

Московский государственный технический университет
имени Н.Э. Баумана

И.Г. Боровик, В.М. Недашковский, С.А. Сакулин

Проектирование информационной системы

*Учебно-методическое пособие по выполнению домашнего задания по курсу
«Безопасность систем баз данных»*

Факультет «Информатика и системы управления»

Кафедра «Информационная безопасность»

Рекомендовано Научно-методическим советом

МГТУ им. Н. Э. Баумана в качестве учебно-методического пособия

Москва

2022

Оглавление

Оглавление	2
Предисловие.....	3
Введение	4
1. Теоретическая часть.....	5
2. Задание	11
3. Требования к содержанию расчетно-пояснительной записки.....	14
4. Контрольные вопросы.....	15
5. Требования к защите	16
6. Список рекомендуемой литературы.....	16
Приложение 1. Иллюстративные (Учебные) примеры	18
П1.1. Фрагменты этапа Подготовка	18
Шаг Описание предметной области	18
Шаг Глоссарий предметной области	22
Шаг Видение	25
П1.2. Фрагменты этапа Анализ.....	27
Шаг Концептуальная модель предметной области.....	27
Шаг Функциональные требования.....	28
Шаг Диаграммы вариантов использования	32
Шаг Краткое и подробное описание вариантов использования	33
Шаг Проектирование пользовательского интерфейса	37
П1.3. Фрагменты этапа Проектирование	41
Шаг Диаграммы анализа.....	41
Шаг Диаграммы последовательности	42
Шаг Диаграммы классов этапа проектирования	46
Шаг Схема (структура) базы данных.....	50

Предисловие

Издание содержит методические указания по выполнению домашнего задания, которое посвящено реализации этапов подготовки, анализа и проектирования процесса разработки информационной системы в рамках дисциплины «Безопасность систем баз данных» для специальностей (уровень специалитета): 10.05.01 Компьютерная безопасность, 10.05.03 Информационная безопасность автоматизированных систем .

Цель домашнего задания состоит в получении студентами первичных навыков практической реализации этапов подготовки, анализа и проектирования процесса разработки информационной системы и разработки сопроводительной документации.

Главной задачей настоящего домашнего задания является формирование у студентов практических навыков грамотного проектирования информационной системы и разработки проектной документации с использованием языка UML в среде свободно распространяемых CASE-средств на примере реализации этапов подготовки, анализа и проектирования процесса разработки информационной системы.

Издание включает, задание на разработку проектной документации этапов подготовки, анализа и проектирования процесса разработки информационной системы, краткое изложение последовательности шагов этапов подготовки, анализа и проектирования процесса разработки информационной системы, требования к содержанию расчетно-пояснительной записки, примеры возможных тем домашнего задания, примеры реализации шагов этапов подготовки, анализа и проектирования процесса разработки информационной системы, вопросы для самоконтроля.

Для успешной защиты домашнего задания необходимо сделать краткое сообщение о проделанной работе, продемонстрировать реализованные примеры диаграмм (диаграмм классов предметной области, диаграмм вариантов

использования, диаграмм анализа, диаграмм последовательности, диаграмм классов этапа проектирования, диаграмм классов базы данных), ответить на вопросы.

После выполнения домашнего задания студенты смогут достаточно оперативно осуществлять разработку необходимой проектной документации этапов подготовки, анализа и проектирования процесса создания имеющих практическую ценность компактных информационных систем с использованием Case-средств и универсального языка моделирования UML.

Введение

В настоящее время информационные системы применяются практически во всех организациях независимо от направления их деятельности и количества сотрудников. При этом не только постоянно расширяются области применения информационных систем, но и активно обновляются технологии и средства, используемые профессиональными разработчиками при создании информационных систем новых поколений.

Поэтому в рамках учебного процесса будущим разработчикам информационных систем необходимо получить такой объем знаний и первичных практических навыков, который бы позволял им в сжатые сроки самостоятельно или небольшим коллективом создавать информационные системы, имеющие практическую ценность.

Совокупность перечисленных факторов делает актуальным выполнение домашнего задания, результатом которого будет достаточно качественная проектная документация этапов подготовки, анализа и проектирования процесса создания компактной информационной системы с видимым приобретенным практическим навыком разработки некоторых типов проектной документации.

1. Теоретическая часть

В этой части дано краткое изложение последовательности шагов этапов подготовки, анализа и проектирования процесса разработки информационной системы, а также полезные краткие рекомендации.

Разработку информационной системы Вам надо начинать с Этапа подготовки.

1. Этап подготовки Вам следует начинать со словесного описания (в свободной форме) предметной области, выделяя для наглядности жирным шрифтом слова (или фразы из двух или трех слов), которые в дальнейшем с большой вероятностью можно будет рассматривать в качестве понятий (сущностей, объектов, классов) предметной области. Вам надо описывать только ту часть предметной области, которая относится к создаваемой информационной системе. Этот шаг является фактически самым главным шагом при разработке информационной системы. Давая словесное описание предметной области обязательно надо иметь в виду следующее. В задании в учебных целях предполагается, что в описываемой предметной области (т.е. в магазине, на складе, в подразделении предприятия, в гостинице, в мастерской, в кинотеатре и т.д.) нет компьютеров. Поэтому (хозяйственная) деятельность в выбранной Вами предметной области организована с помощью использования только бумажных документов (журналы, каталоги, списки, справочники и т.д.) и поэтому при словесном описании обратите внимание на тот факт, что Ваше словесное описание предметной области и создаваемая Вами концептуальная модель предметной области должны содержать в основном бумажные документы, поскольку только они подлежат компьютеризации в информационной системе. Следовательно, только бумажные документы позволяют Вам разработать информационную систему. Если в Вашем словесном описании предметной области без компьютеров нет бумажного документооборота, то и нет необходимости в разработке информационной

системы. Вы там не нужны и Вам для курсового проекта надо выбирать другую предметную область, в которой Вы сможете описать бумажный документооборот с приемлемым для разработки информационной системы количеством бумажных документов.

2. Затем Вам надо составить глоссарий предметной области. На данном шаге необходимо описать основные термины и понятия, которые встречаются в словесном описании Вашей предметной области. Вам следует особое внимание уделять тем терминам и понятиям, которые важны с точки зрения создаваемой информационной системы, т.е. названиям бумажных документов, основным понятиям, фигурирующим в бумажных документах, действующим лицам, формирующим бумажные документы и выполняющих дальнейшие действия с бумажными документами.

3. После этого Вам надо создать документ Видение, в котором описывается видение выявленной Вами значимой проблемы в предметной области и предложения по решению этой значимой проблемы. Этот документ фактически обосновывает необходимость разработки предлагаемой Вами информационной системы.

На этом Вы закончите Этап подготовки и приступите к Этапу анализа.

4. Этап анализа Вам надо начинать с разработки концептуальной модели предметной области. Концептуальная модель является формализованным описанием предметной области, следовательно она должна содержать основные понятия предметной области. В концептуальную модель Вы можете включать, только те понятия предметной области, которые фигурируют в составленном Вами словесном описании Вашей предметной области и в глоссарии предметной области. Эти понятия являются классами в концептуальной модели предметной области, а классы в UML обозначаются прямоугольниками, следовательно, концептуальная модель предметной области является диаграммой классов.

5. Затем Вам надо сформулировать функциональные требования к информационной системе. Здесь Вам рекомендуем сначала сформулировать

несколько больше функциональных требований к информационной системе (предварительных функциональных требований), а затем выделить основные (ключевые) на Ваш взгляд функциональные требования.

6. Только после этого Вы можете приступить к разработке диаграмм вариантов использования (прецедентов, сценариев), имея в виду, что диаграммы прецедентов являются визуальной формой представления сформулированных Вами основных (ключевых) функциональных требований к информационной системе. Следовательно, каждое основное (ключевое) функциональное требование к информационной системе должно быть представлено на диаграмме вариантов использования в виде варианта использования.

7. Вот только теперь Вы можете приступить к краткому, а затем подробному описанию вариантов использования (прецедентов, сценариев).

8. Затем Вам надо спроектировать пользовательский интерфейс. На этом шаге Вам надо создать шаблоны (макеты) пользовательского интерфейса, а также описать возможные переходы между страницами или окнами проектируемой системы. Шаблоны (макеты) пользовательского интерфейса Вам надо создавать на основе анализа Ваших подробных описаний вариантов использования (прецедентов, сценариев) разрабатываемой Вами информационной системы.

На этом Вы закончите Этап анализа и приступите к Этапу проектирования.

9. Этап проектирования Вы обязательно должны начинать с разработки диаграмм анализа (раздел Диаграммы анализа), неукоснительно выполняя правила построения диаграмм анализа. Если Вы нарушите правила построения диаграмм анализа и не построите достаточного количества диаграмм анализа, то Вы построите неправильные диаграммы последовательности. На диаграмме анализа изображают объекты (экземпляры классов), участвующие в варианте использования, и способы взаимодействия этих объектов. Диаграммы анализа облегчают выявление объектов, которые необходимы для выполнения варианта использования. На диаграммах анализа объекты вместо обычных в UML символов объектов изображаются пиктограммами трех видов (см. рис. 1).



Рис. 1. Символы на диаграммах анализа

Граничные объекты актеры используют для взаимодействия с системой.

Сущностные объекты – это обычно объекты из концептуальной модели предметной области. Управляющие объекты (обычно называются *контроллерами*, так как им ничего не соответствует в реальном мире) выполняют функции «клея» между граничными и сущностными объектами.

Существуют четыре основных правила:

1. Актеры могут общаться только с граничными объектами.
2. Граничные объекты могут общаться только с актерами и контроллерами.
3. Сущностные объекты могут общаться только с контроллерами.
4. Контроллеры могут общаться с граничными объектами, сущностными объектами и другими контроллерами, но не с актерами.

На рис. 2 показаны правила построения диаграмм анализа.

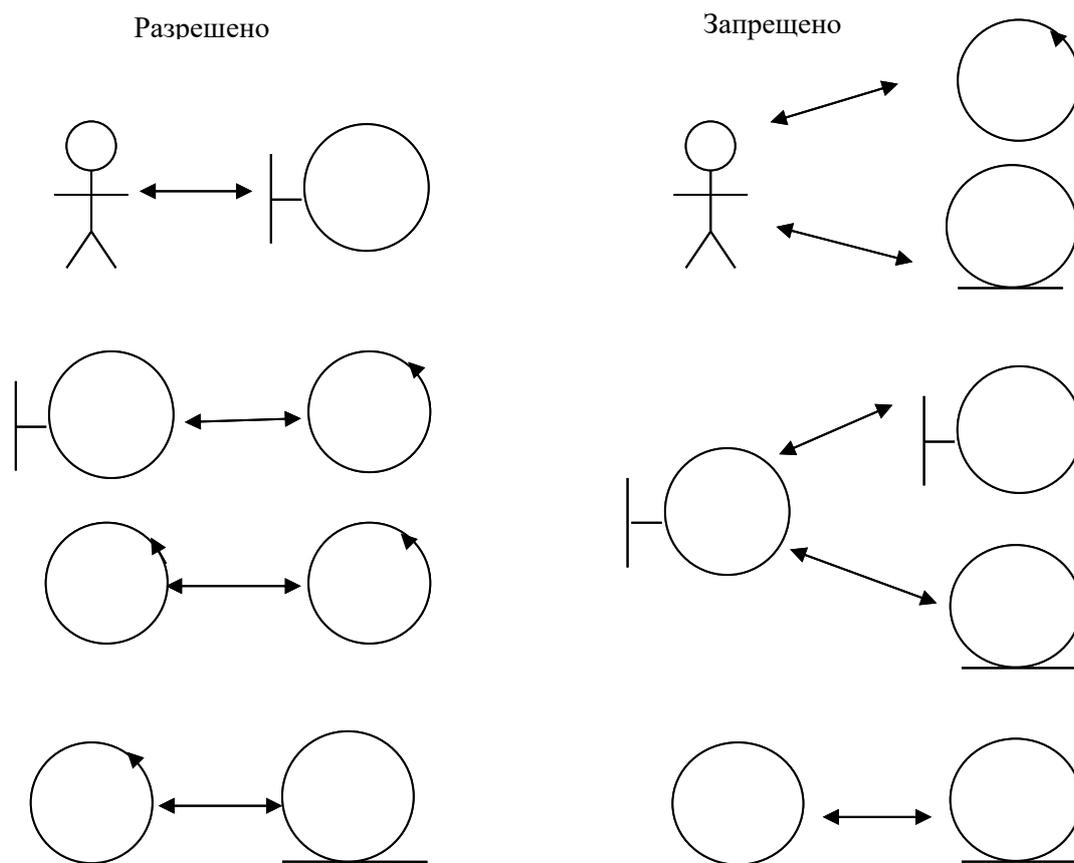


Рис 2. правила построения диаграмм анализа

Разделение объектов диаграммы анализа на три вида (границные, сущностные, управляющие) позволяет предположить, что будущая программная система может иметь трехуровневую архитектуру, которая предписывает необходимость четкого разделения программной системы на три уровня:

- уровень представления (интерфейса),
- уровень приложения (бизнес-логики),
- уровень хранения данных (базы данных).

К граничным объектам (уровень представления) относятся: экранные формы, диалоговые окна, меню. Если у разработчиков есть прототип графического интерфейса пользователя, то граничные объекты легко можно извлечь из текстов вариантов использования.

Сущностные объекты (уровень хранения данных) часто отображаются на таблицы базы данных и файлы, содержащие информацию, которая должна

«пережить» время выполнения варианта использования. Отдельные сущностные объекты, например, результаты поиска, являются временными и исчезают, когда вариант использования завершается. Многие сущностные объекты приходят из модели предметной области.

Управляющие объекты (уровень приложения) не только выполняют функции «клея» между граничными и сущностными объектами, но и заключают в себе логику приложения или бизнес-логику (например, вычисление чего-то, получение списка чего-либо, удаление или создание чего-то).

