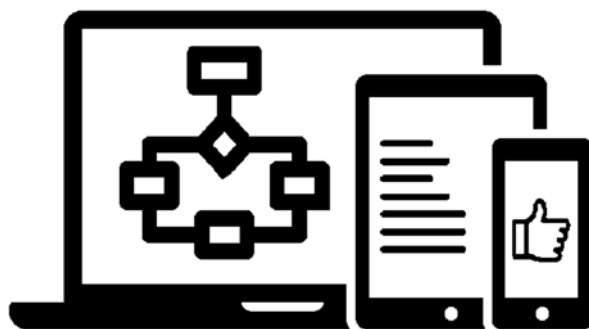


МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Программное обеспечение информационных технологий»

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

*Методические рекомендации
для студентов направлений подготовки
09.03.01 «Информатика и вычислительная техника»
и 09.03.04 «Программная инженерия»
очной формы обучения*



Могилев 2021

УДК 621.04
ББК 36.4
В87

Рекомендовано к изданию
учебно-методическим отделом
Белорусско-Российского университета

Одобрено кафедрой «Программное обеспечение информационных технологий» «30» сентября 2021 г., протокол № 2

Составители: канд. техн. наук, доц. В. В. Кутузов;
канд. техн. наук, доц. С. К. Крутолевич;
канд. техн. наук, доц. Э. И. Ясюкович;
канд. техн. наук К. В. Захарченков;
канд. техн. наук А. Е. Мисник

Рецензент канд. техн. наук, доц. С. В. Болотов

В методических рекомендациях изложены содержание, цель и задачи выпускной квалификационной работы, даны указания по выполнению отдельных её разделов. Предназначены для использования студентами направлений подготовки 09.03.01 «Информатика и вычислительная техника» и 09.03.04 «Программная инженерия» очной формы обучения.

Учебно-методическое издание

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

Ответственный за выпуск	В. В. Кутузов
Корректор	И. В. Голубцова
Компьютерная верстка	Н. П. Полевничая

Подписано в печать . Формат 60×84/16. Бумага офсетная. Гарнитура Таймс.
Печать трафаретная. Усл. печ. л. . Уч.-изд. л. . Тираж 26 экз. Заказ №

Издатель и полиграфическое исполнение:
Межгосударственное образовательное учреждение высшего образования
«Белорусско-Российский университет».
Свидетельство о государственной регистрации издателя,
изготовителя, распространителя печатных изданий
№1/156 от 07.03.2019.
Пр-т Мира, 43, 212022, г. Могилев.

© Белорусско-Российский
университет, 2021

Содержание

1 Цели и задачи ВКР	4
2 Организация процесса выполнения ВКР	5
2.1 Преддипломная практика	5
2.2 Дипломное проектирование	5
3 Содержание ВКР	10
4 Оформление ВКР	11
5 Методические рекомендации	12
5.1 Описание предметной области	12
5.2 Проектирование архитектуры программно-информационной системы	15
5.3 Реализация программно-информационной системы	19
5.4 Результаты	22
5.5 Заключение	22
Список литературы	23
Приложение А	25
Приложение Б	26
Приложение В	27
Приложение Г	28
Приложение Д	29
Приложение Е	30
Приложение Ж	31

1 Цели и задачи ВКР

Выпускная квалификационная работа (ВКР) является обязательной формой итоговой государственной аттестации, самостоятельно выполняемой обучающимися на завершающем этапе подготовки по специальности.

Подготовка и защита ВКР требуют от студента применения всего комплекса приобретенных им знаний и навыков за весь период обучения.

Назначение ВКР – определение уровня и выявление степени готовности будущего молодого специалиста к решению научных, теоретических и практических задач в рамках своей специальности и имеет своей целью:

- закрепление и углубление теоретических и практических знаний по специальности и применение их для решения конкретных задач;
- подтверждение владения основными компетенциями в соответствии с требованиями Государственного образовательного стандарта;
- выяснение подготовленности студента для работы в условиях современного производства по специальности «инженер по информационным технологиям».

Защита ВКР является основанием для присвоения студенту соответствующей квалификации и выдачи диплома государственного образца.

2 Организация процесса выполнения ВКР

2.1 Преддипломная практика

Целью преддипломной практики является изучение и документирование бизнес-процессов разрабатываемой программной системы.

Задачи практики:

- изучение организационной и функциональной структуры подразделения предприятия;
- изучение функциональных обязанностей работников;
- документирование существующих бизнес-процессов формирования документов.

Преддипломная практика проводится по месту распределения студентов или на предприятиях, в учреждениях и организациях, на материалах которых и выполняется ВКР.

Содержание преддипломной практики определяется темой ВКР и характером индивидуального задания. Сбор материалов для специальной части ВКР проводится по следующему плану:

- анализ документооборота и сбор форм документов;
- выявление тех, кто принимает решения, несет ответственность, принимает участие в формировании документооборота;
- составление диаграммы классов (рисунок В.1) документов;
- выявление алгоритмов интеллектуальной обработки и анализа данных.

В качестве отчета по преддипломной практике представляется «Техническое задание на разработку программно-информационной системы».

2.2 Дипломное проектирование

Выполнение ВКР осуществляется студентом на основе выданного ему задания. Кафедра ежегодно разрабатывает тематику дипломного проектирования, которая после обсуждения ее на кафедре и утверждения заведующим предлагается для выбора студентом. При разработке тематики принимается во внимание: актуальность темы; участие в её разработке студентов на предыдущих этапах учебного процесса (производственных практиках, курсовом проектировании, СНИР и т. п.), предложения студентов.

Очень желательно наличие письма из организации, по материалам которой выполняется ВКР.

В ВКР указываются:

- точное наименование организации, по материалам которой выполняется дипломный проект;
- наименование структурного подразделения, использующего разработанный программный продукт;
- наименование автоматизируемых бизнес-процессов.

Например. Разработка программно-информационной системы «Формирование портфеля заказов ЧУП КУВО».

Для закрепления выбранных тем ВКР студенты подают заявление на имя заведующего кафедрой. В заявлении указывается адрес электронной почты студента и телефон.

По представлению кафедры ректор университета издает приказ об утверждении тем и назначении руководителей проектов и консультантов по разделам проектов.

Студент проводит сверку документов в учебном отделе университета. Приказ издается в течение пяти дней после завершения преддипломной практики. Изменение темы дипломного проекта далее возможно в исключительных случаях, распоряжением ректора по письменному заявлению студента с объяснением причины.

Руководитель проекта разрабатывает задание по ВКР и представляет его на утверждение заведующему кафедрой. Один экземпляр утвержденного задания выдается студенту, а второй хранится на кафедре.

Руководитель ВКР и консультанты по отдельным разделам выдают студенту-дипломнику необходимые исходные данные, рекомендуют литературу, справочные материалы и другие источники по теме проекта, осуществляют контроль за ходом выполнения графика работ по проекту, проводят систематические консультации по утвержденному расписанию, проверяют качество выполненных работ и их соответствие заданию и методическим рекомендациям.

Студент-дипломник несет личную ответственность за качество и своевременность выполнения проекта и за соответствие его утвержденному заданию.

В мае проводится смотр ВКР, где студенты показывают свои разработки комиссии в составе заведующего кафедрой и преподавателя-руководителя ДП.

Законченная и оформленная ВКР, подписанная студентом и консультантами, за 10 дней до защиты представляется руководителю на рассмотрение. Срок рассмотрения и подготовки отзыва руководителем не должен превышать трех дней. При рецензировании руководитель должен отметить каждую ошибку и неточность с указанием, в чем заключается их сущность. Недопустима расстановка вопросительных и других знаков без соответствующих разъяснений. В отзыве руководителя должен быть представлен подробный анализ недостатков и ошибок, конкретно и четко сформулированы все требования, которые должен выполнить студент. Выставляется количество набранных баллов по модульно-рейтинговой системе (МРС) оценки учебных достижений студента.

Соответствие между количеством погрешностей и ошибок, обнаруженных при рецензировании, и баллами МРС представлено в таблице 1.

При повторном рассмотрении руководитель ВКР проверяет исправление предыдущих замечаний. Новые замечания не допускаются. Если проект удовлетворяет требованиям, предъявляемым к нему, он допускается к защите, о чем руководитель делает надпись на чертежах и записке.

ВКР направляется на доработку, если количество ошибок и погрешностей позволяет отнести ее к низкому уровню соответствия (рейтинг меньше 36).

Таблица 1 – Допустимые погрешности и ошибки, выявленные при рецензировании

Шкала соответствия	Уровень соответствия	Балл МРС	Погрешности несущественные/ существенные /ошибки
Соответствие	Высокий	58–60	1/0/0
		55–57	2/1/0
	Средний	51–54	3/1/1
		47–50	4/2/1
		42–46	5/2/3
		39–41	6/3/2
Минимальный	36–38	7/4/3	
Несоответствие	Низкий	26–35	8/5/4
		15–25	9/6/5
		5–14	10/10/10

Несущественными погрешностями при определении учебных достижений считаются:

- наличие грамматических ошибок;
- написание пояснительной записки с несоблюдением требований к оформлению текстовых документов;
- отсутствие ссылок на использованные источники.

