Диаграмма последовательности просмотра списка источников приведена на рис. 9.15, а диаграмма кооперации – на рис. 9.16.

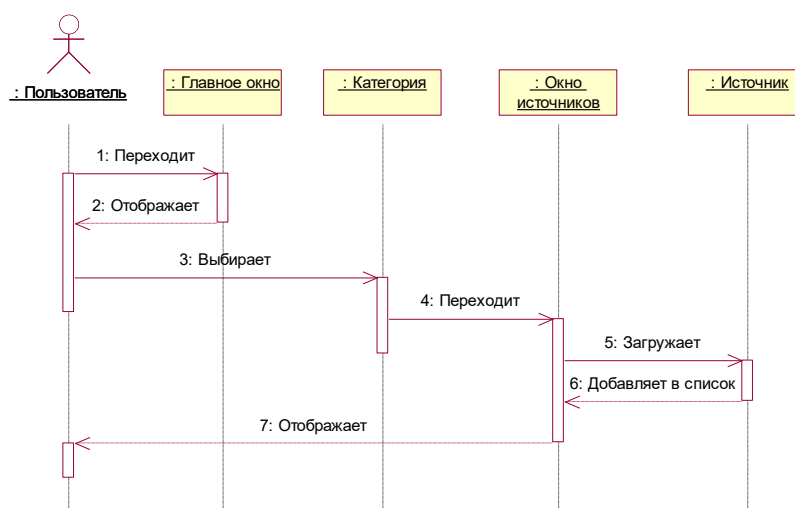


Рис. 9.15. Диаграмма последовательности просмотра списка источников

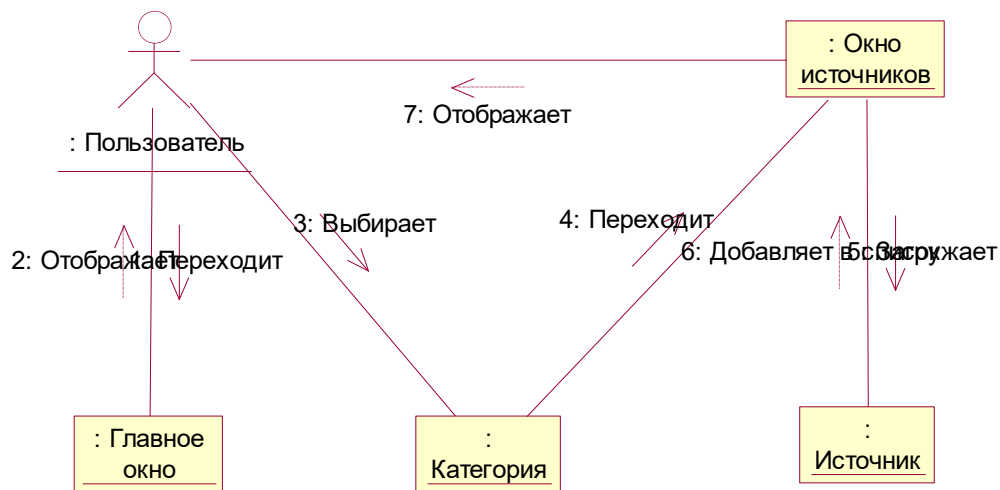


Рис. 9.16. Диаграмма кооперации просмотра списка источников

Пользователь переходит на главную страницу сайта, система отображает начальную страницу, содержащую гиперссылки на страницы поиска, авторизации и просмотра данных по некоторым критериям.

Пользователь выбирает интересующий его критерий, переходит на страницу с источниками, соответствующими выбранному критерию.

Диаграмма последовательности удаления из базы приведена на рис. 9.17, а диаграмма кооперации – на рис. 9.18.