Согласно правилам построения диаграмм анализа не допускаются на уровне представления вызовы методов и инициализации у классов, относящихся к уровню приложения; исключением являются лишь контроллеры, которые должны передавать информацию нужным классам. Следовательно, если необходимо что-то посчитать, получить список чего-либо, что-то удалить или создать, то сначала нужно с помощью контроллера уведомить об этом уровень приложения, а уж затем классы уровня приложения сделают то, что необходимо.

10. Затем Вы должны построить диаграммы последовательности, которые позволят Вам выявить все классы (и их методы), которые необходимы для того, чтобы разрабатываемая информационная система удовлетворяла предъявленным к ней требованиям.

11. После этого Вам надо построить диаграмму классов этапа проектирования. Вам надо будет все классы, входящие в диаграмму классов этапа проектирования, разбить на три группы:

- граничные классы,
- управляющие классы (контроллеры),
- сущностные классы.

12. После этого Вам надо будет разработать схему (структуру) базы данных, используя все сущностные классы этапа проектирования. При разработке структуры базы данных необходимо учитывать и применять положения теории реляционных баз данных, касающиеся, в частности, нормализации структур. В

результате работ, выполненных Вами на данном шаге, у Вас должна появиться схема (структура) базы данных и спецификация таблиц, полей таблиц и отношений между таблицами.

На этом Вы закончите Этап проектирования.

В Приложении Иллюстративные (Учебные) примеры

приведены примеры, полезные для Вас при реализации шагов этапов подготовки, анализа и проектирования процесса разработки информационной системы.

2. Задание

Необходимо выполнить и описать шаги этапов подготовки, анализа и проектирования процесса разработки информационной системы с использованием объектно-ориентированных технологий.

Предметная область, назначение, режимы, алгоритмы обработки информации, область данных системы назначаются исполнителем и согласовываются с преподавателем.

В целях закрепления навыков проектирования необходимо обеспечить выполнение следующего требования к базе данных:

описание предметной области должно содержать не менее 5 - 6 понятий, которые, затем фигурируют в глоссарии предметной области, в концептуальной модели предметной области, в виде сущностных классов в диаграмме классов этапа проектирования, в виде таблиц в схеме базы данных. Объем расчетно-пояснительной записки 30 – 35 листов формата А4.

Примеры возможных тем домашнего задания

- 1 Информационная система автосалона
- 2 Информационная система школы «Электронный дневник»

- 3 Информационная система пункта гарантийного обслуживания бытовой техники
- 4 Информационная система магазина продажи бытовой техники
- 5 Информационная система сети ресторанов
- 6 Информационная система учета штрафов ГИБДД
- 7 Информационная система сети кинотеатров
- 8 Информационная система туристической фирмы
- 9 Информационная система музыкальных событий города
- 10 Информационная система театра
- 11 Информационная система городской библиотеки
- 12 Информационная система оператора сотовой связи
- 13 Информационная система больницы
- 14 Информационная система сети аптек
- 15 Информационная система сети продуктовых магазинов
- 16 Информационная система сети магазинов одежды
- 17 Информационная система поликлиники
- 18 Информационная система военно-исторического архива
- 19 Информационная система союза малых городов России
- 20 Информационная система музыкального магазина
- 21 Информационная система страховой компании
- 22 Информационная система компьютерных курсов
- 23 Информационная система курсов робототехники
- 24 Информационная система кафедры
- 25 Информационная система международного турнира по бальным танцам
- 26 Информационная система ломбарда
- 27 Информационная система «Студенческий форум»
- 28 Информационная система музея
- 29 Информационная система детской школы искусств
- 30 Информационная система парка городского транспорта

- 31 Информационная система почтового отделения
- 32 Информационная система продуктовой базы
- 33 Информационная система строительной фирмы
- 34 Информационная система автосервиса
- 35 Информационная система спортивных соревнований
- 36 Информационная система банковских вкладов физических лиц
- 37 Информационная система книжного магазина
- 38 Информационная система продажи железнодорожных билетов
- 39 Информационная система автовокзала
- 40 Информационная система «Московский метрополитен»
- 41 Информационная система заповедника
- 42 Информационная система детской спортивной школы
- 43 Информационная система гостиницы
- 44 Информационная система спортивного магазина
- 45 Информационная система магазина для художника
- 46 Информационная система магазина ткани
- 47 Информационная система магазина по продаже компьютерных игр
- 48 Информационная система магазина по продаже художественных произведений
- 49 Информационная система клуба по автомобильному спорту
- 50 Информационная система Центра реабилитации слепых и слабовидящих детей
- 51 Информационная система речного вокзала
- 52 Информационная система продаж авиабилетов
- 53 Информационная система авиакомпании
- 54 Информационная система хлебозавода
- 55 Информационная система фитнес центра
- 56 Информационная система кондитерской фабрики
- 57 Информационная система медиа-издательства

- 58 Информационная система научной конференции
- 59 Информационная система общества филателистов
- 60 Информационная система зоомагазина

3. Требования к содержанию расчетно-пояснительной записки

Содержание расчетно-пояснительной записки.

Титульный лист, который является первой страницей и заполняется строго по определенной форме.

Аннотация, которая отражает краткое содержание домашнего задания.
оглавление;

Введение, в котором обосновывается актуальность выбранной темы, формулируются цель работы и содержание поставленной задачи.

Основной раздел, который содержит исследовательскую часть.

В исследовательской части рекомендуется изложить материал этапов

Подготовка (описание предметной области, глоссарий предметной области, документ Видение), Анализ (концептуальная модель предметной области, требования к информационной системе, диаграммы вариантов использования, краткое и полное описание вариантов использования, пользовательский интерфейс), Проектирование (диаграммы анализа, диаграммы последовательности, диаграммы классов уровня проектирования, структура базы данных).

Заключение, которое должно содержать основные выводы по выполненному домашнему заданию, практическую ценность и перспективы развития разрабатываемой информационной системы.

Список литературы, который должен содержать все использованные источники, в том числе доступные в Интернете электронные версии учебников, учебных пособий, монографий и др.

При оформлении расчетно-пояснительной записки соблюдайте требования ГОСТ на оформление текстовой документации. Например, все схемы, рисунки, таблицы должны быть снабжены названиями и пронумерованы.

Грамматические ошибки влияют на оценку домашнего задания.

При разработке проекта возможно использование различных редакторов и других средств CASE.

Расчетно-пояснительную записку в формате .docx, студент должен перед защитой домашнего задания послать преподавателю по его электронному адресу.

4. Контрольные вопросы

1. Основная идея объектно-ориентированного анализа и проектирования.
2. Шаги этапа подготовки
3. Шаги этапа анализа
4. Шаги этапа проектирования
5. Диаграммы прецедентов
6. Диаграммы анализа
7. Глоссарий предметной области
8. Функциональные требования к системе
9. Технические требования к системе
10. Расширение и обобщение прецедентов
11. Концептуальная модель
12. Диаграммы последовательностей
13. Диаграмма кооперации
14. Диаграмма классов
15. Схема (структура) базы данных

5. Требования к защите

На защиту студентом предоставляются расчетно-пояснительная записка. Студент делает краткое сообщение о проделанной работе, которое включает постановку задачи, обзор выполненной работы, примеры диаграмм (диаграмм классов предметной области, диаграмм вариантов использования, диаграмм анализа, диаграмм последовательности, диаграмм классов этапа проектирования), выводы, а также отвечает на вопросы.

6. Список рекомендуемой литературы

1. Скотт К. UML. Основные концепции. – М.: Издательский дом «Вильямс», 2002. -144 с.
2. Мацяшек Л.А. Т. Анализ требований и проектирование систем: Разработка информационных систем с использованием UML. –М.: Издательский дом «Вильямс», 2002.-432 с.
3. Розенберг Д., Скотт К. Применение объектного моделирования с использованием UML и анализ прецедентов: Пер. с англ. –М.: ДМК Пресс, 2002. – 160 с.
4. Советов Б. Я., Цехановский В. В., Чертовской В. Д. Базы данных: теория и практика : учебник для вузов / Советов Б. Я., Цехановский В. В., Чертовской В. Д. - 2-е изд. - М. : Юрайт, 2014. - 462 с. : ил. - (Бакалавр. Базовый курс). - Библиогр.: с. 459-460. - ISBN 978-5-9916-2940-9, <https://biblio-online.ru/book/bazy-dannyh-382861>.
5. Карпова И. П. Базы данных. Курс лекций и материалы для практических занятий : учеб. пособие для вузов / Карпова И. П. - СПб. : Питер, 2018. - 240 с. : ил. - (Учеб. пособие). - Библиогр.: с. 233-234. - ISBN 978-5-496-00546-3.
6. Григорьев Ю. А., Плутенко А. Д., Плужникова О. Ю. Реляционные базы данных и системы NoSQL : учеб. пособие / Григорьев Ю. А., Плутенко А.

- Д., Плужникова О. Ю. ; Амурский гос. ун-т, МГТУ им. Н. Э. Баумана. - Благовещенск : Изд-во Амурского гос. ун-та, 2018. - 420 с. : ил. - Библиогр.: с. 417-420. - ISBN 978-5-93493-308-2.
7. Ревунков Г. И., Ковалева Н. А., Силантьева Е. Ю. Проектирование баз данных : учеб. пособие / Ревунков Г. И., Ковалева Н. А., Силантьева Е. Ю. ; МГТУ им. Н. Э. Баумана. - М. : Изд-во МГТУ им. Н. Э. Баумана, 2018. - 45 с. : ил. - Библиогр.: с. 19. - Режим доступа: <http://ebooks.bmstu.ru/catalog/254/book1681.html> (дата обращения: 26.06.2018). - ISBN 978-5-7038-4718-3.
 8. Боровик И. Г. Управление данными [Электрон. ресурс] : метод. указания к курсовому проектированию / Боровик И. Г. ; МГТУ им. Н. Э. Баумана. - М. : Изд-во МГТУ им. Н. Э. Баумана, 2009. - ФГУП "Информрегистр" №0320901015.
 9. В.В. Кириллов. Основы проектирования реляционных баз данных. СУБД - Учебные пособия и обзоры. ЦИТ. — учебное пособие СПБИТМО, 2011.
 10. Документация Microsoft Developer Network. Программные интерфейсы управления базами данных (ODBC, DAO, ActiveX Data Objects).2011.
 11. Дейт К. Дж. Введение в системы баз данных = Introduction to Database Systems — 9-е изд. — М.: Вильямс, 2006. — 1328 с. — ISBN 5-8459-0788-8 (рус.) 0-321-19784-4 (англ.).
 12. Петкович Д. Microsoft SQL Server 2008. Руководство для начинающих — С.: БХВ-Петербург, 2009. — С. 752. — ISBN 978-5-9775-0149-1.
 13. Роберт Э. Уолтерс, Майкл Коулс SQL Server 2008: ускоренный курс для профессионалов = Accelerated SQL Server 2008 — М.: «Вильямс», 2008. — С. 768. — ISBN 978-5-8459-1481-1.

Приложение 1. Иллюстративные (Учебные) примеры

В этом приложении приведены в основном фрагменты разработки иллюстративного (учебного) примеров «Информационная система свадебного салона» на этапах Подготовка, Анализ, Проектирование . Начнем с фрагментов этапа Подготовка.

П1.1. Фрагменты этапа Подготовка

Этап Подготовка включает шаги Описание предметной области, Глоссарий предметной области, Видение. Поэтому первым шагом в процессе создания информационной системы является описание предметной области, для которой разрабатывается информационная система.

Шаг Описание предметной области

Описание предметной области информационной системы свадебного салона

Свадебный салон является агрегатором по предоставлению некоторых услуг сетью салонов красоты.

Приведем полезный поясняющий рисунок (см. рис. П1.1).

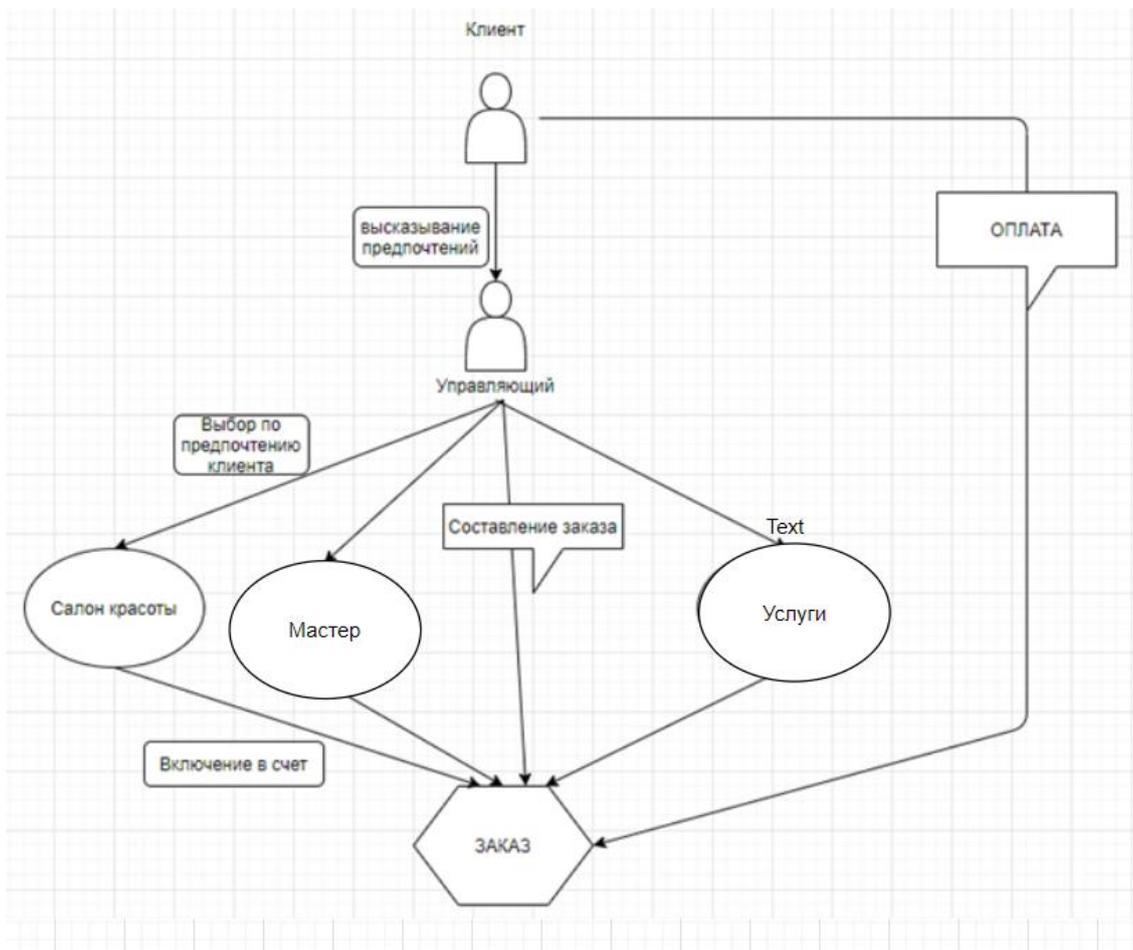


Рис. П1.1. Предметная область

Чтобы точнее понять предметную область информационной системы рассмотрим, в чём состоит работа свадебного салона.