К существенным погрешностям относятся:

- ошибки при проектировании диаграмм UML;
- неточности в определении параметров функций и типов возвращаемых значений;
- неточности изложений алгоритмов вычисляемых процедур;
- несоответствие имён классов на диаграммах UML;
- отсутствие всех необходимых объектов на диаграммах архитектуры АСОИ и диаграммах поведения АСОИ.

К ошибкам относятся:

- отсутствует какой-либо раздел пояснительной записки или лист графической части;
- база данных не соответствует третьей нормальной форме;
- не определены источники данных для всех полей документов (таблицы, запросы, вычислительные процедуры);
- не приведены алгоритмы всех вычислительных процедур;
- на диаграммах последовательности (рисунок Г.1) формирования документов не указаны альтернативные пути и ошибки;
- на диаграммах последовательности представлены не все элементы архитектуры программной системы (формы, запросы, вычислительные процедуры);
- на диаграммах классов представлены не все элементы архитектуры программной системы (формы, запросы, вычислительные процедуры);

- обоснована трудоемкость разработки не всех элементов архитектуры программной системы;
- не обосновано формирование календарного плана разработки;
- отчёт программиста не соответствует календарному плану;
- в отчете программиста не приведен код формирования форм, запросов и вычислительных процедур;
- разработанное программное обеспечение не соответствует UML-модели;
- разработанное программное обеспечение не выполняет необходимых функций;
- инструкция пользователя не содержит описания всех вариантов использования (рисунок А.1) программной системы;
- размер шрифта на диаграммах UML в графической части менее 5 мм.

После рецензирования и одобрения ВКР руководитель подписывает ее, направляет на нормоконтроль и составляет письменный отзыв о проекте.

В отзыве руководителя ВКР должны быть отмечены:

- актуальность выбранной темы;
- умение студента использовать специальную литературу;
- степень решения поставленных в работе задач, полнота использования фактического материала и источников;
- способности студента к инженерной или исследовательской работе;
- степень самостоятельности и инициативности студента при выполнении работы;
- обоснованность выводов и ценность практических результатов, включая возможность использования полученных результатов на практике;
- наличие публикаций и выступлений на конференциях;
- наиболее удачно раскрытые аспекты;
- замечания (если таковые имеются);
- возможность присвоения выпускнику соответствующей квалификации.

Заведующий кафедрой на основании этих материалов принимает решение о допуске студента к защите ВКР в ГЭК, что отображается соответствующим образом на титульном листе пояснительной записки к ВКР. В случае, если заведующий кафедрой не считает возможным допустить студента к защите ВКР, этот вопрос рассматривается на заседании кафедры с участием руководителя. Протокол заседания кафедры через деканат факультета представляется на утверждение ректору университета.

ВКР, допущенная к защите, направляется на рецензию к рецензенту – внешнему специалисту, назначенному приказом ректора. Рецензент после детального ознакомления со всеми материалами составляет обоснованную критическую рецензию.

В рецензии должны быть отмечены:

- актуальность темы ВКР;
- степень соответствия ВКР заданию;
- логичность построения пояснительной записки;
- наличие по теме ВКР критического обзора литературы, его полнота и последовательность анализа;

- полнота описания методики расчета или проведенных исследований, изложения собственных расчетных, теоретических и экспериментальных результатов, оценка достоверности полученных выражений и данных;
- наличие аргументированных выводов по результатам ВКР;
- практическая значимость ВКР, возможность использования полученных результатов;
- недостатки и слабые стороны ВКР;
- замечания по оформлению пояснительной записки к ВКР и стилю изложения материала;
- оценка ВКР: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Рецензент имеет право затребовать у студента-автора ВКР дополнительные материалы, относящиеся к проделанной работе. Студент должен быть ознакомлен с рецензией до защиты работы в ГЭК.

Рецензия подписывается рецензентом с указанием фамилии, имени, отчества, ученого звания, степени, места работы, должности, даты. Подпись рецензента заверяется по месту его основной работы.

Защита ВКР производится на открытом заседании Государственной комиссии, утвержденной министром образования РБ (РФ). Защита состоит в коротком докладе (8–10 мин) студента по выполненному проекту и в ответах на вопросы членов комиссии и замечаний внешнего рецензента.

После доклада выпускник отвечает на вопросы членов ГЭК. Вопросы могут касаться как темы выполненной работы, так и носить общий характер в пределах дисциплин специальности и специализации, изучаемой на протяжении обучения в вузе. После членов ГЭК с разрешения председателя вопросы могут задавать все присутствующие на защите. Затем выступает рецензент или зачитывается его рецензия. При имеющихся замечаниях рецензента выпускник должен ответить на них.

После этого выступает со своим отзывом руководитель или при его отсутствии зачитывается отзыв.

После окончания защит работ ГЭК продолжает свою работу на закрытой части заседания, на которой с согласия председателя комиссии могут присутствовать руководители и рецензенты ВКР.

Количество баллов по МРС оценки учебных достижений студента выставляется в соответствии с таблицей 2.

Таблица 2 – Баллы МРС при защите дипломной работы

Этап защиты	Минимум	Максимум
Доклад	5	15
Ответы на вопросы	10	25
Защита	15	40

Итоговая оценка выставляется в соответствии с положением о МРС путем суммирования баллов за рецензию и защиту ВКР.

ВКР после защиты сдаются секретарем Государственной комиссии по защите в архив университета, где они должны храниться в течение пяти лет.

3 Содержание ВКР

ВКР состоит из пояснительной записки и графической части. Структура пояснительной записки, которую рецензирует руководитель ВКР, приведена в таблице 3. Основными требованиями к пояснительной записке являются четкость и логическая последовательность изложения материала, убедительность аргументации, краткость и ясность формулировок. В тексте записки не должно быть общих фраз, очевидных выводов и т. п. Объем пояснительной записки – не более 100 страниц текста.

Таблица 3 – Структура пояснительной записки

Наименование	Рекомендуемый объем, с.
Титульный лист	1
Задание на проектирование	1
Содержание	1
Введение	1
1 Описание предметной области	5–10
1.1 Обзор литературы	
1.2 Постановка задачи	
1.3 Обоснование актуальности задачи	
1.4 Обоснование используемых технологий, принципов, методик решения задачи	
2 Проектирование архитектуры программно-информационной системы	20–30
2.1 Установление требований	
2.2 Составление спецификации требований	
2.3 Проектирование структуры слоя доступа к данным	
2.4 Проектирование подсистемы интеллектуальной обработки данных (не обязательный раздел)	
3 Реализация программно-информационной системы	5–10
3.1 Описание архитектуры программно-информационной системы	
3.2 Пример использования программно-информационной системы	
4 Результаты	1–2
Заключение	1
Литература	1

Графическая часть проекта включает диаграммы и иллюстрации (плакаты). Диаграммы раскрывают структуру программного обеспечения и алгоритм обработки информации. На плакатах отображаются вопросы постановки задач,

применения математических моделей, а также достигнутые результаты. Перечень графических материалов проекта указан в таблице 4. Рекомендуется выполнять графическую часть на листах формата А4–А3. Объем графического материала представлен в таблице 4.

Для доклада готовятся: файл презентации в формате *.ppt. и видеоролик в формате *.avi, который отражает процесс создания документов и отчетов. Длительность ролика – до 1 мин.

Таблица 4 – Структура графического материала

Наименование	Количество диаграмм/экземпляров
Диаграмма вариантов использования	1
Макеты документов	По числу документов
Диаграмма классов. Структура базы данных	1
Диаграмма классов. Документы	По числу документов
Диаграмма последовательности	По числу вариантов использования
Диаграмма классов. Диалоговые формы	По числу форм
Диаграмма классов. Запросы	По числу запросов
Диаграмма классов. Вычислительные процедуры	По числу процедур
Диаграмма состояний	1
Диаграмма компонентов и размещения	1
Копии экрана. Формы и документы	По числу форм и документов

4 Оформление ВКР

Оформление ВКР должно соответствовать требованиям ГОСТ 2.105–2019. Текстовая часть пояснительной записки выполняется шрифтом Times New Roman с высотой букв 14 пунктов через полуторный интервал. Размер листа графической части выбирается из следующих условий: изображение должно занимать не менее 70 % площади листа, высота букв должна быть не меньше 5 мм.

После утверждения тем ВКР ответственный сотрудник по кафедре за нормоконтроль проводит собрание со студентами и информирует о правилах оформления пояснительной записки и графической части ВКР.

5 Методические рекомендации

Содержание разделов пояснительной записки и графической части ВКР должно соответствовать приведенным методическим рекомендациям.

5.1 Описание предметной области

5.1.1 Введение.

Во введении дается краткое обоснование выбора темы ВКР (не более двух страниц), тезисно обозначается актуальность работы, инженерной или исследовательской задачи, которая решается в работе, выделяются объект и предмет автоматизации, кратко описываются цель ВКР и задачи, которые необходимо решить для достижения поставленной цели, методы и средства решения поставленных задач, назначение работы и преимущества предлагаемого решения по сравнению с аналогами.

