СОДЕРЖАНИЕ

[1. Введение 5](#_Toc501927127)

[2. Описание предметной области 6](#_Toc501927128)

[3. Диаграммы IDEF0 для модели «Как есть» 9](#_Toc501927129)

[4. Диаграммы IDEF0 для модели «Как должно быть» 14](#_Toc501927130)

[5. Техническое задание на разработку ИС по ГОСТ 34.602-89 17](#_Toc501927131)

[6. Модель информационной системы 27](#_Toc501927132)

[6.1. Диаграммы вариантов использования 27](#_Toc501927133)

[6.2. Диаграммы состояний 30](#_Toc501927134)

[6.3. Диаграммы классов 35](#_Toc501927135)

[6.4. Диаграммы последовательности 37](#_Toc501927136)

[6.5. Диаграмма компонентов 39](#_Toc501927137)

[6.6. Диаграмма развертывания 42](#_Toc501927138)

[7. Заключение 44](#_Toc501927139)

[Библиографический список 45](#_Toc501927140)

# Введение

Проектирование является одним из основных этапов в создании программной системы. Данному этапу уделяется большое внимание так как он лежит у самого истока создания системы. Неверный выбор архитектуры может привести к серьезным проблемам в процессе разработки поэтому каждый разработчик ПО должен владеть данными навыками.

Целью данной курсовой работы является проектирование информационно справочной системы агентства перевозок груза. А также получение навыков самостоятельного проектирования информационной системы и получение опыта работы в среде проектирования Visual Paradigm и Ramus.

# Описание предметной области

Агентство предоставляет клиентам данные об имеющихся рейсах (куда, объем и вес груза, цена, сервис, продолжительность), оформляет договора на грузоперевозку, учитывает работу агентов и выплачивает им зарплату. Бухгалтерия оформляет путевки, учитывает денежные средства, ведет список агентов.

Модуль «АРМ агента перевозок груза»

Каждый договор ведет выделенный сотрудник (агент), который получает комиссионные от суммы договора. Договор может быть заключен на несколько однотипных перевозок. Агент находит клиентов, заключает с ними договора, предоставляет рекламную информацию о перевозках. В конце месяца он составляет отчет о работе для бухгалтерии.

Программное обеспечение АРМ должно позволять -

* + - 1. хранить данные о клиентах, о заключенных с ними договорах, справочники рейсов,
      2. выводить в удобной форме данные по следующим запросам пользователя:
* поиск заданного клиента и его договора по номеру паспорта или номеру договора;
* выборка данных о рейсах в заданный район с ценой в указанных пределах;
* выборка списка клиентов, которые оформили договора на большие объемы товара;
* диаграмма – количество грузоперевозок по районам;
* расчет суммарной стоимости договоров и суммы комиссионных для заданного агента и месяца,
* количество заключенных договоров по районам и месяцам года (перекрестный);
  + - 1. автоматизировать обработку информации при следующих бизнес-операциях:
* заключение договора с новым клиентом (ввод данных о клиенте и договоре, вывод печатной копии договора);
* формирование ежемесячных отчетов о работе;
* коррекция договора с клиентом (изменение данных о количестве грузоперевозок);
* изменение данных о стоимости доставки груза, например, увеличение в связи с погодными условиями или изменение сроков;
* архивация в конце года (удаление в архив данных о выполненных договорах);
  + - 1. выводить следующие данные на печать – договор с клиентом, отчет агента за месяц, список клиентов для заданного агента; диаграмма.
      2. выводить сведения об авторе и назначении программы.

Модуль «АРМ калькулятор расчета»

Модуль позволяет пользователю рассчитать стоимость доставки груза.

Программное обеспечение АРМ должно позволять -

* + - 1. хранить данные о пунктах отправки, об отклонениях от расписания;
      2. выводить в удобной форме данные по следующим запросам пользователя:
* изменение стоимости в зависимости от объема и веса груза;
* изменение стоимости в зависимости от пункта отправления и прибытия;
* расчет времени прибытия и отправления заданного груза с учетом известной задержки;
  + - 1. автоматизировать обработку информации при следующих бизнес-операциях:
* регулярное обновление текущих данных для расчета стоимости груза;
  + - 1. выводить следующие данные на печать – итоговую стоимость перевозки груза.

Модуль «АРМ бухгалтера по учету работы агентов»

Бухгалтер ведет учет работы агентов и начисляет им заработную плату в соответствии с комиссионными.

Программное обеспечение АРМ должно позволять -

* + - 1. хранить данные о заключенных договорах, об агентах и условиях их работы (комиссионные), о начисленной ежемесячно заработной плате;
      2. выводить в удобной форме данные по следующим запросам пользователя:
* поиск информации о заданном агенте по его ФИО или личному коду;
* выборка данных о договорах с сортировкой по агентам и месяцам;
* ежемесячный расчет заработной платы агентов;
* диаграмма – стоимость заключенных договоров по агентам;
* расчет суммарной стоимости заработной платы за год;
* распределение зарплаты агентов по месяцам (перекрестный);
  + - 1. автоматизировать обработку информации при следующих бизнес-операциях:
* ежемесячное формирование, коррекция и вывод ведомости заработной платы;
* проверка отчета агента (сравнение отчета с договорами);
* изменение стоимости перевозки груза по весу или по объему на заданный процент;
* удаление устаревшей информации (ведомости за истекший год архивируются и удаляются из текущей БД);
  + - 1. выводить следующие данные на печать – ведомость заработной платы за месяц; список агентов, контракт с агентом, диаграмма.
      2. выводить сведения об авторе и назначении программы.

АРМ руководителя (мобильное приложение) позволяет:

* получать статистические отчеты по деятельности компании;
* прогнозировать прибыль/убыток на будущий период.
* печать отчетов.

# Диаграммы IDEF0 для модели «Как есть»

IDEF0 - методология функционального моделирования. С помощью наглядного графического языка IDEF0, изучаемая система предстает перед разработчиками и аналитиками в виде набора взаимосвязанных функций (функциональных блоков - в терминах IDEF0). Как правило, моделирование средствами IDEF0 является первым этапом изучения любой системы. [1]

В основе диаграммы IDEF0 лежит принцип декомпозиции. Сложная система разбивается ее более простые составляющие.

На рисунке 1 представлена контекстная диаграмма. Точка зрения: руководитель. Входы: данные о рейсах, данные о клиентах и данные об агентах. Выходы: договор на грузоперевозку, отчет по работе и ведомость о зарплате. Механизмы: клиент, агенты, бухгалтеры и автоматизированная система. Управление: политика компании, цели и задачи компании.

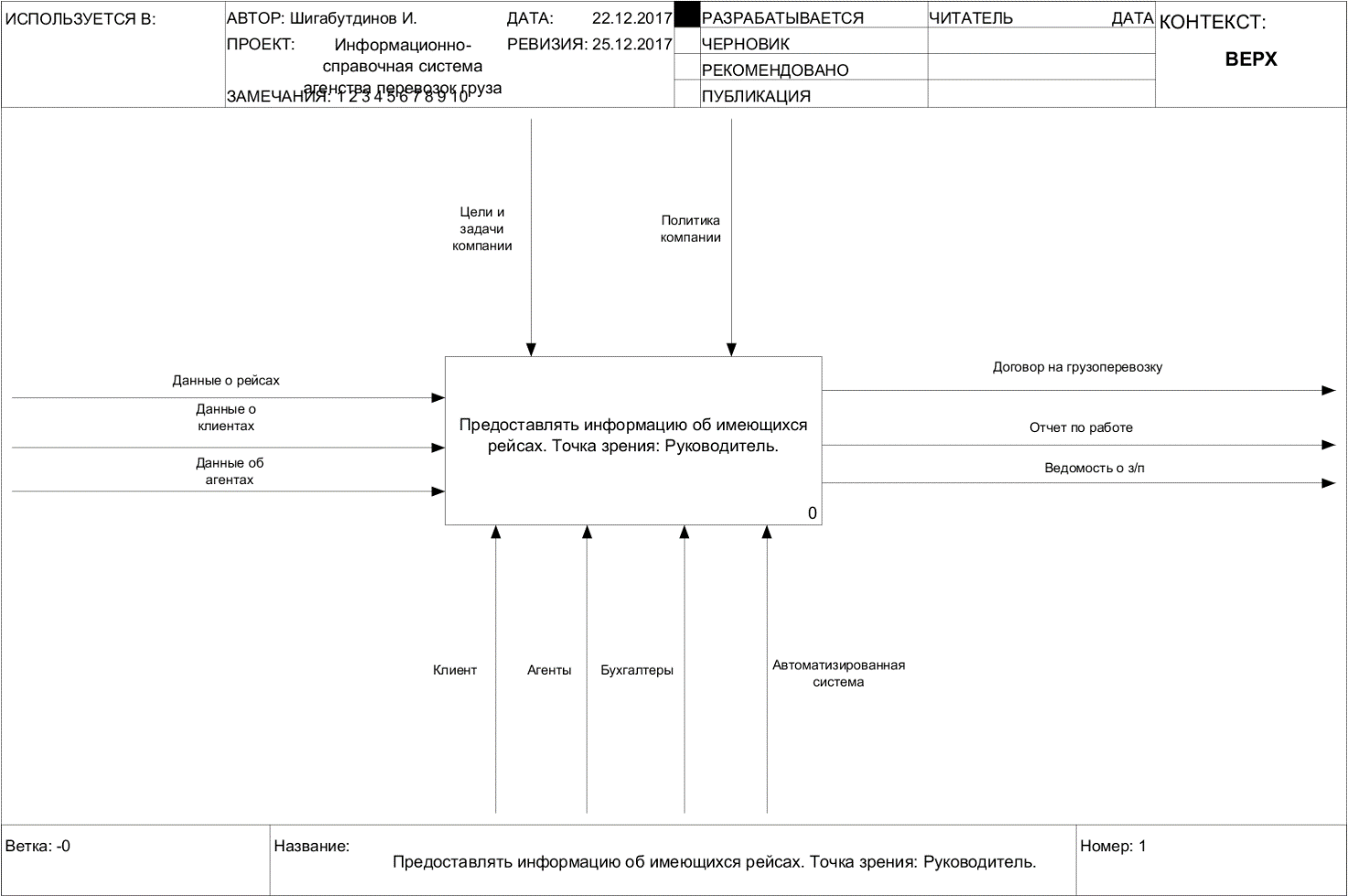


Рисунок 1. Контекстная диаграмма

На рисунке 2 представлена декомпозиция контекстной диаграммы. В ней функциональный блок разделяется на 3 блока: «Формирование заявки», «Заключение договора» и «Работа с отчетами». Клиент формирует заявку и отправляет ее, агент рассматривает заявку и заключает по ней договор, бухгалтер учитывает проделанную работу агента по отчетам.

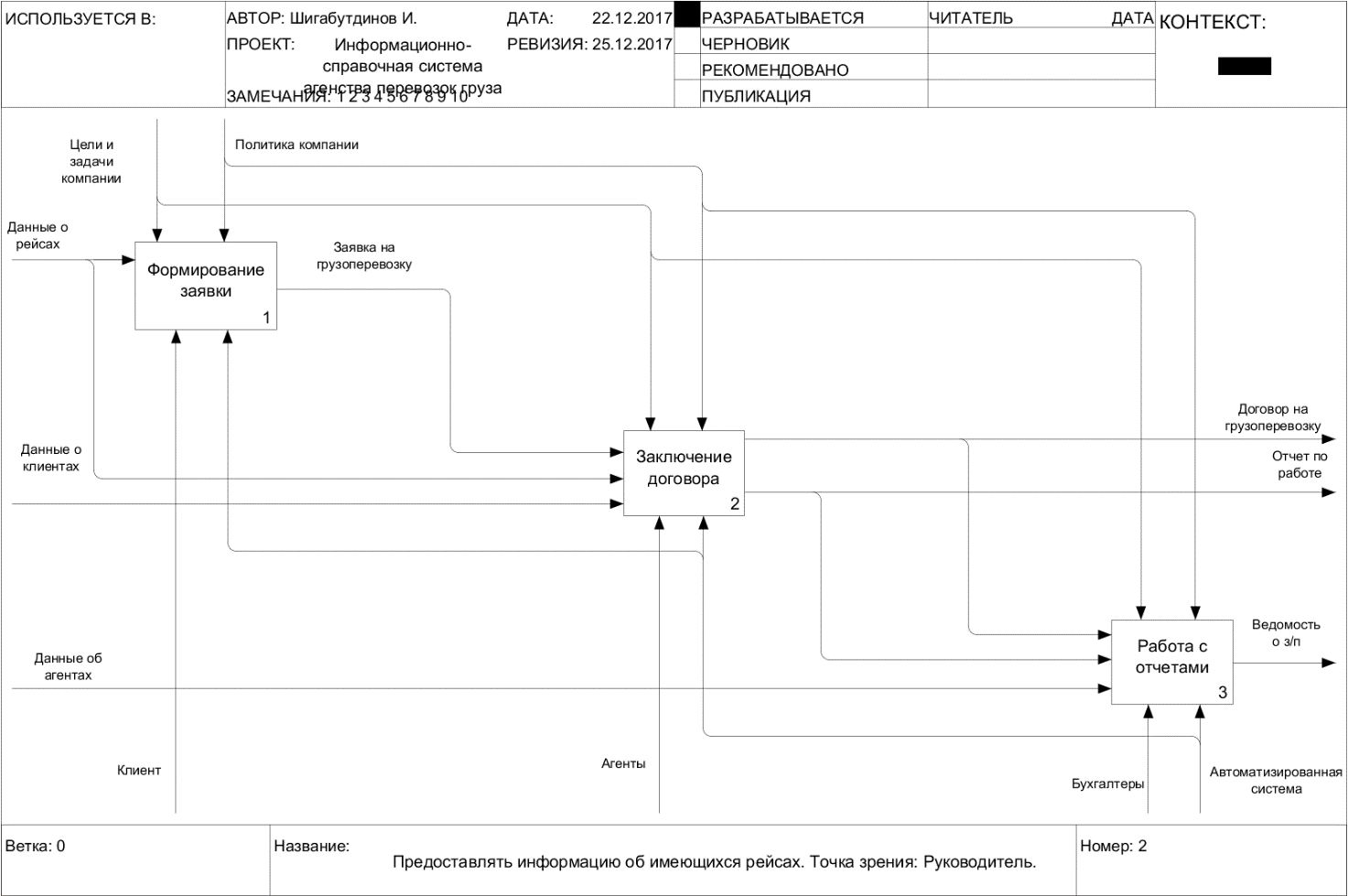


Рисунок 2. Декомпозиция диаграммы IDEF0. Уровень 1

На рисунке 3 представлена декомпозиция функционального блока «Формирование заявки». Данный функциональный блок разбивается на 3 блока: «Предоставление и регулярное обновление информации», «Формирование заявки на грузоперевозку», «Проверка и отправка заявки на грузоперевозку».