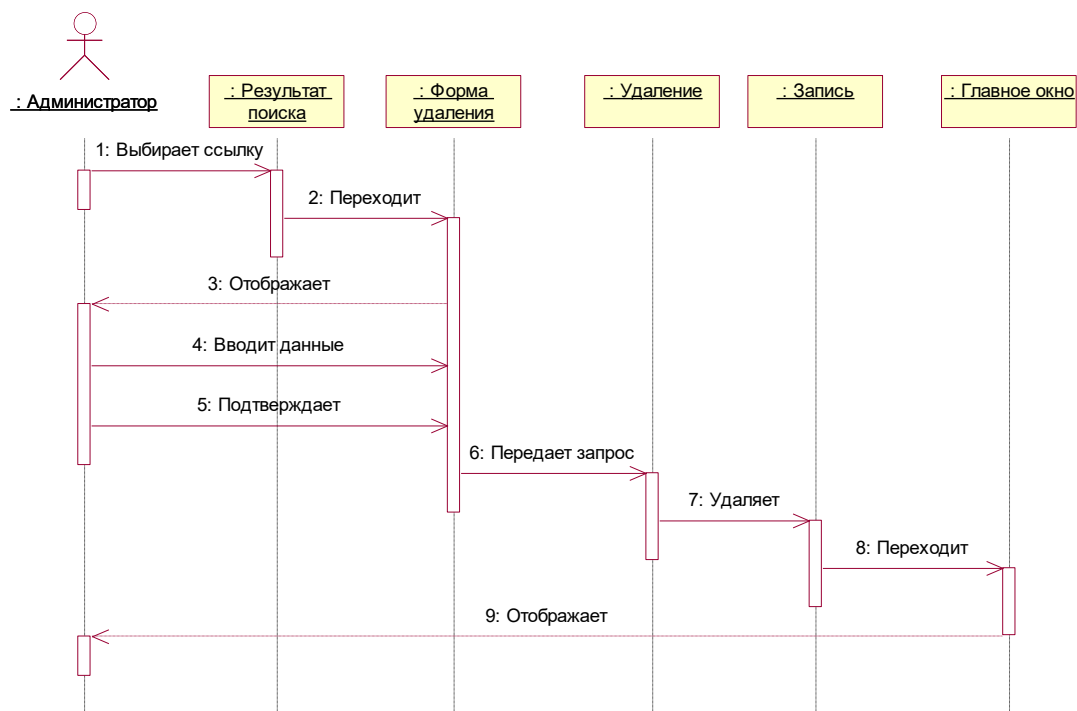


Рис. 9.17. Диаграмма последовательности удаления из базы

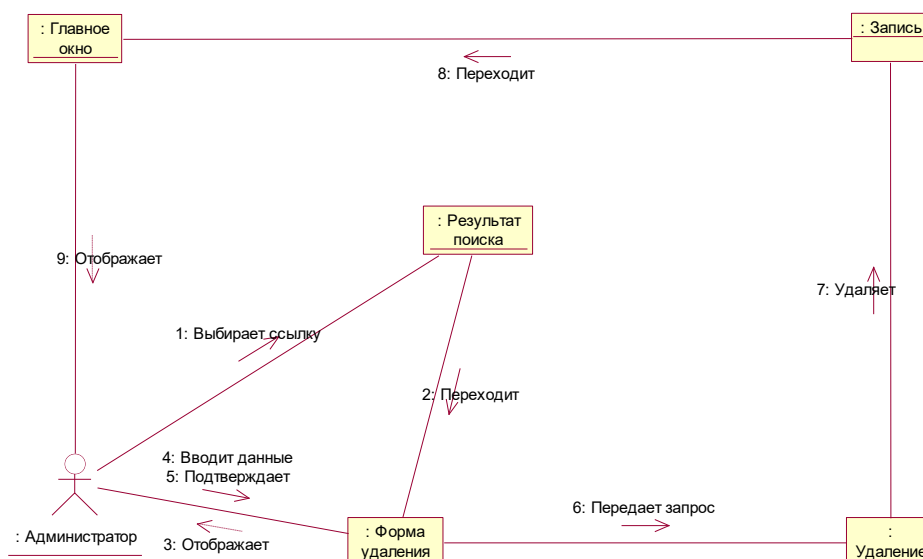


Рис. 9.18. Диаграмма кооперации удаления из базы

Администратор после проведения поиска выбирает определенную ссылку, нажимает кнопку удалить, система отображает окно удаления источников.

Администратор вводит данные, подтверждает проведение операции, система передает запрос на удаление, выбранная запись удаляется, после успешного выполнения осуществляется переход на главную страницу.

Диаграммы деятельности (*Activity Diagram*) раскрывают детали алгоритмической реализации операций, выполняемых системой. Диаграммы деятельности применяются для визуализации алгоритмов, по которым работают операции классов.

Диаграмма деятельности приведена на рис. 9.19.

После запуска системы открывается главное окно, из которого пользователю предоставляется несколько вариантов:

- внести данные в форму авторизации и войти в систему (последующее доступно только после входа в систему);
- открыть список источников, выбрать интересующий источник и ознакомиться с содержанием;
- провести поиск по источникам;
- внести источник на рассмотрение.

В любом из вариантов пользователь может закрыть систему.

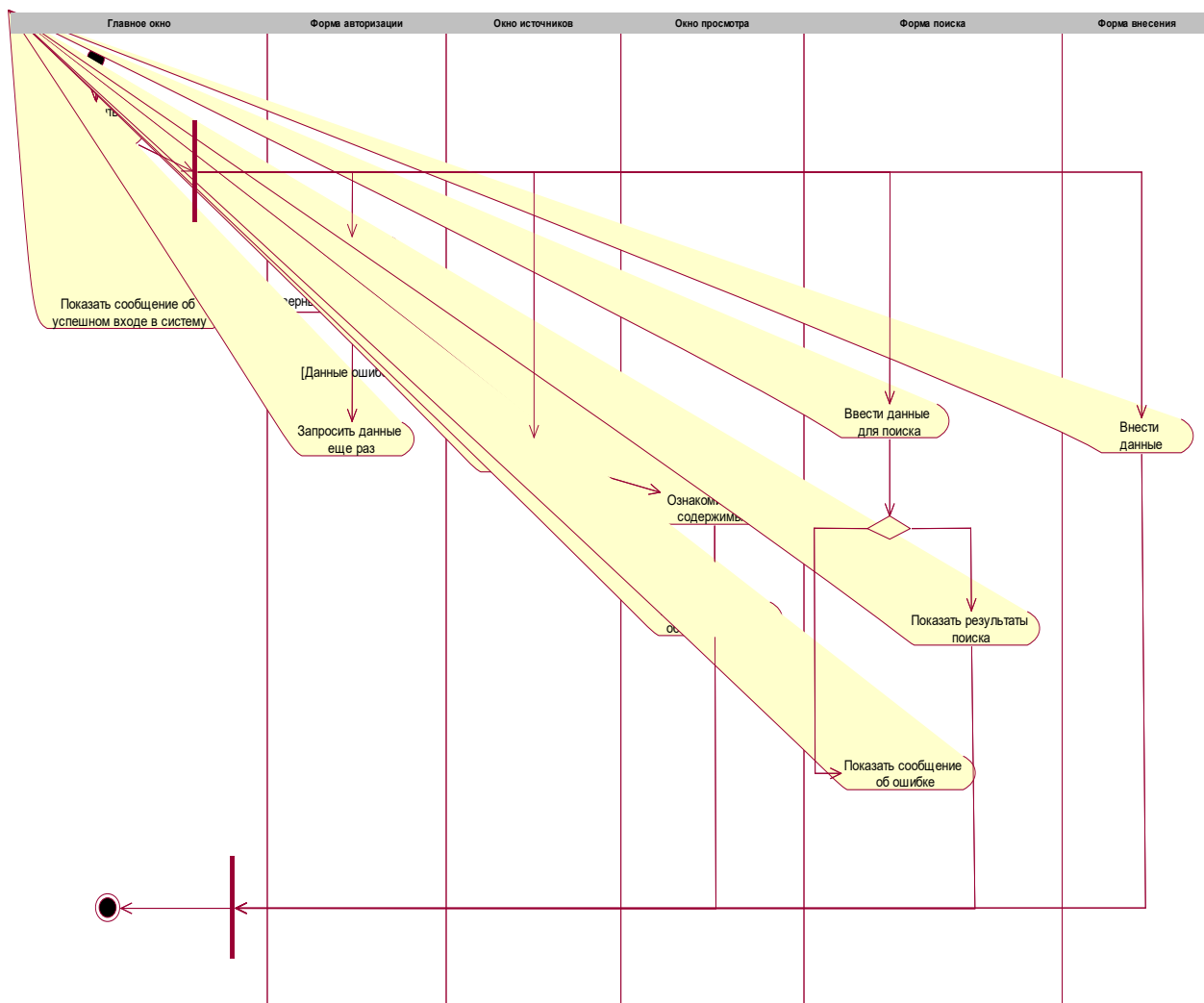


Рис. 9.19. Диаграмма деятельности

Диаграммы состояний (*Statechart Diagram*) применяются для пояснения работы сложных объектов, а точнее переходов их из одного состояния в другое. Говоря иначе, она показывает автомат, фокусируя внимание на потоке управления.

Диаграмма состояний приведена на рис. 9.20.

Начальное состояние системы: открыта главная страница, ожидается действие пользователя.

Далее возможно несколько вариантов действий.