Целью большинства ВКР является автоматизация, информатизация или цифровизация процессов в выбранной предметной области. В качестве объектов автоматизации могут выступать структурные подразделения крупных и средних предприятий и организаций (например, склад или производственный цех промышленного предприятия; кафедра вуза); малые предприятия торговли, общественного питания, сферы услуг и т. п.; объекты социального назначения (например, музеи, театры, парки развлечений); социально значимые системы и их подсистемы. Предметом автоматизации являются процессы выбранного объекта, которые обеспечивают повышение эффективности системы в целом или отдельных бизнес-процессов.

Целью ВКР, содержащих исследовательский компонент, является, как правило, обработка и анализ данных, полученных в результате сбора информации из открытых источников в сети Интернет, полученных в результате эксперимента или из других источников. Предметом автоматизации в этом случае являются методы обработки и анализа данных.

В рамках выполнения ВКР по направлению подготовки 09.03.01 «Информатика и вычислительная техника» рекомендуется больше внимания уделять решению задач, связанных с обработкой информации с электронно-цифровых устройств.

Во введении должно быть кратко и сжато изложены основные идеи работы. Рекомендуемый объем введения – не более трех страниц.

5.1.2 Обзор литературы.

В разделе приводятся и описываются сведения об объекте автоматизации из основных источников литературы.

Необходимо рассмотреть источники информации об основных понятиях и аспектах функционирования объекта автоматизации. Обзор литературы должен включать не менее **десяти** источников, описывающих аналоги создаваемой программно-информационной системы, объект и предмет автоматизации.

При этом необходимо:

- выполнить критический анализ существующих аналогов программно-информационной системы, которую планируется создать в рамках ВКР;
- указать наиболее актуальные задачи, которые не могут быть решены средствами аналогичных программно-информационных систем, но будут реализованы в рамках создаваемой системы;
- описать проблематику и важнейшие результаты, которые планируется получить при выполнении ВКР;
- сформировать перечень задач, подлежащих автоматизации.

5.1.3 Постановка задачи.

Постановка задачи – это пункт, в котором описываются процессы и функции, подлежащие автоматизации в процессе выполнения ВКР. Автоматизации подлежат функции обработки информации (формирование документов и отчетов, поиска данных по заданным критериям). Перечень процессов и функций, подлежащих автоматизации, должен соответствовать цели создания программно-информационной системы.

Для каждого процесса и функции, подлежащей автоматизации, должно быть описано, как соответствующий процесс или функция выполнялись до внедрения разрабатываемой системы и что планируется улучшить в выполнении процесса в результате реализации создаваемой системы. В ВКР должно быть реализовано не менее пятнадцати функций обработки информации, из которых минимум две должны автоматизировать формирование документов и минимум пять – формирование отчетов.

В данном пункте должны быть представлены исходные данные для решения задачи и выходные данные, выдаваемые конечным пользователям как результат работы программно-информационной системы.

Входные данные представляют собой информацию, которую вводит пользователь в программно-информационную систему вручную; информацию, которая загружается в систему из внешних программных модулей, баз данных, интернет-ресурсов, электронно-цифровых устройств.

Выходная информация может быть представлена в виде документов и отчетов, автоматизация формирования которых осуществляется в процессе работы программно-информационной системы; web-форм, использующихся для вывода данных пользователям; информации в базе данных; выходного сигнала электронно-цифровых устройств.

Для исследовательских задач в данном пункте должен быть кратко описан математический аппарат, использующийся для обработки данных.

5.1.4 Обоснование актуальности задачи.

Актуальность задачи представляет собой степень ее важности и востребованности для решения конкретной проблемы, вопроса или задачи.

При обосновании актуальности темы ВКР необходимо объяснить причину назревания темы исследования именно в данный момент, указать, что являлось препятствием правильному раскрытию её ранее, выявить, насколько обращение к данной теме обусловлено развитием науки, накоплением новой информации и

методов исследования по данной проблеме, недостатками в изученности проблемы в уже проведённых исследованиях, необходимостью изучения проблемы исследования в новой экономической ситуации, применением новых методов и методик исследования и т. д.

Как правило, актуальность исследования представляется как противоречивая ситуация, требующая своего решения. Решение данной проблемы должно непосредственно быть связано с практической необходимостью. Это значит, что обращаясь к той или иной проблеме, необходимо чётко представлять, на какие вопросы практики могут дать ответ результаты работы. При этом актуальность должна соответствовать современному уровню научных знаний, иметь практическое значение для социально-экономической жизни страны. Обоснование актуальности не должно быть ни слишком кратким, ни излишне объемным.

5.1.5 Обоснование используемых технологий, принципов, методик решения задачи.

Выбор методов решения задач осуществляется преимущественно для ВКР, содержащих исследовательский компонент. При выборе методов решения задач производится анализ существующих подходов к решению поставленных задач с учетом специфики объекта автоматизации. В процессе выбора методов решения задач следует провести обзор литературы, по результатам которого должен быть дополнен список литературных источников. По тексту ВКР должны быть ссылки на литературу, содержащую методы решения задачи. Выбор метода решения каждой задачи выполняется на основе обоснования преимуществ выбранного метода по сравнению с аналогами.

Обоснование средств решения задач ВКР заключается в формировании требований к аппаратному, системному (общему) и специальному прикладному программному обеспечению и в выборе на основе этих требований соответствующих компонентов программного обеспечения (ПО).

Требования к аппаратному обеспечению должны описывать, на каких типах устройств должна работать создаваемая программно-информационная система (настольных персональных компьютерах и ноутбуках, планшетах, смартфонах, умных часах и т. п.).

При обосновании выбора системного ПО целесообразно:

- дать классификацию операционной системы, указать факторы, влияющие на выбор конкретного класса и его версии, и обосновать выбор операционной системы;

- дать классификацию, привести факторы и обосновать выбор используемой СУБД.

При обосновании выбора средств решения задач ВКР необходимо учитывать возможности сред разработки программного обеспечения, возможность работы создаваемых приложений на выбранных аппаратных платформах, операционных системах, наличие стандартных средств интеграции с выбранной СУБД. Выбор средств решения задач ВКР предполагает выбор среды разработки, фреймворков и библиотек, использующихся в процессе решения

задач ВКР. Для обоснования выбора средств решения задач ВКР сравниваются возможности существующих средств разработки ПО и выбирается среда разработки, обеспечивающая решение всех задач ВКР с минимальными трудозатратами в минимальные сроки.

При обзоре средств решения задач ВКР обязательно делаются ссылки по тексту на используемые источники литературы. Результаты анализа возможностей средств разработки ПО представляются в виде таблицы, отражающей наличие / отсутствие в каждой среде возможностей для решения.

При обосновании выбора технологий решения задач ВКР необходимо сформулировать требования, которые должны выполняться при решении задач ВКР (надежности, эффективности, понятности пользователю, защиты информации, модифицируемости, мобильности, масштабируемости, минимизации затрат на сопровождение и поддержку и т. д.).

Формулировку требований к технологиям проектирования и разработки ПО необходимо производить с учетом выдвинутых требований к аппаратному обеспечению, системному ПО и средствам разработки. При обосновании проектных решений по выбору технологий решения задач ВКР необходимо:

- дать классификацию и обосновать выбор методов (например, структурное, модульное проектирование, метод «сверху вниз» или объектно-ориентированное проектирование и т. д.) и средств проектирования специального (функционального) ПО (например, использование библиотеки прикладных программ, генератора программ или какого-либо языка программирования);

- определить возможности выбранных технологий, при использовании которых выполняются все требования к решению задач ВКР (например, возможность организации удобного интерфейса, оптимизации запросов к данным и т. п.).

Выбор технологий проектирования и разработки, по возможности, необходимо аргументировать, сравнивая их с аналогичными технологиями, существующими на рынке, четко указав преимущества выбранных технологий проектирования и разработки по сравнению с аналогами. При оценке технологий разработки для решения задач ВКР обязательно делаются **ссылки по тексту на используемые источники литературы**.

Для решения задач ВКР рекомендуется выбирать преимущественно самые современные технологии. Выбор устаревших технологий оправдан преимущественно для доработки существующих автоматизированных систем и должен быть подробно аргументирован и обоснован.

5.2 Проектирование архитектуры программно-информационной системы

5.2.1 Установление требований.

В разделе описываются требования, которые предъявляются к программной системе.

Ядро документа описания требований состоит из формулировок (изложения) требований. Требования могут быть сгруппированы в виде формулировок

сервисов (зачастую называемых функциональными требованиями) и формулировок ограничений. Формулировки сути сервисов могут быть затем разделены на требования к функциям (function requirements) и требования к данным (data requirements).

Документ описания требований должен обращаться к проектным вопросам. Обычно проектные вопросы рассматриваются в начале, а затем в конце документа.

Во вводной части документа рассматривается бизнес-контекст проекта, включая цель проекта, участников проекта и основные ограничения. Ближе к заключительной части документа поднимаются другие проектные вопросы, включая план-график выполнения проектных работ, бюджет, риски, документацию и т. д.

