

ПЕРМСКИЙ
ГОСУДАРСТВЕННЫЙ
НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ

ОСНОВЫ ПРОЕКТИРОВАНИЯ
И РЕАЛИЗАЦИИ
ИНФОРМАЦИОННЫХ СИСТЕМ



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное
образовательное учреждение высшего образования
«ПЕРМСКИЙ ГОСУДАРСТВЕННЫЙ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»

ОСНОВЫ ПРОЕКТИРОВАНИЯ И РЕАЛИЗАЦИИ ИНФОРМАЦИОННЫХ СИСТЕМ

*Допущено методическим советом
Пермского государственного национального
исследовательского университета в качестве
учебно-методического пособия для студентов,
обучающихся по направлению подготовки бакалавров
«Фундаментальная информатика и информационные технологии»*



Пермь 2021

УДК 32.97(075.8)
ББК 004.434: 004.94
Д215

Дацун Н. Н.

Д215 Основы проектирования и реализации информационных систем [Электронный ресурс] : учебно-методическое пособие / Н. Н. Дацун, А. В. Отинов, Т. С. Гашева, Д. И. Власов ; Пермский государственный национальный исследовательский университет. – Электронные данные. – Пермь, 2021. – 5,78 Мб ; 132 с. – Режим доступа: <http://www.psu.ru/files/docs/science/books/uchebnie-posobiya/Dacun-Otinov-Gasheva-Vlasov-Osnovy-roektirovaniya-i-realizacii-informacionnyh-sistem.pdf>. – Заглавие с экрана.

ISBN 978-5-7944-3703-4

Пособие представляет собой методические рекомендации по выполнению комплекса лабораторных работ и проведения практических занятий по дисциплине «Основы проектирования и реализации информационных систем» при объектно-ориентированном подходе к моделированию. Сформулированы общие методические указания и требования к комплексу работ. Для каждой лабораторной работы и практических занятий приведены цель, методические указания, сроки выполнения и общая оценка, пример выполнения лабораторной работы (домашнего задания), задания, содержание отчета, критерии оценивания. Справочный материал содержит информацию, необходимую для выполнения парного оценивания проектов, созданных студентами, и список типичных ошибок в студенческих моделях.

Издание предназначено для студентов бакалавриата очной и заочной форм обучения направления подготовки «Фундаментальная информатика и информационные технологии», изучающих дисциплину «Основы проектирования и реализации информационных систем». Учебное пособие будет полезно студентам и другим ИТ-направлений и специальностей.

УДК 32.97(075.8)
ББК 004.434: 004.94

*Издается по решению ученого совета механико-математического факультета
Пермского государственного национального исследовательского университета*

Рецензенты: кафедра информационных технологий в бизнесе НИУ ВШЭ – Пермь (зав. кафедрой, д-р пед. наук, профессор **Е. Г. Плотникова**, канд. физ.-мат. наук, доцент **Л. В. Шестакова**, канд. техн. наук **О. Л. Викентьева**); доцент кафедры высшей математики Санкт-Петербургского государственного архитектурно-строительного университета, канд. физ.-мат. наук **Л. Ю. Уразаева**

© ПГНИУ, 2021

© Дацун Н. Н., Отинов А. В., Гашева Т. С.,
Власов Д. И., 2021

ISBN 978-5-7944-3703-4

СОДЕРЖАНИЕ

Введение	7
1 Входной контроль по дисциплине «ОПРИС»	9
2 Общие методические указания и требования	10
2.1 Персонификация файлов документов.....	10
2.2 Оформление документов в соответствии с ГОСТ	10
2.3 Документирование моделей	10
2.4 Представление моделей.....	10
2.5 Работа в команде очной формы обучения.....	11
2.6 Работа студентов заочной формы обучения	11
2.7 Заимствование	11
2.8 Порядок предоставления отчетов по ЛР и ДЗ	11
2.9 Оценивание и самооценивание ЛР и ДЗ, его связь с графиком учебного процесса по дисциплине «ОПРИС»	12
3 Практическое занятие № 1. Требования к ИС	13
3.1 Цель практического занятия № 1	13
3.2 Методические указания к проведению практического занятия № 1 ..	13
3.3 Краткие теоретические сведения к проведению практического занятия № 1	13
3.4 Пример: описание предметной области	15
3.5 Домашнее задание № 1	19
3.6 Вопросы для самопроверки	20
4 Лабораторная работа № 1. Моделирование на этапе анализа ИС: функциональность ИС	23
4.1 Цель лабораторной работы № 1	23
4.2 Общие методические указания к выполнению ЛР № 1	23
4.3 Сроки выполнения и представление отчета по ЛР № 1	23
4.4 Пример выполнения ЛР № 1	23
4.5 Требования, указания и задания к ЛР № 1	27
4.6 Содержание отчета по ЛР № 1	27
4.7 Оценивание ЛР № 1	27
5 Практическое занятие № 2. Объектно-ориентированный подход к моделированию: моделирование функциональных требований	29
5.1 Цель практического занятия № 2	29
5.2 Методические указания к проведению практического занятия № 2 ..	29
5.3 Пример описания модели функциональных требований	29
5.4 Домашнее задание № 2.....	34
5.5 Вопросы для самопроверки	34
6 Практическое занятие № 3. Объектно-ориентированный подход к моделированию: моделирование бизнес-процессов	36
6.1 Цель практического занятия № 3	36
6.2 Методические указания к проведению практического занятия № 3 ..	36

6.3	Пример описания модели бизнес-процессов	36
6.4	Домашнее задание № 3.....	38
6.5	Вопросы для самопроверки	38
7	Практическое занятие № 4. Объектно-ориентированный подход к моделированию: концептуальная модель	40
7.1	Цель практического занятия № 4	40
7.2	Методические указания к проведению практического занятия № 4 ..	40
7.3	Пример создания концептуальной модели	40
7.4	Домашнее задание № 4.....	49
7.5	Вопросы для самопроверки	51
8	Лабораторная работа № 2. Моделирование на этапе анализа ИС при объектно-ориентированном подходе: моделирование поведения системы.....	53
8.1	Цель лабораторной работы № 2	53
8.2	Общие методические указания к выполнению ЛР № 2.....	53
8.3	Сроки выполнения и представление отчета по ЛР № 2.....	53
8.4	Пример выполнения ЛР № 2.....	54
8.5	Пример определения системных операций.....	56
8.6	Требования, указания и задания к ЛР № 2	63
8.7	Содержание отчета по ЛР № 2	63
8.8	Вопросы для самопроверки	64
8.9	Оценивание ЛР № 2.....	64
9	Практическое занятие № 5. Этап проектирования ИС при объектно-ориентированном подходе: моделирование поведения системы	66
9.1	Цель практического занятия № 5	66
9.2	Методические указания к проведению практического занятия № 5 ..	66
9.3	Пример модели поведения системы на этапе проектирования.....	68
9.4	Домашнее задание № 5.....	68
9.5	Вопросы для самопроверки	68
10	Лабораторная работа № 3. Этап проектирования ИС при объектно-ориентированном подходе: моделирование структуры системы	73
10.1	Цель лабораторной работы № 3	73
10.2	Общие методические указания к выполнению ЛР № 3.....	73
10.3	Сроки выполнения и представление отчета по ЛР № 3.....	73
10.4	Пример выполнения ЛР № 3.....	74
10.5	Пример описания некоторых классов.....	79
10.6	Указания и задания к ЛР № 3	81
10.7	Содержание отчета по ЛР № 3	81
10.8	Оценивание ЛР № 3	81
11	Лабораторная работа № 4. Этап реализации ИС при объектно-ориентированном подходе	83
11.1	Цель лабораторной работы № 4	83
11.2	Общие методические указания к выполнению ЛР № 4.....	83