Вызов формы авторизации: система ожидает ввода данных, проверяет введенную информацию и либо открывает доступ, либо запрашивает повторное введение. (После этого пользователь может выполнять любое из нижеследующих действий)

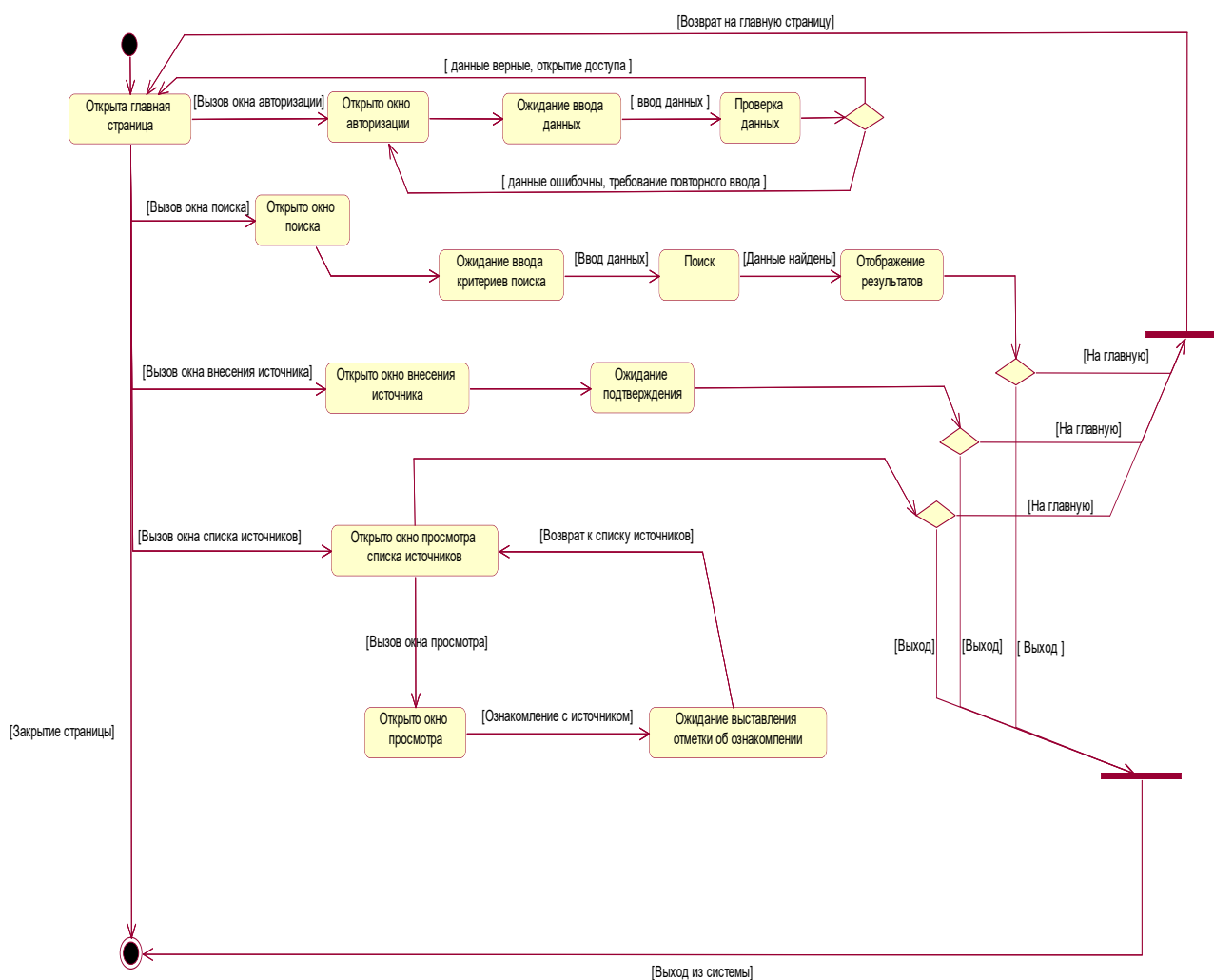


Рис. 9.20. Диаграмма состояний

Если осуществляется вызов списка источников, то после система ожидает выбор интересующего источника и выставление отметки об ознакомлении, после непосредственного прочтения.

Если осуществляется вызов окна поиска по источникам, то система ожидает ввода критериев, после чего осуществляет поиск, затем отображает найденный результат.

Если осуществляется вызов формы внесения источника на рассмотрение, то после ввода данных ожидается подтверждение правильности.

В любом из вариантов после выполнения действий система может перейти либо к главному окну, либо выполнить выход, в зависимости от действий пользователя.

На рисунке 9.21 приведена диаграмма компонентов.

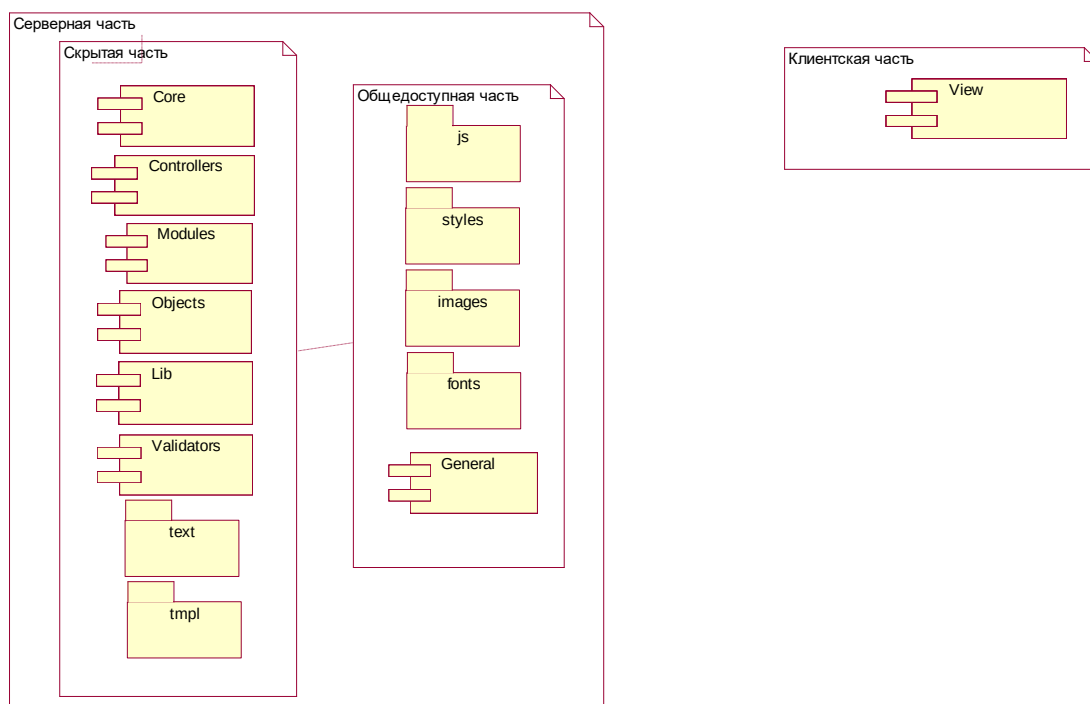


Рис. 9.21. Диаграмма компонентов

Модель размещения (*Deployment Diagrams*) – модель физической архитектуры системы, отображает технологическую конфигурацию системы.