Шаблон документа описания требований определяет структуру документа с подробными указаниями о содержании каждого из разделов документа. Указания могут включать содержание вопросов, мотивацию, примеры и дополнительные соображения.

Документ описания требований должен создать прецедент для системы. В частности, необходимо упомянуть обо всех усилиях, приложенных для обоснования актуальности системы на этапе ее планирования. Документ описания требований должен прояснить вопрос о том, каким образом предлагаемая система может способствовать достижению деловых целей и решению задач организацией.

Описание требований может быть выполнено по следующему шаблону.

- 1 Предварительные замечания к проекту.
 - 1.1 Цели и рамки проекта.
 - 1.2 Деловой контекст.
 - 1.3 Участники проекта.
 - 1.4 Идеи в отношении решений.
 - 1.5 Обзор документа.
- 2 Системные сервисы.
 - 2.1 Рамки системы.
 - 2.2 Функциональные требования.
 - 2.3 Требования к данным.
- 3 Системные ограничения.
 - 3.1 Требования к интерфейсу.
 - 3.2 Требования к производительности.
 - 3.3 Требования к безопасности.
 - 3.4 Эксплуатационные требования.
 - 3.5 Политические и юридические требования.
 - 3.6 Другие ограничения.
- 4 Проектные вопросы.
 - 4.1 Открытые вопросы.
 - 4.2 Предварительный план-график.
 - 4.3 Предварительный бюджет.

5.2.2 Составление спецификации требований.

Требования необходимо специфицировать (т. е. задать) графически или каким-либо иным формальным способом. Всесторонняя спецификация системы может потребовать использования многих типов моделей. Язык UML изобилует интегрированными методами моделирования, способными помочь бизнес-аналитику справиться с этой задачей. Спецификация, подобно процессу разработки ПО в целом, – итеративный процесс с пошаговым наращиванием уровня детализации моделей. Немаловажную роль в успешном моделировании играет использование CASE-средств.

В результате спецификации требований вырабатываются три категории моделей: модели состояний (рисунок Д.1), модели поведения и модели изменения состояния. Для каждой из категорий существует несколько методов работы с ними. Далее объясняются и иллюстрируются на примерах все основные методы моделирования языка UML.

Принципы спецификации требований. Спецификация требований связана с доскональным моделированием требований заказчиков, определенных в процессе установления требований. При этом рассматриваются только услуги, которые стремятся получить от системы заказчики (формулировки сервисов).

В качестве входной информации процесса спецификации требований выступают неформальные требования заказчиков, а результатом этого процесса являются модели спецификации проектных конструкций. Эти модели дают более формальное определение различных сторон (представлений) системы. Обычно требования пользователей в процессе спецификации подразделяются на две основные категории: функциональные требования и требования к данным.

В качестве результата этапа спецификации выступает расширенный («детально проработанный») документ описания требований. Новый документ часто называют документом спецификации требований.

Модели спецификаций можно разделить на три группы:

- 1) модели состояний;
- 2) модели поведения;
- 3) модели изменения состояний.

Модели состояний «детализируют» требования к данным. Модели поведения обеспечивают детализированные спецификации для функциональных требований. Модели изменения состояний охватывают два вида требований. Они призваны объяснить, каким образом действие функций приводит к изменению данных.

Модели представляются в виде диаграмм на языке визуального моделирования (Visual Modeling Language) – это язык UML. Обычно диаграмма служит целям моделирования одной из сторон системы – состояний, поведения или изменения состояний. Заметное исключение составляет диаграмма классов, которая определяет все три аспекта – состояние и поведение объектов и, косвенно, изменения состояний объектов.

Каждая диаграмма дает представление об определенной стороне системы. Взятые вместе диаграммы дают возможность разработчикам и пользователям взглянуть на предлагаемое решение с разных точек зрения, выделяя одни его

стороны и игнорируя другие. Ни одна из диаграмм в отдельности не дает полного определения системы. Систему можно понять только через взаимосвязанный набор диаграмм.

5.2.3 Проектирование структуры слоя доступа к данным.

На стадии анализа моделируются только основные классы, относящиеся к предметной области, т. е. таблицы базы данных (рисунок Е.1). Следует убедиться, что отсутствуют избыточность хранимой информации и аномалии в организации данных, т. е. модель данных необходимо привести к третьему нормальному виду.

В разделе должен быть сформирован документ «Структура базы данных». В документе представлены имена полей каждой таблицы базы данных, тип данных каждого поля и условия назначения.

Для контроля возможности выполнения программной системы функциональных требований заполняется специальная таблица, которая позволяет сопоставить наименование полей в документах и отчетах с полями таблиц базы данных. В таблицах базы данных не должно быть таблиц, информация из которых нигде не используется. Все вычисляемые поля должны иметь ссылки на алгоритмы.

Пример диаграммы классов базы данных представлен на рисунке Б.1.

В заключении по разделу делаются выводы:

- о возможности определить значения всех полей разрабатываемых документов и отчетов;
- об отсутствии в базе таблиц, значение полей которых нигде не используется;
- о приведении базы данных к третьей нормальной форме.

5.2.4 Проектирование подсистемы интеллектуальной обработки данных.

Данный пункт не является обязательным и присутствует в ВКР, содержащих исследовательский компонент.

Интеллектуальная обработка данных предполагает использование цифровых технологий для решения задач, возникающих на различных этапах управления данными в организации: распознавание документов, преобразование в структурированный вид, классификация, анализ информации и ее поиск в различных источниках.

В рамках данного пункта студенты при выполнении ВКР могут реализовывать машинное обучение (классификация и кластеризация данных, полносвязанные и глубинные нейронные сети), методы визуализации данных, методы обработки текстов, методы обработки изображений. При выполнении ВКР, связанных с интеллектуальной обработкой данных, информация может считываться с электронно-цифровых устройств и использоваться для интеллектуального управления различными устройствами.

В процессе интеллектуальной обработки данных студент может разрабатывать и применять специализированное программно-математическое обеспечение для проведения исследований и решения инженерных задач;

разрабатывать модели подсистем и процессов, автоматизируемых при выполнении ВКР; проверять на практике адекватность созданных моделей; использовать пакеты прикладных программ анализа и синтеза электронно-цифровых устройств; выполнять экспериментальные исследования для решения научно-исследовательских и производственных задач с использованием современной аппаратуры и методов исследования; участвовать в научных исследованиях в группе; ставить задачи и выбирать методы экспериментальных исследований.

Для интеллектуальной обработки данных могут быть использованы следующие методы:

- 1) линейные методы классификации данных:
 - линейная и логистическая регрессия;
 - методы, основанные на деревьях принятия решения, в том числе метод случайного леса;
 - метод k-ближайших соседей;
 - методы бустинга и стеккинга;
 - методы оценки результатов классификации: точность, полнота, f-мера, матрица ошибок;
- 2) методы кластеризации данных:
 - метод k-средних;
 - спектральная кластеризация;
 - методы, основанные на оценке плотности распределения точек в пространстве;
- 3) обработка текстов на естественном языке:
 - лексический и синтаксический анализ текстов;
 - выделение именованных сущностей, фактов;
 - классификация и кластеризация текстов;
- 4) «плотные» нейронные сети;
- 5) глубинное обучение нейронных сетей.

Для визуального представления основных результатов интеллектуальной обработки данных рекомендуется использовать диаграммы рассеяния, размаха, гистограммы, эпюры, отображение трехмерных данных, отображение последовательностей.

5.3 Реализация программно-информационной системы

Под программно-информационной системой понимается прикладная подсистема в программной реализации, которая ориентирована на сбор, систематизацию, хранение, поиск и обработку информации требуемого типа.

В состав любой программно-информационной системы входят программные компоненты, которые реализуют следующие функции:

- организовывается диалоговый ввод-вывод информации;
- прописывается логика диалога;
- задается логика обработки данных;
- прописываются алгоритмы управления данными;

– фиксируются операции по манипулированию файлами и (или) базами данных.

Разработка информационных систем базируется на соблюдении следующих принципов (поэтапность):

- определение целей проектирования;
- выбор типа информационной системы;
- разработка и согласование технического задания;
- непосредственная программная реализация;
- внедрение информационной системы;
- сопровождение работающей информационной системы;
- оценка эффективности, эффекта, который приносят данные информационные системы.

5.3.1 Описание архитектуры программно-информационной системы.

Архитектура программного обеспечения – совокупность важнейших решений об организации программной системы. Архитектура включает:

– выбор структурных элементов и их интерфейсов, с помощью которых составлена система, а также их поведения в рамках сотрудничества структурных элементов;

– соединение выбранных элементов структуры и поведения во всё более крупные системы;

– архитектурный стиль, который направляет всю организацию – все элементы, их интерфейсы, их сотрудничество и их соединение.

Документирование архитектуры программного обеспечения (ПО) упрощает процесс коммуникации между разработчиками, позволяет зафиксировать принятые проектные решения и предоставить информацию о них эксплуатационному персоналу системы, повторно использовать компоненты и шаблоны проекта в других.

Архитектура ПО обычно содержит несколько видов, которые аналогичны различным типам чертежей в строительстве зданий.