11.3 Сроки выполнения и представление отчета по ЛР № 4.....	88
11.4 Пример выполнения ЛР № 4 (диаграмма компонентов)	88
11.5 Пример выполнения ЛР № 4 (диаграмма развертывания)	92
11.6 Содержание отчета по ЛР № 4	93
11.7 Оценивание ЛР № 4.....	94
12 Варианты описания предметных областей индивидуальных заданий.....	95
13 Критерии парного оценивания моделей, созданных студентами	100
14 Выписки из учебных планов.....	105
Список использованных источников	106
ПРИЛОЖЕНИЕ А Пример оформления титульного листа отчета по лабораторной работе в соответствии с ГОСТ 7.32	108
ПРИЛОЖЕНИЕ Б Инструкция по экспорту файлов модели из инструмента GenMyModel после создания UML диаграмм.....	109
ПРИЛОЖЕНИЕ В Типичные ошибки в UCD студентов	111
ПРИЛОЖЕНИЕ Г Типичные ошибки в AD студентов	117
ПРИЛОЖЕНИЕ Д Типичные ошибки в CD студентов	125

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ

В настоящем пособии применяют следующие сокращения и обозначения

CASE	– <i>англ.</i> Computer-Aided Software Engineering – инструмент, который позволяет автоматизировать процесс разработки информационной системы и программного обеспечения
DDD	– <i>англ.</i> Domain-driven design – предметно-ориентированное проектирование [1]
LMS	– <i>англ.</i> Learning Management System – система управления обучением
Moodle	– <i>англ.</i> Modular Object-Oriented Dynamic Learning Environment – модульная объектно-ориентированная динамическая обучающая среда [2]
англ	– английский
БД	– база данных
ГОСТ	– межгосударственный стандарт
ГОСТ Р	– национальный стандарт Российской Федерации
грейдер	– студент, оценивающий выполнение работы другого студента
ДЗ	– домашнее задание
ДП	– диаграмма последовательностей
др.	– другие
ЕТИС	– единая телеинформационная система ПГНИУ
ИС	– информационная система
КТ	– контрольная точка
ЛР	– лабораторная работа
напр.	– например
ООП	– объектно-ориентированное программирование
ОПРИС	– основы проектирования и реализации информационных систем
ПГНИУ	– Пермский государственный национальный исследовательский университет
ПИ	– пользовательский интерфейс
ПО	– программное обеспечение
пр.	– прочее
ПрО	– предметная область
СО	– системная операция
СУБД	– система управления базой данных
т.д.	– так далее
т. е.	– то есть
т.к.	– так как
т. п.	– тому подобное
ФИО	– фамилия и инициалы
ЯП	– язык программирования

ВВЕДЕНИЕ

В вариативной части учебных планов направления подготовки «Фундаментальная информатика и информационные технологии» (специализация «Открытые информационные системы») очной и заочной форм обучения предусмотрено изучение дисциплины «Основы проектирования и реализации информационных систем». Целью этой дисциплины является подготовка студентов, способных анализировать задачи, стоящие перед ИС, принимать проектные решения, документировать и реализовывать их на различных этапах жизненного цикла ИС. Практический блок дисциплины «ОПРИС» направлен на приобретение практических навыков применения методологий моделирования, проектирования и реализации информационных систем для решения прикладных задач, на выработку на основе приобретенных знаний и навыков необходимых компетенций для решения конкретных профессиональных задач.

Комплекс лабораторных работ, практических занятий и самостоятельной работы студентов по дисциплине «Основы проектирования и реализации информационных систем» решает задачи формирования следующих компетенций: владеть современным математическим аппаратом, фундаментальными концепциями и системными методологиями; способность создавать, анализировать, реализовывать математические и информационные модели с применением современных вычислительных систем; способность применять современные парадигмы и методологии программирования, языки программирования и языки баз данных; способность разрабатывать алгоритмические и программные решения в области системного и прикладного программирования, тесты и средства тестирования систем на соответствие стандартам и исходным требованиям.

В пособии представлены:

- информация об организации и проведении входного контроля по дисциплине;
- общие методические указания и требования к организации практической части дисциплины;
- цель, методические указания, пример выполнения задания по темам:
 - 1) требования к ИС;
 - 2) моделирование на этапе анализа ИС: функциональность ИС;
 - 3) объектно-ориентированный подход к моделированию: моделирование функциональных требований;

4) объектно-ориентированный подход к моделированию: моделирование бизнес-процессов;

5) объектно-ориентированный подход к моделированию: концептуальная модель;

6) моделирование на этапе анализа ИС при объектно-ориентированном подходе: моделирование поведения системы;

7) этап проектирования ИС при объектно-ориентированном подходе: моделирование поведения системы;

8) этап проектирования ИС при объектно-ориентированном подходе: моделирование структуры системы;

9) этап реализации ИС при объектно-ориентированном подходе.

– варианты описания ПрО индивидуальных заданий для моделирования ИС;

– критерии парного оценивания моделей, созданных студентами.

Целью комплекса занятий практического блока является создание моделей этапа анализа, проектирования и реализации информационной системы при объектно-ориентированном подходе. Предусматривается работа студентов очной формы обучения в командах из двух-трех человек. Результат моделирования команд оформляется в виде презентации, которая проходит процедуру парного оценивания и публичной защиты с ответами на замечания рецензентов работ.

Пособие содержит список печатных и электронных источников.

В приложениях представлен образец документа, необходимого для оформления отчетов по лабораторным работам и домашним заданиям, инструкция по экспорту модели из инструмента создания UML диаграмм; списки типичных ошибок, которые допускают студенты при создании диаграмм вариантов использования, активностей и классов.

Текст методического пособия подготовлен в соответствии с ГОСТ 7.32 [3] и Р 7.0.5 [4].

Разделы 3, 4, приложение В подготовлены Дацун Н.Н., Отиновым А.В., раздел 5, приложение Г – Дацун Н.Н., Гашевой Т.С., раздел 10, приложение Д – Дацун Н.Н., Власовым Д.И. Остальные разделы пособия подготовлены Дацун Н.Н.

1 Входной контроль по дисциплине «ОПРИС»

Задание 1. Типы моделей системы.

Задание 2. Классификация систем моделирования.

Задание 3. Классификация ИС по типу хранимых данных.

Задание 4. Модели жизненного цикла ИС.

Задание 5. Понятие «активность» системы.

Задание 6. Стандарт, содержащий описание стадий и этапов работы по каноническому проектированию ИС.

Задание 7. Набор технологий, опирающихся на модель СОМ.

Задание 8. Понятие «событие» в системе.

Задание 9. В пристанционном буфете автостанции «Култаево» работает один продавец. Среднее время обслуживания одного покупателя 4 мин. Стоянка автобуса рейса № 109 на этой станции 5 мин., стоянка автобуса рейса № 339 – 25 мин. Если продавец занят обслуживанием пассажира рейса № 339 в тот момент, когда в буфет вошел пассажир рейса № 109, то этот пассажир рейса № 109 уедет без покупки. Постройте граф состояний, описывающий эту систему массового обслуживания.

Задание 10. Базовый класс `CPoint` определяет понятие «точка на плоскости с координатами x и y ». Производный класс `CCircle` определяет понятие «окружность с центром в указанной точке с заданным радиусом R ». Опишите классы `CPoint` и `CCircle` на языке `C++`.

Входной контроль по дисциплине «Основы проектирования и реализации информационных систем» у студентов очной формы обучения проводится во время лабораторных работ в виде теста на платформе LMS Moodle [2] при первом посещении студентом занятий по дисциплине. В ЕТИС это КТ1; этот результат в ЕТИС заносится, но в итоговой оценке по дисциплине не учитывается. Оценивание входного контроля по дисциплине «ОПРИС»: «зачтено» (5–10 баллов) и «не зачтено» (меньше 5 баллов).

2 Общие методические указания и требования

2.1 Персонафикация файлов документов

Файлы отчетов по лабораторным работам и домашним заданиям должны иметь семантические имена, которые содержат:

- номер ЛР или ДЗ,
- ФИО студента и шифр группы.