На модели размещения представлена конфигурация обрабатывающих узлов системы и размещенных в них компонентов, она приведена на рис. 9.22.

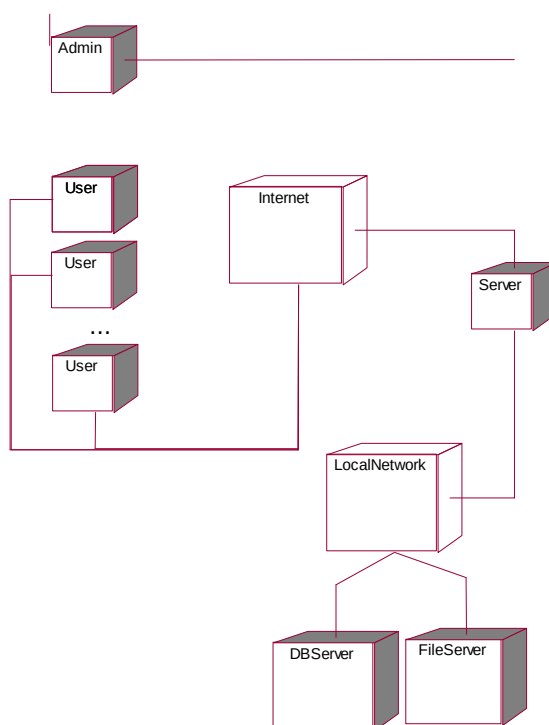


Рис. 9.22. Диаграмма размещения

User – компьютеры пользователей, которые работают с электронной библиотекой.

Admin – компьютер администратора, используется для занесения и изменения данных в базе.

Internet – группа устройств, предоставляющих доступ к глобальной сети Интернет.

Server – сервер электронной библиотеки.

Local Network – устройства локальной сети, обеспечивающие связь серверов.

DBServer – сервер БД, который содержит информацию о книгах, патентах и т.п.

FileServer – файловый сервер, который хранит все материалы.

10. ДОКУМЕНТИРОВАНИЕ ПРОГРАММНЫХ СРЕДСТВ В ИТ-ПРОЕКТЕ

При разработке создается и используется большой объем разнообразной документации. Она необходима как средство передачи информации между разработчиками, как средство управления разработкой и как средство передачи пользователям информации, необходимой для применения и сопровождения. Документацию можно разбить на две группы:

- документы управления разработкой;
- документы, входящие в состав программного средства.

Документы управления разработкой управляют и протоколируют процессы разработки и сопровождения, обеспечивая связи внутри коллектива разработчиков и между коллективом разработчиков и менеджерами программного средства (Software Managers или Program Managers) – лицами, управляющими разработкой. Эти документы могут быть следующих типов:

- планы, оценки, расписания (создаются менеджерами для прогнозирования и управления процессами разработки и сопровождения);
- отчеты об использовании ресурсов в процессе разработки (создаются менеджерами, служат для отчета перед заказчиком, исследуются при аудите);
- стандарты (предписывают принципы, правила, соглашения, регламентирующие процесс разработки, могут быть как международными или национальными, так и специально созданными для организации, в которой ведется разработка);
- рабочие документы (являются основными техническими документами, обеспечивающими связь между разработчиками, содержат идеи и проблемы, возникающие в процессе разработки, описание используемых стратегий и подходов, а также промежуточные рабочие версии документов, которые должны войти в состав программного средства);

– заметки и переписка (в них фиксируются различные детали взаимодействия между менеджерами и разработчиками).

Документы, входящие в состав программного средства, описывают программы этого программного средства как с точки зрения их применения пользователями, так и с точки зрения их разработчиков и сопроводителей. Эти документы будут использоваться не только на стадии эксплуатации (в ее фазах применения и сопровождения), но и на стадии разработки для управления процессом разработки (вместе с рабочими документами) – во всяком случае, они должны быть проверены (протестированы) на соответствие программам. Эти документы образуют два комплекта с разным назначением:

- пользовательская документация ПД (П-документация);
- документация по сопровождению ПС (С-документация).

Пользовательская документация программных средств.

Пользовательская документация (User Documentation) объясняет пользователям, как они должны действовать, чтобы применить разрабатываемое программное средство. Она необходима, если программное средство предполагает какое-либо взаимодействие с пользователями. К такой документации относятся документы, которыми должен руководствоваться пользователь при инсталляции (при установке программ с соответствующей настройкой на среду применения), при применении программ для решения своих задач и при управлении программным средством (например, когда программное средство будет взаимодействовать с другими системами). Эти документы частично затрагивают вопросы сопровождения, но не касаются вопросов, связанных с модификацией программ.

В связи с этим следует различать две категории пользователей: ординарных пользователей и администраторов программного средства.

Ординарный пользователь (end-user) использует программное средство для решения своих задач (в своей предметной области). Это может быть инженер, проектирующий техническое устройство, или кассир, продающий желез-

нодорожные билеты с помощью программного средства. Он может и не знать многих деталей работы компьютера или принципов программирования.

Администратор программного средства (System Administrator) управляет использованием средства ординарными пользователями и осуществляет сопровождение программного средства, не связанное с модификацией программ. Например, он может управлять правами доступа ординарных пользователей к программному средству, поддерживать связь с поставщиками программного средства или выполнять определенные действия, чтобы поддерживать программное средство в рабочем состоянии.

Состав пользовательской документации зависит от аудиторий пользователей, на которые ориентировано программное средство, и от режима использования документов. Под аудиторией здесь понимается контингент пользователей, у которого есть необходимость в пользовательской документации. Удачный пользовательский документ существенно зависит от точного определения аудитории, для которой он предназначен. Пользовательская документация должна содержать информацию, необходимую для каждой аудитории. Под режимом использования документа понимается способ, определяющий, каким образом используется этот документ. Обычно пользователю достаточно больших программных систем требуются либо документы для изучения программного средства (использование в виде инструкции), либо для уточнения некоторой информации (использование в виде справочника).

Типичным является следующий состав пользовательской документации:

- общее функциональное описание (дает краткую характеристику функциональных возможностей программного средства, предназначено для пользователей, которые должны решить, насколько необходимо им данное средство);
- руководство по инсталляции (предназначено для администраторов, должно детально предписывать, как устанавливать системы в конкретной среде, в частности, должно содержать описание носителей данных, на которые поставляется программное средство, описание файлов, представляющих программное средство, и требования к минимальной конфигурации аппаратуры);

– инструкция по применению (предназначена для ординарных пользователей, содержит необходимую информацию по применению программного средства, организованную в форме удобной для ее изучения);

– справочник по применению (предназначен для ординарных пользователей, содержит необходимую информацию по применению программного средства, организованную в форме удобной для избирательного поиска отдельных сведений);

– руководство по управлению программным средством (предназначено для администраторов, содержит рекомендации по поддержанию программного средства в рабочем состоянии и восстановлению его работоспособности после сбоев).