Архитектурный вид состоит из двух компонентов: элементы и отношения между элементами.

Архитектурные виды можно поделить на три основных типа:

1) модульные виды (англ. *module views*) – показывают систему как структуру из различных программных блоков;

2) компоненты-и-коннекторы (англ. *component-and-connector views*) – показывают систему как структуру из параллельно запущенных элементов (компонентов) и способов их взаимодействия (коннекторов);

3) размещение (англ. *allocation views*) – показывает размещение элементов системы во внешних средах (рисунок Ж.1).

Примеры модульных видов.

Декомпозиция (англ. *decomposition view*) – состоит из модулей в контексте отношения «является подмодулем».

Использование (англ. *uses view*) – состоит из модулей в контексте отношения «использует» (т. е. один модуль использует сервисы другого модуля).

Вид уровней (англ. layered view) – показывает структуру, в которой связанные по функциональности модули объединены в группы (уровни).

Вид классов/обобщений (англ. class/generalization view) – состоит из классов, связанных через отношения «наследуется от» и «является экземпляром».

Примеры видов компонентов-и-коннекторов.

Процессный вид (англ. process view) – состоит из процессов, соединённых операциями коммуникации, синхронизации и/или исключения.

Параллельный вид (англ. concurrency view) – состоит из компонентов и коннекторов, где коннекторы представляют собой «логические потоки».

Вид обмена данными (англ. shared-data (repository) view) – состоит из компонентов и коннекторов, которые создают, сохраняют и получают постоянные данные.

Вид клиент-сервер (англ. client-server view) – состоит из взаимодействующих клиентов и серверов и коннекторов между ними (например, протоколов и общих сообщений).

Примеры видов размещения.

Развертывание (англ. deployment view) – состоит из программных элементов, их размещения на физических носителях и коммуникационных элементов.

Внедрение (англ. implementation view) – состоит из программных элементов и их соответствия файловым структурам в различных средах (разработческой, интеграционной и т. д.).

Распределение работы (англ. work assignment view) – состоит из модулей и описания того, кто ответственен за внедрение каждого из них.

Примеры архитектурных шаблонов.

Многоуровневый шаблон (Layered pattern). Система разбивается на уровни, которые на диаграмме изображаются один над другим. Каждый уровень может вызывать только уровень на один ниже него. Таким образом, разработку каждого уровня можно вести относительно независимо, что повышает модифицируемость системы. Недостатками данного подхода являются усложнение системы и снижение производительности.

Шаблон посредника (Broker pattern). Когда в системе присутствует большое количество модулей, их прямое взаимодействие друг с другом становится слишком сложным. Для решения проблемы вводится посредник (например, шина данных), по которой модули общаются друг с другом. Таким образом, повышается функциональная совместимость модулей системы. Все недостатки вытекают из наличия посредника: он понижает производительность, его недоступность может сделать недоступной всю систему, он может стать объектом атак и узким местом системы.

Шаблон «Модель-Представление-Контроллер» (Model-View-Controller pattern). Так как требования к интерфейсу меняются чаще всего, то возникает потребность часто его модифицировать, при этом сохраняя корректное взаимодействие с данными (чтение, сохранение). Для этого в шаблоне Model-View-Controller (MVC) интерфейс отделён от данных. Это позволяет менять интерфейсы, равно как и создавать их разные варианты. В MVC система разделена на:

- модель, хранящую данные;
- представление, отображающее часть данных и взаимодействующее с пользователем;
- контроллер, являющийся посредником между видами и моделью.

Однако концепция MVC имеет и свои недостатки. В частности, из-за усложнения взаимодействия падает скорость работы системы.

Клиент-серверный шаблон (Client-Server pattern). Если есть ограниченное число ресурсов, к которым требуется ограниченный правами доступ большого числа потребителей, то удобно реализовать клиент-серверную архитектуру. Такой подход повышает масштабируемость и доступность системы. Но при этом сервер может стать узким местом системы, при его недоступности становится недоступна вся система.

В данном разделе необходимо выполнить описание архитектуры проектируемой программно-информационной системы.

5.3.2 Пример использования программно-информационной системы.

В разделе приводятся основной пример использования разработанной программной системы, снимки экранов и подробно описывается принцип использования программной системы.

5.4 Результаты

В разделе приводятся:

- полученные результаты в ходе выполнения ВКР, важнейшие предложения;
- обобщенные данные о достигнутых результатах;
- оценка выполнения поставленных в ВКР целей, задач, предполагаемых результатов;
- пути реализации предложений по повышению эффективности деятельности предприятий, которые будут эксплуатировать разработанную программную систему.

5.5 Заключение

В заключении приводятся основные результаты выполнения ВКР, дается их оценка, достигнутые результаты сопоставляются с требованиями задания на ВКР. В заключении логически последовательно обосновываются и излагаются собственные позиции студента. Этот раздел содержит теоретические и практические выводы и предложения, к которым пришел студент в результате исследования. Они должны быть краткими и четкими, дающими полное представление о содержании, значимости, обоснованности и эффективности разработок. Эти предложения оформляются в виде тезисов (по пунктам) и должны отражать основные выводы по теории вопроса, по проведенному анализу и всем предлагаемым направлениям совершенствования проблемы с оценкой их эффективности по конкретному объекту исследования.

Список литературы

- 1 **МИ БРУ 2.003–2018.** Выпускная квалификационная работа. Организация выполнения и защита: методическая инструкция Белорусско-Российского университета. – Могилев: Белорус.-Рос. ун-т, 2018. – 34 с.
- 2 **Брайант, Р. Дэвид.** Компьютерные системы: архитектура и программирование. Взгляд программиста: пер. с англ. / Р. Дэвид Брайант, О. Халларон. – Санкт-Петербург : БХВ-Петербург, 2012. – 1104 с.
- 3 **Орлов, С. А.** Технологии разработки программного обеспечения. Разработка сложных программных систем : учебник / С. А. Орлов. – 3-е изд. – Санкт-Петербург : Питер, 2004. – 527 с.
- 4 **Леоненков, А. В.** Самоучитель UML / А. В. Леоненков. – 2-е изд., перераб. и доп. – Санкт-Петербург : БХВ-Петербург, 2012. – 432 с.
- 5 **Буч, Г.** UML. Классика computer science: пер. с англ. / Г. Буч, А. Якобсон, Дж. Рамбо; под ред. С. Орлова. – 2-е изд. – Санкт-Петербург: Питер, 2006. – 736 с.
- 6 **Буч, Г.** Язык UML. Руководство пользователя: пер. с англ. / Г. Буч, Д. Рамбо, И. Якобсон. – 2-е изд. – Москва : ДМК Пресс, 2007. – 496 с.
- 7 **Рамбо, Дж.** UML 2.0. Объектно-ориентированное моделирование и разработка / Дж. Рамбо, М. Блаха. – 2-е изд. – Санкт-Петербург: Питер, 2007. – 544 с.
- 8 **Заботина, Н. Н.** Методы и средства проектирования информационных систем [Электронный ресурс]: учебное пособие / Н. Н. Заботина. – Москва: ИНФРА-М, 2020. – 331 с. – Режим доступа: <https://znanium.com/catalog/product/1043093>. – Дата доступа: 01.11.2021.
- 9 **Шустова, Л. И.** Базы данных [Электронный ресурс]: учебник / Л. И. Шустова, О. В. Тараканов. – Москва : ИНФРА-М, 2021. – 304 с. – Режим доступа: <https://znanium.com/catalog/product/1189322>. – Дата доступа: 01.11.2021.
- 10 **Агальцов, В. П.** Базы данных: в 2 кн. Кн. 2: Распределенные и удаленные базы данных: учебник / В. П. Агальцов. – Москва: ФОРУМ; ИНФРА-М, 2020. – 271 с. – Режим доступа: <https://znanium.com/catalog/product/1093648>. – Дата доступа: 01.11.2021.
- 11 **Гагарина, Л. Г.** Технология разработки программного обеспечения [Электронный ресурс]: учебное пособие / Л. Г. Гагарина, Е. В. Кокорева, Б. Д. Сидорова-Виснадул ; под ред. Л. Г. Гагариной. – Москва : ФОРУМ; ИНФРА-М, 2021. – 400 с. – Режим доступа: <https://znanium.com/catalog/product/1189951>. – Дата доступа: 01.11.2021.
- 12 **Макконнелл С.** Совершенный код. Мастер-класс=Code Complete. Second Edition : пер. с англ. / С. Макконнелл. – Санкт-Петербург: БХВ, 2020. – 896 с.
- 13 **Макаровских, Т. А.** Документирование программного обеспечения. В помощь техническому писателю : учебное пособие / Т. А. Макаровских. – 2-е изд. – Москва : ЛЕНАНД, 2015. – 266 с.
- 14 **Бедердинова, О. И.** Моделирование информационных систем на платформе SOFTWARE IDEAS MODELER [Электронный ресурс]: учебное посо-

бие / О. И. Бедердинова, Л. В. Кремлева, С. В. Протасова. – Москва: ИНФРА-М, 2019. – 166 с. – Режим доступа: <https://znanium.com/catalog/product/1020362>. – Дата доступа: 01.11.2021.