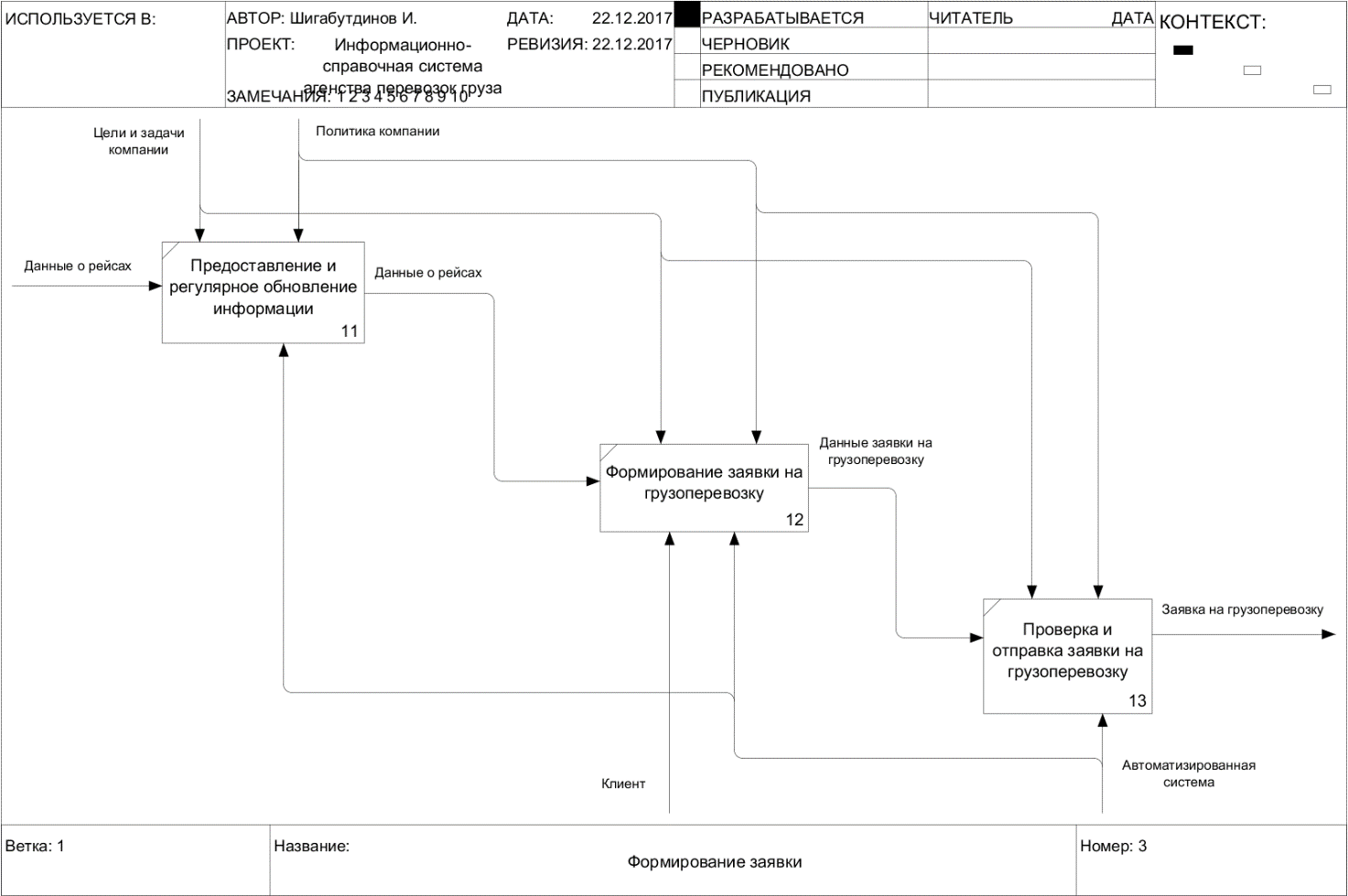


Рисунок 3. Декомпозиция диаграммы IDEF0. Уровень 2

На рисунке 4 представлена декомпозиция функционального блока «Заключение договора». Данный функциональный блок разбивается на 6 блоков: «Работа с клиентами», «Работа с рейсами», «Работа с заявками», «Работа с договорами», «Работа с отчетами» и «Архивация».

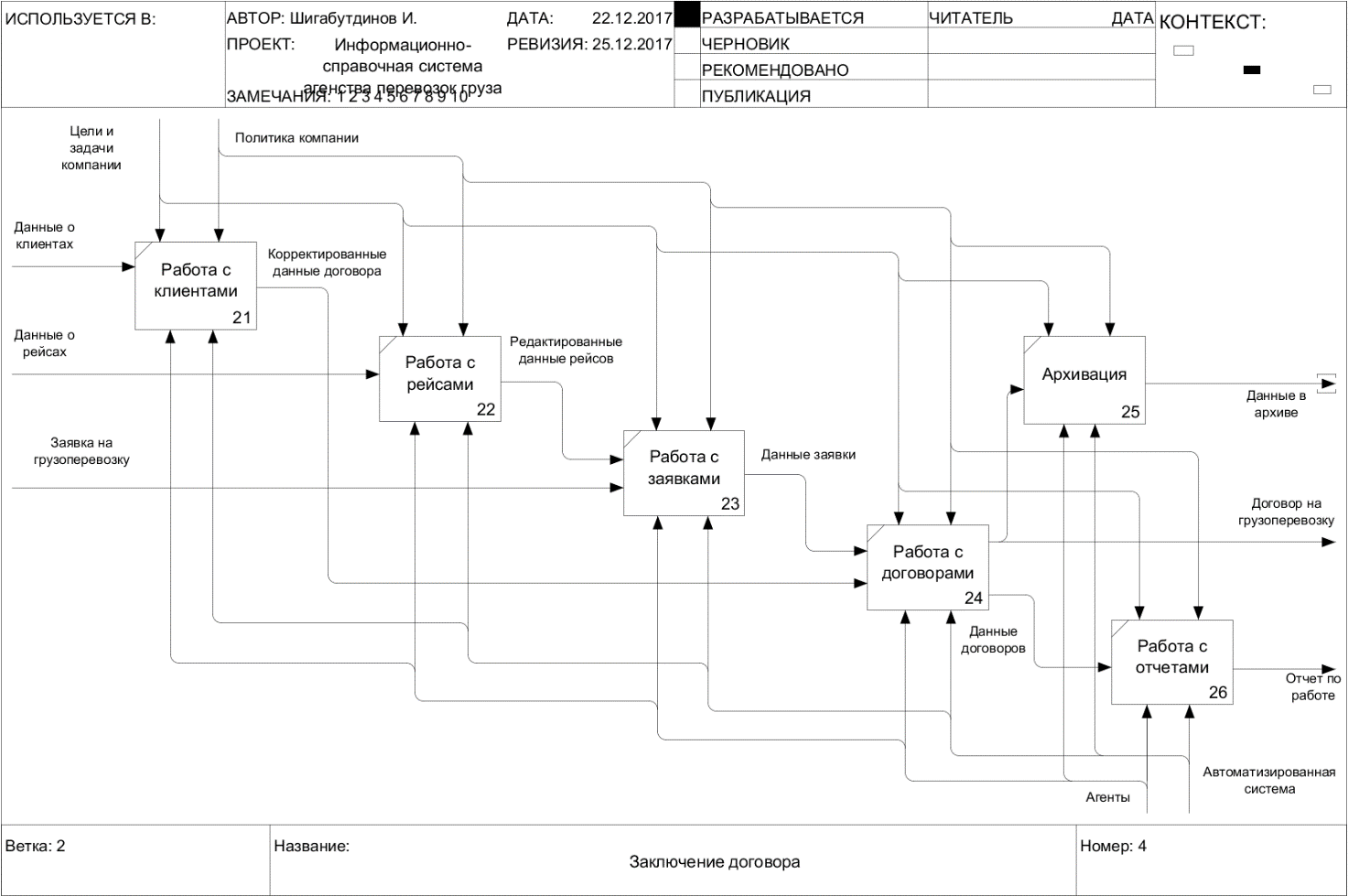


Рисунок 4. Декомпозиция диаграммы IDEF0. Уровень 2

На рисунке 5 представлена декомпозиция функционального блока «Работа с отчетами». Данный функциональный блок разбивается на 5 блоков: «Работа с агентами», «Работа с договорами», «Работа с отчетами», «Работа с ведомостями», и «Архивация».

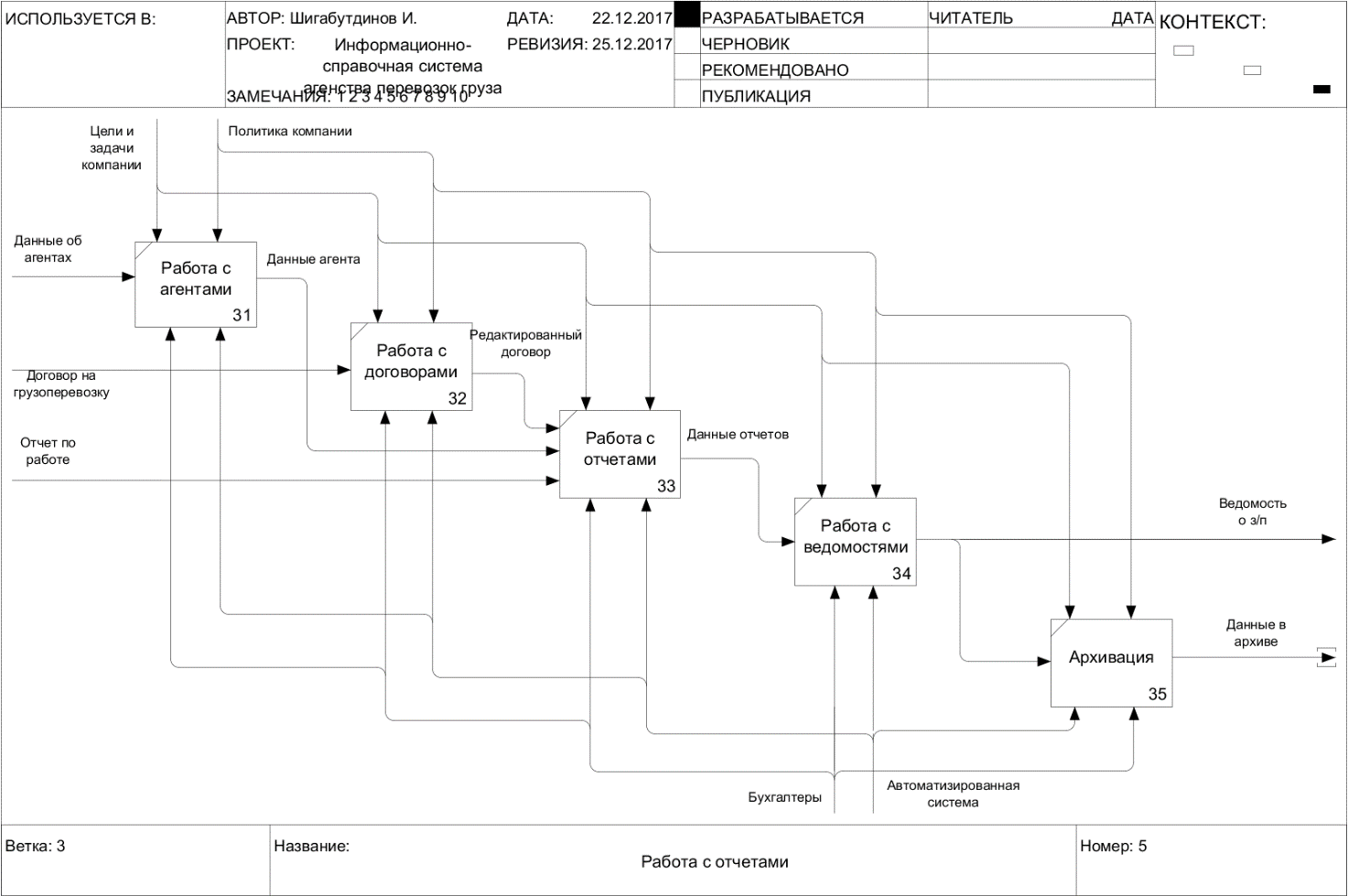


Рисунок 5. Декомпозиция диаграммы IDEF0. Уровень 2

# Диаграммы IDEF0 для модели «Как должно быть»

На рисунке 6 представлена контекстная диаграмма. На диаграмме добавился механизм «Руководитель». Входы: «Запрос статистики» и «Запрос прогноза». Выходы: «Статистические отчеты» и «Прогноз прибыли/убытков».

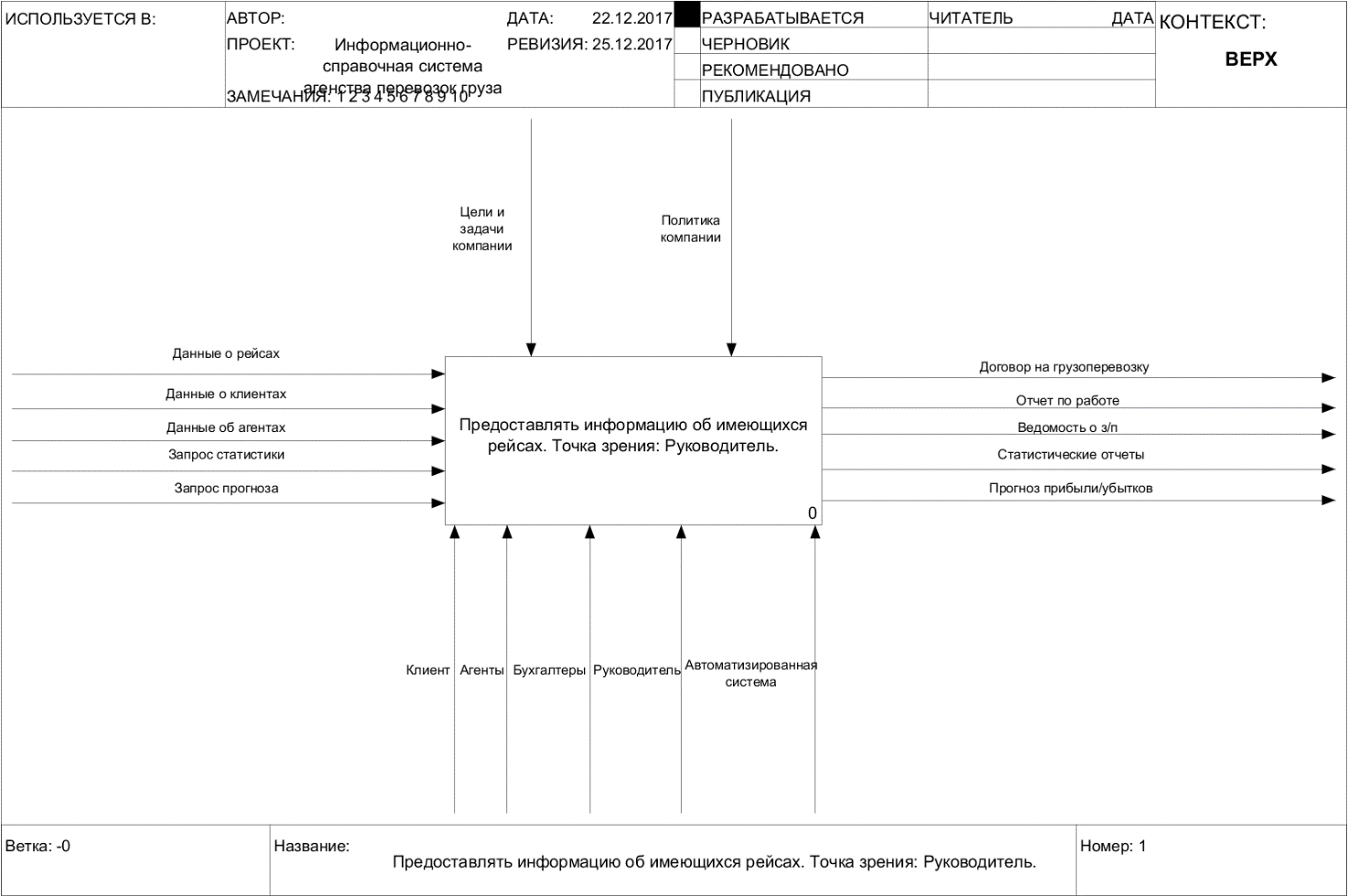


Рисунок 6. Контекстная диаграмма

На рисунке 7 представлена декомпозиция контекстной диаграммы. Данная диаграмма разбивается на 4 функциональных блока: «Формирование заявки», «Заключение договора», «Работа с отчетами» и новый блок – «Прогноз прибыли/убытков».