Документация по сопровождению программных средств (System Documentation) описывает программное средство с точки зрения его разработки. Эта документация необходима на стадии сопровождения программного средства. В случае необходимости модернизации программного средства к этой работе привлекается специальная команда разработчиков-сопроводителей. Этой команде придется иметь дело с такой же документацией, которая определяла деятельность команды первоначальных (основных) разработчиков, с той лишь разницей, что эта документация для команды разработчиков-сопроводителей будет, как правило, чужой (она создавалась другой командой). Чтобы понять строение и процесс разработки модернизируемого программного средства, команда разработчиков-сопроводителей должна изучить эту документацию, а затем внести в нее необходимые изменения, повторяя в значительной степени технологические процессы, с помощью которых создавалось первоначальное ПС.

Документация по сопровождению ПС можно разбить на две группы:

– документацию, определяющую строение программ и структур данных ПС и технологию их разработки;

– документацию, помогающую вносить изменения в ПС.

Документация первой группы содержит итоговые документы каждого технологического этапа разработки ПС:

- внешнее описание (Requirements Document);
- описание архитектуры (Description of the System Architecture), включая внешнюю спецификацию каждой ее программы (подсистемы);
- описание модульной структуры каждой программы, включая внешнюю спецификацию каждого модуля;
- спецификацию и описание строения каждого модуля (Design Description);
- исходные тексты модулей (Program Source Code Listings);
- документы для валидации программного средства.

Документы для валидации программного средства включают в себя документацию по тестированию (схема тестирования и описание комплекта тестов), а также могут включать и результаты других видов проверки ПС, например доказательства свойств программ. Для обеспечения приемлемого качества этой документации полезно следовать общепринятым рекомендациям и стандартам.

Документация второй группы представляет собой руководство по сопровождению (System Maintenance Guide), которое описывает особенности реализации (в частности, трудности, которые пришлось преодолевать) и как учтены возможности развития программного средства в его строении (конструкции). В нем также фиксируются, какие части ПС являются аппаратно- и программно-зависимыми.

Требования к содержанию документов, разрабатываемых при создании АС, установлены указаниями РД 50-34.698-90 (Руководящий документ по стандартизации. Методические указания. Информационная технология. Комплекс стандартов и руководящих документов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов), а также

соответствующими государственными стандартами Единой системы программной документации (ЕСПД) ГОСТ 19, Единой системы конструкторской документации (ЕСКД) ГОСТ 2.

Содержание документов, разрабатываемых на предпроектных стадиях.

1. Стадия «Формирование требований к АС».

1.1. На стадии разрабатывают отчет по ГОСТ 7.32 и заявку на разработку АС.

1.2. Основная часть отчета содержит разделы:

- 1) характеристика объекта и результатов его функционирования;
- 2) описание существующей информационной системы;
- 3) описание недостатков существующей информационной системы;
- 4) обоснование необходимости совершенствования информационной системы объекта;
- 5) цели, критерии и ограничения создания АС;
- 6) функции и задачи создаваемой АС;
- 7) выводы и предложения.

1.3. В разделе «Характеристика объекта и результатов его функционирования» описывают тенденции развития, требования к объему, номенклатуре и качеству результатов функционирования, а также характер взаимодействия объекта с внешней средой.

При выявлении фактических показателей функционирования определяют существующие показатели и тенденции их изменения во времени.

1.4. Раздел «Описание существующей информационной системы» содержит описание функциональной и информационной структуры системы, качественных и количественных характеристик, раскрывающих взаимодействие ее компонентов в процессе функционирования.

1.5. В разделе «Описание недостатков существующей информационной системы» приводят результаты диагностического анализа, при котором оцени-

вают качество функционирования и организационно-технологический уровень системы, выявляют недостатки в организации и технологии функционирования информационных процессов и определяют степень их влияния на качество функционирования системы.

1.6. В разделе «Обоснование необходимости совершенствования информационной системы объекта» при анализе соответствия показателей функционирования объекта предъявляемым требованиям оценивают степень соответствия прогнозируемых показателей требуемым и выявляют необходимость совершенствования информационной системы путем создания АС.

1.7. Раздел «Цели, критерии и ограничения создания АС» содержит:

- 1) формулировку производственно-хозяйственных, научно-технических и экономических целей и критериев создания АС;
- 2) характеристику ограничений по созданию АС.

1.8. Раздел «Функции и задачи создаваемой АС» содержит:

- 1) обоснование выбора перечня автоматизированных функций и комплексов задач с указанием очередности внедрения;
- 2) требования к характеристикам реализации функций и задач в соответствии с действующими нормативно-техническими документами, определяющими общие технические требования к АС конкретного вида;
- 3) дополнительные требования к АС в целом и ее частям, учитывающие специфику создаваемой АС.

1.9. Раздел «Ожидаемые технико-экономические результаты создания АС» содержит:

- 1) перечень основных источников экономической эффективности получаемых в результате создания АС (в том числе – экономия производственных ресурсов, улучшения качества продукции, повышений производительности труда и т.д.) и оценку ожидаемых изменений основных технико-экономических

и социальных показателей производственно-хозяйственной деятельности объекта (например, показателей по номенклатуре и объемам производства, себестоимости продукции, рентабельности, отчислениям в фонды экономического стимулирования, уровни социального развития);

2) оценку ожидаемых затрат на создание и эксплуатацию АС с распределением их по очередям создания АС и по годам;

3) ожидаемые обобщающие показатели экономической эффективности АС.

1.10. Раздел «Выводы и предложения» рекомендуется разделять на подразделы:

1) выводы о производственно-хозяйственной необходимости и технико-экономической целесообразности создания АС;

2) предложения по совершенствованию организации и технологии процесса деятельности;

3) рекомендации по созданию АС.

1.11. Подраздел «Выводы о производственно-хозяйственной необходимости и технико-экономической целесообразности создания АС» содержит:

1) сопоставление ожидаемых результатов создания АС с заданными целями и критериями создания АС (по целевым показателям и нормативным требованиям);

2) принципиальное решение вопроса о создании АС (положительное или отрицательное).

1.12. Подраздел «Предложения по совершенствованию организации и технологии процесса деятельности» содержит предложения по совершенствованию:

1) производственно-хозяйственной деятельности;

2) организационной и функциональной структур системы, методов деятельности, видов обеспечения АС.

1.13. Подраздел «Рекомендации по созданию АС» содержит рекомендации по:

- 1) виду создаваемой АС, ее совместимости с другими АС и неавтоматизируемой частью соответствующей системы;
- 2) организационной и функциональной структуре создаваемой АС;
- 3) составу и характеристикам подсистем и видов обеспечения АС;
- 4) организации использования имеющихся и приобретению дополнительных средств вычислительной техники;
- 5) рациональной организации разработки и внедрения АС;
- 6) определению основных и дополнительных, внешних и внутренних источников и видов объемов финансирования и материального обеспечения разработок АС;
- 7) обеспечению производственных условий создания АС;
- 8) другие рекомендации по созданию АС.