Клиент, который хочет получить услуги свадебного салона, обращается к **управляющему (или администратору)**. В соответствии с пожеланиями клиента он выбирает удобный **салон красоты**, узнает, свободен ли он в нужное время. В соответствии с пожеланиями клиента он также выбирает **услуги** и **мастера**, который окажет ему данные услуги. Управляющий составляет **заказ** и записывает клиента в определенный салон красоты, к определенному мастеру в определенную дату и время.

Суть или особенность свадебного салона состоит в том, что любой мастер на выбор клиента приезжает в любой **салон красоты** на выбор клиента в

удобное для клиента время, то есть мастер не привязан к какому-либо конкретному салону красоты.

Имеются следующие особенности:

- Свадебный салон сотрудничает со многими салонами красоты.
- Свадебный салон предлагает различные варианты услуг, которые могут понадобиться невесте.
- Свадебный салон предлагает разных мастеров.
- Общая стоимость складывается из всех выбранных вариантов из вышеперечисленных трёх пунктов.

В настоящее время работа такого свадебного салона никак не автоматизирована. Таким образом, в настоящее время администратор свадебного салона не может как-либо систематизировать информацию о разных заказах, о занятости того или иного мастера или салона красоты.

Если проводить автоматизацию работы такого свадебного салона, то в первую очередь хотелось бы автоматизировать следующие виды работы свадебного салона:

- хранение и возможность редактирования информации о текущем состоянии занятости мастеров и салонов красоты по **общему журналу заказов;**
- хранение и возможность редактирования информации о текущих расценках на разные услуги разных мастеров в разных салонах красоты по **общему журналу расценок;**
- получение статистических отчетов по разным критериям (самых популярных, самых продаваемых, самых часто заказываемых специалистов/салонов красоты)
- хранение информации о текущих заказах

- хранение информации про каждого клиента с возможностью анализа данных каждого клиента
- хранение и возможность редактирования информации по разным салонам, мастерам и услугам
- возможно, какую-то бонусную программу для постоянных клиентов в будущем
- возможность хранения и редактирования рейтинга мастера в соответствии с отзывами клиентов об этом мастере

В проекте свадебный салон является местом, в котором можно оформить заказ на создание свадебного образа, включив в этот заказ на выбор клиента следующие пункты:

- услуги – выбор услуг разных типов (маникюр, педикюр, макияж, причёска и т.д.) в зависимости от желания клиента;
- мастер – выбор в зависимости от опыта работы, стоимости за свои услуги, примеров работ (портфолио), стиля работы и отзывов на сайте;
- салон красоты – выбор в зависимости от месторасположения, наличия свободных мест и времени, уровня сервиса и статуса салона;

Подобный свадебный салон может принадлежать отдельному предпринимателю, который работает в сфере красоты. Салоны красоты могут быть собственностью предпринимателя, а могут просто сотрудничать по договору с данным свадебным салоном для увеличения клиентов. Клиентами являются обычные люди, которые нуждаются в оказании данных услуг. Мастерами являются люди с соответствующим образованием, сотрудничающие с данным свадебным салоном. Это могут быть как самостоятельные мастера, так и мастера, работающие в салоне красоты, который сотрудничает с данным свадебным салоном.

Более подробно процесс функционирования свадебного салона можно описать следующим образом.

По определенным критериям осуществляется поиск салонов красоты и мастеров, с которыми данный свадебный салон будет сотрудничать. С ними заключается определенный договор, после чего информация по каждому мастеру и салону красоты вносится в базу данных. Каждая услуга, которую предоставляют мастера, заносится в базу данных с подробным описанием данной услуги и её стоимости у каждого мастера, который её делает.

Для записи в свадебный салон клиент должен выбрать услугу, которую хочет получить, выбрать мастера, который её сделает и салон красоты, удобный для этого клиента. Если мастер свободен в выбранную клиентом дату и время, то производится **запись клиента**.

Шаг Глоссарий предметной области

Глоссарий предметной области свадебного салона

Название термина	Описание термина
Клиент	Человек, который обращается в компанию, чтобы сделать свадебный образ.
Управляющий/администратор	Человек, который владеет или управляет салонами красоты, занимается свадебной стилистикой, ведет базу клиентов, помогает клиентам в записи.
Салон красоты	Специальное помещение, специально оборудованное для проведения процедур по созданию свадебных образов.
Услуги	Например: маникюр, педикюр, макияж, причёска и т.д.
Мастер	Человек, который является специалистом в области макияжа, маникюра, педикюра, создания

	образа с помощью специальных средств, искусства макияжа.
Цена	Цена за какую-либо отдельную услугу у определенного мастера.
Заказ	Документально зафиксированный заказ, включающий в себя информацию об месте, мастере и списка услуг, выбранных клиентом, дате и времени проведения процедур и общей стоимости создания образа
Общая стоимость	Общая стоимость за все выбранные пункты (салон красоты, тип прически и макияжа и стилист).
Каталог услуг	Картотека услуг, где для каждой услуги есть свой бланк с подробным описанием, с возможностью заполнения нового бланка услуги, редактирования уже существующей услуги, поиска и удаления бланка услуги из Каталога услуг.
Каталог мастеров	Картотека мастеров, где для каждого мастера есть свой бланк с подробным описанием, с возможностью заполнения нового бланка мастера, редактирования уже существующего мастера, поиска и удаления бланка мастера из Каталога мастеров.
Каталог салонов	Картотека салонов, где для каждого салона есть свой бланк с подробным описанием, с возможностью заполнения нового бланка салона, редактирования уже существующего салона,

	поиска и удаления бланка салона из Каталога салонов.
Каталог клиентов	Картотека клиентов, где для каждого клиента есть свой бланк с подробным описанием, с возможностью заполнения нового бланка клиента, редактирования уже существующего клиента, поиска и удаления бланка клиента из Каталога клиентов.
Журнал заказов	Журнал заказов, в котором для каждого заказа описаны точные данные клиента, услуги, мастера и салона, даты и времени проведения заказа, с возможностью регистрации нового заказа, редактирования уже существующего заказа, поиска и удаления заказа из журнала заказов.
Корзина	Интерфейс, куда пользователь может добавлять выбранные им параметры заказа (услуги, мастера, салоны и т.д.), которые он собирается заказать. В данном интерфейсе реализованы возможности добавления параметра, изменения параметра, удаления параметра и выбор нескольких параметров для добавления в заказ и последующего формирования заказа.
Поиск	Поиск среди наименований по каталогам мастеров, салонов, услуг и клиентов по любому из параметров.
Отзыв	Отзыв клиента о мастере, который получил какую-то услугу этого мастера, по 10-бальной

	шкале, основанный на качестве предоставления данной услуги.
Рейтинг	«Усредненный отзыв» - среднее арифметическое от всех отзывов клиентов для каждого мастера.
Менеджер	Человек, который владеет или управляет одним конкретным салоном красоты, отвечает за запись в своем салоне красоты.
Журнал расценок	Журнал расценок, в котором для каждого мастера описаны услуги, которые он совершает и цены за эти услуги, с возможностью регистрации новой услуги от нового мастера, редактирования уже существующих, поиска и удаления существующих из журнала расценок.

Шаг Видение

- **Совладельцы системы**

Совладельцами данной системы является:

Администратор свадебного салона. Данный совладелец ведет административные работы на торговой в салоне. Он принимает клиентов, помогает им сделать выбор из каталога услуг, записывает, бронируя для них определенные время и мастера, вносит в базу заказов и так далее. Администратор салона будет выступать в качестве единственного пользователя системы.

- **Границы системы**

Так как система является однопользовательской, то ее границы определены достаточно четко. Они определяются рабочим местом администратора свадебного салона.

- **Проблемы предметной области**

Элемент	Описание
Проблема	Трудности в выборе у клиента, в связи с ограниченным выбором специалистов, стилей и времени записи в одном салоне красоты или же разброс этих качественных услуг по разным салонам красоты, не сотрудничающим вместе
Воздействует на	Клиента и выполняющий заказы персонал
Результатом чего является	неудовлетворенность клиентов, плохое мнение о салоне, плохие отзывы, низкий рейтинг салона, что влечет за собой уменьшение прибыли.
Выигрыш от	новой системы организации записи на создание свадебного образа, направленной на решение данной проблемы
может состоять в следующем	<ul style="list-style-type: none"> - бóльшем выборе разных салонов/мастеров/услуг для клиента - удовлетворенность клиента качественной работой салона с большим выбором услуг/мастеров/времени и мест записи в рамках одной системы - повышение точности в заказах клиента (договоренность по мельчайшим деталям) - удобство для клиента в возможности организации самых лучших услуг и мастеров в одном месте в подходящее для клиента время

	- в конечном счёте, увеличение оборота клиентов и, соответственно, прибыли
--	--

- **Возможности системы**

Основными возможностями системы будут являться:

- ввод, хранение и возможность редактирования информации о заказах клиентов по всем параметрам;
- организация большого выбора услуг, мастеров, салонов красоты по разным параметрам в рамках одной системы для клиента;
- возможность анализа спроса на тот или иной вид услуг/мастеров/салонов красоты.
- возможность хранения, добавления, редактирования и удаления услуги, мастера или салона красоты из соответствующих каталогов.
- возможность поиска среди наименований по каталогам мастеров, салонов и услуг по любому из параметров.
- возможность оставления клиентами отзывов на мастеров, в зависимости от качества предоставления услуг данным мастером.
- формирование рейтинга мастера в зависимости от отзывов клиента.

П1.2. Фрагменты этапа Анализ

Шаг Концептуальная модель предметной области

Фрагмент концептуальной модели предметной области информационной системы свадебного салона

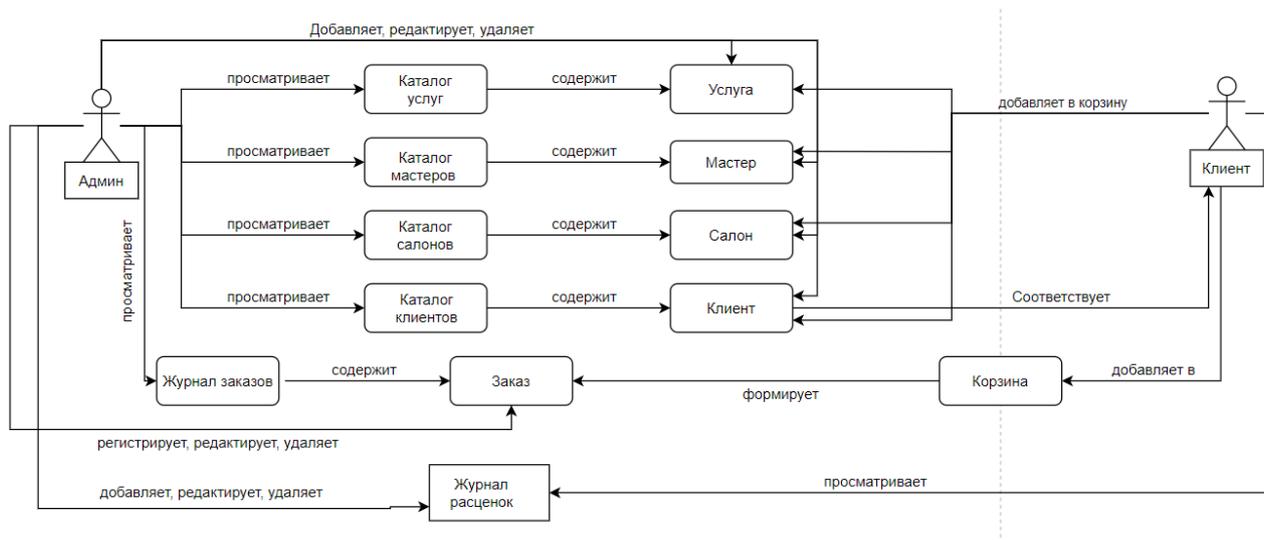


Рис. П1.2. Фрагмент концептуальной модели предметной области информационной системы свадебного салона

Шаг Функциональные требования

Фрагмент таблицы функциональных требований

Номер	Текст
FR010	Система должна позволять просматривать информацию о заказе.
FR020	Система должна позволять вводить информацию о новом заказе.
FR030	Система должна позволять удалять информацию о заказе.
FR040	Система должна позволять обновлять информацию о заказе.
FR050	Система должна позволять просматривать информацию о салонах красоты.