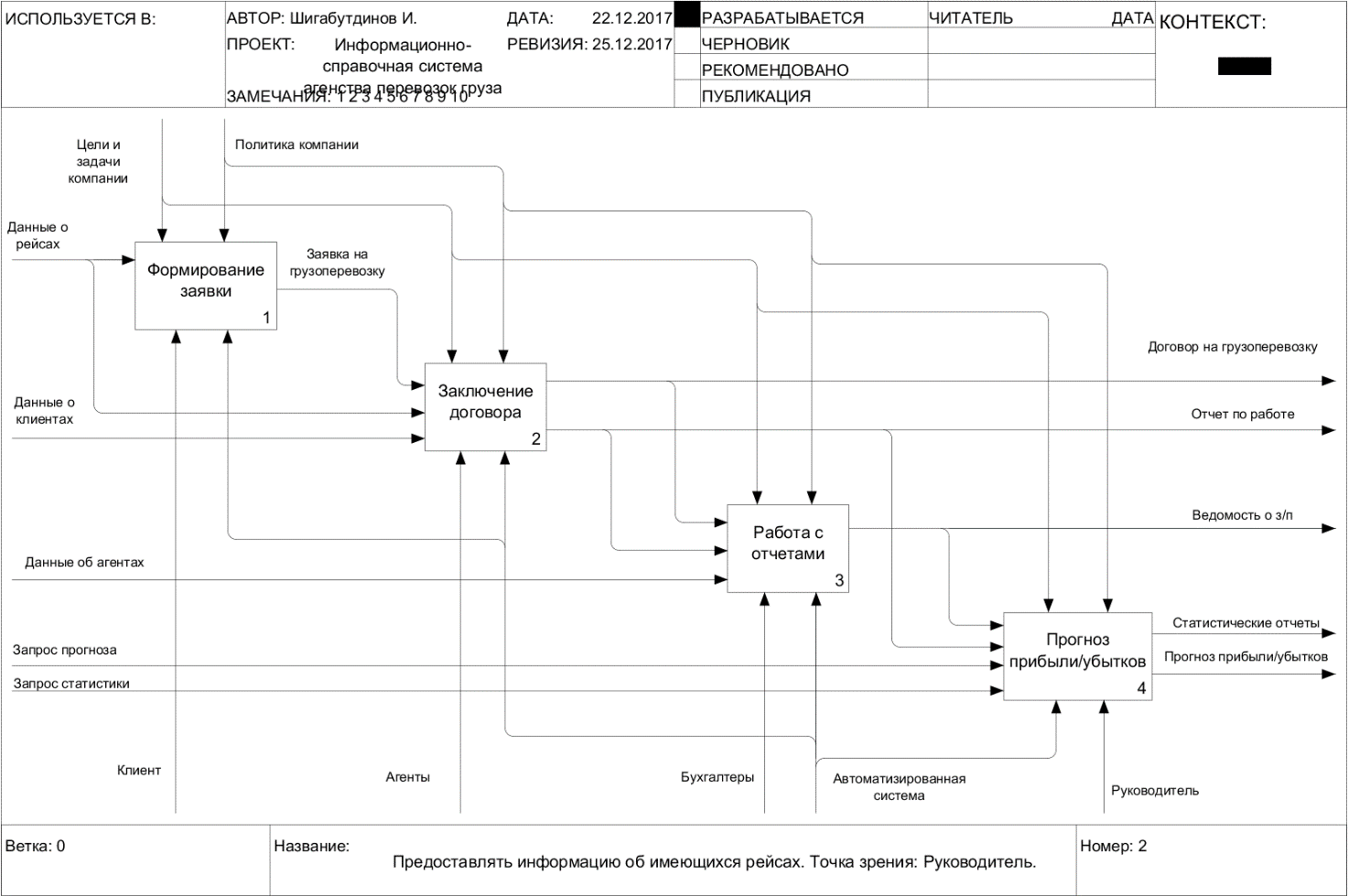


Рисунок 7. Декомпозиция контекстной диаграммы

Декомпозиция функциональных блоков: «Формирование заявки», «Заключение договора» и «Работа с отчетами» была показана выше.

На рисунке 8 представлена декомпозиция функционального блока «Прогноз прибыли/убытков». Данный функциональный блок разбивается на 3 блока: «Формирование отчета о доходах», «Формирование отчета о расходах» и «Прогнозирование прибыли/убытков».

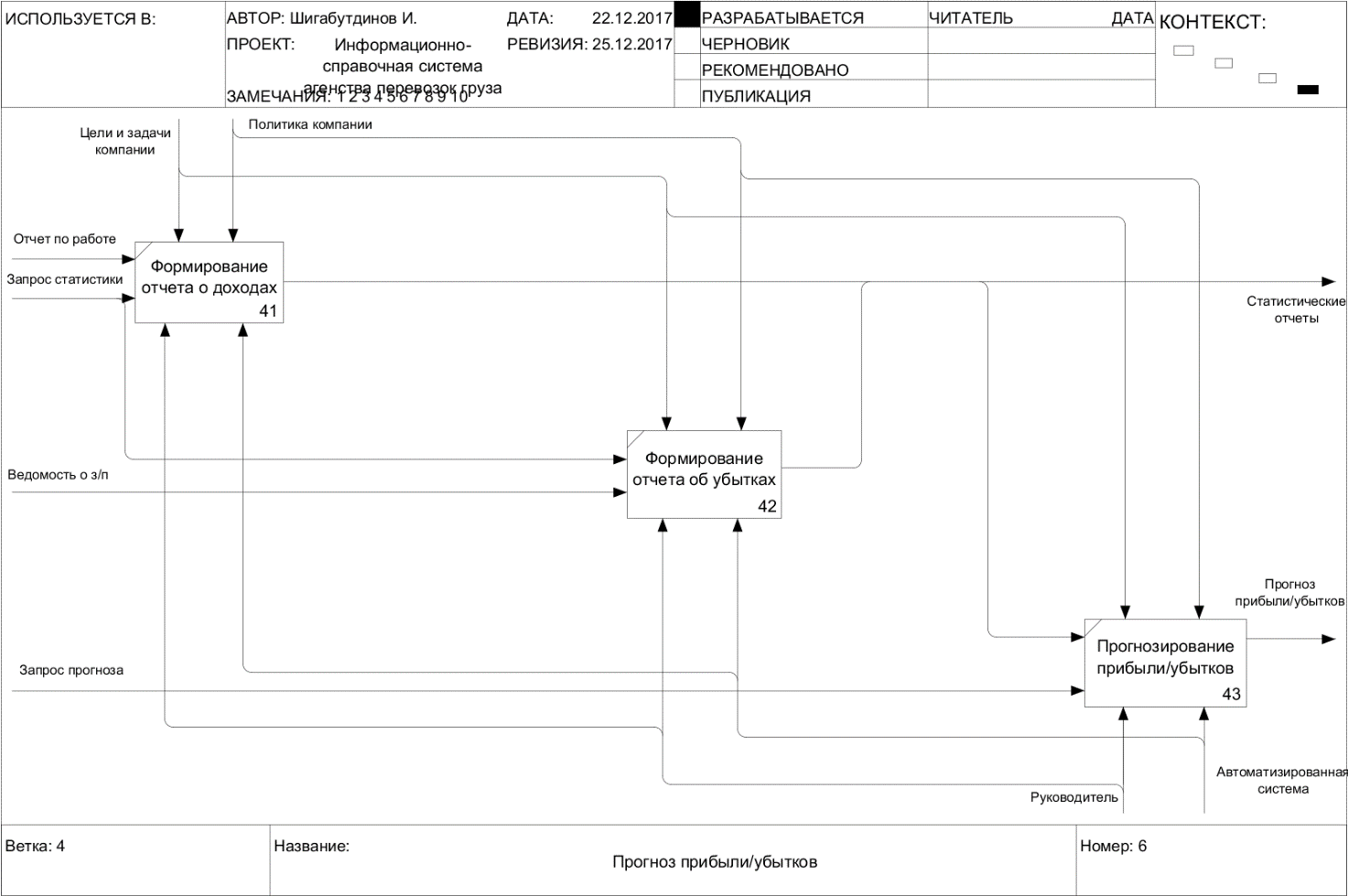


Рисунок 8. Декомпозиция диаграммы IDEF0 Уровень 2

# Техническое задание на разработку ИС по ГОСТ 34.602-89

1. Общие сведения [8].
   1. Полное наименование: Информационно справочная система агентства перевозок груза
   2. Краткое наименование: Информационно справочная система
   3. Разработчик: Студент ПИбд-31
   4. Перечень документов: Техническое задание (ТЗ), пояснительная записка.
   5. Начало работ – 01.09.2017. Окончание работ – 30.12.2017.
   6. Работы по созданию системы сдаются проектировщиком в конце срока работ. По окончании работ Разработчик сдает заказчику соответствующие отчетные документы, состав которых определены договором.
2. Назначение и цели создания системы
   1. Назначение системы:

Автоматизация бизнес-процессов информационно-справочной системы агентства грузоперевозок.

* 1. Цели создания системы:
     + 1. автоматизация деятельности агентства;
       2. учёт прибыли/убытков;
       3. автоматизация бизнес-процессов;
       4. автоматизация обработки информации;
       5. хранение важной информации.

1. Характеристика объекта автоматизации
   1. Агентство грузоперевозок – это агентство, предоставляющее услуги перевозки грузов и сервиса.
   2. Сведения об условиях эксплуатации объекта автоматизации и характеристиках окружающей среды.
      1. Условия эксплуатации объектов автоматизации

Персональные компьютеры средней мощности (требования указаны ниже)

Мобильное устройство средней мощности (требования указаны ниже)

* + 1. Характеристики окружающей среды

Операционная система – Windows(для десктоптых приложений) и Android (для руководителя и клиента)

1. Требования к системе
   1. Требования к системе в целом

Система представляет собой клиент-серверную модель, содержащую удаленную БД на сервере. БД общая для всех клиентов. В основе структуры приложений лежит пanтерн MVC. Язык программирования для десктопных приложений – C#, для мобильного приложения – Java.

* + 1. Требования к надёжности

Надёжность обеспечивает валидация данных на стороне клиента, взаимодействие с базой данных происходит по сетевому протоколу TCP/IP.

* + 1. Требования к безопасности

База данных должна быть защищена от внешних атак, благодаря использованию фаерволов. Также безопасность обеспечивает авторизация пользователей.

* + 1. Требования к производительности

Архивация данных позволяет удалять из базы данных устаревшую информацию, что повышает скорость выполнения запросов и, соответственно, производительность. Архив сохраняется в виде xml-файлов на компьютерах пользователей. Также производительность обеспечивается за счёт кэширования данных во всех АРМ и приложении клиентов.

* 1. Требования к функциям (задачам), выполняемым системой (с указанием исходных данных и результатов выполнения каждой функции);
     1. АРМ агента

|  |  |  |
| --- | --- | --- |
| Функция | Задача | Результат выполнения функции |
| Работа с заявками | Принять заявку | Обработка заявок |
| Отклонить заявку |
| Просматривать заявки |
| Работа с договорами | Просматривать договоры | Обработка договоров |
| Заключить договор |
| Редактировать договор |
| Распечатать договор |
| Архивировать договор |
| Работа с отчетами | Создать отчет | Обработка отчетов |
| Распечатать отчет |
| Работа с клиентами | Найти клиента | Обработка клиентов |
| Распечатать список клиентов |
| Работа с рейсами | Получить список рейсов | Обработка рейсов |
| Изменить стоимость |
| Просмотреть диаграмму |
| Распечатать диаграмму |

* + 1. АРМ бухгалтера

|  |  |  |
| --- | --- | --- |
| Функция | Задача | Результат выполнения функции |
| Работа с отчетами | Просмотреть список отчетов | Обработка отчетов |
| Проверить отчет |
| Посчитать зарплату |
| Рассчитать распределение ЗП агентов по месяцам |
| Работа с агентами | Получить список агентов | Обработка агентов |
| Получить агента по ФИО |
| Получить агента по ID |
| Печать списка агентов |
| Печать контракта агента |
| Работа с ведомостями | Получить список ведомостей | Обработка ведомостей |
| Создать ведомость |
| Распечатать ведомость |
| Редактировать ведомость |
| Посчитать ЗП за год |
| Архивация |
| Работа с договорами | Получить список договоров | Обработка договоров |
| Редактировать договор |
| Просмотреть диаграмму |
| Напечатать диаграмму |
| Напечатать договор |

* + 1. АРМ Руководителя

|  |  |  |
| --- | --- | --- |
| Функция | Задача | Результат выполнения функции |
| Работа с отчетами | Просмотреть отчет о доходах | Обработка отчетов |
| Распечатать отчет |
| Работа с ведомостями | Просмотреть отчет об убытках | Обработка ведомостей |
| Распечатать отчет |
| Работа с прогнозом | Просмотреть прогноз | Обработка прогноза |

* + 1. Приложение клиента

|  |  |  |
| --- | --- | --- |
| Функция | Задача | Результат выполнения функции |
| Работа с грузом | Просмотреть груз | Обработка груза |
| Создать груз |
| Редактировать груз |
| Работа с рейсами | Получить список рейсов | Обработка рейсов |
| Получить список рейсов по пункту отправки |
| Получить список рейсов по пункту прибытия |
| Получить список рейсов по времени отправки |
| Получить список рейсов по времени прибытия |
| Получить список рейсов по сервису |
| Получить список сервисов |
| Работа с пунктами | Получить список пунктов | Обработка пунктов |
| Просмотреть карту |
| Работа с заявкой | Создать заявку | Обработка заявки |
| Распечатать заявку |
| Отправить заявку |
| Задать груз |
| Задать рейс |
| Задать количество поездок |

* 1. Требования к видам обеспечения
     1. Требования к интерфейсу:

Валидация полей ввода влияет на такой параметр качества, как Надежность. Каждое поле проверяется на корректность введенных данных. После проверки данные передаются в классы контроллеры. Удобство интерфейса достигается за счет использования стандартных компонентов WindowsForms.