Примеры:

- ЛР1 Якин П.С., ФИТ-2-2019.docx,
- ДЗ4 Якушин В.П., ММЗ-31-2017.zip.

2.2 Оформление документов в соответствии с ГОСТ

Отчеты по лабораторным работам и домашним заданиям необходимо оформлять в соответствии с ГОСТ 7.32 [3] и ГОСТ Р 7.0.5 [4]. Пример оформления титульного листа отчета приведен в приложении А. Список использованных источников в данном учебном пособии оформлен в соответствии со стандартом ГОСТ Р 7.0.5, поэтому его можно использовать как шаблон при оформлении любых отчетов.

2.3 Документирование моделей

Документы моделей должны содержать комментарии о названии модели, и ее авторе. Достаточно к моделям на диаграммах языка UML добавлять в верхнем левом углу соответствующую заметку, в которой следует указать:

- название модели,
- название ИС,
- ФИО студента, шифр группы;
- дату создания / модификации.

2.4 Представление моделей

Отчеты по лабораторным работам и домашним заданиям, которые предусматривают разработку моделей, должны сопровождаться файлами модели, созданными и экспортированными средствами GenMyModel [5]:

- файл модели в формате xmi,
- графический файл модели в формате png.

В случае создания альбома диаграмм для каждой из них в соответствующий элемент «Задание» сайта moodle.movs.psu.ru необходимо загрузить пару – xmi- и png-файлы.

Инструкция по экспорту согласованных xmi- и png- файлов модели из GenMyModel приведена в приложении Б.

2.5 Работа в команде очной формы обучения

Работа над созданием модели ИС выполняется студентами очной формы обучения в команде из 2-3 человек. Рекомендуется чередовать распределение ролей между участниками команды.

2.6 Работа студентов заочной формы обучения

Работа над созданием модели ИС выполняется студентами заочной формы обучения индивидуально.

2.7 Заимствование

В случае выявления заимствований в тексте отчета по ЛР или ДЗ соответствующая КТ студентам не засчитывается, им выдается для выполнения новое задание.

2.8 Порядок предоставления отчетов по ЛР и ДЗ

Варианты заданий по ЛР и ДЗ в соответствии с п. 2.5 для команд студентов очной формы обучения выдаются преподавателем, который проводит лабораторные работы и/или практические занятия, или выбираются студентами самостоятельно.

Варианты заданий по ЛР и ДЗ для студентов заочной формы обучения выдаются преподавателем, который проводит лабораторные работы и/или практические занятия. Номер варианта задания соответствует порядковому номеру студента в журнале успеваемости в ЕТИС.

Информация о вариантах заданий будет доступна весь период обучения:

– для студентов очной формы обучения на курсе «Основы проектирования и реализации информационных систем (гр. ФИТ)» <https://moodle.movs.psu.ru/course/view.php?id=16> (далее «ОПРИС»),

– для студентов заочной формы обучения на курсе «Основы проектирования и реализации информационных систем (ФИТ, заочники)» <https://moodle.movs.psu.ru/course/view.php?id=37> (далее «ОПРИС-3»).

Отчеты по выполнению ЛР и ДЗ должны быть размещены каждым участником команды студентов очной формы обучения в соответствующих элементах «Задание» курса «ОПРИС» на moodle.movs.psu.ru. Отчеты должны

быть размещены для проверки преподавателями не позже, чем в 15:00 за один рабочий день до даты защиты работы.

Отчеты по выполнению ЛР и ДЗ должны быть размещены каждым студентом заочной формы обучения в соответствующих элементах «Задание» курса «ОПРИС-3» на moodle.movs.psu.ru. Отчеты должны быть размещены для проверки преподавателями не позже, чем в 15:00 за один рабочий день до даты защиты работы.

2.9 Оценивание и самооценивание ЛР и ДЗ, его связь с графиком учебного процесса по дисциплине «ОПРИС»

Для каждой ЛР или ДЗ указаны сроки выполнения в соответствии с графиком учебного процесса по дисциплине «ОПРИС» и критерии оценивания.

В случае отставания студентами очной формы обучения от установленного графика, опоздание приводит к снижению общей оценки за ЛР и ДЗ (каждая учебная неделя – минус один балл).

Итоговый результат работы каждой команды студентов очной формы обучения оформляется в виде презентации, которая отражает все этапы проектирования и реализации ИС, выполненные в рамках ЛР и ДЗ. Эти презентации проходят процедуру парного оценивания. При этом оценивание выполненной работы каждого студента выполняют 2-3 других студента этой учебной группы с помощью элемента «Парное оценивание работ студентов» курса «ОПРИС» на moodle.movs.psu.ru.

Защита проекта информационной системы каждой командой проводится публично в часы последнего практического занятия. Презентации, представленные на публичную защиту в форуме «Презентации проектов для оценивания студентами из других команд» курса «ОПРИС» на moodle.movs.psu.ru, должны быть исправлены с учетом замечаний рецензентов. Степень участия в парном оценивании, качество этого оценивания, представление и защита проекта учитываются в итоговой оценке дисциплины.

3 Практическое занятие № 1. Требования к ИС

3.1 Цель практического занятия № 1

Практическое занятие направлено на изучение требований к ИС, использование методов выявления требований и формирование соответствующей документации. Выполняется студентами очной формы обучения.

3.2 Методические указания к проведению практического занятия № 1

Для проведения практического занятия №1 необходимо предварительно изучить ГОСТ 19.201 [6] и ГОСТ 34.602 [7].

3.3 Краткие теоретические сведения к проведению практического занятия № 1

3.3.1 Определение границ информационной системы. Информационные системы являются открытыми, т. е. взаимодействующими с внешней средой. Функции системы проявляются в процессе ее взаимодействия с внешней средой. При этом важно определить границу между внешней средой и создаваемой системой. Разработчик должен определить границы системы, полагая, что цель ее функционирования известна. Необходимо в состав системы включить те элементы, которые своим функционированием обеспечивают реализацию заданной цели. Все то, что не вошло в состав системы, относят к окружающей среде. Окружающая среда включает в себя другие системы, которые реализуют свои цели функционирования. На рисунке 3.1 «система₁», «система₂», ... «система_n» представляют собой внешнюю среду.

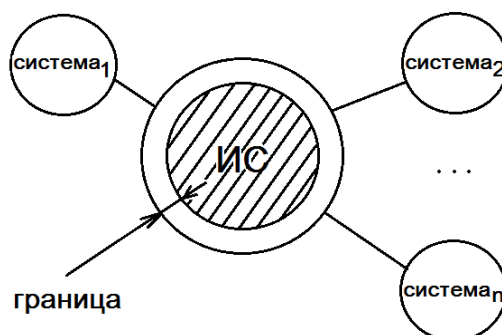


Рис. 3.1. Информационная система, ее граница и внешняя среда

Следует отметить, что одним из основных свойств процесса проектирования являются:

- дивергенция (расширение границ проектной ситуации с целью обеспечения более обширного пространства поиска решения),
- трансформация (стадия создания принципов и концепций (исследование структуры проблемы)),
- конвергенция (охватывает традиционное проектирование, т. е. программирование, отладка, проработка деталей).

3.3.2. Требования к системе. Требование – это условие, которому должна удовлетворять система, или свойство, которым она должна обладать, чтобы удовлетворить потребность пользователя в решении некоторой задачи и удовлетворить требования контракта, стандарта или спецификации. Стандартами, регламентирующими работу с требованиями, являются:

- ГОСТ 19.201-78. Единая система программной документации. Техническое задание. Требования к содержанию и оформлению [6],
- ГОСТ 34.602-89. Информационная технология. Техническое задание на создание автоматизированной системы [7].

Среди требований выделяют функциональные и нефункциональные.

Функциональные требования определяют действия, которые должна выполнять система, без учета ограничений, связанных с ее реализацией (описывают поведение системы в процессе обработки информации).