15 **Магазанник, В. Д.** Человеко-компьютерное взаимодействие [Электронный ресурс]: учебное пособие / В. Д. Магазанник. – 2-е изд., доп. – Москва : Университетская книга, 2020. – 408 с. – Режим доступа: <https://znanium.com/catalog/product/1214481>. – Дата доступа: 01.11.2021.

16 **Хорев, П. Б.** Объектно-ориентированное программирование с примерами на С# [Электронный ресурс]: учебное пособие / П. Б. Хорев. – Москва: ФОРУМ; ИНФРА-М, 2020. – 200 с. – Режим доступа: <https://znanium.com/catalog/product/1069921>. – Дата доступа: 01.11.2021.

17 **Федорова, Г. Н.** Разработка, внедрение и адаптация программного обеспечения отраслевой направленности [Электронный ресурс]: учебное пособие / Г. Н. Федорова. – Москва: КУРС; ИНФРА-М, 2021. – 336 с. – Режим доступа: <https://znanium.com/catalog/product/1138896>. – Дата доступа: 01.11.2021.

Приложение А (рекомендуемое)

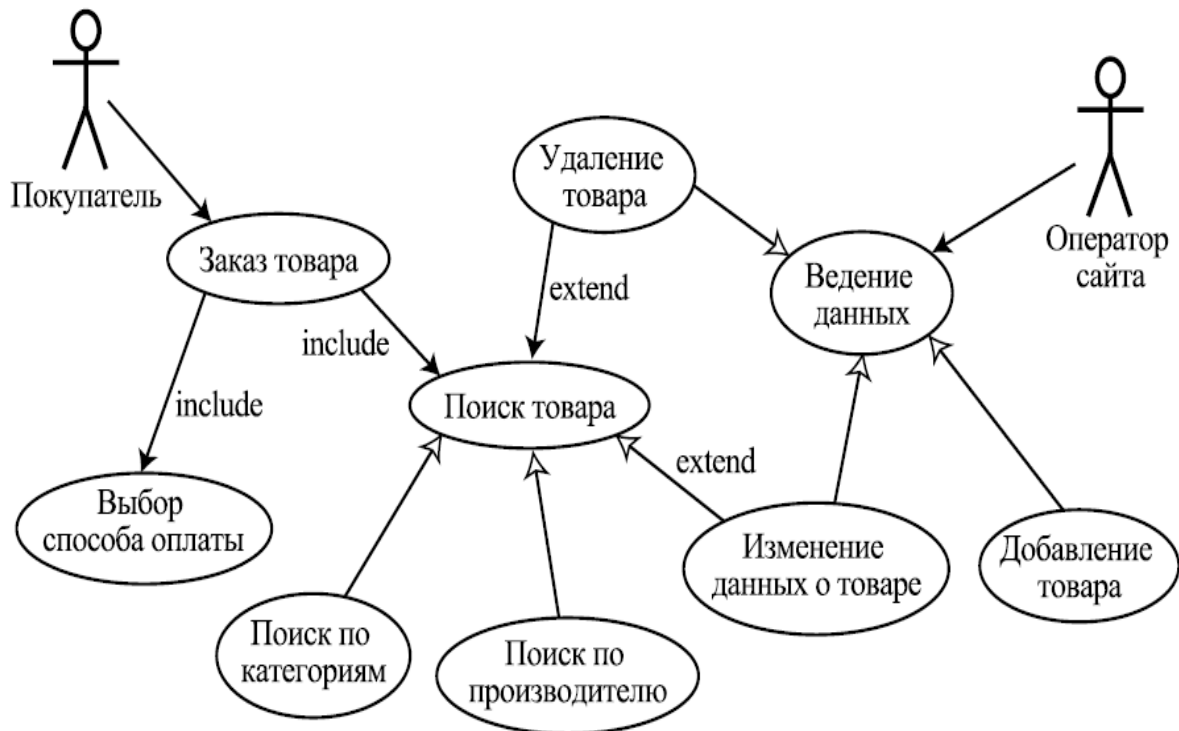


Рисунок А.1 – Диаграмма вариантов использования

Приложение Б (рекомендуемое)

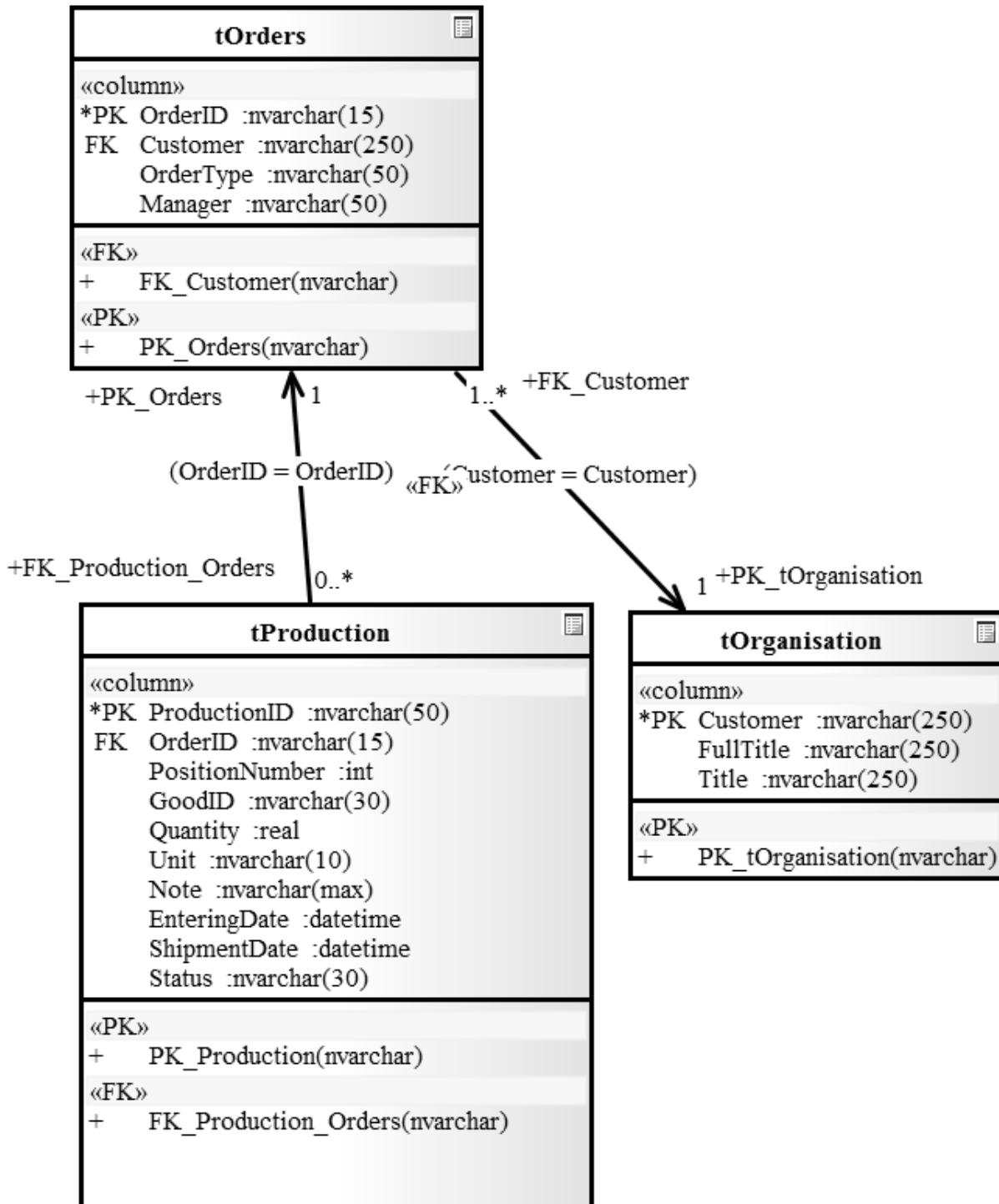


Рисунок Б.1 – Диаграмма классов базы данных

Приложение В (рекомендуемое)