Также удобство достигается за счет реализации кэширования на клиентах – буфер обмена.

* + 1. Требование к контроллеру

Кэширование и управление исключениями влияют на такой параметр качества, как Надежность и Безопасность. Данные кэшируются на один сеанс. Кэширование обеспечивает быстроту работы системы в целом. Управления исключениями обрабатывает ошибки.

* + 1. Требование к связи

Протоколирование влияет на такие показатели качества, как Надежность и Безопасность. Протоколирование применяется при соединении с сервером с помощью протокола TCP/IP. Так же должен быть доступ в интернет.

* + 1. Требование к серверу

Аутентификация на сервере (Роли) влияет на такой параметр качества, как Безопасность, Надежность. Аутентификация ролей разграничивает доступ к данным на сервере, так же исключает негативное влияние со стороны.

* + 1. Спецификации по безопасности
    - Информация Системы должна быть расположена в закрытой БД, доступ к которой разрешен только из программных модулей Системы.
    - Доступ к закрытым областям Системы разрешен только пользователям, имеющим повышенный статус.
    - На уровне СУБД должно быть реализовано разграничение доступа к данным в БД.
    1. Требования к надежности и сохранности информации:

1) Сервер должен иметь блок бесперебойного питания, обеспечивающие функционирование системы в случае возникновения аварийной ситуации.

* + 1. Требования к операционному и программному окружению, к техническим средствам, к средствам разработки, сопровождения и адаптации АС к условиям конкретного объекта.

• Требования к операционному и программному окружению. Сервер БД: ОС - Linux CentOS, СУБД - MariaDB.

• Требования к техническим средствам: Требования к серверу:

|  |  |
| --- | --- |
| Ресурс, единица измерения | Требуется |
| CPU (процессор) | Минимальные требования: CPU Intel Xeon X5550 2.66  GHz\4 core  Рекомендуемые требования: CPU Intel Xeon E5-2680 2.8 GHz\10 core |
| Memory (оперативная память) | Минимальные требования: 16Gb Рекомендуемые требования: 64Gb |
| Количество HDD (жестких дисков) и свободное пространство на них | Минимальные требования:  1×3Tb    Рекомендуемые требования: 2x3Tb |
| Требуется доступ в интернет | |

Требования к клиентскому месту (ПК):

|  |  |
| --- | --- |
| Ресурс, единица измерения | Требуется |
| CPU (процессор) | 32-разрядный (x86) или 64-разрядный (x64) процессор с тактовой частотой 1 гигагерц (ГГц) или выше. |
| Memory (оперативная память) | 1 гигабайт (ГБ) (для 32-разрядной системы) или 2 ГБ  (для 64-разрядной системы) оперативной памяти (ОЗУ). |
| Свободное пространство на HDD | 16 гигабайт (ГБ) (для 32-разрядной системы) или 20 ГБ (для 64-разрядной системы) пространства на жестком диске. |
| Требуется доступ в интернет | |

Требования к клиентскому месту (Android):

|  |  |
| --- | --- |
| Ресурс, единица измерения | Требуется |
| ОС | Android 4.4 и выше |
| Свободное пространство памяти | 69 МБ |
| Требуется доступ в интернет | |

Требования к средствам разработки, сопровождения и адаптации Системы:

Среда разработки Visual Studio версии 2010 и выше.

Среда разработки Android Studio версии 3.0 и выше.

* + 1. Для работы с Системой предъявляются следующие требования к квалификации Пользователя:
  1. Основные навыки пользования ПК и ОС Windows.
  2. Основные навыки использования мобильного устройства и ОС Android.
  3. Хорошее знание предметной области (информационного наполнения информационно-справочной системы).

1. Состав и содержание работ по созданию системы
   1. перечень документов, предъявляемых по окончании соответствующих стадий и этапов работ:

Техническое задание на разработку информационно-справочной системы, пояснительная записка к курсовому проекту.

* 1. программа работ, направленных на обеспечение требуемого уровня надежности разрабатываемой системы (при необходимости):

Валидация волей ввода на клиентах

Устойчивый доступ к БД через протокол TCP/IP

1. Порядок контроля и приемки системы (при необходимости)

Контроль и приёмка системы осуществляется за счёт договорённости между разработчиком и заказчиком без привлечения сторонних приёмочных комиссий. Испытания системы проводятся на базе разработчика в соответствии с требованиями заказчика и оговариваются с заказчиком, ставясь в соответствии с требованиями к функционированию системы.

1. Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие (при необходимости)

В офисе агентства необходимо наличие минимум двух ПК, сервера и одного Android устройства

Клиенту необходимо иметь в наличии Android устройство с доступом к сети Интернет.

1. Требования к документированию.

Документация должна быть составлена на русском языке, а также быть подготовлена как в печатном, так и в электронном виде.

1. Источники разработки.
   1. Разработка Системы выполняется с учетом требований следующих нормативных документов:
      * + ГОСТ 19.201-78. Единая система программной документации. Техническое задание. Требования к содержанию и оформлению.
        + ГОСТ 34.602-89 Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы .
        + ГОСТ 25123-82. Машины вычислительные и системы обработки данных. Техническое задание. Порядок построения, изложения и оформления.
   2. В соответствии с полученным заданием, студент проводит проектирование и дальнейшее планирование:
      * + Среда программирования: Visual Studio, Android Studio
        + CASE-средства: Ramus, Visual Paradigm
        + Текстовый редактор Word
        + Листы и ручка

# Модель информационной системы

Спроектированная информационная система базируется на клиент-серверной архитектуре. На сервере расположена база данных. Сервер лишь управляет доступом к БД и производит контрольную валидацию, а вся логика обрабатывается на стороне клиента, что представляет собой модель «Толстый клиент». Использование данной модели способствует снижению нагрузки на сервер. [2]

Система спроектирована с использованием паттерна MVC так как данный паттерн позволяет разделить логику от представлений, что необходимо для кросс платформенной разработки и позволяет с небольшими усилиями масштабировать проект. [3]

Для АРМ агента и бухгалтера были спроектированы десктопные приложения на языке C#, а для приложения клиента и АРМ руководителя – мобильные приложения на языке Java. Данные языки реализуют объектно-ориентированный подход что является одним из условий поставленной задачи.

# Диаграммы вариантов использования

Диаграмма вариантов использования (Use Case Diagram) представляет из себя набор действий (вариантов использования), а также актера (пользователя). В процессе функционирования программы актер взаимодействует с системой посредством вариантов использования.

На рисунке 9 представлена диаграмма вариантов использования для АРМ агента. Агент рассматривает заявки клиентов на грузоперевозку, заключает с ними договоры, отчитывается о проделанной работе.

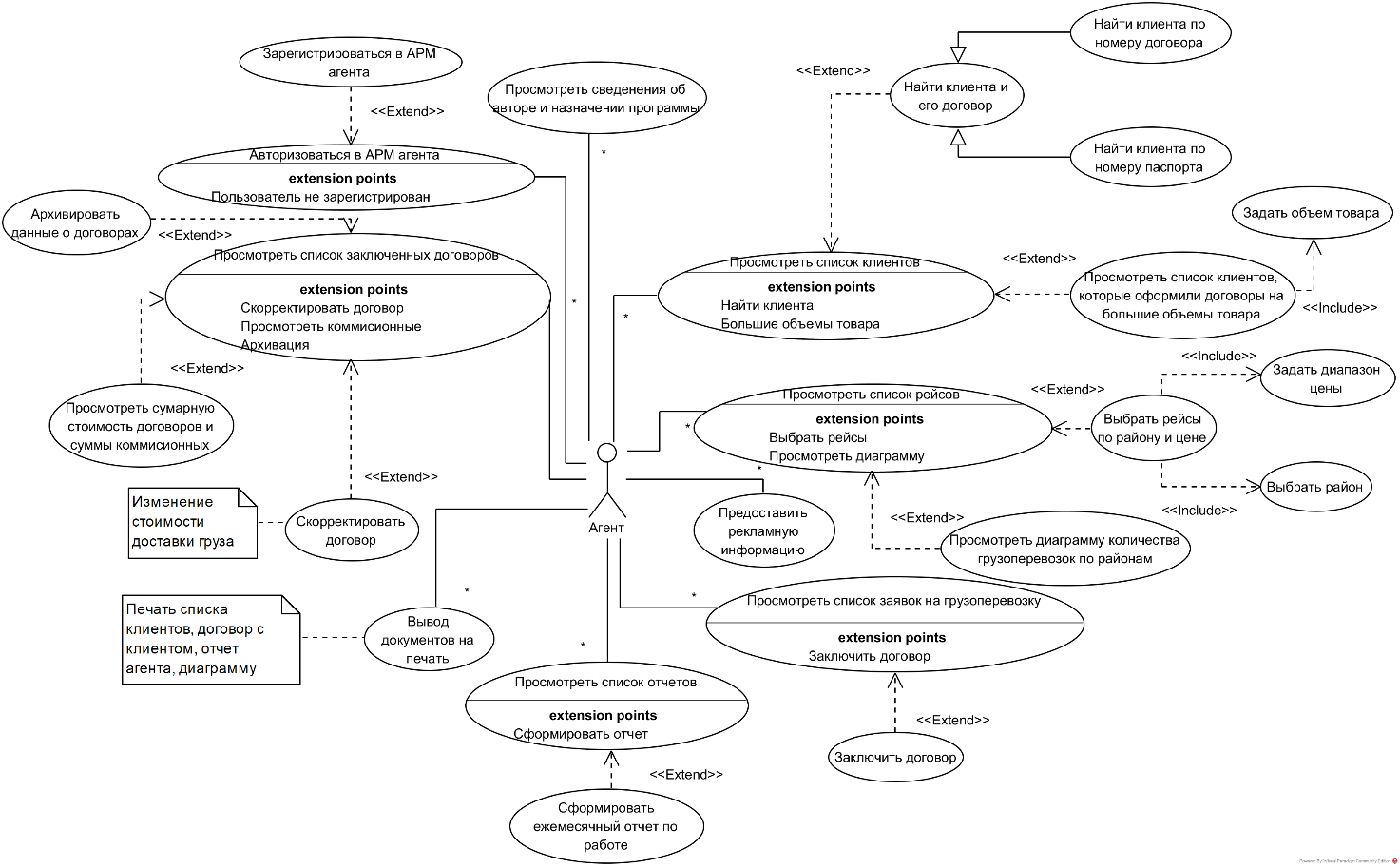


Рисунок 9. Диаграмма вариантов использования для АРМ агента

На рисунке 10 представлена диаграмма вариантов использования для АРМ бухгалтера. Бухгалтер проверяет отчеты агентов, рассчитывает заработную плату, формирует ведомости.

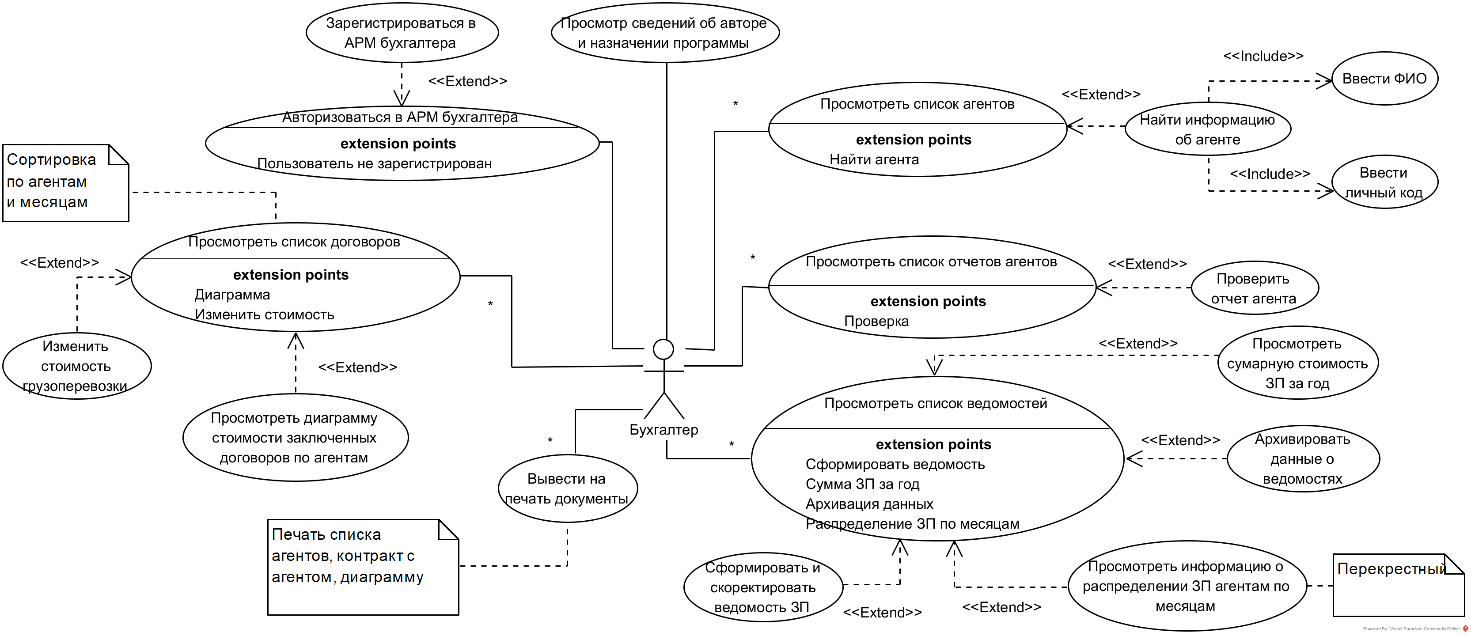


Рисунок 10. Диаграмма вариантов использования для АРМ бухгалтера

На рисунке 11 представлена диаграмма вариантов использования для приложения клиента. Клиент формирует и отправляет заявку на грузоперевозку.