Нефункциональные требования описывают атрибуты ИС или атрибуты системного окружения:

- требования к применению (определяют качество пользовательского интерфейса, документации и т.п.),
- требования к производительности (накладывают ограничения на функциональные требования, задавая необходимую эффективность использования ресурсов, пропускную способность и время реакции),
- требования к реализации (предписывают использовать определенные стандарты, ЯП, операционную среду и т.п.),
- требования к надежности (определяют допустимую частоту и воздействие сбоев, возможности восстановления),
- требования к интерфейсу (определяют внешние сущности, с которыми может взаимодействовать система, и регламент этого взаимодействия).

Для выявления требований используются (в сочетании) пять методов:

- 1) собеседование (интервьюирование),
- 2) анкетирование,
- 3) моделирование и анализ бизнес-процессов,
- 4) сессии по выявлению требований (мозговой штурм),

5) создание и демонстрация пользователям работающих прототипов приложений.

Требования, выявленные в результате применения этих методов, оформляют в виде трех документов:

- 1) словарь предметной области (глоссарий),
- 2) концепция,
- 3) дополнительные спецификации.

Глоссарий определяет общую терминологию для всех моделей и описаний требований к системе. Он предназначен для описания терминологии ПрО и может быть использован как словарь данных системы.

Концепция определяет глобальные цели проекта и основные особенности разрабатываемой системы. Частью концепции является постановка задачи, определяющая требования к выполняемым системой функциям.

Дополнительные спецификации (технические требования) представляют собой описание нефункциональных требований: надежность, юзабилити, производительность, сопровождаемость и пр.

В объектно-ориентированном подходе функциональные требования моделируют и документируют с помощью вариантов использования (Use Case, UC) [8]–[19].

3.4 Пример: описание предметной области

3.4.1 Предметная область для проектирования и реализации информационной системы парного оценивания используется при проведении практических занятий №1–5 и выполнении ЛР №1–4.

Название ПрО: Разработка системы парного оценивания. Название ИС: PAsystem (Система парного оценивания работ обучающихся).

3.4.2 Глоссарий PAsystem содержит термины:

- LMS (Learning Management System) – система управления обучением;
- MOOC (Massive Open Online Course) – массовый открытый онлайн-курс, онлайн-курс, не имеющий ограничений для участия и имеющий открытый доступ через Интернет;

- MOOC платформа – программно-аппаратный комплекс организационных, информационных и технических решений, обеспечивающих взаимодействие создателя, инструктора и обучающегося MOOC через электронные каналы связи (напр., [20]–[23]), может быть реализована на базе LMS (напр., [24]);

- парное оценивание (Peer Assessment, PA или Peer Grading, PG) или взаимное оценивание (коллегиальное оценивание, оценивание сверстников) – метод оценивания, при котором работу оценивает не преподаватель

(вручную) или MOOC платформа (автоматически), а несколько сверстников, которые обучаются на этом же курсе (дисциплине). Это специфический вид аутсорсинга в оценивании работ, когда каждый обучающийся оценивает несколько работ других обучающихся, назначенных ему, и может применяться в заданиях с открытым ответом (доказательство теоремы, эссе, заявка на проект, модель ИС и т.д.);

- пользователь – инструктор или обучающийся;
- инструктор – создатель курса или преподаватель, использующий курс, например, в смешанном обучении;
- обучающийся – персона, проходящая обучение на курсе;
- грейдер (grader), оценщик – обучающийся, который оценивает задание, представленное другим обучающимся;
- рубрики (rubrics), категории, шаблоны оценивания – инструкции для оценщика, указывающие критерий и степень (чаще всего баллы) соответствия оцениваемой работы данному критерию;
- оценка – количество баллов, которое обучающийся получил за выполнение работы;
- количественное оценивание – метод оценивания работы, при котором за работу выставляются баллы;
- порядковое оценивание – метод оценивания, при котором нужно сравнить работы и выбрать, какая из них лучше, а какая хуже (ранжирование);
- консолидированная оценка – оценка, которая вычисляется на основе нескольких оценок;
- смешанное обучение (Blended Learning) – образовательный подход, который совмещает обучение с учителем – «лицом к лицу» (face-to-face, face to face, f4f) – и онлайн обучение, который предполагает элементы самостоятельного контроля учеником образовательного маршрута, времени, места и темпа обучения, а также интеграцию опыта обучения с учителем и онлайн;
- Moodle (Modular Object-Oriented Dynamic Learning Environment) – система управления обучением (LMS, Learning Management System), виртуальная обучающая среда (VLE, Virtual Learning Environment), система управления курсами (CMS, course management system) – распространяется бесплатно как программное обеспечение с открытым кодом (Open Source) под лицензией GNU Public License [2] и позволяет создавать сайты для электронного обучения (e-learning);
- юзабилити – удобство использования, одно из нефункциональных требований [25].

Пример использования основных понятий парного оценивания представлен в разделе 13.

3.4.3 Концепция PAsystem представлена ниже.

Система парного оценивания должна предоставлять возможность:

- инструктору: задавать сроки отправки и оценивания работы обучающихся,
- инструктору: настраивать параметры оценивания работы обучающихся,
- обучающемуся: загружать работу, которая является ответом на задание,
- инструктору: распределять грейдерам работы, загруженные обучающимися, для дальнейшего оценивания,
- грейдеру: оценивать назначенные ему работы,
- инструктору: вычислять итоговые оценки всех обучающихся и выгружать их в LMS Moodle,
- обучающемуся: получать оценки за выполнение задания и за парное оценивание.

ИС использует уже существующую БД Moodle, которая содержит всю информацию о курсах, пользователях (инструкторах и обучающихся) и результатах деятельности обучающихся. Эта БД поддерживается реляционной СУБД. Создаваемая система будет работать с существующей БД:

- в режиме доступа без обновлений при аутентификации пользователей и при импорте работ на оценивание,
- в режиме доступа с обновлениями при экспорте в LMS Moodle итоговых оценок за парное оценивание.

Для входа в систему парного оценивания пользователи должны использовать логин и пароль, с которыми они зарегистрированы в LMS Moodle.

Необходимость импорта-экспорта данных между PAsystem и LMS Moodle приводит к тому, что далее при проектировании и реализации будут учитываться соответствующие БД.

В системе парного оценивания существует период времени, когда обучающиеся должны представить свои работы на оценивание, и период времени для выполнения самого оценивания.

При выполнении парного оценивания обучающимся рекомендуется оставлять текстовые комментарии, объясняющие соответствующие оценки.

В конце выполнения парного оценивания обучающиеся должны иметь доступ к системе для просмотра результатов своей деятельности и к своим оценкам на курсе в LMS Moodle. Эта информация конфиденциальная, поэтому система должна обеспечить ее защиту от несанкционированного доступа.

3.4.4 Нефункциональными требованиями к ИС PAsystem являются:

- требования к применению:

- 1) ИС должна быть кроссплатформенной;
- 2) ИС должна быть доступна с мобильных устройств;
 - требования к производительности – ИС должна поддерживать несколько тысяч одновременно работающих с БД пользователей, т.к. на курсе могут обучаться несколько тысяч студентов;
 - требования к реализации:
 - 1) язык программирования ИС должен быть совместимым с языком реализации LMS Moodle;
 - 2) ИС должна быть кроссплатформенной, как и LMS Moodle;
 - требования к надежности:
 - 1) ИС должна функционировать на протяжении всего периода доступности к текущему курсу в LMS Moodle (24×7);
 - 2) ИС должна функционировать на протяжении всей сессии пользователя в LMS Moodle (по умолчанию 45 мин.);
 - 3) ИС должна выдерживать нагрузку не менее 1000 пользователей;
 - 4) ИС должна сохранять целостность входных данных;
 - требования к интерфейсу:
 - 1) ИС должна взаимодействовать с LMS Moodle;
 - 2) ИС должна выгружать оценки парного оценивания в LMS Moodle;
 - 3) ПИ ИС должен быть веб-ориентированным;
 - 4) ПИ ИС должен быть согласован с интерфейсом LMS Moodle;
 - требования к безопасности:
 - 1) ИС не должна позволять обучающемуся изменять никакую регистрацию в системе, кроме собственной;
 - 2) ИС не должна позволять инструктору модифицировать информацию о парном оценивании никаких курсов, кроме тех, на которых он имеет роль «Управляющий (Учитель)»;
 - 3) только инструктор может ставить итоговую оценку за парное оценивание;
 - 4) только администратор может изменять любую информацию об обучающихся.