1.14. Заявка на разработку АС составляется в произвольной форме и содержит предложения организации-пользователя к организации-разработчику на проведение работ по созданию АС и его требования к системе, условия и ресурсы на создание АС.

2. Стадия «Разработка концепции АС».

2.1. На стадии разрабатывают отчет по ГОСТ 7.32.

2.2. В основной части отчета приводят:

- 1) описание результатов изучения объекта автоматизации;
- 2) описание и оценку преимуществ и недостатков разработанных альтернативных вариантов концепции создания АС;
- 3) сопоставительный анализ требований пользователя к АС и вариантов концепции АС на предмет удовлетворения требованиям пользователя;
- 4) обоснование выбора оптимального варианта концепции и описание предлагаемой АС;

5) ожидаемые результаты и эффективность реализации выбранного варианта концепции АС;

6) ориентировочный план реализации выбранного варианта концепции АС;

7) необходимые затраты ресурсов на разработку, ввод в действие и обеспечение функционирования;

8) требования, гарантирующие качество АС;

9) условия приемки системы.

Содержание организационно-распорядительных документов:

1. Акт завершения работ.

Документ содержит:

1) наименование завершенной работы (работ);

2) список представителей организации-разработчика и организации-заказчика, составивших акт;

3) дату завершения работ;

4) наименование документа(ов), на основании которого(ых) проводилась работа;

5) основные результаты завершенной работы;

6) заключение о результатах завершенной работы.

2. Акт приемки в опытную эксплуатацию.

Документ содержит:

1) наименование АС (или ее части), принимаемой в опытную эксплуатацию, и соответствующего объекта автоматизации;

2) наименование документа, на основании которого разработана АС;

3) состав приемочной комиссии и основание для ее работы (наименование, номер и дату утверждения документа, на основании которого создана комиссия);

- 4) период времени работы комиссии;
- 5) наименование организации-разработчика, организации-соисполнителя и организации-заказчика;
- 6) состав функций АС (или ее части), принимаемых в опытную эксплуатацию;
- 7) перечень составляющих технического, программного, информационного и организационного обеспечений, проверяемых в процессе опытной эксплуатации;
- 8) перечень документов, предъявляемых комиссии;
- 9) оценку соответствия принимаемой АС техническому заданию на ее создание;
- 10) основные результаты приемки в опытную эксплуатацию;
- 11) решение комиссии о принятии АС в опытную эксплуатацию.

3. Акт приемки в промышленную эксплуатацию.

3.1. Документ содержит:

- 1) наименование объекта автоматизации и АС (или ее части), принимаемой в промышленную эксплуатацию;
- 2) сведения о статусе приемочной комиссии (государственная, межведомственная, ведомственная), ее составе и основание для работы;
- 3) период времени работы комиссии;
- 4) наименование организации-разработчика, организации-соисполнителя и организации-заказчика;
- 5) наименование документа, на основании которого разработана АС;
- 6) состав функций АС (или ее части), принимаемой в промышленную эксплуатацию;
- 7) перечень составляющих технического, программного, информационного и организационного обеспечений, принимаемых в промышленную эксплуатацию;

8) список ответственных представителей организаций, выполняющих наладочные работы;

9) указания о порядке устранения ошибок монтажа и лицах, ответственных за выполнение этих работ.

4. Документ «Приказ о начале опытной эксплуатации АС (ее частей)» содержит:

1) наименование АС в целом или ее частей, проходящей опытную эксплуатацию;

2) наименование организации разработчика, организаций-соисполнителей;

3) сроки проведения опытной эксплуатации;

4) список должностных лиц организации-заказчика и организации-разработчика, ответственных за проведение опытной эксплуатации;

5) перечень подразделений организации-заказчика, участвующих в проведении опытной эксплуатации.

5. Документ «Приказ о вводе в промышленную эксплуатацию АС (ее частей)» должен содержать:

1) состав функций АС или ее частей, технических и программных средств, принимаемых в промышленную эксплуатацию;

2) список должностных лиц и перечень подразделений организации-заказчика, ответственных за работу АС;

3) порядок и сроки введения новых форм документов (при необходимости);

4) порядок и сроки перевода персонала на работу в условиях функционирования АС.

6. Приказ о составе приемочной комиссии.

6.1. Документ содержит:

1) наименование принимаемой АС в целом или ее частей;

2) сведения о составе комиссии;

3) основание для организации комиссии;

- 4) наименование организации-заказчика;
- 5) наименование организации-разработчика, организаций-соисполнителей;
- 6) назначение и цели работы комиссии;
- 7) сроки начала завершения работы комиссии;
- 8) указание о форме завершения работы комиссии.

7. Протокол испытаний.

7.1. Документ содержит:

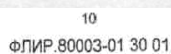
- 1) наименование объекта испытаний;
- 2) список должностных лиц, проводивших испытания;
- 3) цель испытаний;
- 4) сведения о продолжительности испытаний;
- 5) перечень пунктов технического задания на создание АС, на соответствие которым проведены испытания;
- 6) перечень пунктов «Программы испытаний», по которым проведены испытания;
- 7) сведения о результатах наблюдений за правильностью функционирования АС;
- 8) сведения об отказах, сбоях и аварийных ситуациях, возникающих при испытаниях;
- 9) сведения о корректировках параметров объекта испытания и технической документации.

8. Протокол согласования.

8.1. Документ содержит:

- 1) перечень рассмотренных отклонений с указанием документа, отклонения от требований которого являются предметом согласования;
- 2) перечень должностных лиц, составивших протокол;
- 3) обоснование принятых отклонений от проектных решений;
- 4) перечень согласованных отклонений и сроки внесения необходимых изменений в техническую документацию.

Шаблон формуляра на одиночную ПЭВМ приведен на рис. 10.3 и 10.4.



4. КОМПЛЕКТНОСТЬ

4.3. Комплектность поставки приведена в таблице 1.

Таблица 1

Обозначение	Наименование	Кол.	Зав. №	Примечание
ФЛИР.80003-01	КП «ОФИС»	1		Компакт-диск с загрузочным модулем
	Комплект эксплуатационной документации согласно ФЛИР.80003-01 20 01	1		Книги

2
ФЛИР.80003-01 30 01

СОДЕРЖАНИЕ

1. Общие указания.....	3
2. Общие сведения.....	4
3. Основные характеристики	5
4. Комплектность	10
5. Периодический контроль основных характеристик при эксплуатации и хранении ..	11
6. Свидетельство о приемке.....	12
7. Свидетельство об упаковке	14
8. Гарантийные обязательства.....	15
9. Сведения о рекламациях.....	16
10. Сведения о хранении	17
11. Сведения о закреплении при эксплуатации	18
12. Сведения об установке программного средства на ЭВМ.....	20
13. Особые отметки	21

1. ОБЩИЕ УКАЗАНИЯ

1.2. Ввод в эксплуатацию КП «ОФИС» проводится в соответствии с настоящим формуляром.