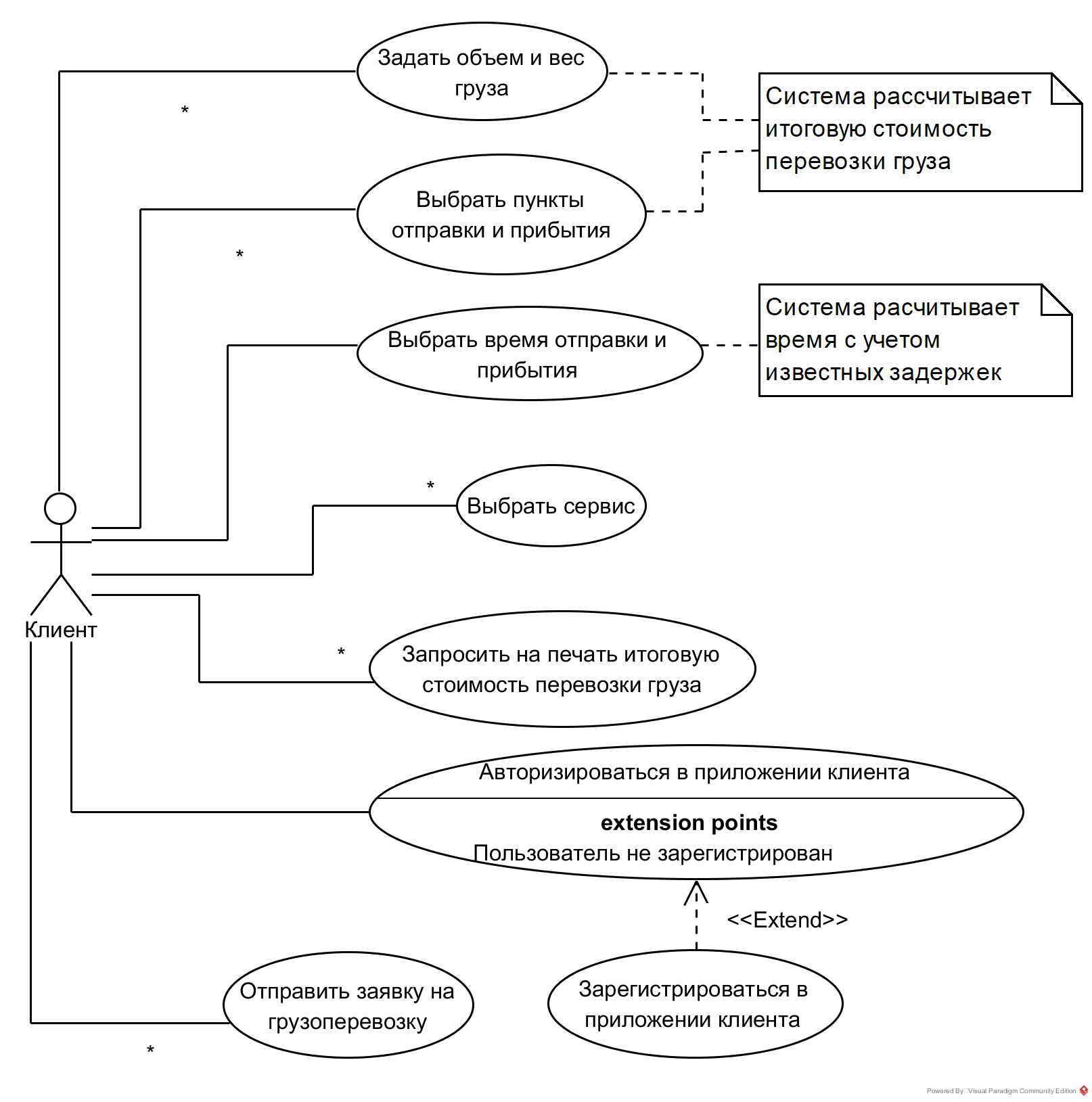


Рисунок 11. Диаграмма вариантов использования для приложения клиента

На рисунке 12 представлена диаграмма вариантов использования для АРМ руководителя. Руководитель получает статистические отчеты, прогнозирует прибыль/убыток.

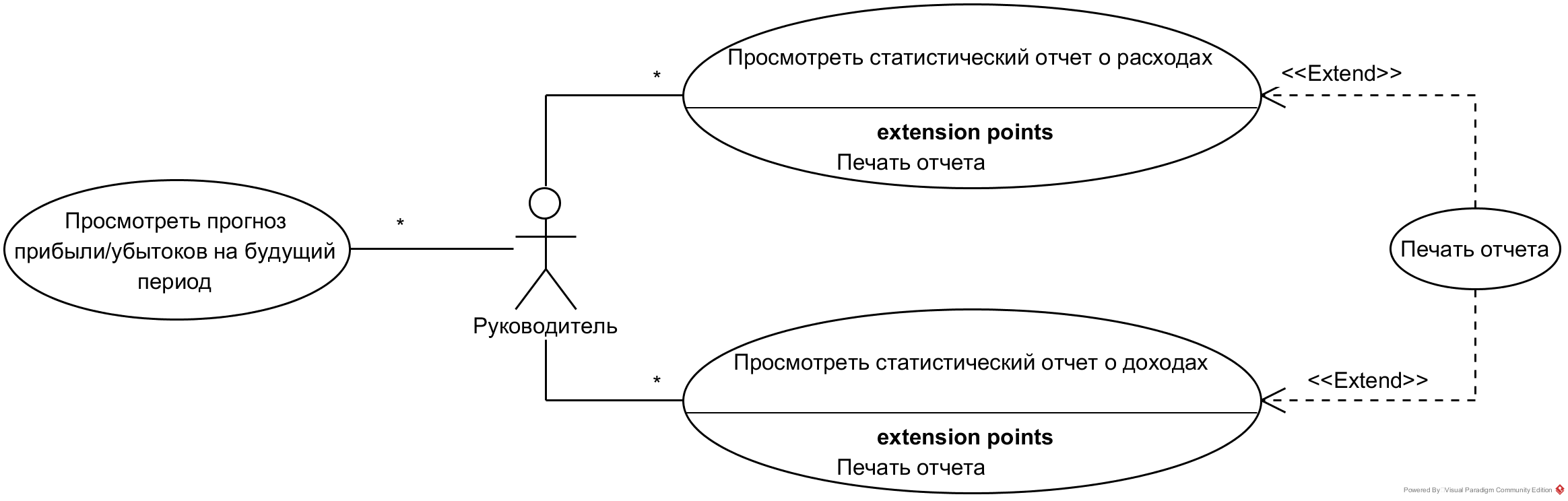


Рисунок 12. Диаграмма вариантов использования для АРМ руководителя

# Диаграммы состояний

По существу, диаграмма состояний представляет собой диаграмму состояний из теории автоматов со стандартизированными обозначениями. Диаграммы состояний применяются для того, чтобы объяснить, каким образом работают сложные объекты. Диаграммы состояний служат для моделирования динамических аспектов системы Данная диаграмма описывает возможные последовательности состояний и переходы, которые в совокупности характеризуют поведение элемента модели в течение его жизненного цикла. [5]

На рисунке 13 представлена диаграмма состояний для АРМ агента. Состояния данной диаграммы представляют из себя активные формы пользовательского интерфейса. На диаграмме присутствуют и сторожевые условия.

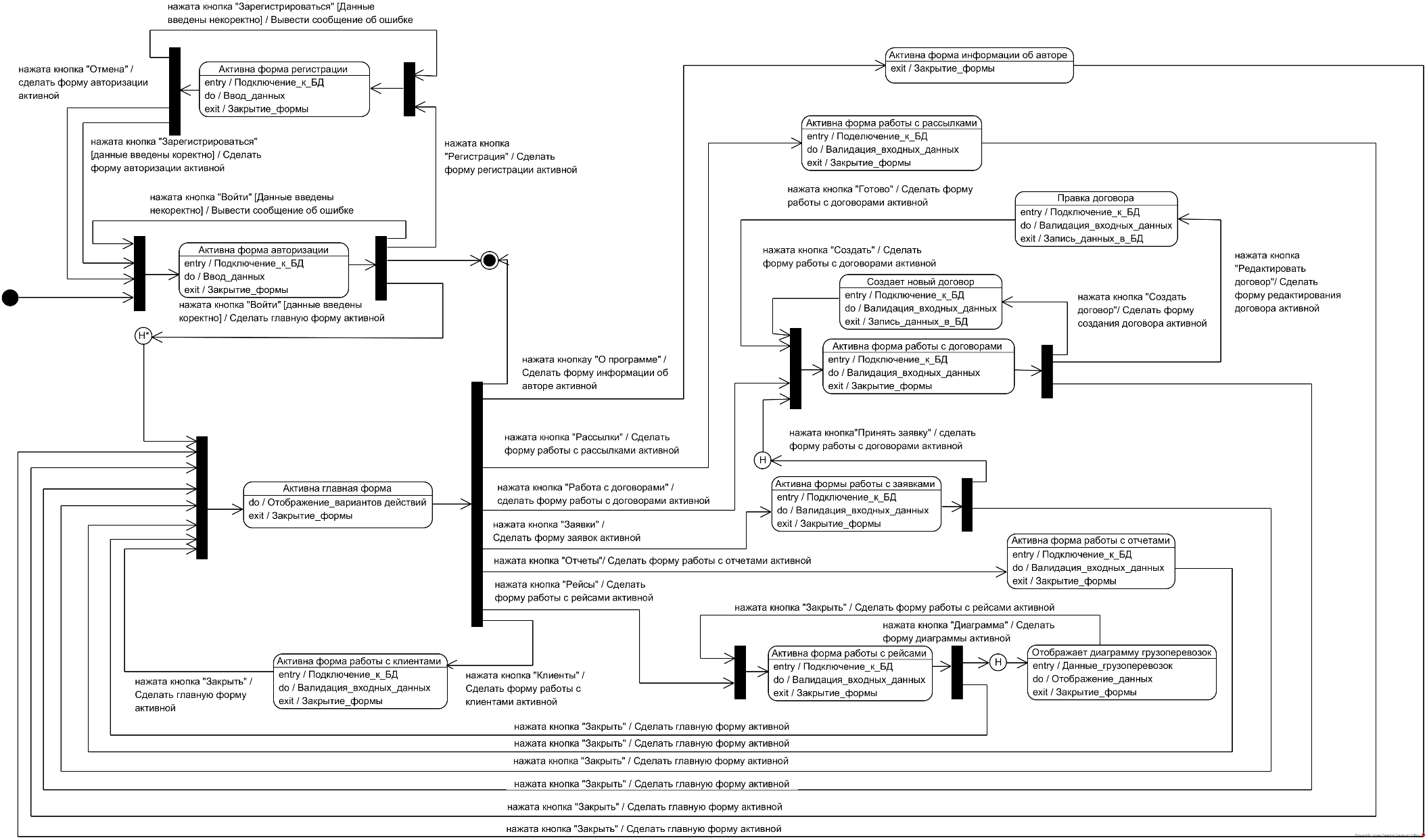


Рисунок 13. Диаграмма состояний для АРМ агента

На рисунке 14 представлена диаграмма состояний для АРМ бухгалтера. Как и на предыдущей диаграмме здесь в качестве состояний служат формы пользовательского интерфейса. На диаграмме присутствуют несколько сторожевых условий.

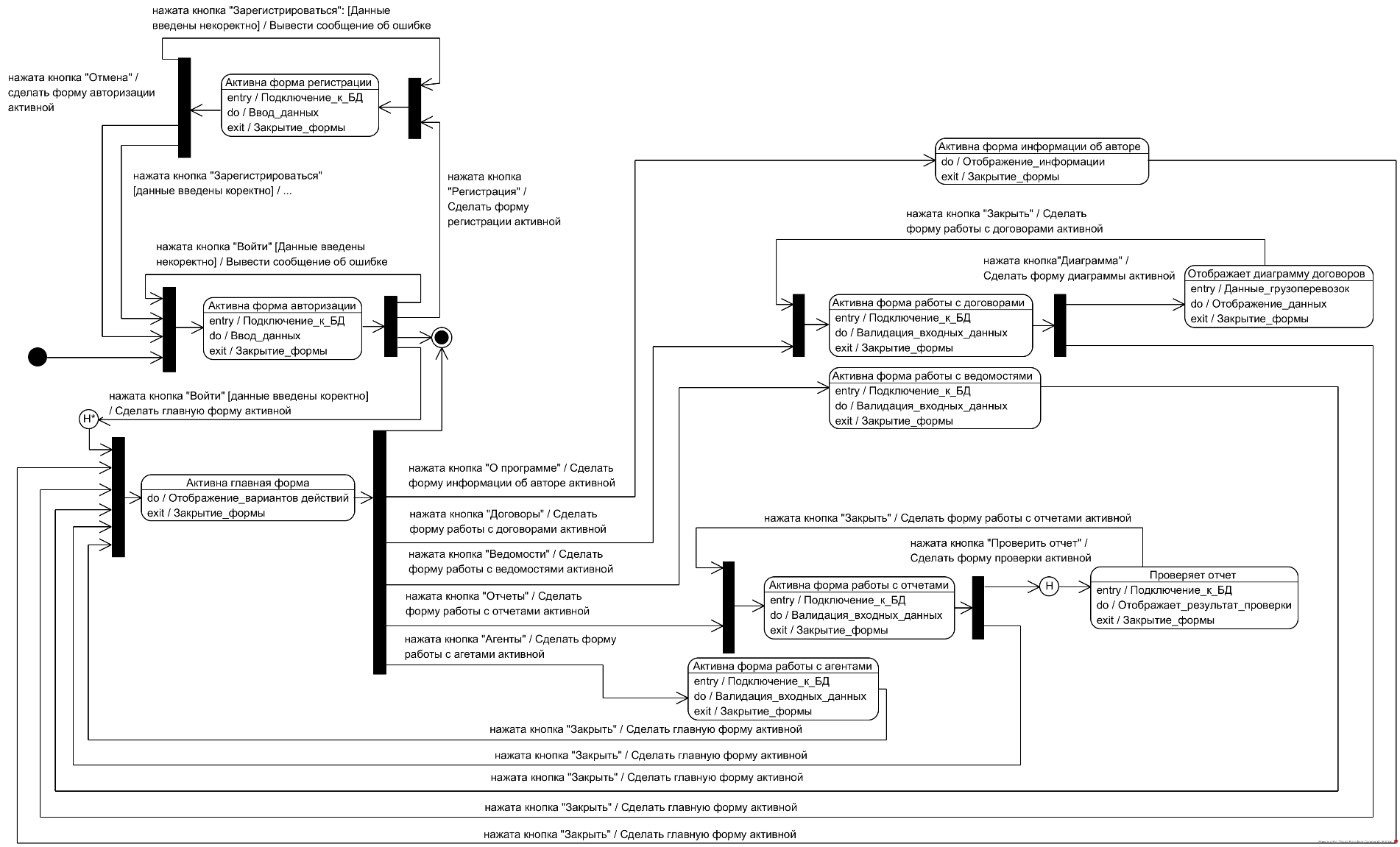


Рисунок 14. Диаграмма состояний для АРМ бухгалтера

На рисунке 15 представлена диаграмма состояний для АРМ руководителя. Состояния диаграммы так же представляют себя формы пользовательского интерфейса.