3.4.5 Проектная организация RAsystem. ИС должна использовать реляционную БД с персональными данными пользователей, зарегистрированных на соответствующем курсе в Moodle; хранить шаблоны оценивания, работы обучающихся, представленные для оценивания, назначение грейдеров, результаты оценивания каждым грейдером и итоговые оценки обучающихся. ИС должна быть интегрирована с существующей LMS Moodle, имеющей в своем составе реляционную СУБД.

3.4.6 Граница системы показана на рисунке 3.2. Из этого рисунка видно, что PAsystem интегрируется с LMS Moodle, но по отношению к PAsystem система управления обучением является внешней средой.

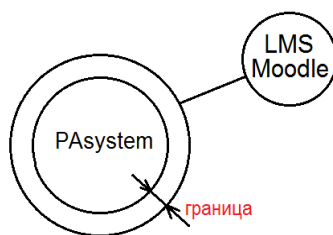


Рис. 3.2. Информационная система PAsystem, ее граница и внешняя среда

3.4.7 Действующими лицами (Акторами, Actors) ИС PAsystem являются:

- Инструктор (создает рубрики, назначает оценщиков и вычисляет итоговые оценки всех обучающихся),
- Обучающийся (отправляет работу для оценивания, читает свою итоговую оценку),
- Грейдер (выполняет оценивание назначенных ему работ в соответствии с инструкцией для оценщика).

3.4.8 Описание функциональных требований выполняют с помощью диаграммы прецедентов (Use Case Diagram, UCD) языка UML. Исходя их потребностей действующих лиц, выделены восемь вариантов использования:

- 1) задание сроков отправки и оценивания работ,
- 2) настройка критериев оценивания,
- 3) отправка работы,
- 4) назначение грейдеров,
- 5) оценивание работы,
- 6) вычисление итоговых оценок всех обучающихся,
- 7) экспорт оценок,
- 8) чтение личной итоговой оценки.

Диаграмма прецедентов приведена на рисунке 3.3. Граница PAsystem подчеркивает, что в рамках дисциплины «ОПРИС» проектированием и реализацией БД ИС мы не занимаемся. На диаграмме прецеденты размещены по вертикали в порядке их возникновения в ИС. Альтернативным вариантом является создание отдельных диаграмм прецедентов для каждого из Акторов, что иллюстрируют рисунки 3.4–3.5.

3.5 Домашнее задание № 1

3.5.1 Для предметной области сформулировать требования к ИС.

3.5.2 Срок выполнения: 1-я неделя. Срок защиты: 2-я неделя. Максимальная оценка составляет 10 баллов, зачетная – 4,1 балла. Представление отчета через элемент «Отчет ДЗ1» курса «ОПРИС» на moodle.movs.psu.ru.

3.5.3 Содержание отчета:

- а) словесное описание предметной области,
- б) название Про, название ИС и ее граница,
- в) глоссарий,
- г) концепция,
- д) требования (нефункциональные и функциональные),
- е) диаграмма вариантов использования.

3.6 Вопросы для самопроверки

1. Определение понятия «требование к системе».
2. Стандарты, регламентирующие работу с требованиями.
3. Группы требований к ИС.
4. Что определяют функциональные требования?
5. Что описывают нефункциональные требования?
6. Какие ошибки допущены на диаграмме прецедентов, представленной на рисунке 3.6?