FR060	Система должна позволять вводить информацию о новых салонах красоты.
FR070	Система должна позволять удалять информацию о салонах красоты.
FR080	Система должна позволять обновлять информацию о салонах красоты.
FR090	Система должна позволять просматривать информацию об услугах.
FR100	Система должна позволять вводить информацию об услугах.
FR110	Система должна позволять удалять информацию об услугах.
FR120	Система должна позволять обновлять информацию об услугах.
FR130	Система должна позволять просматривать информацию о мастерах.
FR140	Система должна позволять вводить информацию о новых мастерах.
FR150	Система должна позволять удалять информацию о мастерах.
FR160	Система должна позволять обновлять информацию о мастерах.
FR170	Система должна иметь функцию отображения состояния заказа – оплачен или не оплачен.
FR180	При просмотре списка салонов необходимо предусмотреть поиск какого-либо салона по названию, цене и другой информации.

FR190	При просмотре списка услуг необходимо предусмотреть поиск варианта по коду, типу услуги, и др.
FR200	При просмотре списка мастеров необходимо предусмотреть поиск какого-либо мастера по стоимости, отзывам и другой информации.
FR210	Система должна автоматически считать общую стоимость за организацию подготовки свадебного образа.
FR220	<p>Заказ должен содержать следующую информацию:</p> <ul style="list-style-type: none"> • № заказа • ФИО клиента • Салон красоты • № Услуги • Id мастера • Время и дата • Общее время занятости • Общая стоимость • Статус оплаты
FR230	<p>База данных салона красоты должна содержать следующую информацию:</p> <ul style="list-style-type: none"> • Название салона красоты • Адрес • Телефон • ФИО менеджера • Стоимость места за час
FR240	<p>Услуги должны содержать следующую информацию:</p> <ul style="list-style-type: none"> • № услуги

	<ul style="list-style-type: none"> • Тип • Название • Время исполнения
FR250	<p>Поле «Мастер» должно содержать следующую информацию:</p> <ul style="list-style-type: none"> • Id мастера • ФИО стилиста • Телефон • Рейтинг • Опыт
FR260	<p>Поле «Клиент» должно содержать следующую информацию:</p> <ul style="list-style-type: none"> • ФИО клиента • Телефон
FR270	Система должна позволять просматривать информацию о клиентах.
FR280	Система должна позволять вводить информацию о новых клиентах.
FR290	Система должна позволять удалять информацию о клиентах.
FR300	Система должна позволять обновлять информацию о клиентах.
FR310	При просмотре списка клиентов необходимо предусмотреть поиск какого-либо клиента по ФИО, телефону и другой информации.
FR320	При просмотре журнала заказов необходимо предусмотреть поиск какого-либо заказа по ФИО клиента, мастеру и другой информации.

Шаг Диаграммы вариантов использования

Примеры диаграмм вариантов использования информационной системы свадебного салона

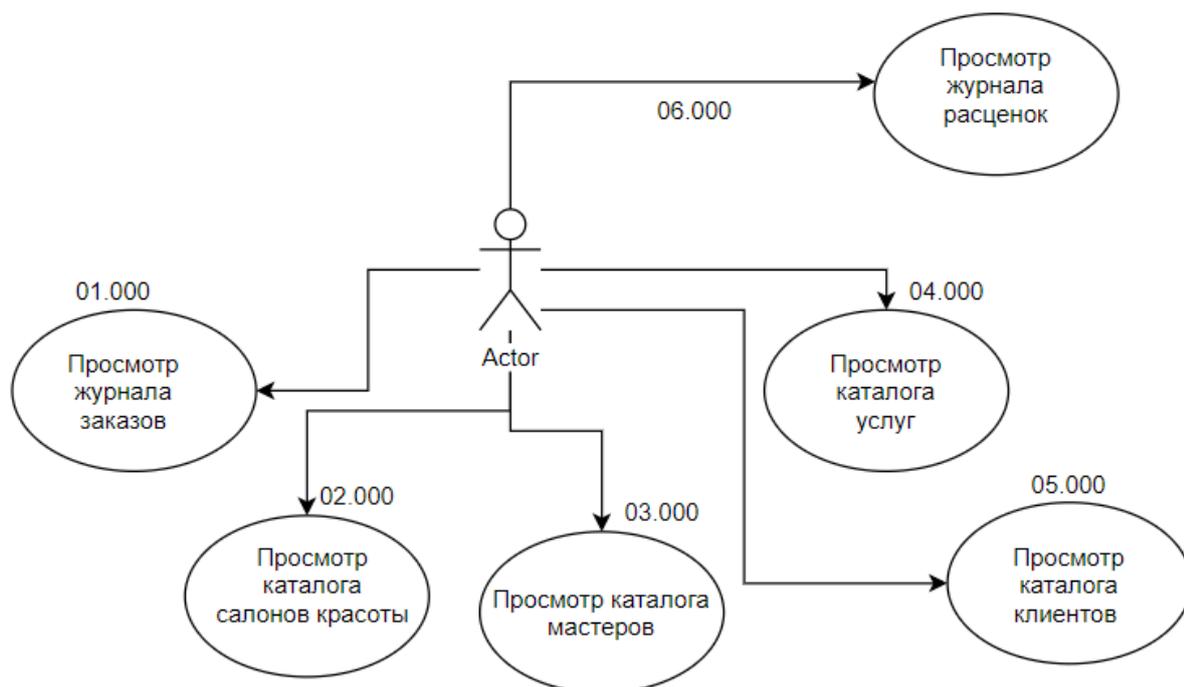


Рис.П1.3. Неполная диаграмма вариантов использования системы для администратора свадебного салона

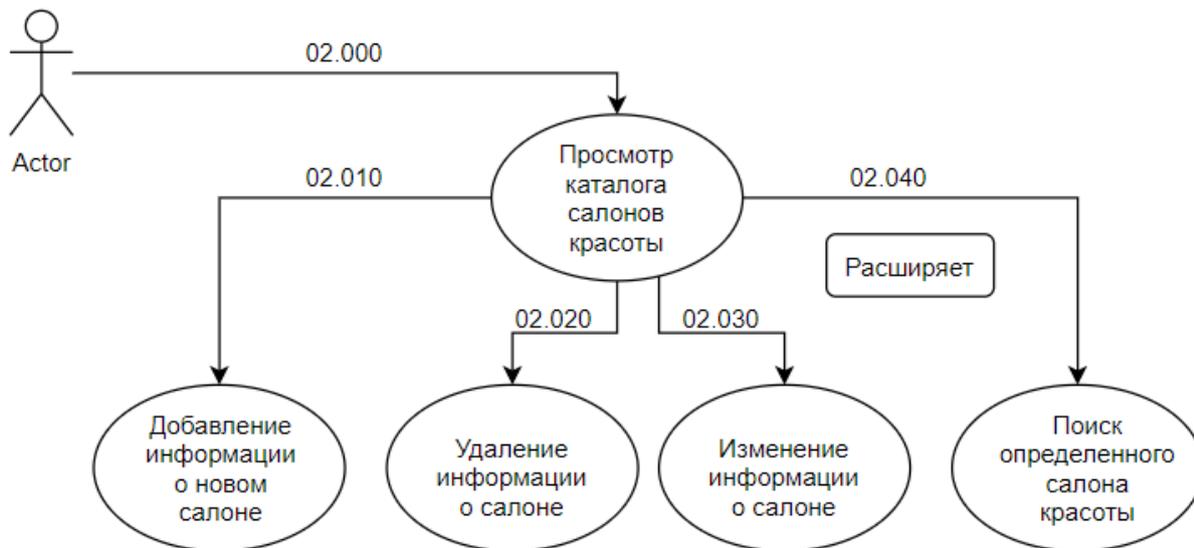


Рис. П1.4. Диаграмма варианта использования (прецедента) Просмотр списка салонов красоты

Шаг Краткое и подробное описание вариантов использования

Примеры краткого и подробного описания вариантов использования (прецедентов)

Просмотр списка салонов красоты

Описание: Данный прецедент описывает процесс просмотра актером списка салонов красоты, которые сотрудничают с нашей компанией и в которых можно организовать создание свадебного образа с указанием для каждого салона адреса, стоимости аренды места за час и другой информации.

Требования: FR050, FR180

Основной ход событий.

Действия актера	Отклик системы
1. Актер вызывает список салонов красоты или вызывает функцию «Просмотреть все салоны красоты» при просмотре результатов поиска.	2. Система отображает на экране отсортированным по названию салона красоты список салонов красоты в виде: <ul style="list-style-type: none"> • Название • Адрес • Телефон • ФИО менеджера • Стоимость места в час
3. Актер просматривает список салонов красоты	

Альтернативный ход событий.

Происходит непредвиденный сбой при отображении списка. Система отображает актеру сообщение об ошибке «Произошел непредвиденный сбой при отображении списка салонов красоты. Список салонов красоты не может быть отображен». Актер подтверждает прочтение сообщения. Список салонов отображается пустым. Выполнения варианта использования заканчивается.

Точки расширения.

Название	Описание	Вызываемый вариант использования
Форма поиска		02.040 Поиск определенного салона красоты

Элемент управления «Редактировать информацию»	При вызове сценария учитывается, какой салон красоты выбран в списке.	02.030 Изменение информации о салоне
Элемент управления «Новый салон красоты»		02.010 Добавление информации о новом салоне
Элемент управления «Удалить»	При вызове сценария учитывается, какой салон красоты выбран в списке.	02.020 Удаление информации о салоне красоты

Поиск определенного салона красоты 02.040

Описание: Данный прецедент может быть вызван актером при просмотре списка салонов красоты. В прецеденте описывается процесс поиска, определенного салона красоты по его названию или другим критериям, например, стоимость аренды за час. При этом ищутся не удаленные салоны красоты, содержащие введенную строку.

Требования: FR180

Основной ход событий.

Действия актера	Отклик системы
1. В специальном поле актер вводит название салона красоты или адрес	2. Система находит все салоны красоты, в информации о которых содержится введенные данные.

или другой критерий поиска и вызывает функцию поиска	
	3. Система отображает актеру список найденных салонов красоты

Альтернативный ход событий.

Актер сделал ошибку при вводе названия. Система выдает сообщение об ошибке. Актер подтверждает прочтение сообщения. Выполнение варианта переходит на пункт.

Изменение информации о салоне красоты 02.030

Описание: Данный прецедент может быть вызван при просмотре списка салонов красоты. Этот прецедент описывает процесс редактирования информации о салоне красоты, например, изменение стоимости аренды места за час. При этом информация о салоне красоты обновляется.

Требования: FR080

Основной ход событий.

Действия актера	Отклик системы
1. Актер вызывает функцию изменения информации о салоне красоты.	2. Система отображает форму для изменения информации о салоне красоты, при этом она заполнена текущей информацией.

3. Актер вносит изменения о салоне красоты и сохраняет их.	4. Система проверяет, правильно ли заполнены данные: стоимость аренды места за час > 0, не введено ли пустое название салона красоты и др.
	5. Условие выполняется. Система обновляет информацию о аренды места за час.

Альтернативный ход событий.

Актер отказывается от изменений информации о салоне красоты. Выполнение сценария на этом заканчивается.

Данные введены неправильно. Система выдает сообщение об ошибке. Актер подтверждает прочтение сообщения. Выполнение сценария переходит на пункт 3.

Шаг Проектирование пользовательского интерфейса

Пример дерева функций

Дерево функций представляет собой иерархическую структуру действий, реализованных в информационной системе. Состав и классификация функций разрабатываемой системы представлены в виде дерева функций на рис. П1.5.