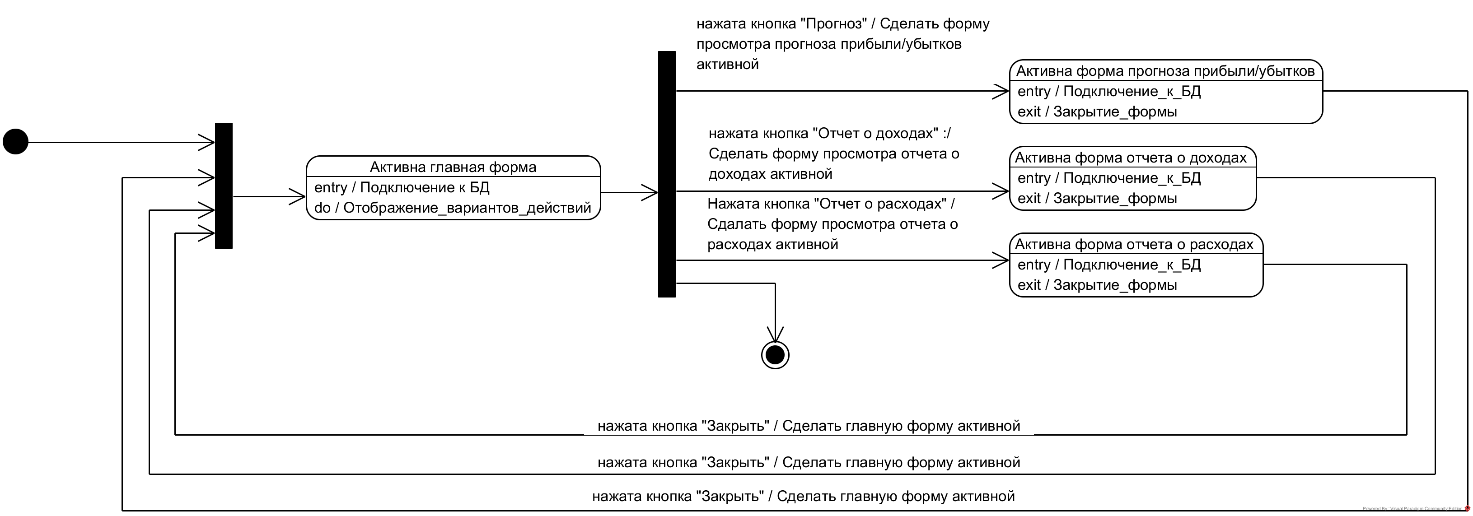


Рисунок 15. Диаграмма состояний для АРМ руководителя

На рисунке 16 представлена диаграмма состояний для приложения клиента. Состояния диаграммы так же представляют себя формы пользовательского интерфейса. На диаграмме присутствуют несколько сторожевых условий.

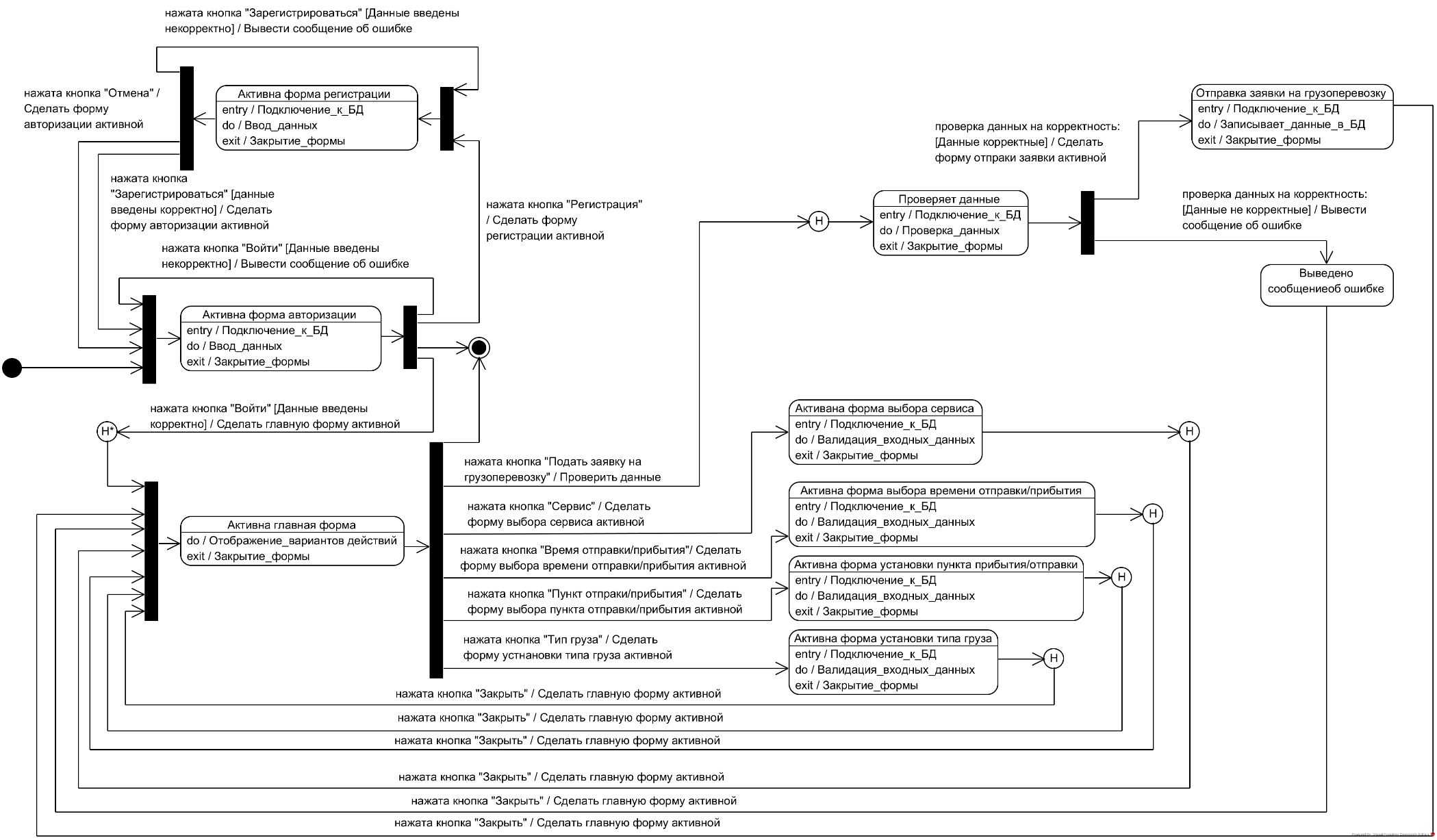


Рисунок 16. Диаграмма состояний для приложения клиента

# Диаграммы классов

На следующих рисунках будут представлены диаграммы классов. В данных диаграммах классы разделены по пакетам, соответствующим паттерну MVC.

Все классы-сущности (Models) наследуются от абстрактного класса «Model». Данный класс содержит в себе методы для работы с базой данных, а также общие атрибуты для классов-сущностей (моделей).

Классы-контроллеры (Controllers) наследуются от интерфейса «IController» который содержит метод кеширования отражающий сквозную функциональность.

Классы-представления (Views) наследуются от интерфейса «IView» который содержит метод валидации входных данных отражающий сквозную функциональность.

На рисунке 17 представлена диаграмма классов для АРМ агента.

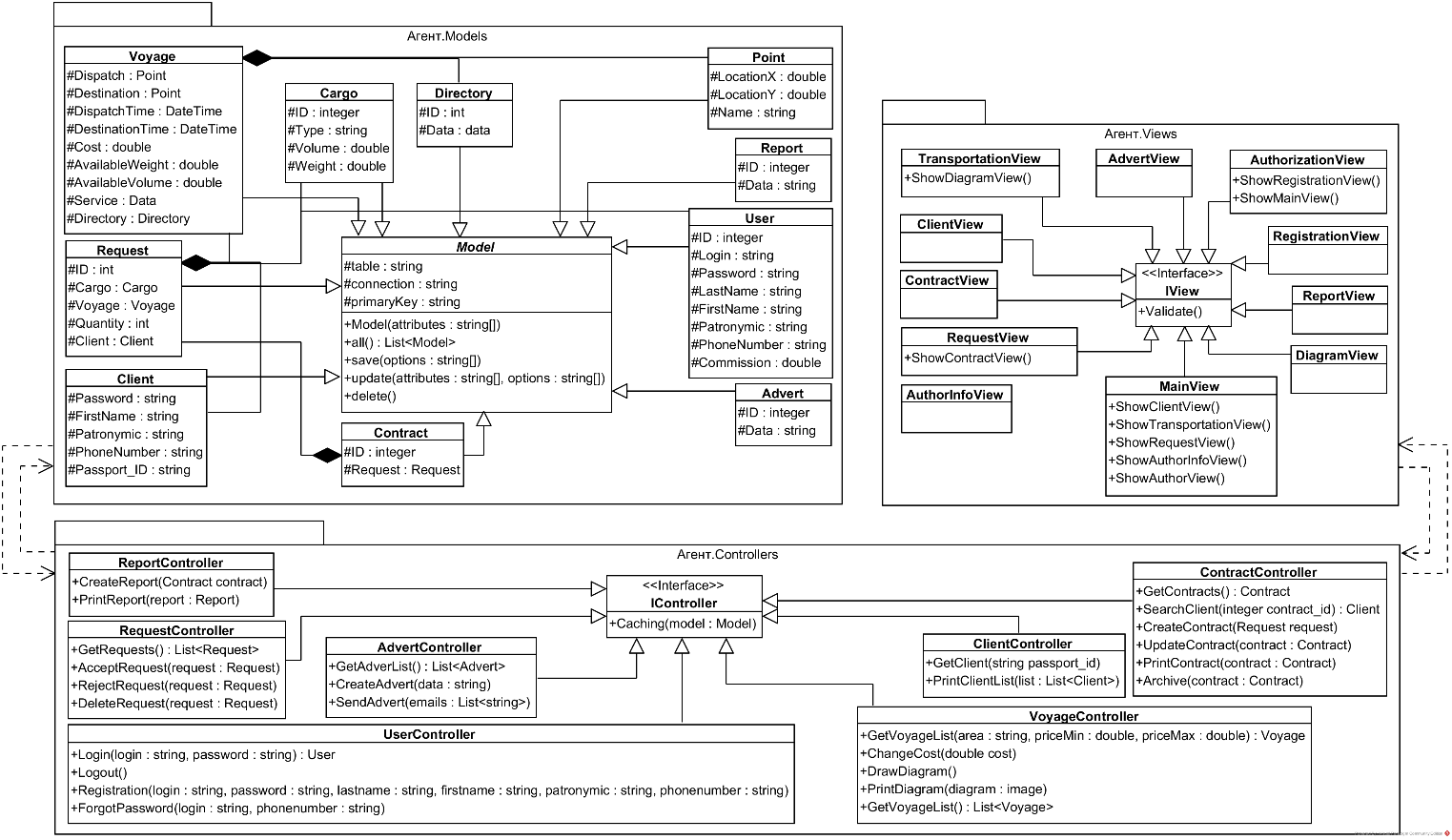


Рисунок 17. Диаграмма классов для АРМ агента

На рисунке 18 представлена диаграмма классов для АРМ бухгалтера.

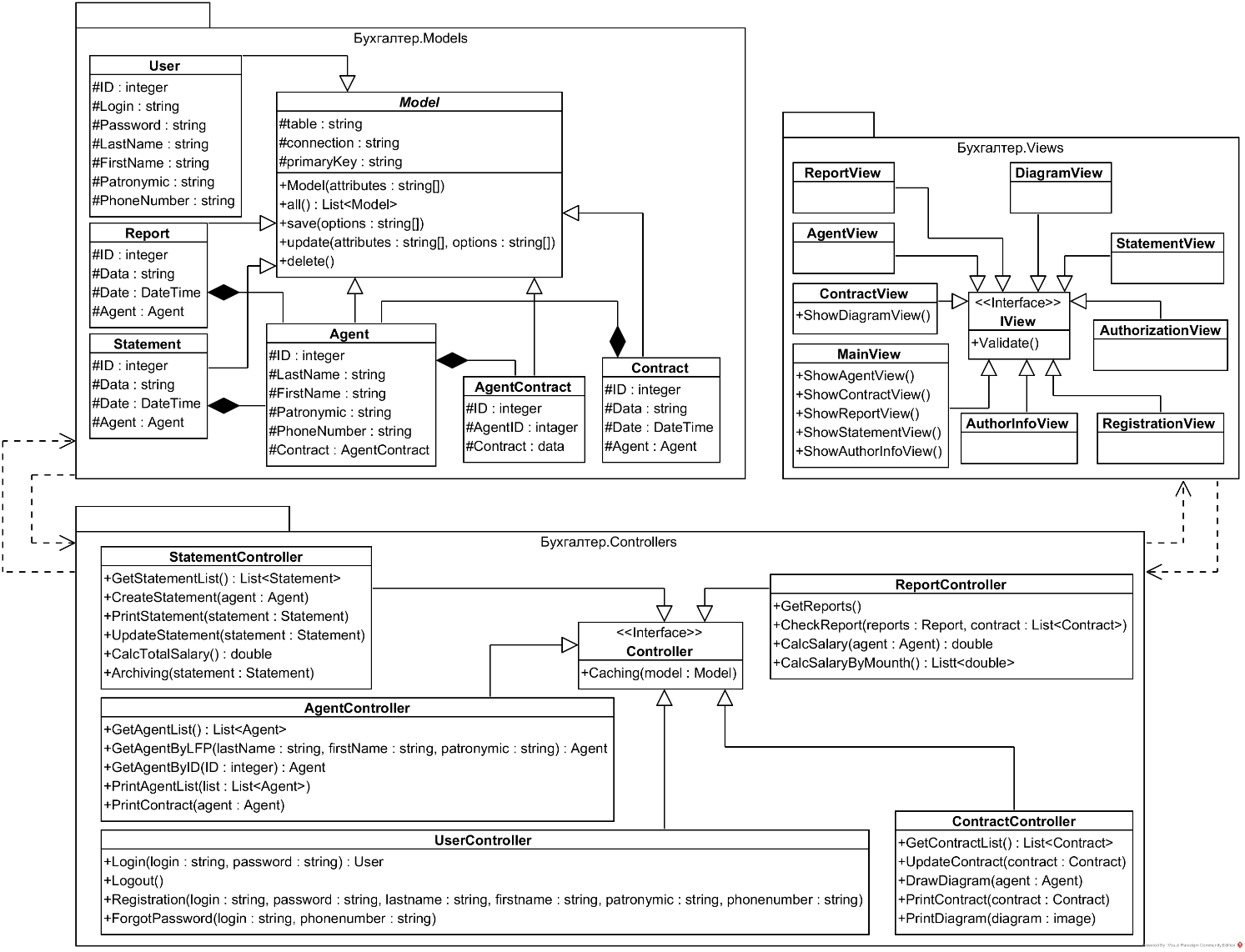


Рисунок 18. Диаграмма классов для АРМ бухгалтера

На рисунке 19 представлена диаграмма классов для АРМ руководителя.