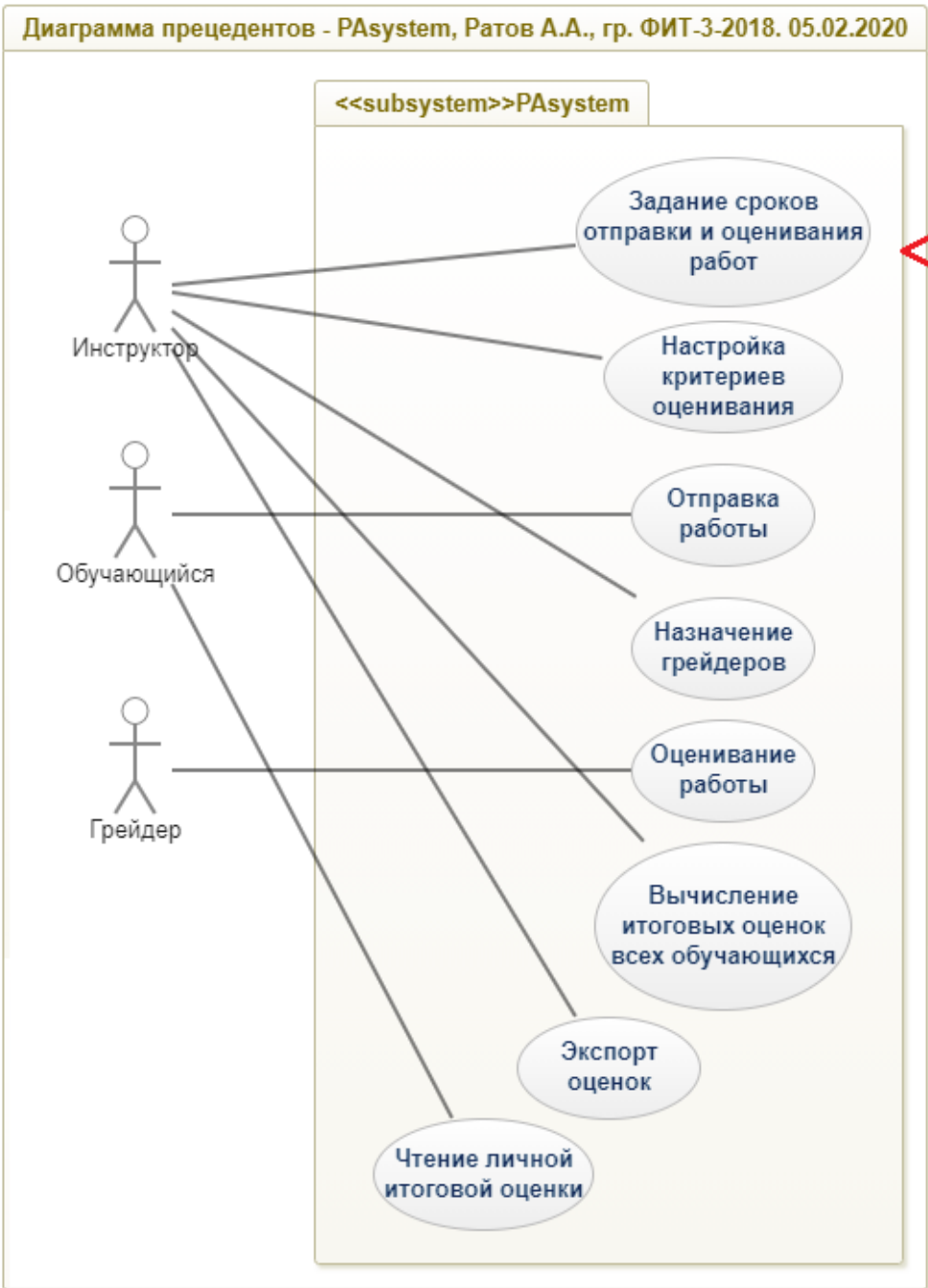


Рис. 3.3. Диаграмма прецедентов ИС парного оценивания



Рис. 3.4. Диаграмма прецедентов ИС для Обучающегося и Грейдера



Рис. 3.5. Диаграмма прецедентов ИС для Инструктора

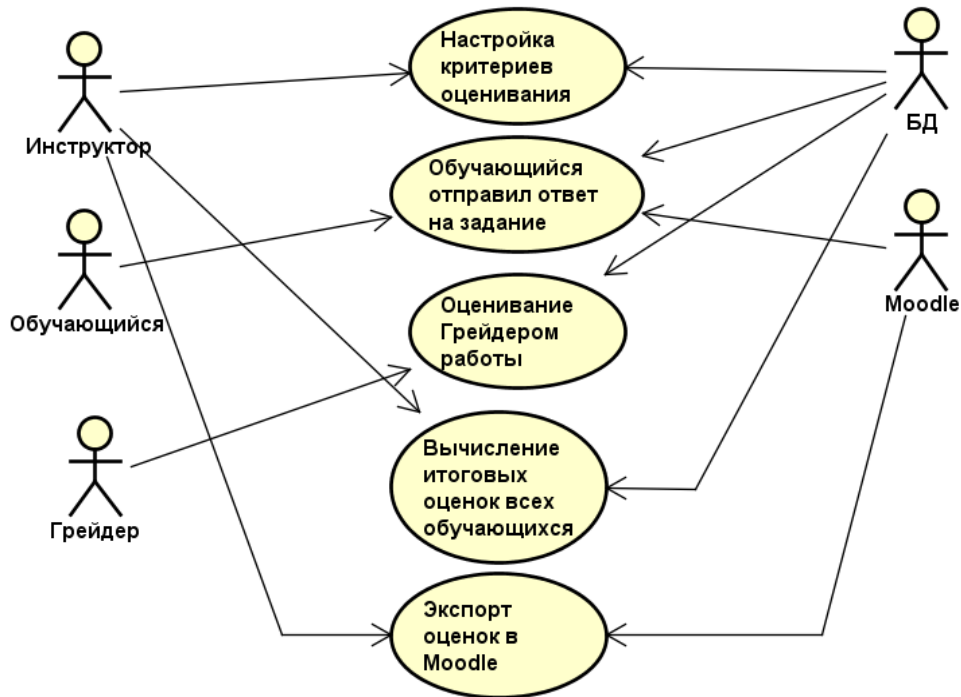


Рис. 3.6. Диаграмма прецедентов с ошибками

4 Лабораторная работа № 1.

Моделирование на этапе анализа ИС: функциональность ИС

4.1 Цель лабораторной работы № 1

Целью выполнения ЛР №1 является изучение основных элементов диаграммы прецедентов, создания диаграммы прецедентов, а также получение навыков работы с инструментальными средствами создания UCD. Выполняется студентами очной и заочной форм обучения.

4.2 Общие методические указания к выполнению ЛР № 1

Для выполнения ЛР № 1 необходимо предварительно изучить:

- стандарт ISO/IEC 19505-1:2012 [8] – [9],
- лексику, синтаксис, семантику и нотацию диаграмм прецедентов языка UML,
- инструмент для создания диаграмм прецедентов языка UML [5].

4.3 Сроки выполнения и представление отчета по ЛР № 1

Срок выполнения студентами очной формы обучения: 2-я и 3-я недели. Срок защиты: 4-я неделя. Представление отчета через элемент «Отчет ЛР1+Д32» курса «ОПРИС» на moodle.movs.psu.ru.

Представление отчета студентами заочной формы обучения через элемент «Отчет ЛР1» курса «ОПРИС-3» на moodle.movs.psu.ru.

4.4 Пример выполнения ЛР № 1

4.4.1 Уточним набор прецедентов ИС RAsystem. Разрабатываемая система является веб-приложением с требованиями о безопасности персональных данных пользователей. Поэтому необходимо кроме основных функциональных требований предусмотреть дополнительно прецеденты «Вход в систему» и «Выход из системы».

4.4.2 С прецедентами «Вход в систему» и «Выход из системы» должен быть ассоциирован еще один Актер системы – Пользователь, наличие которого не было рассмотрено в п. 3. После входа в систему выполняется распределение соответствующих ролей пользователям.

4.4.3 Между Акторами в RAsystem существуют отношения обобщения, которые иллюстрирует рисунок 4.1, а). Участие Актора Пользователь в прецедентах иллюстрирует рисунок 4.1, б). Чтобы убедиться в правильно-

сти отношения обобщения между Акторами, проверяем участие Акторов-потомков в прецедентах Актора-родителя:

- входить в систему и выходить из нее должны Акторы Инструктор и Обучающийся,

- Актор Грейдер перед оцениванием назначенных ему работ должен отправить свою работу как Обучающийся.

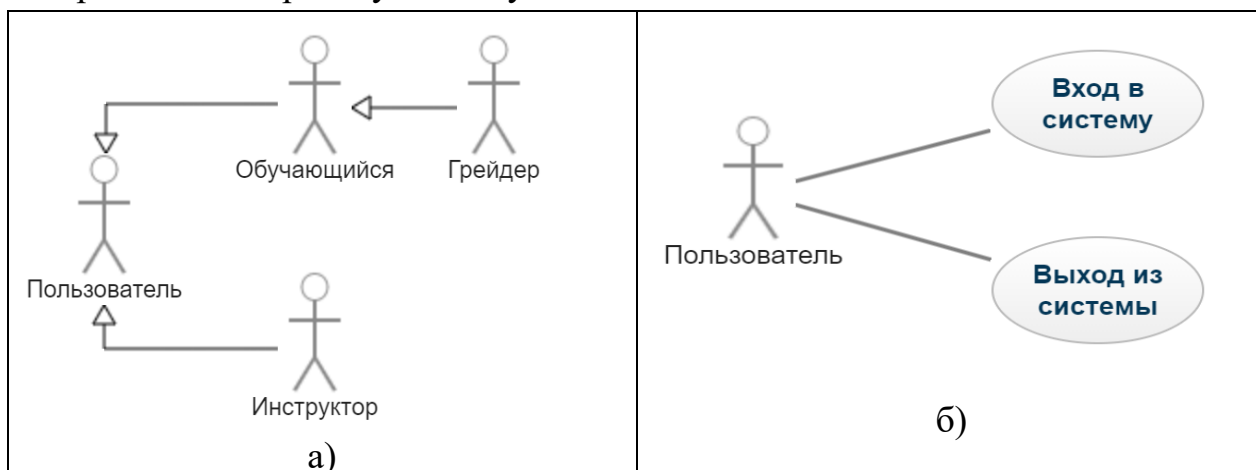


Рис. 4.1. Отношения между Акторами и прецеденты «Пользователя»

Таким образом, отношения обобщения между Акторами построено корректно.

4.4.4 Уточним набор прецедентов, связанный с Актором Инструктор. В прецеденте «Назначение грейдеров» возможны варианты случайного и ручного распределения. Первый из них можно применить при оценивании индивидуальных работ, второй – групповых. Прецедент «Назначение грейдеров» является родительским для прецедентов «Случайное распределение» и «Ручное распределение», т.к. они используют общую схему взаимодействия Актора Инструктор с ИС, и различаются только в конкретных шагах. Описанное отношение обобщения между прецедентами представлено на рисунке 4.2, а).