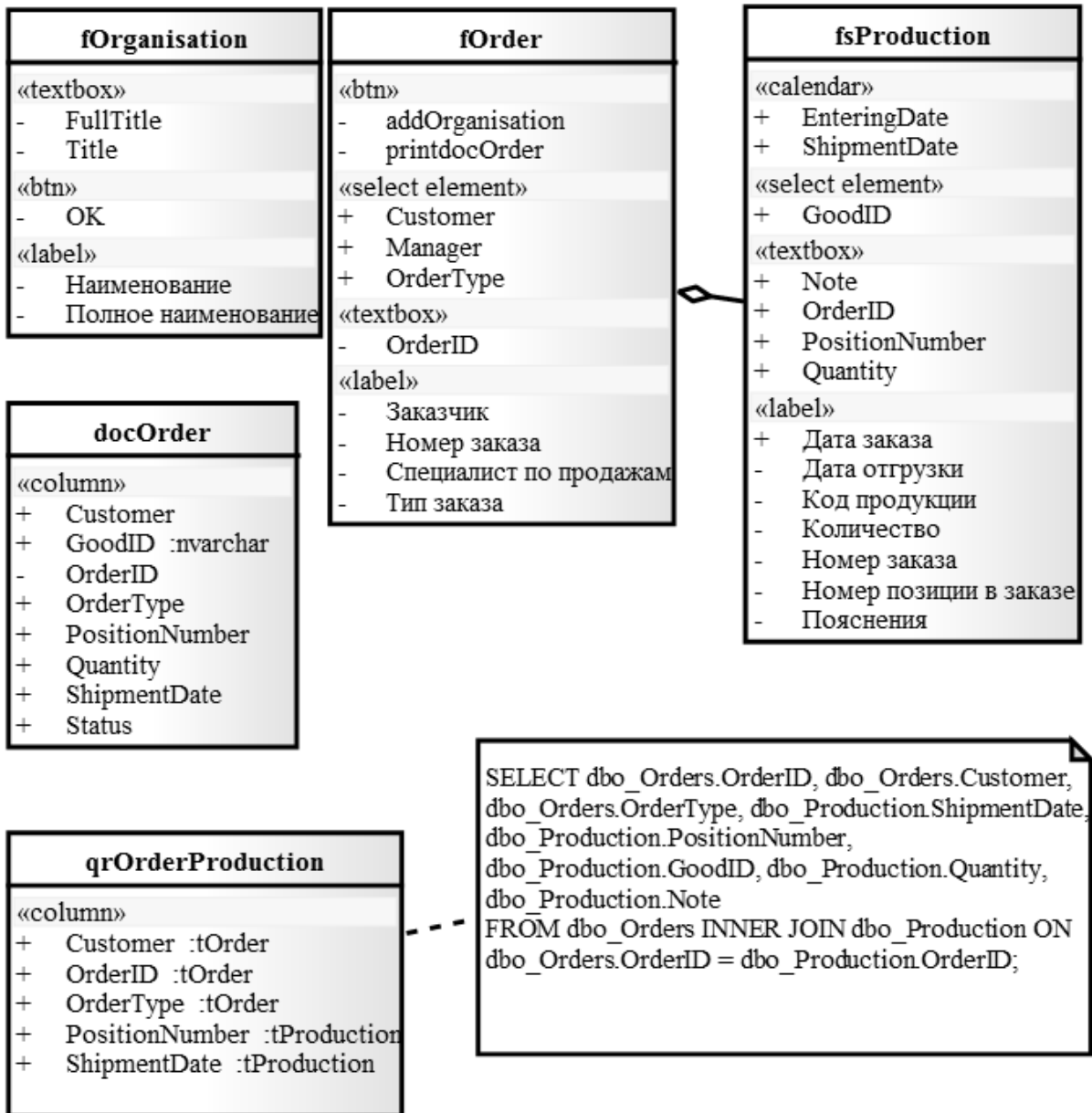


Рисунок В.1 – Диаграмма классов объектов программной системы

Приложение Г (рекомендуемое)

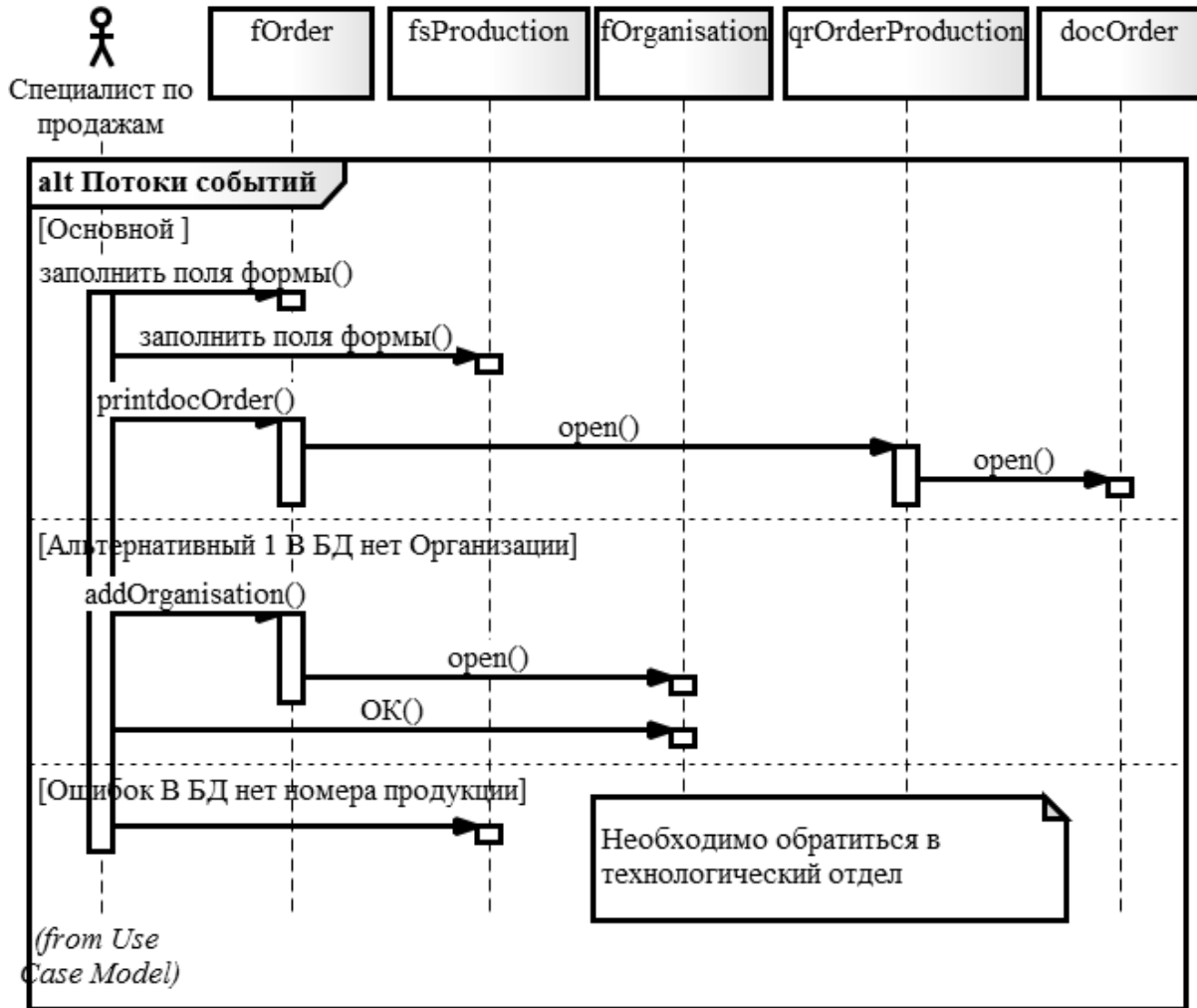


Рисунок Г.1 – Диаграмма последовательности варианта использования

Приложение Д (рекомендуемое)