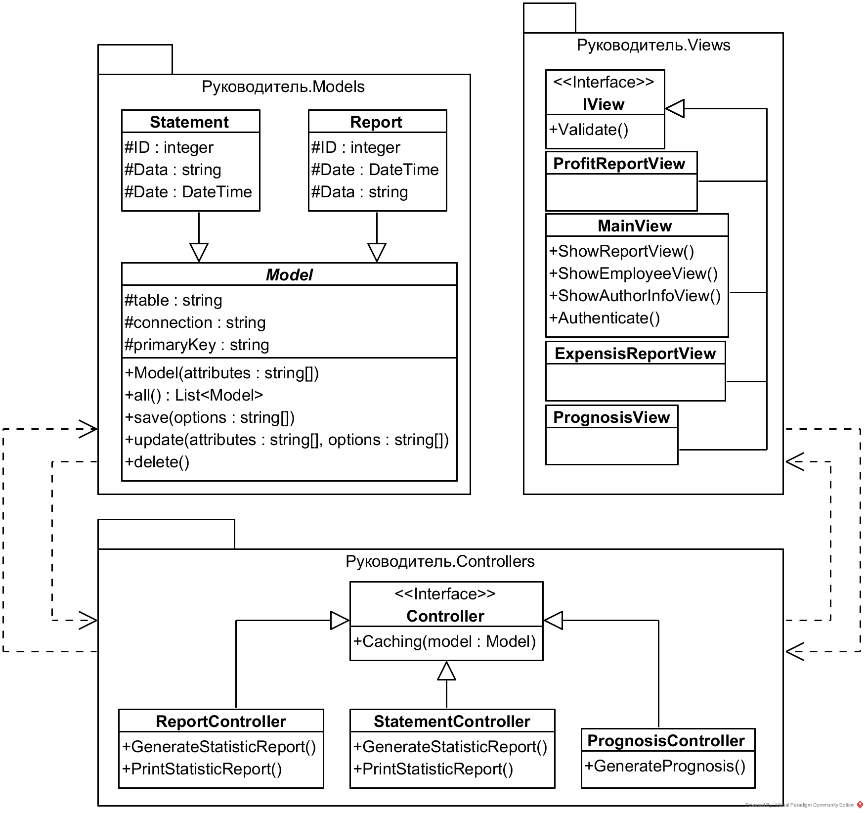


Рисунок 19. Диаграмма классов для АРМ руководителя

На рисунке 20 представлена диаграмма классов для приложения клиента.

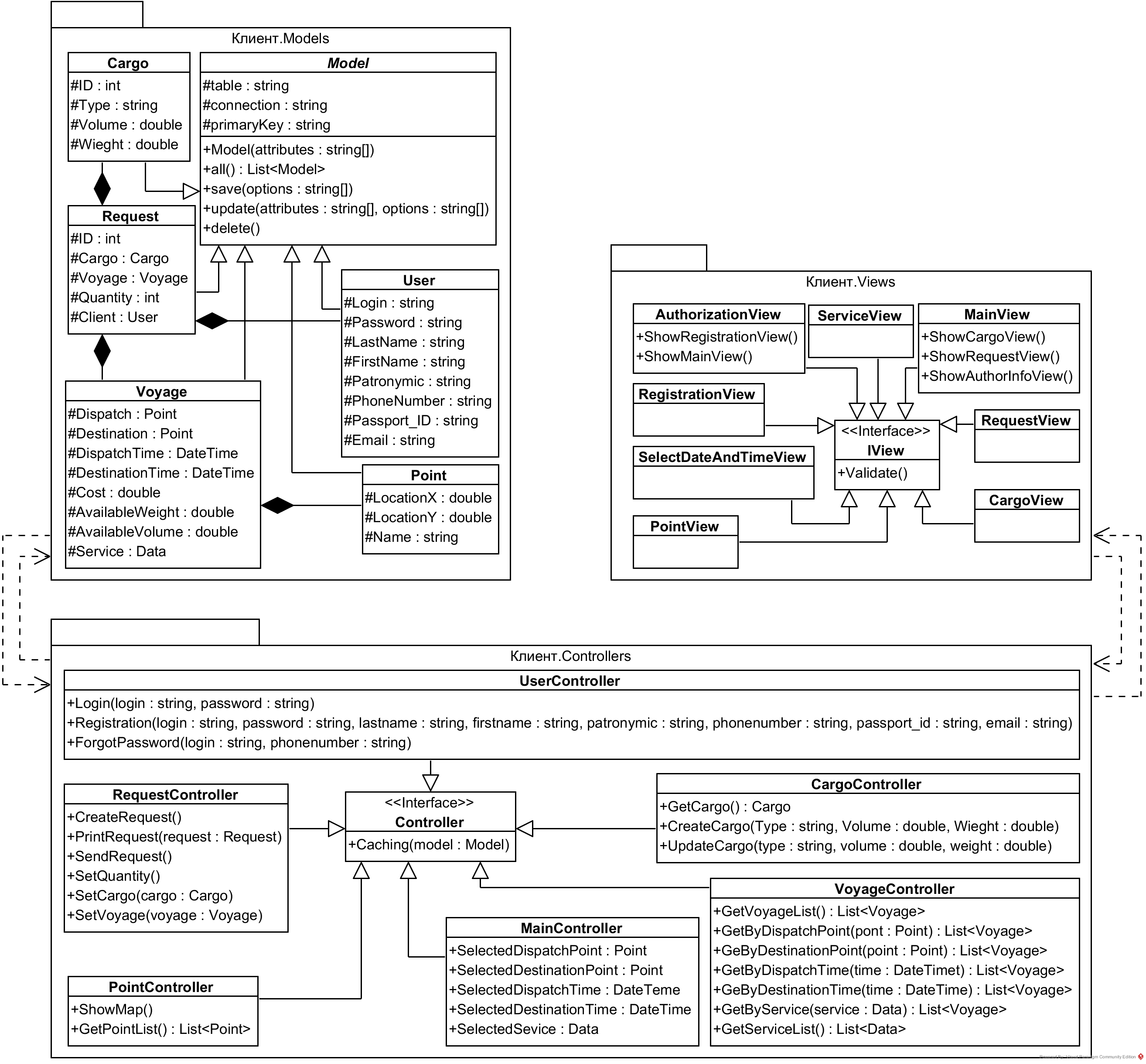


Рисунок 20. Диаграмма классов для приложения клиента

# Диаграммы последовательности

Диаграмма последовательностей относится к диаграммам взаимодействия UML, описывающим поведенческие аспекты системы, но рассматривает взаимодействие объектов во времени. Другими словами, диаграмма последовательностей отображает временные особенности передачи и приема сообщений объектами. Диаграмма последовательностей отображает взаимодействие объектов в динамике [5].

На рисунке 21 представлена диаграмма последовательности для АРМ агента. В ней отображен процесс заключения договора. Агент рассматривает список заявок на грузоперевозку, выбирает одну из них, далее он может отклонить заявку тем самым удалив ее из базы данных, либо может подтвердить заявку и заключить с данным клиентом договор, после чего заявка так же удаляется из базы данных.

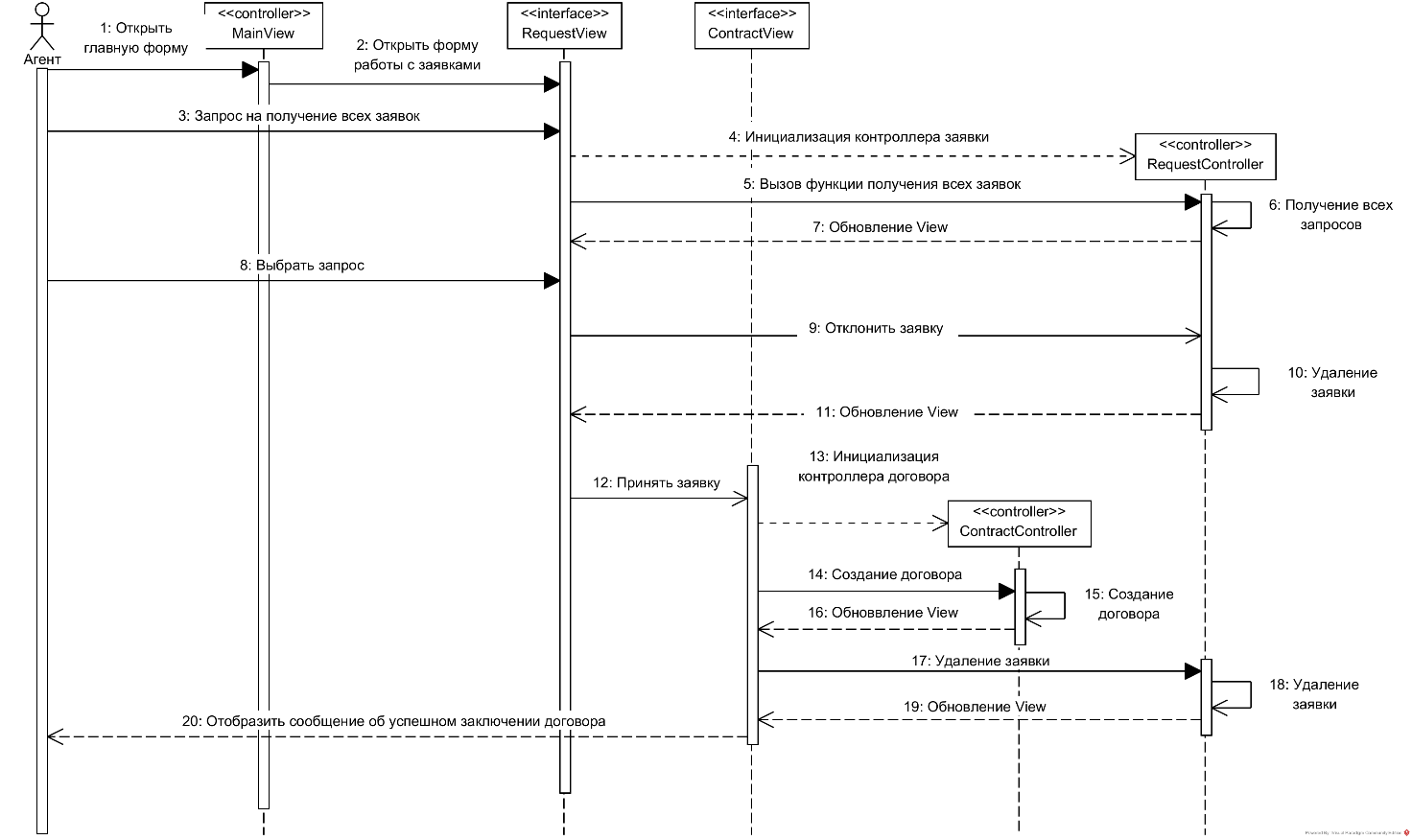


Рисунок 21. Диаграмма последовательности для процесса заключения договора

На рисунке 22 отображен процесс архивации для АРМ агента. Агент просматривает список договоров, выбирает один из них и дает команду на архивацию. В процессе архивации договор сохраняется в файл с формата xml, затем он удаляется из базы данных.

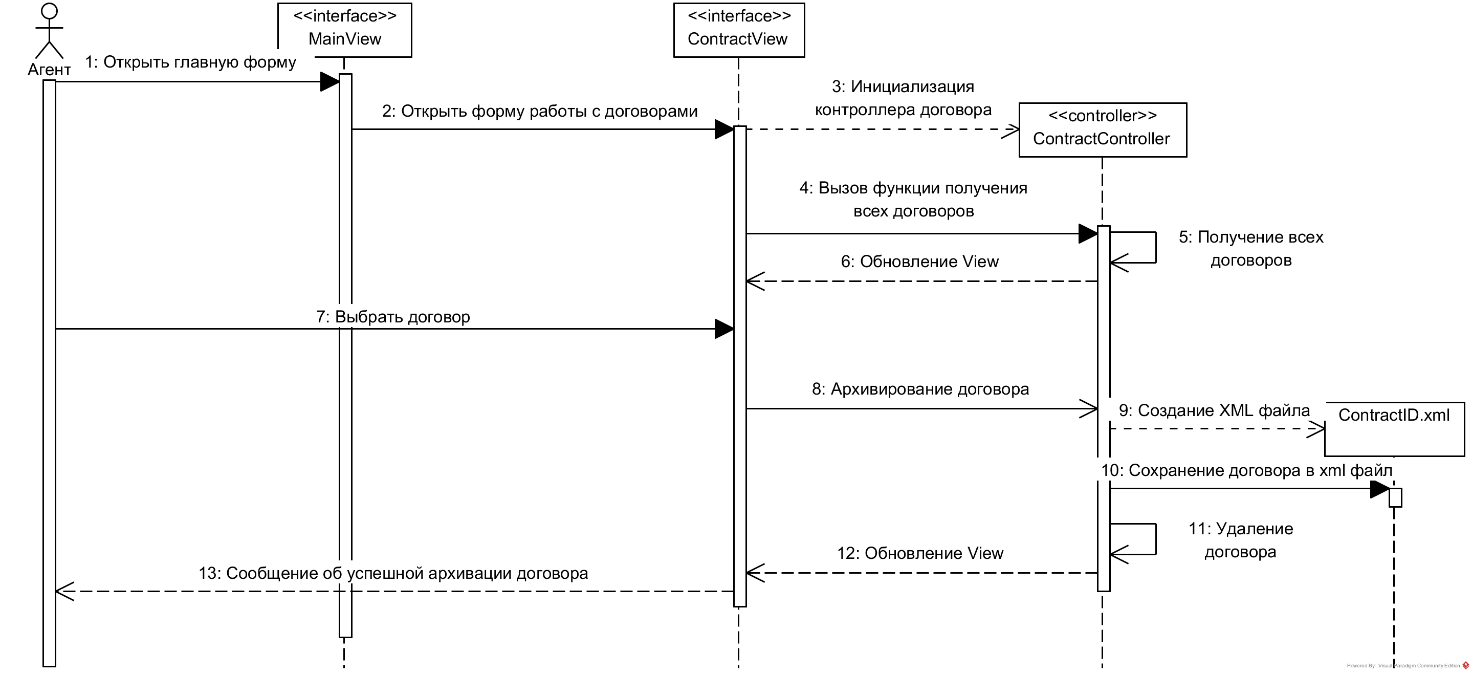


Рисунок 22. Диаграмма последовательности для процесса архивации

# Диаграмма компонентов

Диаграмма компонентов это статическая структурная [диаграмма](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0), она показывает разбиение программной системы на структурные [компоненты](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D0%BE%D0%BD%D0%B5%D0%BD%D1%82_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) и связи (зависимости) между компонентами. В качестве физических компонентов могут выступать [файлы](https://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB), библиотеки, модули, исполняемые файлы, пакеты и т. п. [6]

На рисунке 23 представлена диаграмма компонентов для всей системы в целом, на следующих за ней рисунках отображена декомпозиция данной диаграммы.