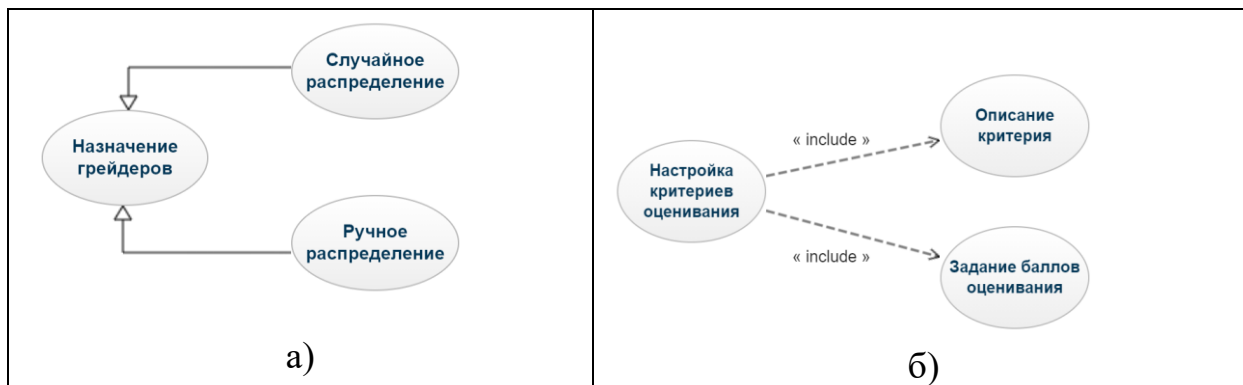


Рис. 4.2. Отношение обобщения и зависимость включения между прецедентами в ИС

4.4.5 Прецедент «Настройка критериев оценивания» предусматривает выполнение Актором Инструктор таких действий как подробное описание критерия, по которым производится оценивание, и указание баллов, которые показывают степень соответствия данной работы конкретному критерию.

Значит, в модели ИС должны быть прецеденты «Описание критерия» и «Задание баллов оценивания», описывающие деятельность Инструктора при работе с каждым из критериев оценивания. Между прецедентами, рассмотренными в этом пункте, существует зависимость включения, показанная в модели на рисунке 4.2, б), т.к. базовый вариант «Настройка критериев оценивания» зависит только от результатов выполнения включаемых в него вариантов использования «Описание критерия» и «Задание баллов оценивания», но не от структуры включаемых в него вариантов.

4.4.6 При оценивании работы, назначенной Актору Грейдеру, обучающемуся может потребоваться помощь, которую он получит при просмотре примера выполнения рецензии. Поэтому между вариантами использования «Оценивание работы» и «Просмотр примера» существует зависимость расширения с точкой расширения «Example», представленная на рисунке 4.3. В этом случае расширяемый прецедент «Оценивание работы» выполняет последовательность действий, которая определяет его поведение. Так как у него имеется точка расширения «Example» на экземпляре другого варианта использования «Просмотр примера», то проверяется логическое условие данного отношения (Grader selected HELP). Если это условие выполняется, исходная последовательность действий по оцениванию работы расширяется посредством включения действий просмотра примера рецензии.

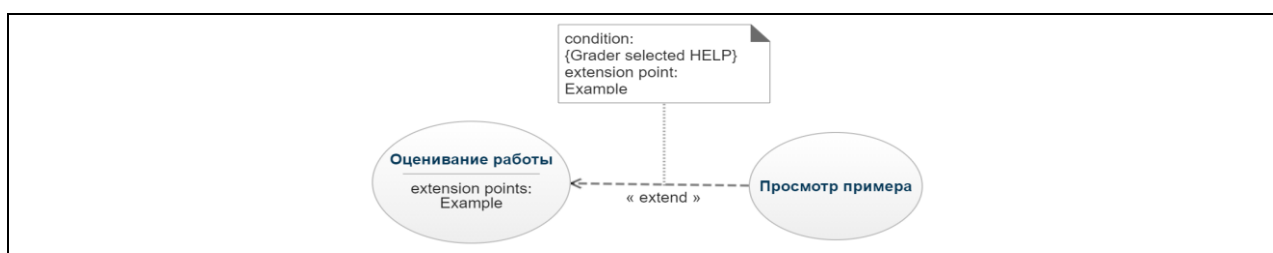


Рис. 4.3. Зависимость расширения между прецедентами в ИС

4.4.7 Названия стереотипов зависимостей включения и расширения могут быть записаны как по-русски, так по-английски (рисунки 4.2, б и 4.3) в зависимости от нотации, допустимой в соответствующем инструменте для создания Use Case Diagram.

4.4.8 Итоговая модель функциональности на этапе анализа ИС представлена на рисунке 4.4.

4.4.9 При построении UCD необходимо помнить об отличиях при работе с отдельной сущностью (понятием, объектом) ПрО и работой с множеством (списком, набором, коллекцией) таких сущностей.

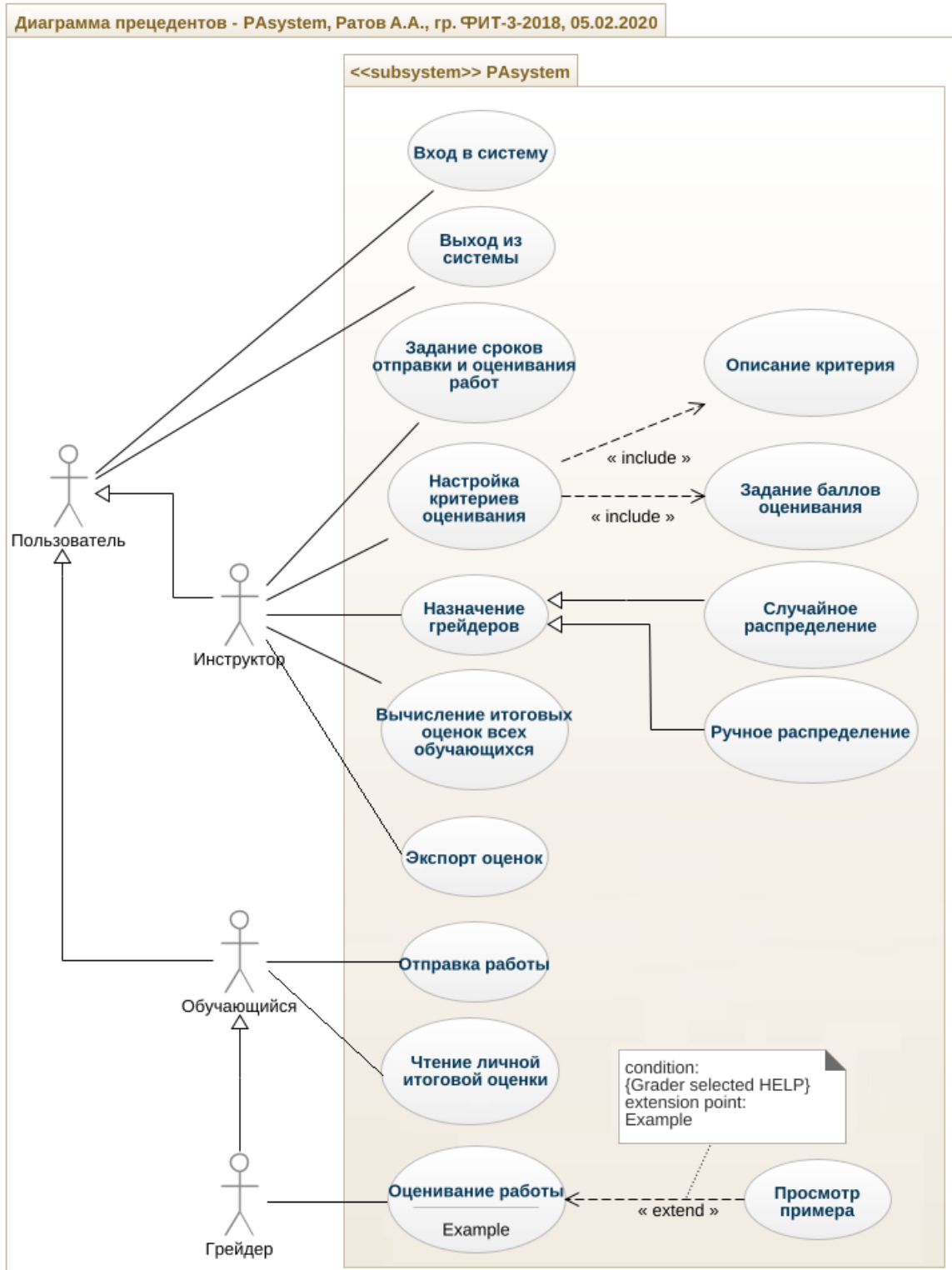


Рис. 4.4. Модель функциональности на этапе анализа ИС «PAsystem»

Работа с отдельной сущностью в общем случае предусматривает такие действия: создание, уничтожение, модификация, просмотр отдельного свойства и т.п.