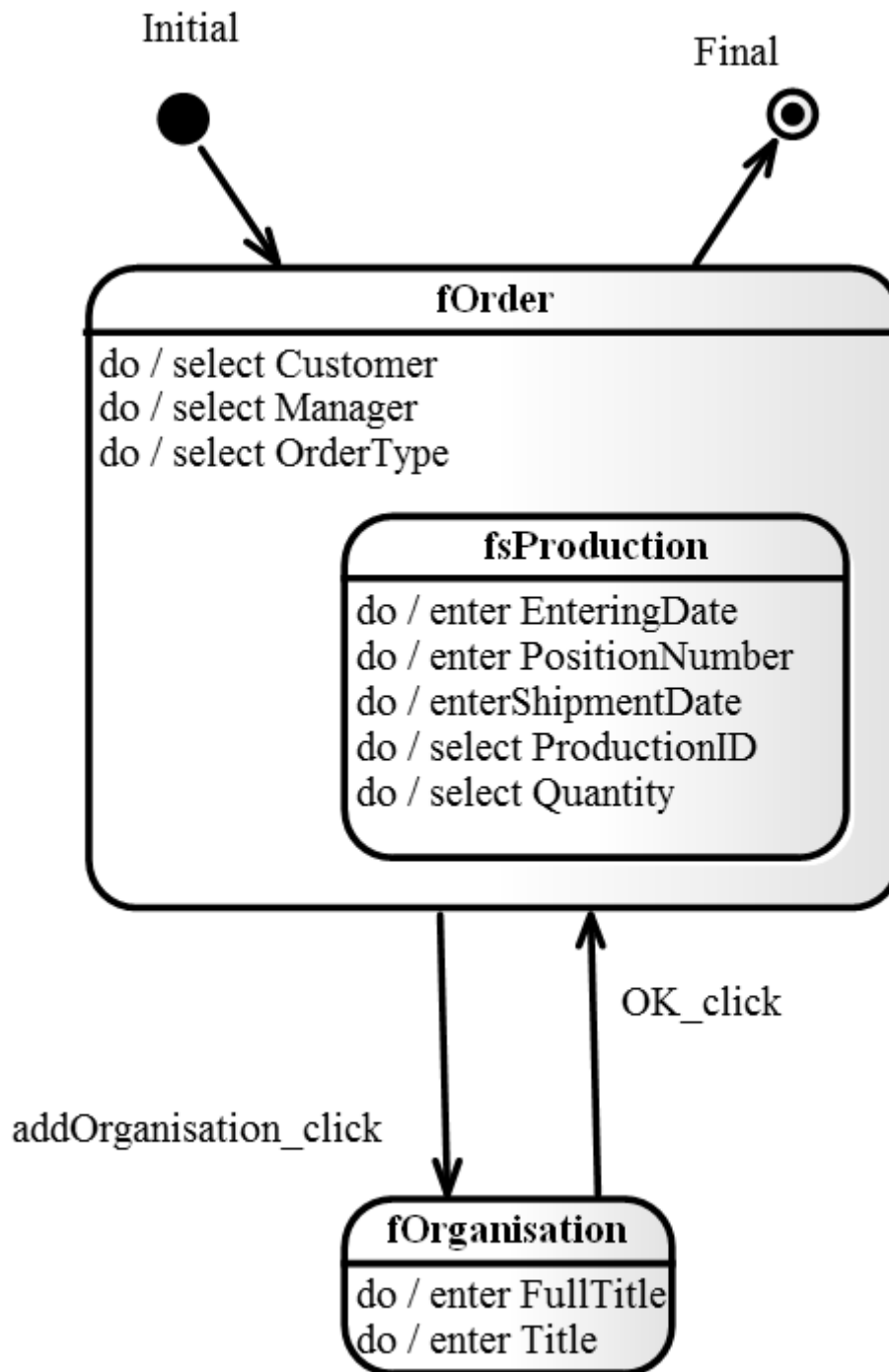


Рисунок Д.1 – Диаграмма состояний

Приложение Е (рекомендуемое)

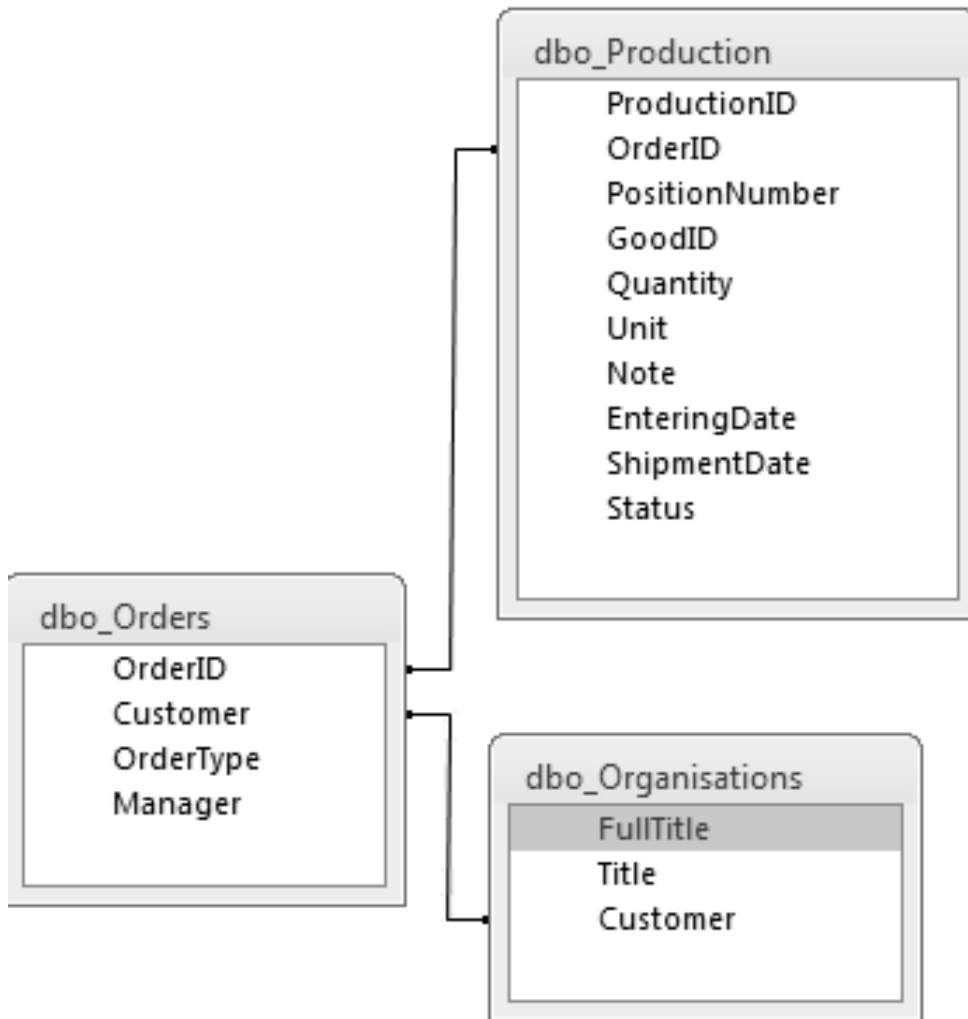


Рисунок Е.1 – Схема данных

Приложение Ж (рекомендуемое)

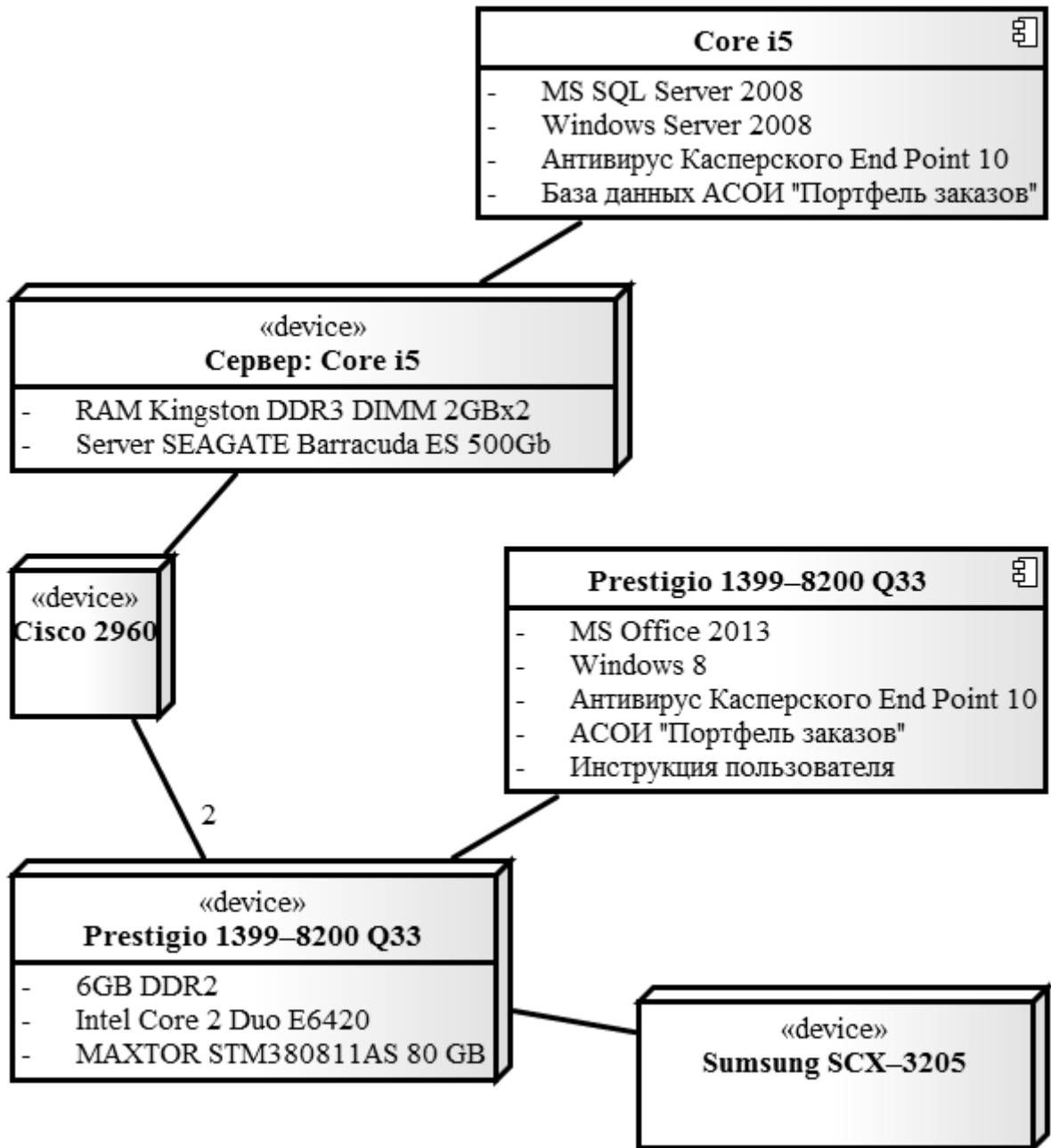


Рисунок Ж.1 – Диаграмма компонентов и размещения