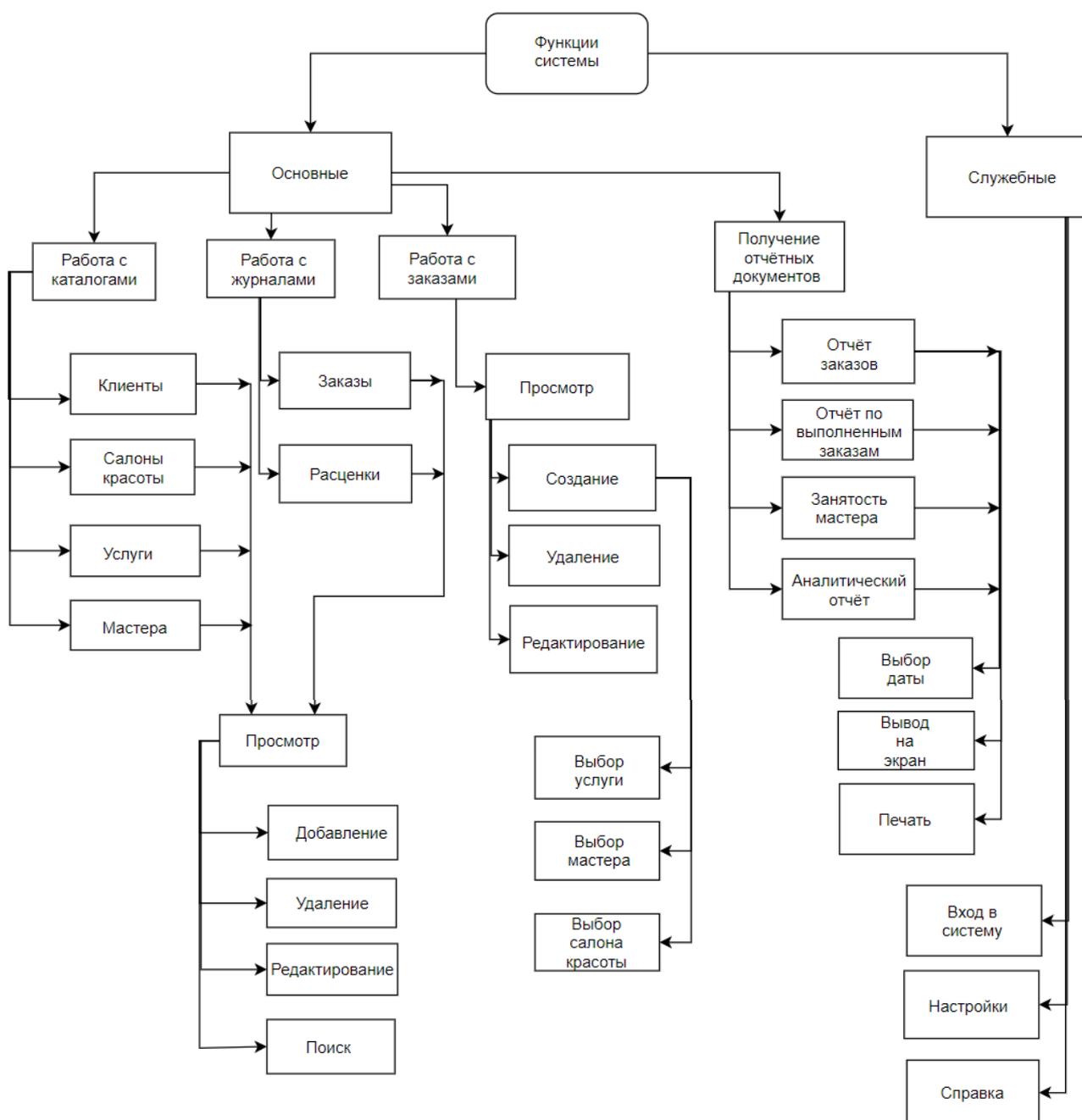


Рис. П1.5. Пример дерева функций информационной системы

Пример сценария диалога с информационной системой

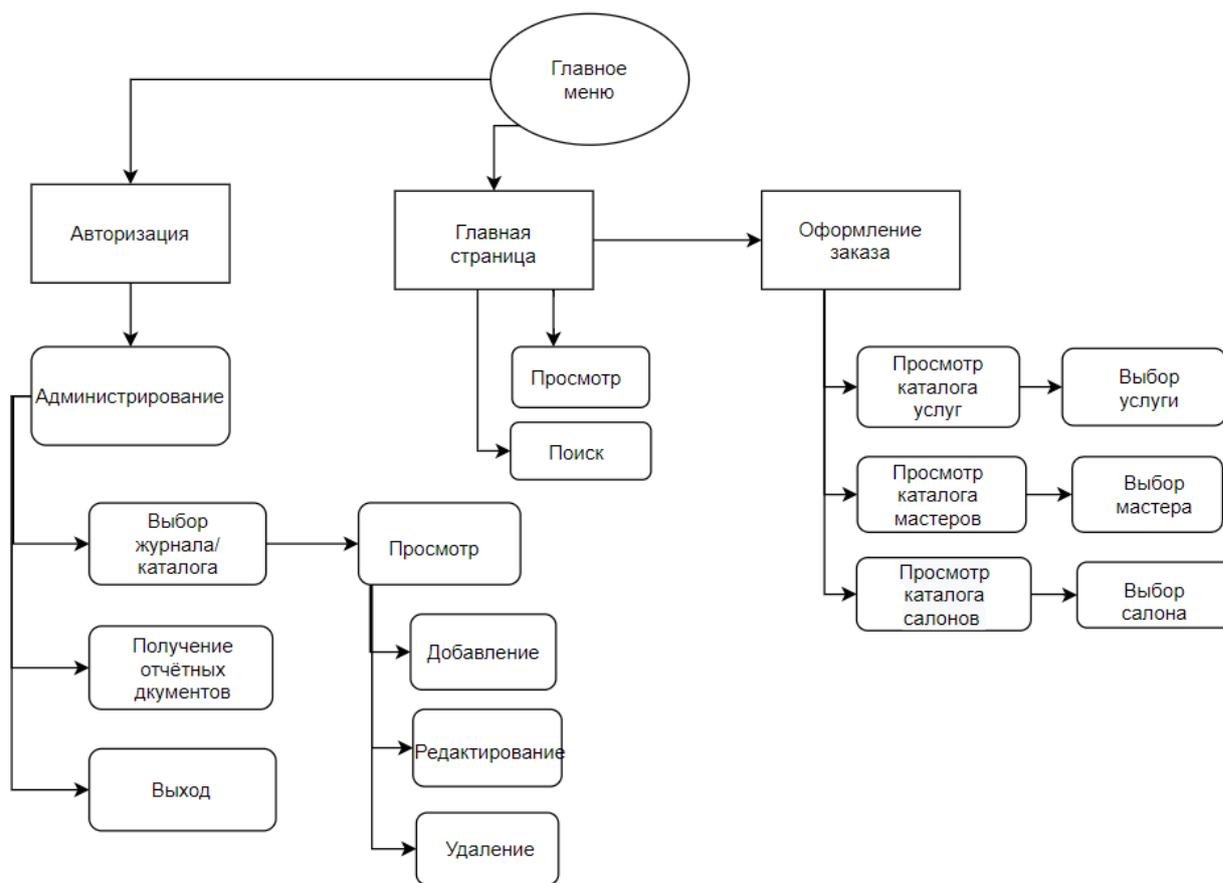


Рис. П1.6. Пример сценария диалога с системой

Примеры макетов интерфейсов

Технически система может состоять, например, из разделов:

- Пользовательский (режим просмотра);
- Административный (режим редактирования);

Пример макета страницы пользовательского раздела

	Свадебный салон «Капризная невеста»
Главное меню	Содержимое выбранного раздела ...
Элемент меню	
Элемент меню	

Элемент меню	
Вход в систему	
Логин: <input style="width: 50px;" type="text"/>	
Пароль: <input style="width: 50px;" type="text"/>	

Рис. П1.7. Пример макета страницы пользовательского раздела

Пример макета страницы административного раздела

<i>Административный раздел</i>				
Выберите опцию:				
Редактирование	Редактирование	Редактирование	...	Редактирование
1	2	3		n

Рис. П1.8. Пример макета страницы административного раздела

При выборе одной из опций появляется окно редактирования, например, изменения или добавления данных в базу данных (см. рис. ниже).

<i>Административный раздел</i>				
Редактирование таблицы N				
Поле 1	Поле 2	Поле 3	Поле 4	Поле 5

Форма для добавления записи:	
Поле 1	<Ввод данных>
Поле 2	<Ввод данных>
...	...
Назад в административный раздел	

Рис.П1. 9. Пример макета страницы административного раздела подраздела редактирования

П1.3. Фрагменты этапа Проектирование

Шаг Диаграммы анализа

На диаграмме анализа изображают:

- объекты, участвующие в варианте использования,
- способы взаимодействия этих объектов.

Это диаграмма, на которой вместо обычных в UML символов классов изображаются пиктограммы трех видов, представленных на рисунке ниже. Граничные объекты актеры используют для взаимодействия с системой. Сущностные объекты – это обычно объекты из модели предметной области. Управляющие объекты (обычно называются контроллерами, так как им ничего не соответствует в реальном мире) выполняют функции «клея» между граничными и сущностными объектами.



Рис. П1.10. Символы на диаграммах анализа

На рис. П1.11 приведен иллюстративный пример диаграммы анализа для варианта использования «Изменение информации о заказе 01.030»

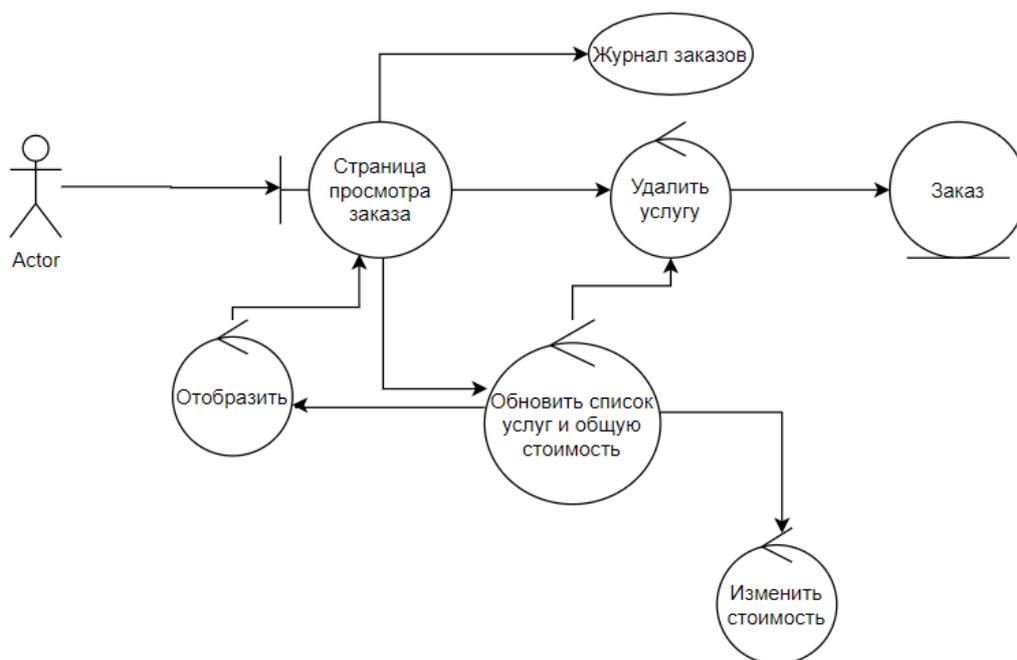


Рис.П1.11. Диаграмма анализа варианта использования «Изменение информации о заказе 01.030»

Шаг Диаграммы последовательности

Диаграммы последовательности отражают поток событий (бизнес-логику), происходящих в рамках варианта использования. С помощью диаграмм

взаимодействия можно определить классы, которые нужно создать, связи между ними, а также операции и ответственности каждого класса.

Диаграммы последовательностей состоят из последовательных откликов системы на различные действия пользователя, взаимодействия отдельных элементов системы между собой и вариантами ответа за запросы.

Диаграммы последовательности упорядочены по времени. Они полезны для того, кто хочет понять логическую последовательность событий в сценарии.

Пример диаграммы последовательности для варианта использования «Просмотр каталога услуг 04.000» представлен на рис. П1.12.

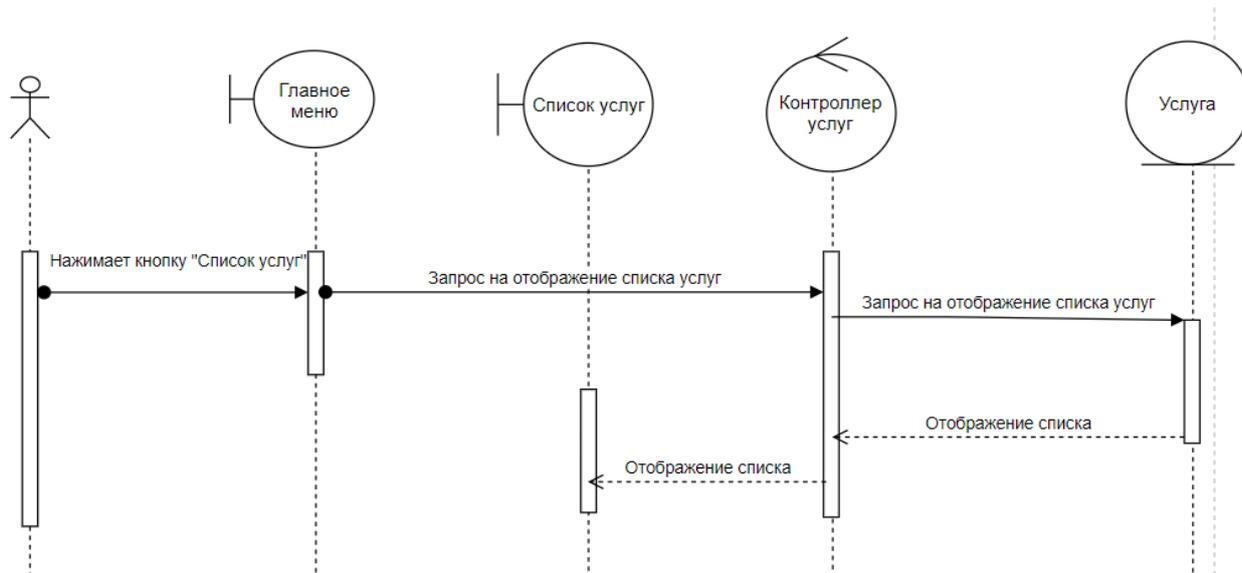


Рис. П1.12 Диаграмма последовательности для прецедента «Просмотр каталога услуг 04.000»

Далее приведены диаграммы последовательности для прецедентов «Просмотра списка объектов», «Редактирования», «Создания нового», «Удаления», «Поиска» в общем виде. (Рисунки 20-24 соответственно).