1.4. Формуляр должен находиться у ответственного должностного лица (администратора службы безопасности информации). Все записи в формуляре должны производиться только черными чернилами отчетливо и аккуратно. Подчистки, помарки и незаверенные исправления НЕ ДОПУСКАЮТСЯ.

Рис. 10.2. Формуляр

"УТВЕРЖДАЮ"

" ____ " _____ 200 ____ г.

**МАШИНА ВЫЧИСЛИТЕЛЬНАЯ
ЭЛЕКТРОННАЯ ПЕРСОНАЛЬНАЯ**

тип: IBM PC
учетный номер _____

ФОРМУЛЯР

5. УСЛОВИЯ ЭКСПЛУАТАЦИИ

ПЭВМ может
являться:

- температура
- атмосферное
- относительная влажность
- содержание пыли в воздухе не более 0.75 мг/м³;
- электрическая составляющая электромагнитного поля помех не должна превышать 0.3 В/м в диапазоне частот от 0.15 до 300 МГц.

6. ОСОБЕННОСТИ ЭКСПЛУАТАЦИИ

ВНИМАНИЕ! КАТЕГОРИЧЕСКИ ЗАПРЕЩАЕТСЯ:

- использование техники несоответственно её назначению;
- эксплуатация ПЭВМ и периферийного оборудования при нестабильном электропитании;
- форматирование жесткого диска форматом низкого уровня;
- изменение конфигурации;
- включение/выключение соединительных проводов при включенном электропитании;
- использование магнитных носителей, приведшее к заражению программой-вирусом;
- эксплуатация ПЭВМ без заземления;
- включение/выключение ПЭВМ с интервалом менее 1 мин.;
- эксплуатация техники при обнаружении механических повреждений.

7. ЗАКРЕПЛЕНИЕ ЗА ЛИЧНЫМ СОСТАВОМ

№ п/п	Основание	Дата	В/звание	Должность	Фамилия И.О.	Роспись

УЧЕТ НЕИСПРАВНОСТЕЙ ПРИ ЭКСПЛУАТАЦИИ

Дата и время появления или его состав- ной части	Характер (внеш- нее проявление) неисправности	Причина неис- правности (отка- за), количество часов работы отказавшего элемента изде- лия	Принятые меры по устранению неисправности	Должность, фамилия и под- пись лица, от- ветственного за устранение не- исправности

9. СВЕДЕНИЯ О РЕМОНТЕ ИЗДЕЛИЯ

Наимено- вание и обозначе- ние со- ставной части изделия	Основание для проведе- ния ремонта	Дата		Наименование ремонтных работ	Должность, фамилия и подпись ответственно- го лица	
		посту- пления в ре- монт	выко- ла из ре- монта		принимает: директор ремонт	принимает: начальник ремонта

Рис. 10.3. Шаблон формуляра на одиночную ПЭВМ

10. СВЕДЕНИЯ О ЗАМЕНЕ СОСТАВНЫХ ЧАСТЕЙ ИЗДЕЛИЯ ЗА ВРЕМЯ ЭКСПЛУАТАЦИИ

Заменяемая часть				Вновь установленная часть		Дата, должность, фамилия и подпись лица ответственного за эксплуатацию
Наименование и обозначение	Учётный номер	Наработка	Причина выхода из строя	Наименование и обозначение	Учётный номер	ПЭВМ

11. СВЕДЕНИЯ О ХРАНЕНИИ ИЗДЕЛИЯ

Дата		Условия хранения	Должность, фамилия и подпись лица, ответственного за хранение	
установки на хранение	снятия с хранения			

3. КОМПЛЕКТАЦИЯ КОМПЬЮТЕРА

Учётный номер _____

№ п/п	Тип	Название	Номер	Прим.
1	Корпус			
2	Блок питания			
3	Материнская плата			
4	Процессор			
5	Модуль памяти			
6	Видео карта			
7	Монитор			
8	Жесткий диск			
9	Дисковод			
10	Дисковод			
11	Клавиатура			
12	Мышь			
13	Звуковая карта			
14	Сетевая карта			
15	Дисковод компакт дисков (CD)			
16	Звуковые колонки			
17	Микрофон			
18				
19				
20				
21				

4. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ



Наименование техн. характеристики	Номинал	Действительное значение	Примечание

2. ОБЩИЕ СВЕДЕНИЯ ОБ ИЗДЕЛИИ

Наименование изделия "Машина вычислительная электронная персональная" (ПЭВМ)

Тип ПЭВМ: IBM PC

Учётный номер _____

Дата выпуска: _____

Предприятие изготовитель: _____

12. ОСОБЫЕ ОТМЕТКИ

Рис. 10.3. Продолжение

На рисунке 10.4 приведен шаблон программы и методики испытаний, на рис. 10.5 – шаблон оформления оптического диска, а на рис. 10.6 – пример оформления оптического диска.

УТВЕРЖДЕН	
XXXX.00614-01 51 01-ЛУ	

Стор. №	
---------	--

Изм. №	
--------	--

Изд. №	
--------	--

Изд. №	
--------	--

2
XXXX.00614-01 51 01

АННОТАЦИЯ

Настоящий документ «Программный комплекс «Формирование СПД». Программа и методика испытаний» XXXX.00614-01 51 01 распространяется на программный комплекс «Формирование сетеориентированного пространства данных» XXXX.00614-01 (далее – ПК «Формирование СПД», программное издание), разработанный ММММММ по Договору с ОАО «НННННН» № ИСТ-28/2011 от 20.10.2011 г. на выполнение основной части опытно-конструкторской работы (СЧ ОКР) «УУУУУУУУУУУУ».

Настоящий документ содержит программу и методики проверок ПК «Формирование СПД» и предназначен для проведения предварительных испытаний опытного образца ПК «Формирование СПД» на соответствие требованиям технического задания (далее – ТЗ) на СЧ ОКР «УУУУУУУУУУУУ» и дополнений № 1 и № 2 к ТЗ.

Настоящий документ разработан в соответствии с ГОСТ 19.301-79.

ПРОГРАММНЫЙ КОМПЛЕКС
«ФОРМИРОВАНИЕ СПД»
Программа и методика испытаний
XXXX.00614-01 51 01
Листов 172

Рис. 10.4. Шаблон программы и методики испытаний

УТВЕРЖДЕН
КОНУС.00001-01 90 01 ЛУ

Макет
программного изделия КОНУС 00001-01

Этикетка
КОНУС.00001-01 90 01

Тамбов
КОНУС-ИТ
20...