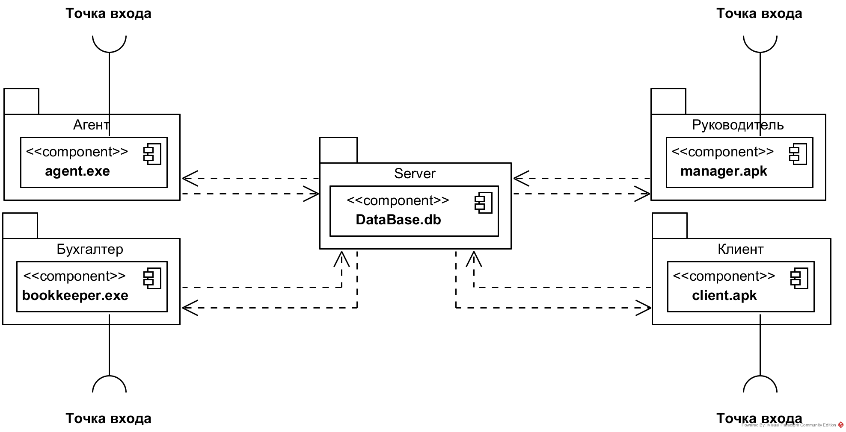


Рисунок 23. Диаграмма компонентов для всей системы в целом.

На рисунке 24 представлена диаграмма компонентов для АРМ агента. Каждый класс из пакета «View» состоит из нескольких файлов: designer, resx и сам класс.



Рисунок 24. Диаграмма компонентов для АРМ агента

На рисунке 25 представлена диаграмма компонентов для АРМ бухгалтера. Каждый класс из пакета «View» состоит из нескольких файлов: designer, resx и сам класс.

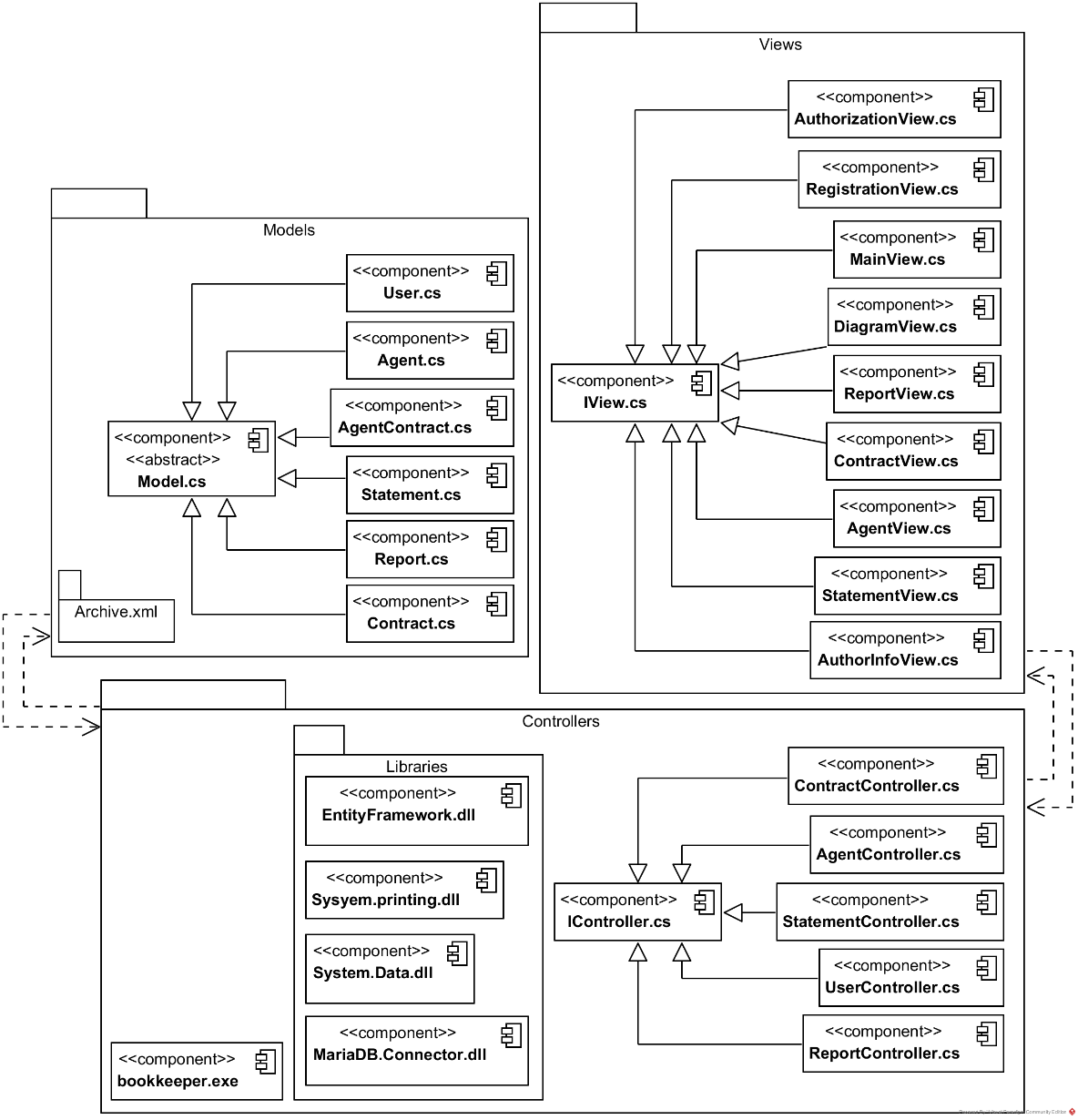


Рисунок 25. Диаграмма компонентов для АРМ бухгалтера.

На рисунке 26 представлена диаграмма компонентов для АРМ руководителя.

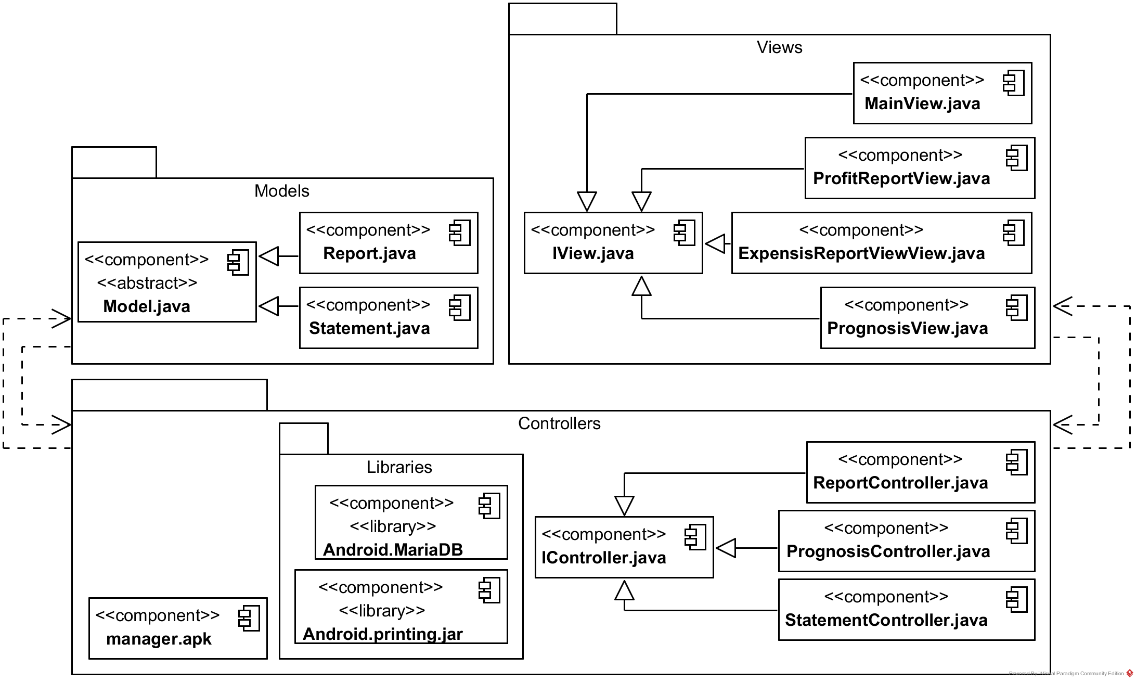


Рисунок 26. Диаграмма компонентов для АРМ руководителя

На рисунке 27 представлена диаграмма компонентов для приложения клиента.

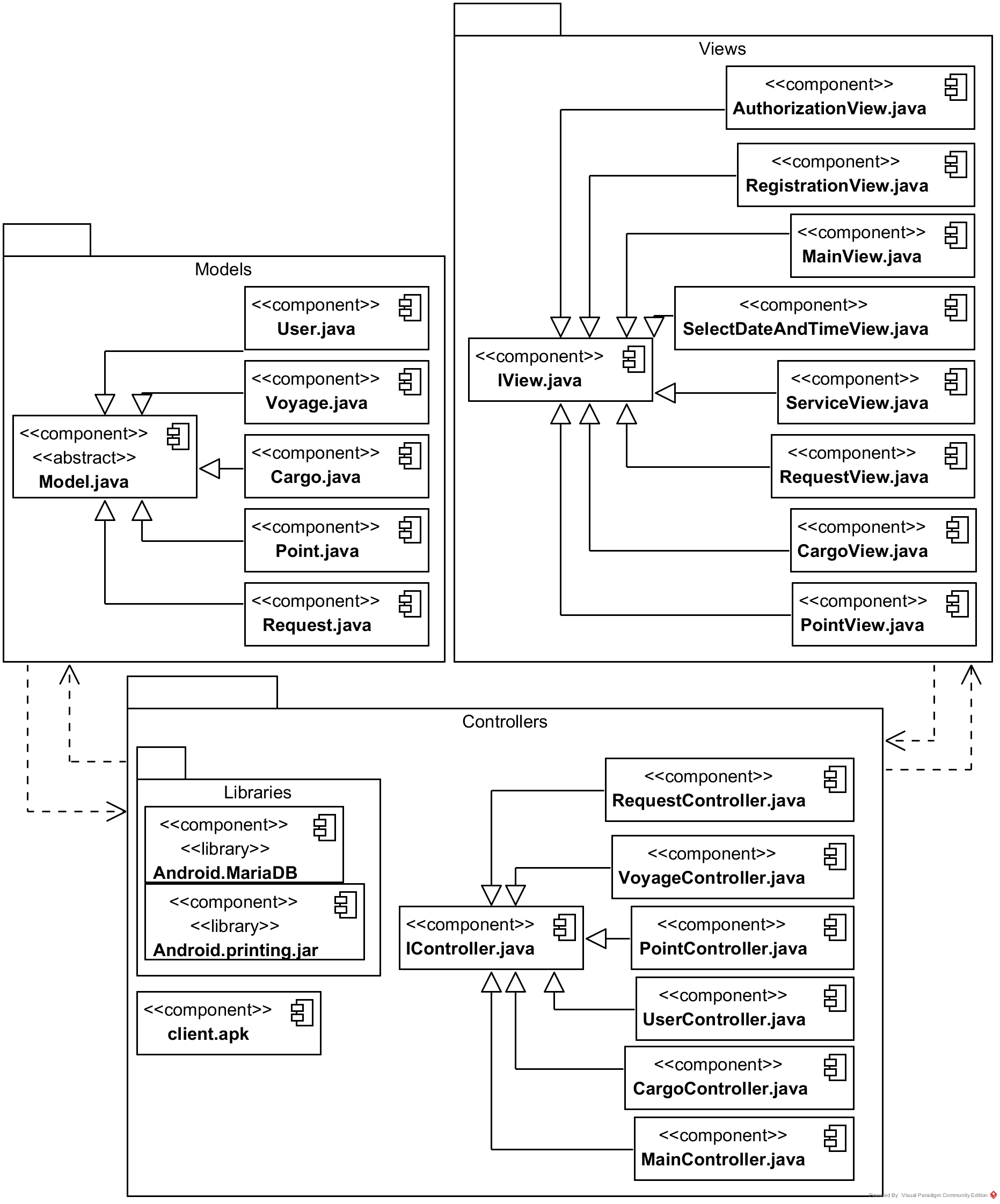


Рисунок 27. Диаграмма компонентов для приложения клиента

# 6.6. Диаграмма развертывания

Диаграмма развертывания применяется для представления общей конфигурации и топологии распределенной программной системы и содержит распределение компонентов по отдельным узлам системы [11].

На рисунке 28 представлена диаграмма развертывания. АРМ агента и бухгалтера находятся в одной локальной сети с сервером, на котором расположена база данных. АРМ руководителя приложение клиента находятся в глобальной сети. Все АРМ а так же приложение клиента взаимодействуют с принтером.



Рисунок 28. Диаграмма развертывания

# Заключение

В ходе выполнения данной курсовой работы была спроектирована информационно-справочная система агентства перевозок груза, предназначенная для автоматизации процесса предоставления услуг грузоперевозки. Были получены новые знания и навыки по проектированию программных систем. Так же были улучшены навыки работы в среде проектирования Visual Paradigm и Ramus.

# Библиографический список

1. insapov.ru «Описание стандарта idef0» [В Интернете] URL:  [http://www.insapov.ru/idef0-standard-description.html](http://www.insapov.ru/idef0-standard-description.html%20idef0)
2. ИНТУИТ «Архитектурные решения» [В Интернете] URL: <https://www.intuit.ru/studies/courses/508/364/lecture/8643?page=1>
3. Professorweb «Паттерн MVC» [В Интернете] URL: <https://professorweb.ru/my/WPF/documents_WPF/level36/36_3.php>
4. Блог программиста «Основы UML – диаграммы использования(use-case)» [В Интернете] URL: <https://pro-prof.com/archives/2594>
5. ИНТУИТ «Виды диаграмм UML» [В Интернете] URL: <https://www.intuit.ru/studies/courses/1007/229/lecture/5954?page=4>
6. ВикипедиЯ «Диаграмма компонентов» [В Интернете] URL: <https://ru.wikipedia.org/wiki/Диаграмма_компонентов>
7. Проектирование и архитектура программных систем: курс лекций / Корунова Н.В. , – Ульяновск: Информационные системы, 2017.
8. ГОСТ 34.602-89. Информационная технология. Техническое задание на создание автоматизированной системы. – Москва: Издательство стандартов, 1989. – 11 с.
9. Выполнение и оформление курсовых проектов (работ) : методические указания / сост. Н. В. Корунова. – Ульяновск : УлГТУ, 2014. – 40 с.
10. 1. Федеральный государственный образовательный стандарт высшего про-фессионального образования по направлению подготовки 230100 Про-граммная инженерия (квалификация (степень) «бакалавр»). Утвержден приказом Министерства образования и науки Российской Федерации от 09 ноября 2009 г. № 542. Москва, 2009.