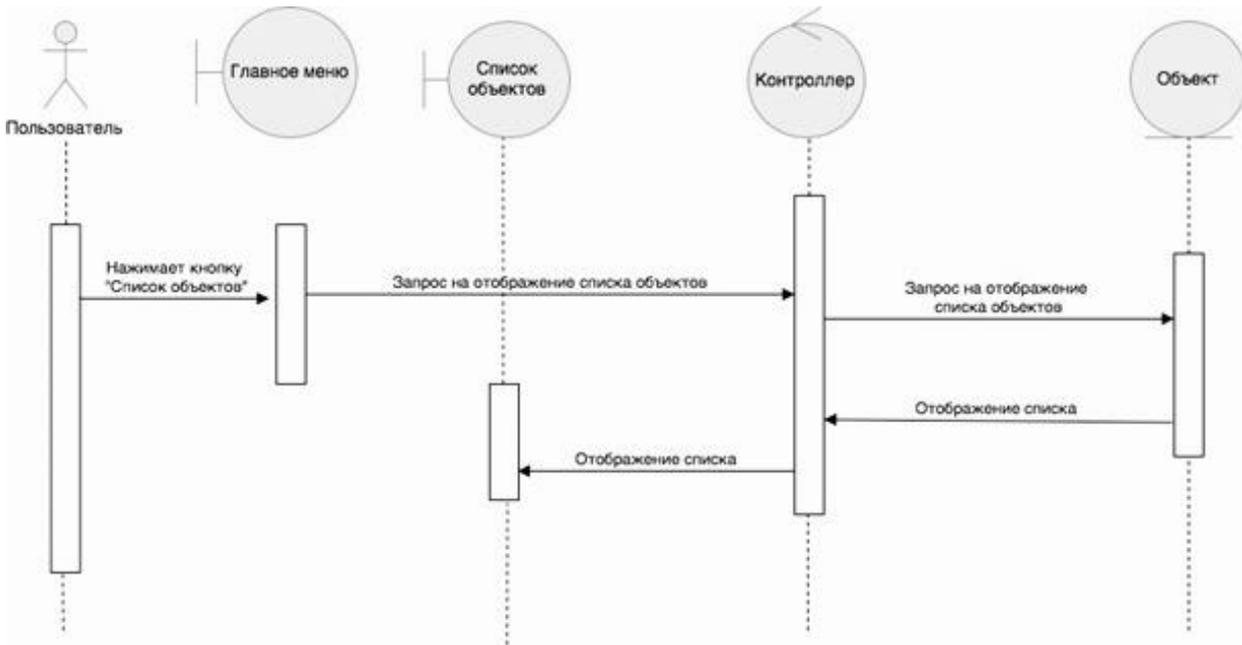


Рис. П1.13. Диаграмма последовательности для прецедента «Просмотр каталога объектов»

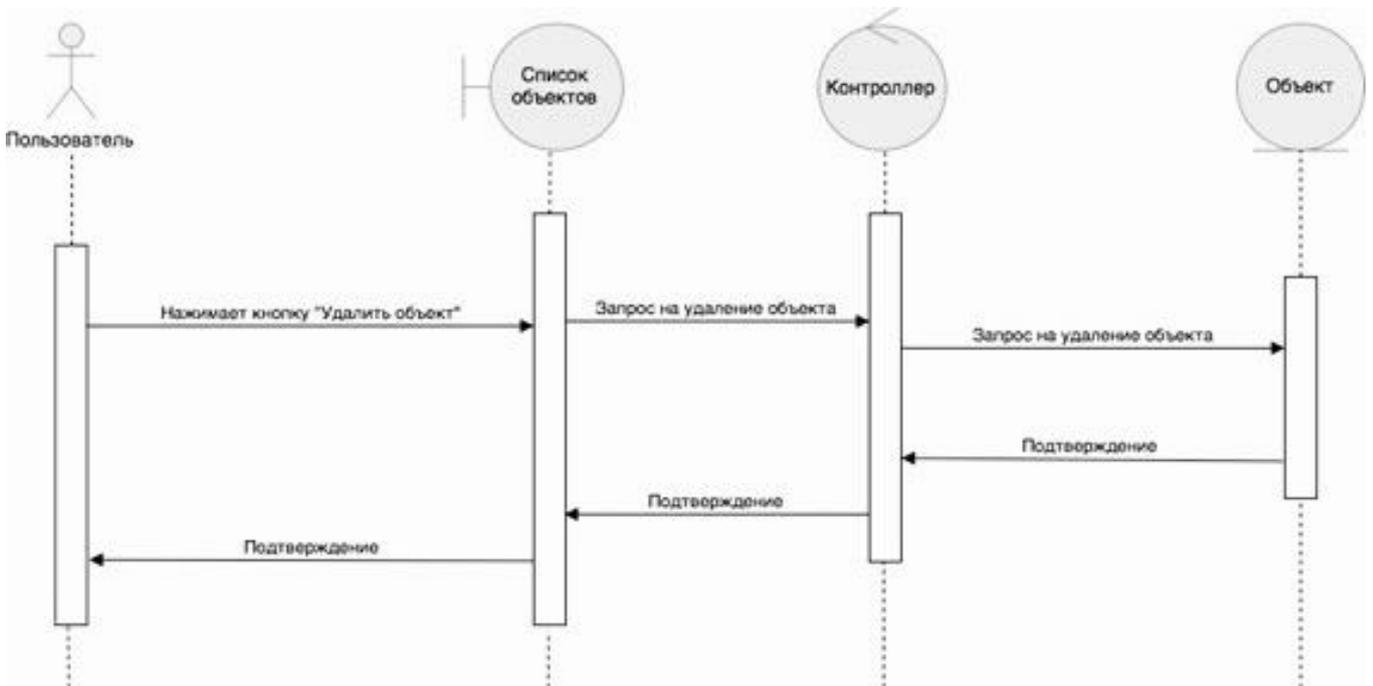


Рис. П1.14. Диаграмма последовательности для прецедента «Удаление объекта»

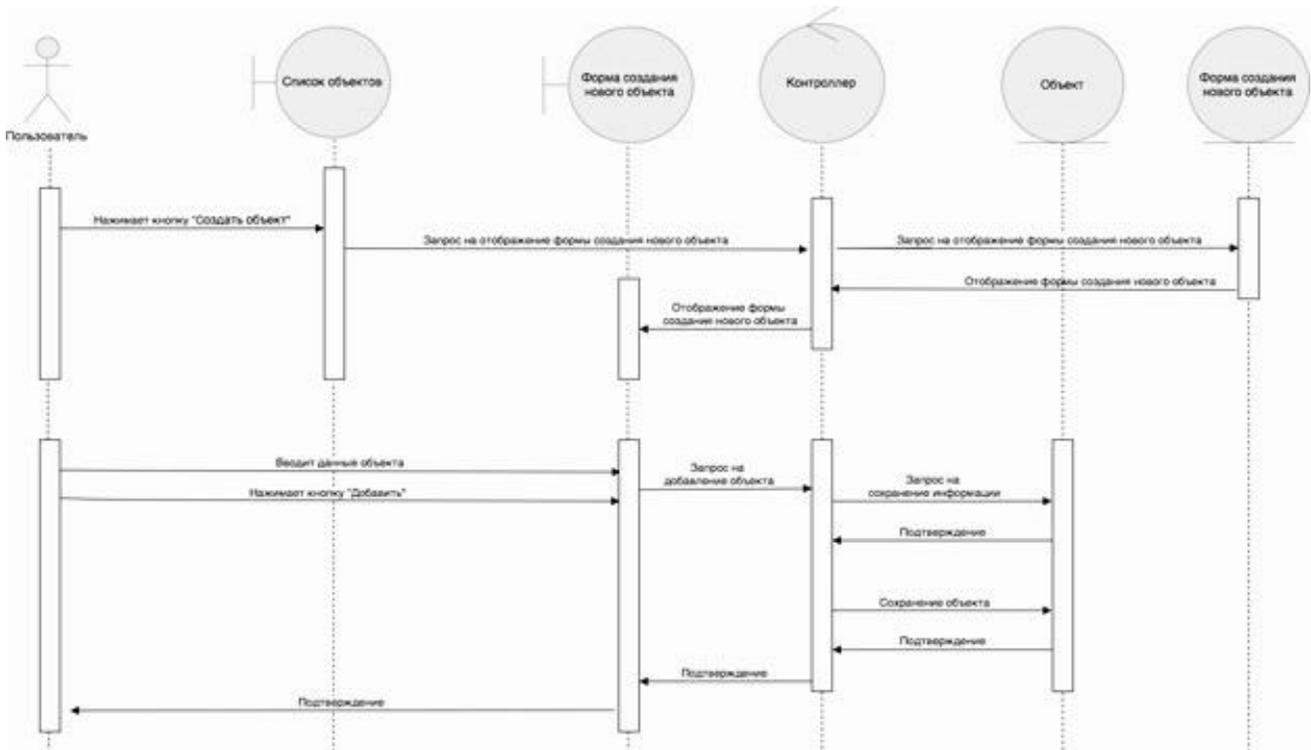


Рис. П1.15. Диаграмма последовательности для прецедента «Создание нового объекта»

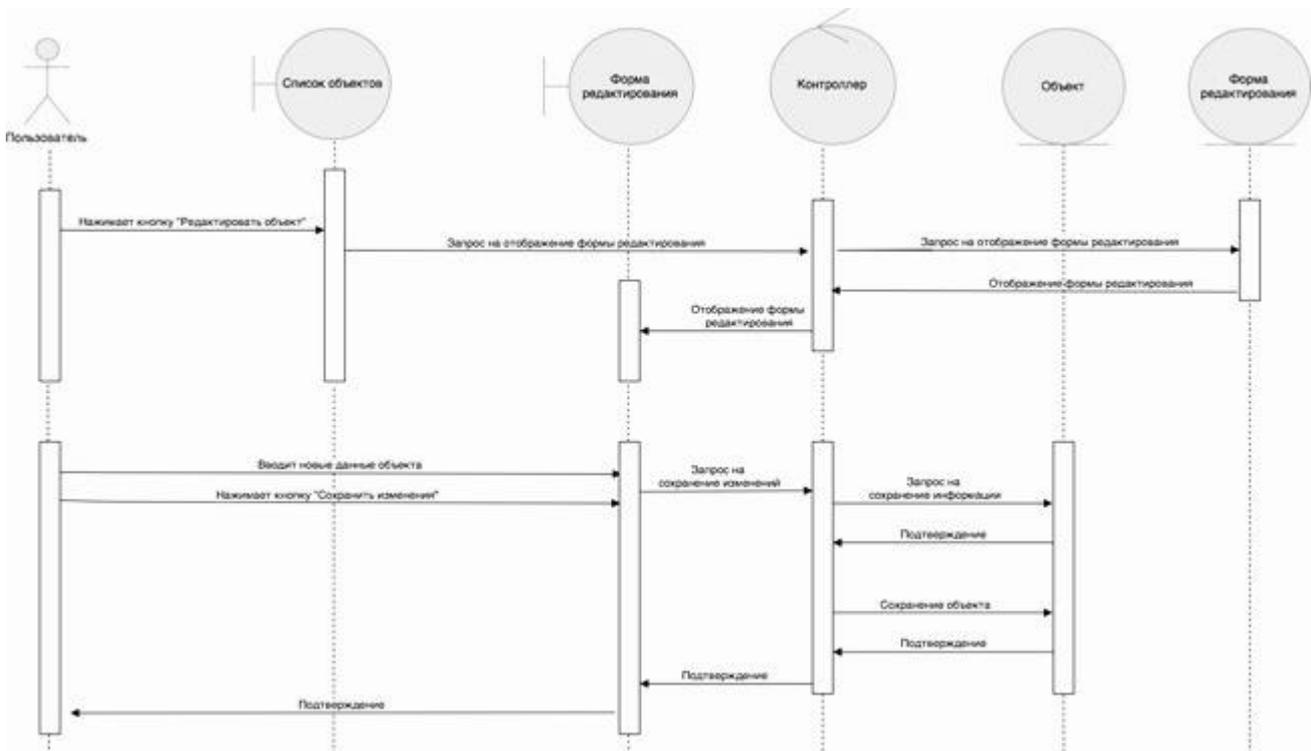


Рис. П1.16. Диаграмма последовательности для прецедента «Редактирование объекта»

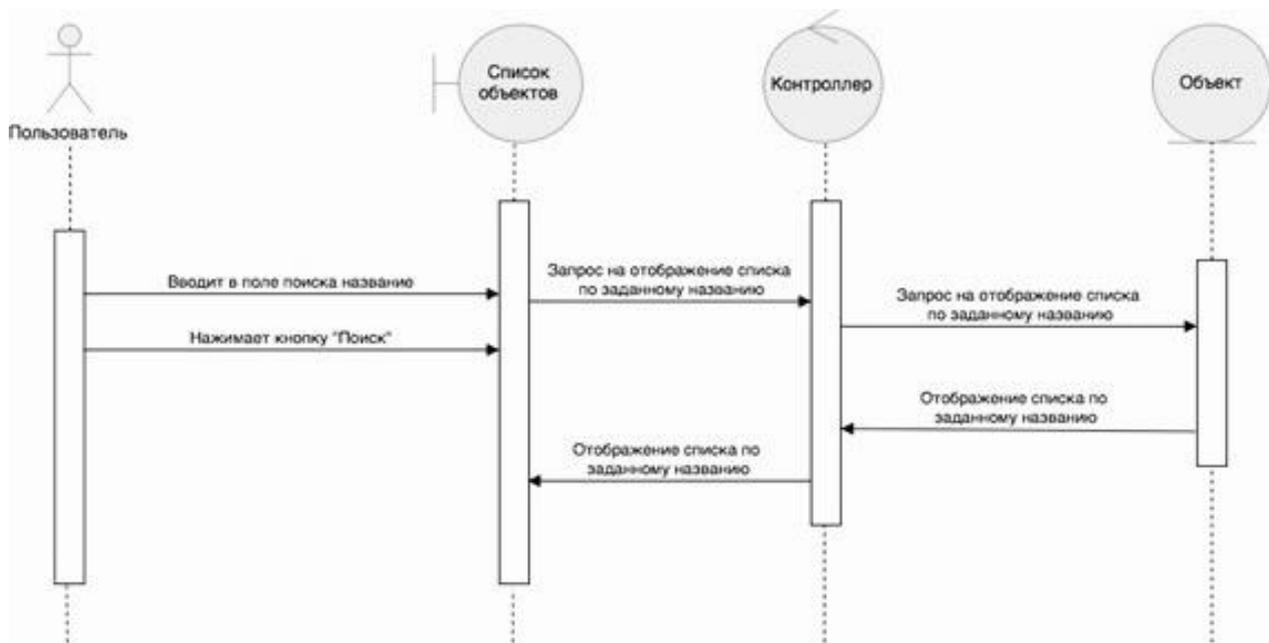


Рис. П1.17. Диаграмма последовательности для прецедента «Поиск объекта по названию»

Шаг Диаграммы классов этапа проектирования

Диаграмма классов этапа проектирования дополняет и расширяет концептуальную модель. На этой диаграмме для каждого отображаемого на диаграмме класса идентифицируются и специфицируются атрибуты (данные), а из диаграмм последовательности выявляются и специфицируются операции (методы) и их параметры. Таким образом, диаграмма классов этапа проектирования содержит в себе детальную информацию по всем входящим в диаграмму классам. Поэтому для лучшей визуализации эту диаграмму рекомендуется представлять в пакетном виде. Напомним, что, имея в виду язык UML, пакеты - это логические "ящики", по которым могут быть разложены, в частности, объединенные в группы классы диаграммы классов этапа проектирования.