Макет программного изделия КОНУС.00001-01 поставляется на одном оптическом компакт-диске и может быть использован в сетевом однопользовательском режиме.

Для установки макета программного изделия выполните следующие действия: в терминале пользователя наберите и выполните команду `dpkg -i` путь до пакета/ifs_calc.deb

Авторы: ООО «КОНУС-ИТ», г. Тамбов

КОНУС.00001-01 90 01
© АО «НИИ ТП»

Макет программного изделия КОНУС.00001-01 поставляется в комплекте:

- оптический компакт-диск с инсталляционным пакетом КОНУС.00001-01;
- оптический компакт-диск с текстом программы КОНУС.00001-01 12 01;
- руководство системы программиста КОНУС.00001-01 32 01;
- руководство оператора КОНУС.00001-01 34 01;
- программа и методика испытаний КОНУС.00001-01 51 01;
- пояснительная записка КОНУС.00001-01 81 01;
- этикетка КОНУС.00001-01 90 01.

Контрольная сумма: _____

392000, Россия, Тамбов
Ул.
тел:
E-mail:|

Макет
программного изделия
КОНУС.00001-01



© АО «НИИ ТП»

Тамбов
КОНУС-ИТ
201...

Рис. 10.5. Шаблон оформления оптического диска



Рис. 10.6. Пример оформления оптического диска

ЗАКЛЮЧЕНИЕ

В пособии изложен теоретический материал с примерами реализации, что позволит студентам получить навыки управления ИТ-проектами.

Управленческие процессы планирования, организации, анализа, регулирования, контроля, учета и др. рассматриваются как виды деятельности конкретных работников, технических и информационных систем.

Модель жизненного цикла по стандарту ISO/IEC 12207 определяет последовательность выполнения и взаимосвязи процессов, действий и задач на протяжении ЖЦ. В пособии сравниваются научные исследования с информационно-технологическими (ИТ) проектами.

В сентябре–октябре 2014 г. приказом Министерства труда и социальной защиты Российской Федерации определены профессиональные стандарты «Менеджер по информационным технологиям», «Руководитель проектов в области информационных технологий», «Руководитель разработки программного обеспечения». В пособии рассмотрены требования к квалификации специалистов в области управления ИТ-проектами.

Подробно рассмотрены процессы управления проектами, вопросы технико-экономического анализа разработки проектов программных средств, методы оценки трудозатрат на разработку программной системы.

В пособии приведен материал о библиотеке ITIL, ITSM и информационном менеджменте, модели системы обеспечения участников ИТ-проекта информацией, документировании программных средств в ИТ-проекте, выборе модели жизненного цикла для ИТ-проекта.

СПИСОК ЛИТЕРАТУРЫ

1. **Методы** анализа информационных систем : монография / Ю. Ю. Громов, В. Е. Дидрих, М. А. Ивановский и др. – Тамбов ; М. ; СПб. ; Баку ; Вена ; Гамбург : Изд-во МИНЦ «Нобелистика». – 2012.
2. **Свидетельство** о государственной регистрации программ для ЭВМ № 2023618463 от 25 апреля 2023 г.
3. **Теория** информационных процессов и систем : учебник / Ю. Ю. Громов, В. Е. Дидрих, О. Г. Иванова, В. Г. Однолько. – Тамбов : Изд-во ФГБОУ ВПО «ТГТУ», 2014.
4. **ГОСТ РД 50-34.698–90.** Автоматизированные системы. Требования к содержанию документов.
5. **ГОСТ Р 50-34.126–92.** Правила проведения работ при создании информационных систем.
6. **ГОСТ 28195–1989.** Оценка качества программных средств. Общие положения и ГОСТ Р ИСО/МЭК 9126–1993 Информационная технология. Оценка программной продукции. Характеристики качества и руководство по их применению.
7. **Грекул, В. И.** Проектирование информационных систем. Курс лекций : учебное пособие / В. И. Грекул, Г. Н. Денищенко, Н. Л. Коровкина. – М. : Интернет-университет ИТ, 2005.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	3
1. УПРАВЛЕНЧЕСКИЕ ПРОЦЕССЫ.....	4
2. МОДЕЛЬ ЖИЗНЕННОГО ЦИКЛА ПО СТАНДАРТУ ISO/IEC 12207.....	5
3. НАУЧНЫЕ ИССЛЕДОВАНИЯ И ИНФОРМАЦИОННО-ТЕХНОЛОГИЧЕСКИЕ (ИТ) ПРОЕКТЫ	8
4. ТРЕБОВАНИЯ МИНИСТЕРСТВА ТРУДА И СОЦИАЛЬНОЙ ЗАЩИТЫ РОССИЙСКОЙ ФЕДЕРАЦИИ К КВАЛИФИКАЦИИ СПЕЦИАЛИСТОВ В ОБЛАСТИ УПРАВЛЕНИЯ ИТ-ПРОЕКТАМИ.....	12
5. ПРОЦЕССЫ УПРАВЛЕНИЯ ПРОЕКТАМИ	16
6. ТЕХНИКО-ЭКОНОМИЧЕСКИЙ АНАЛИЗ РАЗРАБОТКИ ПРОЕКТОВ ПРОГРАММНЫХ СРЕДСТВ.....	37
7. МЕТОДЫ ОЦЕНКИ ТРУДОЗАТРАТ НА РАЗРАБОТКУ ПРОГРАММНОЙ СИСТЕМЫ	39
7.1. Прямой метод оценки трудозатрат	39
7.2 Оценка трудозатрат методом функциональных точек	42
8. МОДЕЛЬ СИСТЕМЫ УПРАВЛЕНИЯ ПЕРСОНАЛОМ	50
9. МОДЕЛЬ СИСТЕМЫ ОБЕСПЕЧЕНИЯ УЧАСТНИКОВ ИТ-ПРОЕКТА ИНФОРМАЦИЕЙ.....	86
10. ДОКУМЕНТИРОВАНИЕ ПРОГРАММНЫХ СРЕДСТВ В ИТ-ПРОЕКТЕ	108
ЗАКЛЮЧЕНИЕ	126
СПИСОК ЛИТЕРАТУРЫ.....	127

Учебное электронное издание

ИВАНОВСКИЙ Михаил Андреевич
ГЛАЗКОВА Инга Александровна

УПРАВЛЕНИЕ ИТ-ПРОЕКТАМИ

Учебное пособие

Редактор Л. В. Комбарова
Графический и мультимедийный дизайнер Т. Ю. Зотова
Обложка, упаковка, тиражирование Л. В. Комбаровой

ISBN 978-5-8265-2876-1



Подписано к использованию 20.02.2025.

Тираж 50 шт. Заказ № 29

Издательский центр ФГБОУ ВО «ТГТУ»
392000, г. Тамбов, ул. Советская, д. 106, к. 14
Телефон 8(4752) 63-81-08
E-mail: izdatelstvo@tstu.ru