Поскольку создаваемая Вами программная система должна иметь трехуровневую архитектуру, то рекомендуем Вам сначала выделить пакеты по

типу их принадлежности к одному из следующих трех уровней: уровню представления (граничные классы), уровню приложения (контроллеры), уровню хранения данных (сущностные классы) (рис. П.1.18).



Рис. П.1.18 Пакеты функциональности диаграммы классов

А затем в выделенных пакетах каждого уровня, разместите диаграммы классов уровня представления, прикладного уровня, уровня хранения данных (базы данных) соответственно (рис. П1.19, П1.20, П1.21).

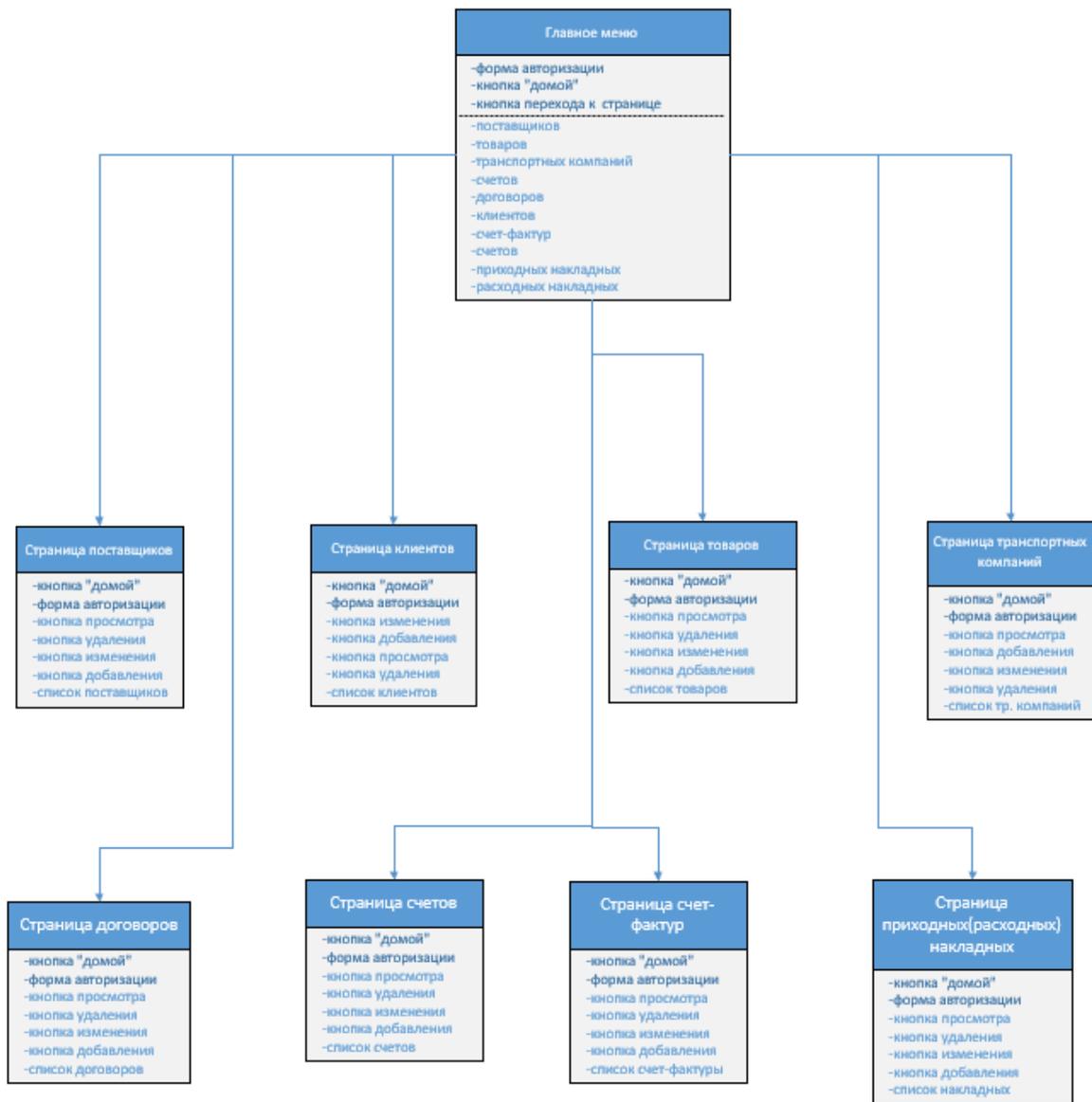


Рис. П1.19. Диаграмма классов уровня представления

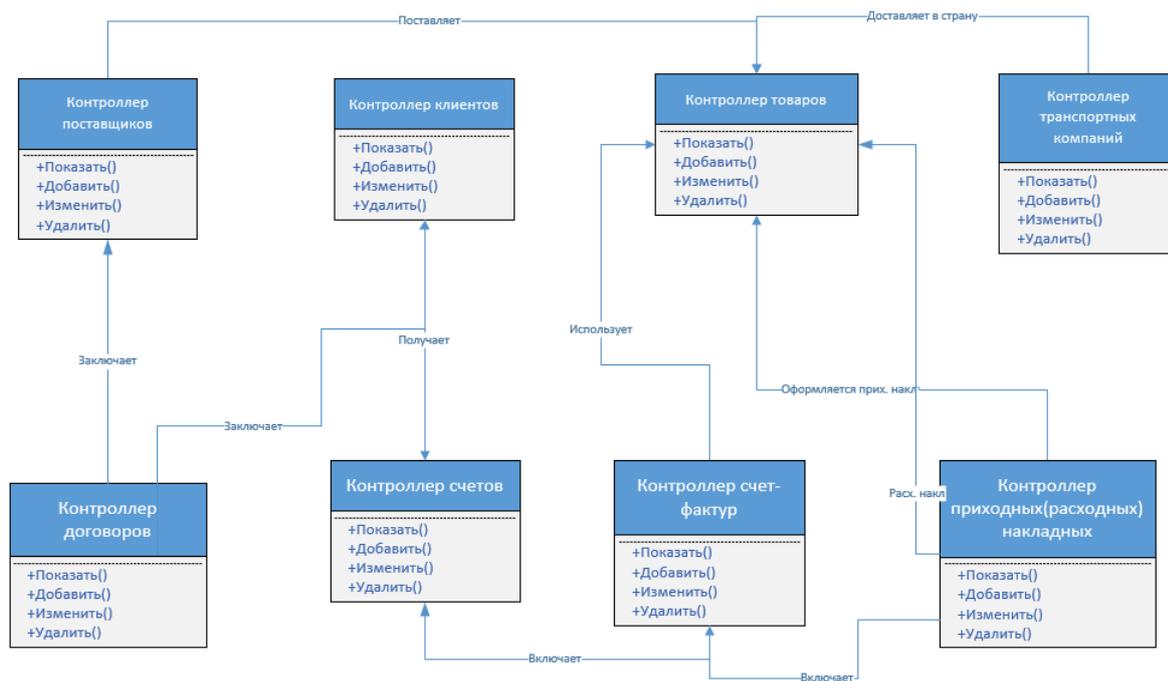


Рис. П1.20. Диаграмма классов прикладного уровня

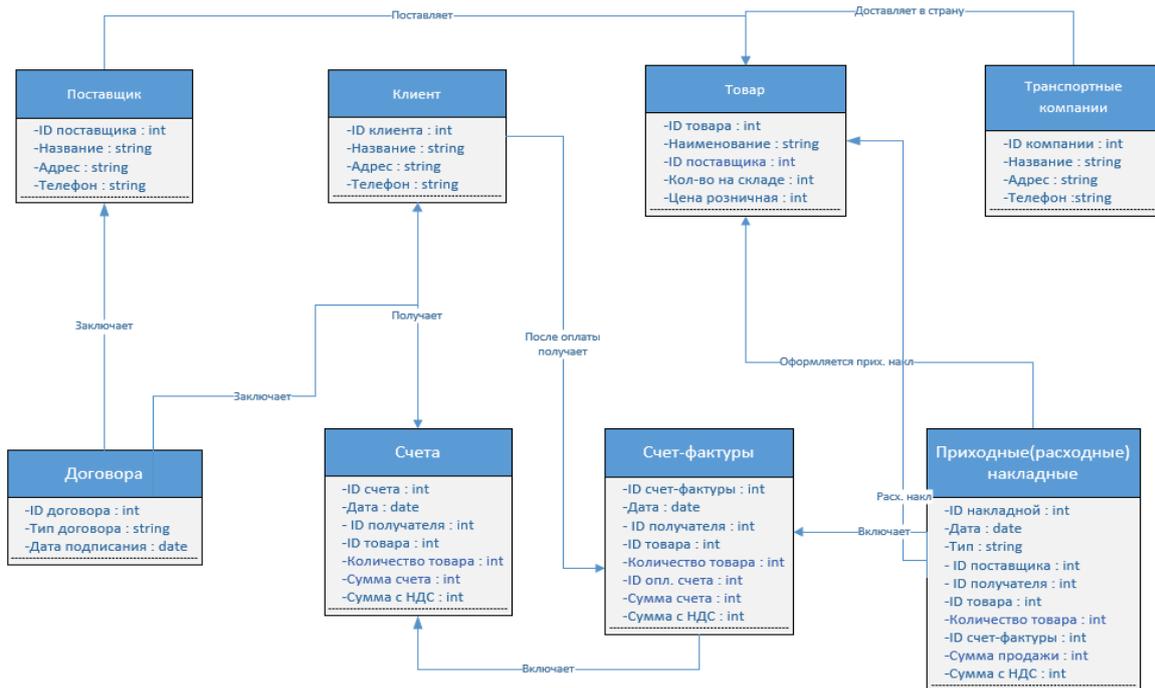


Рис. П1.21. Диаграмма классов уровня хранения данных (базы данных)

Шаг Схема (структура) базы данных

Для отображения атрибутов в столбцы также существует множество способов. Все они влияют не только на отображение столбцов, но и на отображение классов в таблицы. Возможно, у вас есть атрибуты, которых нет в базе данных, или столбцы, которые никогда не отображаются или вообще не обрабатываются в программе, но включены в базу данных для каких-то определенных целей. Иногда атрибут или даже класс преобразуется в несколько столбцов в базе данных.

При отображении атрибутов в столбцы иногда учитываются параметры производительности базы данных, но, как правило, определяющими в этом процессе оказываются требования. В такой сценарий можно даже включить представления базы данных, в которых атрибуты как часть класса отображались бы в таблице или в других столбцах. Это делается, в основном, с целью ускоренного доступа к часто запрашиваемым столбцам или из соображений защиты информации. Во многих базах данных нельзя защитить отдельный

столбец, можно защитить только всю таблицу целиком, поэтому представления играют важную роль в вопросах защиты данных. Очевидно, что не вся информация о служащем должна быть доступна. Например, незачем показывать данные о заработной плате всем пользователям базы данных. Для этого можно создать представление для определенной группы пользователей, в котором данная информация будет доступна. Для пользователей, не принадлежащих к указанной группе, данные по заработной плате служащего будут недоступны.

При отображении атрибутов в столбцы частично происходит и отображение классов в таблицы, но в любом случае их нужно рассматривать как один процесс. Нельзя преобразовать столбцы, не изменив таблиц, и невозможно переделать таблицы, не изменив при этом столбцы. Рассмотрим, например, класс, который содержит метод поиска адресных данных по почтовому индексу. Абсолютно неважно, какой алгоритм поиска используется в базе данных, лишь бы в ней была вся необходимая адресная информация. С другой стороны, в программе этот набор может оказаться в двух классах — один класс для поиска по адресу, а другой для поиска по почтовому индексу (рис. П1.22).

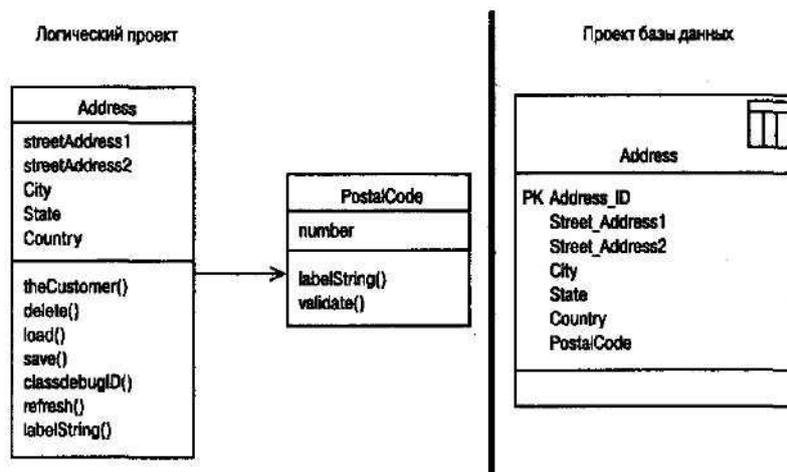


Рис. П1.22. Класс с поиском по почтовому индексу на диаграмме классов и его представление в модели базы данных.

В приложении часто вычисляется общая сумма продаж, однако ее нет в базе данных, где продажи хранятся по конкретным товарам или регионам, а

общая сумма различных комбинаций вычисляется по разным выборкам данных. Атрибуты, которые содержат значения для вычислений, выбранных в базе данных, называются производными атрибутами и не хранятся в базе данных.

При отображении атрибутов в столбцы важно отобразить не только названия столбцов, но также понимать, как типы данных в логической модели будут отображены в типы данных именно в вашем варианте базы данных. Данные в моделях, разработанных на этапе анализа, могут быть либо основного, либо логического типа, поэтому специальные знания по конкретному программному обеспечению (например, Oracle, IBM DB2, Microsoft SQL Server и т.п.) не требуются, достаточно только общих представлений о типах атрибутов. При описании атрибутов необходимо точно знать длину типов данных в отличие от анализа, где это не так важно.

Итак, при отображении атрибутов в конкретные столбцы базы данных необходимо расширить эти столбцы насколько позволяют допустимые значения. Например, для столбца Country (Страна) требуется создать диапазон допустимых значений в виде списка стран, которые можно ввести в этот столбец. Примерно то же требуется сделать при проектировании архитектуры, когда решается, где фактически должны быть реализованы бизнес-правила. В приложении их можно реализовать через методы или группы методов, в базе данных — через диапазоны допустимых значений или вообще реализовать бизнес-правила на бизнес-уровне, чтобы при их изменении не изменять ни программу, ни базу данных.

Приведем пример, где в базе будут храниться данные о производителях, товарах, клиентах, приходных и расходных накладных, а также о счетах. Поэтому соответствующие классы помечаем как таблицы. В результате получим следующую диаграмму (рис.П1.23).

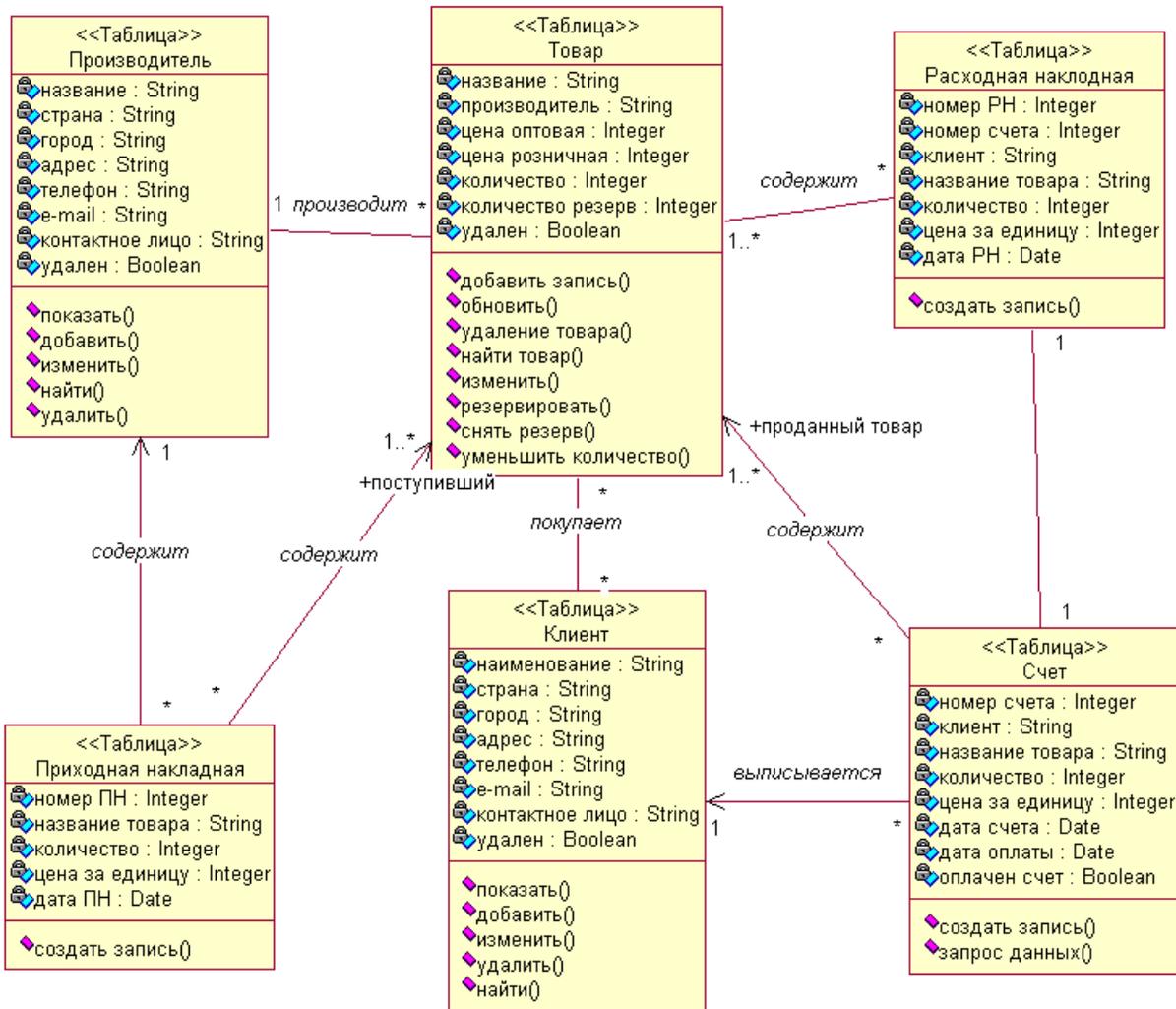


Рис.П1.23. Диаграмма классов базы данных.

Далее нужно произвести нормализацию диаграммы классов, по которой будет создаваться модель базы данных. Т.е. надо проанализировать все ассоциации между классами и заменить связи "много-ко-многим" и "один-к-одному", разбив или объединив таблицы соответственно.

Между классами "Счет" и "Расходная накладная" существует связь "один-к-одному", поэтому эти классы можно объединить в один "Расход". Между классами "Приходная накладная" и "Товар" - связь "много-ко-многим", поэтому чтобы ее изменить введен такой класс как "Строка ПН", в которой будет информация о каждом товаре, включенном в одну приходную накладную. Связь между классом "Клиент" и классом "Товар" является избыточной. Таким

образом, внеся все эти изменения, физическая модель базы данных будет выглядеть следующим образом (рис. П1.24).

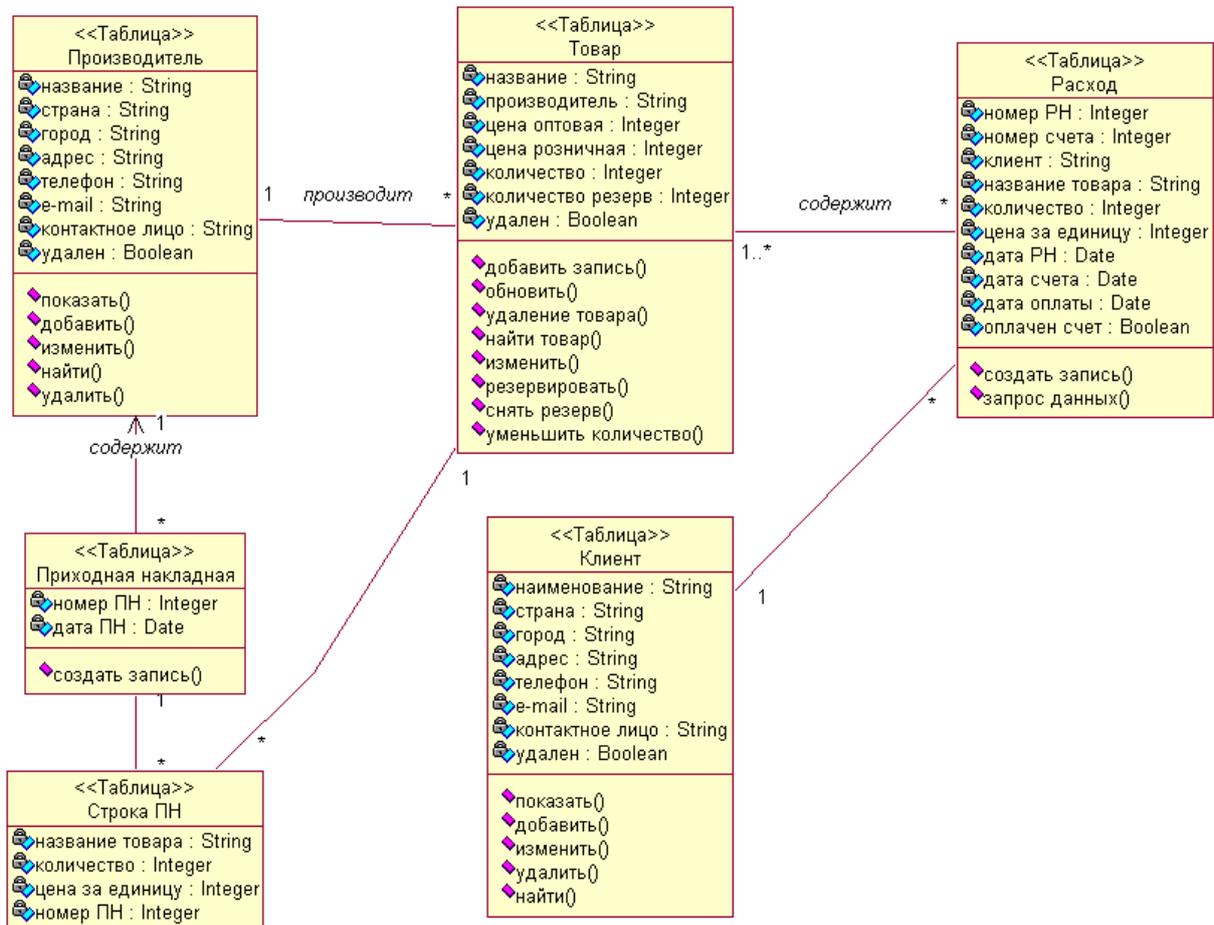


Рис. П1.24. Физическая модель базы данных (промежуточный этап).

Как видно из диаграммы у нас осталась еще одна связь "много-ко-многим" между "Товар" и "Расход". Избавимся от нее аналогично как в случае с "Приходная накладная" и "Товар". Тогда окончательная физическая модель базы данных будет выглядеть так (рис. П1.25):

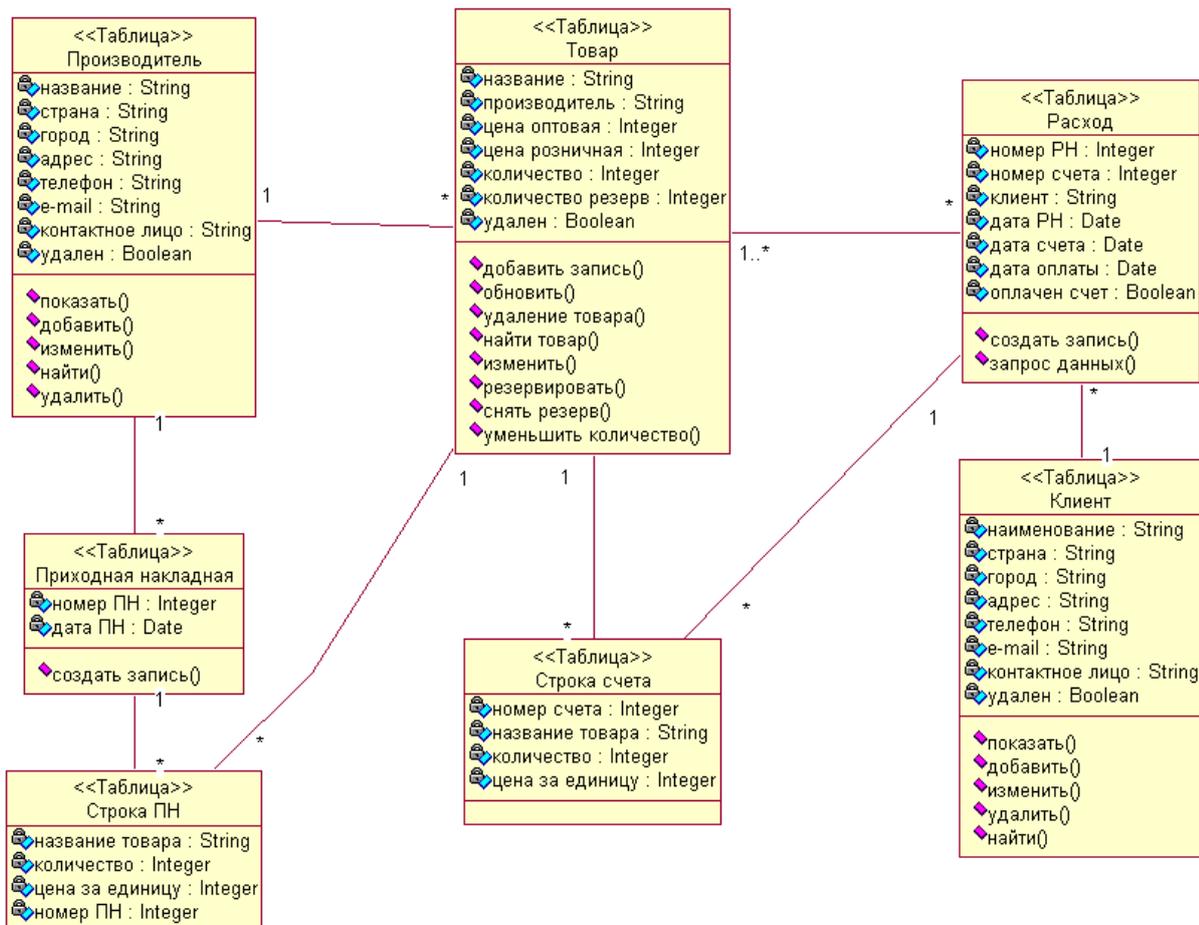


Рис. П1.25. Физическая модель базы данных.

На этом завершаются фрагменты шагов этапа Проектирование.