Работа с множеством сущностей обычно включает: просмотр множества, выбор элемента из множества, добавление элемента в множество, удаление элемента из множества и т.п.

Пример 4.1. Аналогия с работой в главном меню окна приложения текстового процессора/редактора. Файл можно рассматривать как множество элементов (строк/символов текста). Первый элемент главного меню (Files) содержит команды для работы с файлом в целом («Создать», «Открыть», «Сохранить» и т.д.). Второй элемент главного меню (Edit) содержит команды для работы с элементами файла («Вставить», «Изменить», «Удалить» и т.д.).

4.4.10 Список типичных ошибок, которые допускают студенты при создании UCD, приведен в приложении В.

4.5 Требования, указания и задания к ЛР № 1

4.5.1 Построить диаграмму прецедентов (не менее 4 прецедентов), в которой отразить требования к системе, ее основную функциональность.

4.5.2 Определить отношения между акторами (обобщение) и проиллюстрировать их на диаграмме.

4.5.3 Определить отношения между прецедентами и проиллюстрировать их на диаграмме (ассоциация, обобщение, зависимости включения и расширения).

4.6 Содержание отчета по ЛР № 1

Содержание отчета по ЛР № 1:

- а) определение отношений между акторами,
- б) определение отношений между прецедентами,
- в) диаграмма(-ы) вариантов использования.

4.7 Оценивание ЛР № 1

Оценивание выполнения ЛР № 1 приведено в таблицах 4.1–4.2. Максимальная оценка составляет у студентов очной формы обучения 10 баллов, зачетная – 4,1 балла. Максимальная оценка составляет у студентов заочной формы обучения 20 баллов, зачетная – 9 баллов.

Таблица 4.1

Оценивание выполнения ЛР № 1 (очная форма обучения)

Баллы	Выполнение требований к ЛР № 1
1	2
Общий 10, в том числе 9	Выполнены: все требования п. 4.5
Общий 9,9–8. в том числе 8,9–7	Выполнены: до 80% требований п. 4.5 (отсутствует зависимость расширения между прецедентами)
Общий 7,9–6, в том числе 6,9–5	Выполнены: до 60% требований п. 4.5 (отсутствуют зависимости между прецедентами)
Общий 5,9–4,1, в том числе 5,9–3,1	Выполнены: до 40% требований п. 4.5 (присутствуют только ассоциации между прецедентами, нет отношений между акторами)

Таблица 4.2

Оценивание выполнения ЛР № 1 (заочная форма обучения)

Баллы	Выполнение требований к ЛР № 1
1	2
Общий 20, в том числе 19	Выполнены: все требования п. 4.5
Общий 16–18,9 в том числе 15–18,9	Выполнены: до 80% требований п. 4.5 (отсутствует зависимость расширения между прецедентами)
Общий 12–15,9, в том числе 11–15,9	Выполнены: до 60% требований п. 4.5 (отсутствуют зависимости между прецедентами)
Общий 9–11,9, в том числе 8–11,9	Выполнены: до 40% требований п. 4.5 (присутствуют только ассоциации между прецедентами, нет отношений между акторами)

Примечание: обязательным является выполнение п. 2.1 (0,1 балла), 2.2 (0,1 балла), 2.3 (0,1 балла), 2.4 (0,2 балла) и 4.5.1 (0,5 балла); общая сумма – 1 балл.

5 Практическое занятие № 2.

Объектно-ориентированный подход к моделированию: моделирование функциональных требований

5.1 Цель практического занятия № 2

Практическое занятие направлено на изучение классификации вариантов использования, описание модели функциональных требований при объектно-ориентированном подходе к моделированию и формирование соответствующей документации. Выполняется студентами очной формы обучения.

5.2 Методические указания к проведению практического занятия № 2

Для проведения практического занятия № 2 необходимо предварительно изучить тему 6.2 [17].

5.3 Пример описания модели функциональных требований

5.3.1 Исходной является модель функциональных требований ИС RAsystem, построенная в п. 4.4.

5.3.2 Списки Акторов и прецедентов модели представлены в таблице 5.1.

Таблица 5.1

Элементы модели функциональных требований

Список Акторов	Список прецедентов	
Пользователь Инструктор Обучающийся Грейдер	Вход в систему Выход из системы Задание сроков отправки и оценивания работ Настройка критериев оценивания Описание критерия Задание баллов оценивания Назначение грейдеров	Случайное распределение Ручное распределение Вычисление итоговых оценок всех обучающихся Экспорт оценок Отправка работы Оценивание работы Просмотр примера Чтение личной итоговой оценки

5.3.3 Высокоуровневое описание прецедентов приведено на рисунке 5.1.

<p><i>Название:</i> Вход в систему.</p> <p><i>Акторы:</i> Пользователь.</p> <p><i>Описание:</i> Пользователь входит в систему с логином и паролем, с которыми он зарегистрирован в LMS Moodle. В случае успешного входа система начинает сеанс работы, назначает Пользователю соответствующие права.</p>
--

Рис. 5.1. Высокоуровневое описание прецедентов

Название: Выход из системы.

Акторы: Пользователь.

Описание: Пользователь явно выходит из системы. Система завершает сеанс работы Пользователя.

Название: Задание сроков отправки и оценивания работ.

Акторы: Инструктор.

Описание: Инструктор задает крайний срок отправки работ обучающимися. Инструктор задает крайний срок завершения оценивания работ грейдерами. Система контролирует, чтобы срок завершения оценивания был позже, чем срок отправки работ.

Название: Настройка критериев оценивания.

Акторы: Инструктор.

Описание: Инструктор создает элемент парного оценивания работ. Он настраивает критерии оценивания работ.

Название: Описание критерия.

Акторы: Инструктор.

Описание: Инструктор заполняет для каждого критерия оценивания работы его описание.

Название: Задание баллов оценивания.

Акторы: Инструктор.

Описание: Инструктор задает баллы оценивания для каждого критерия.

Название: Назначение грейдеров.

Акторы: Инструктор.

Описание: Инструктор осуществляет распределение грейдеров.

Название: Вычисление итоговых оценок всех обучающихся.

Акторы: Инструктор.

Описание: Инструктор осуществляет вычисление итоговых оценок за оценивание работы. Система вычисляет оценку каждого обучающегося на основании оценок, выставленных ему грейдерами, и оценок, которые он выставил как греjder. Система фиксирует итоговые оценки всех обучающихся.

Название: Экспорт оценок.

Акторы: Инструктор.

Описание: Инструктор дает команду системе выгрузить консолидированные оценки за работу в платформу LMS. Система экспортирует данные оценки в Moodle и заносит в журнал оценок курса.

Продолжение рисунка 5.1

Название: Отправка работы.

Акторы: Обучающийся.

Описание: Обучающийся выполняет работу и отправляет ее на проверку в элемент парного оценивания платформы LMS. Система загружает работу, критерии оценивания и данные о студенте из Moodle.

Название: Оценивание работы.

Акторы: Грейдер.

Описание: Грейдер выбирает работу другого обучающегося для оценивания. Работа и критерии оценивания направляются этому Грейдеру. Грейдер изучает критерии оценивания работы. Грейдер читает работу обучающегося. Грейдер при помощи критериев оценки выставляет оценку данной работе по каждому из предложенных критериев и завершает оценивание. Система вычисляет оценку этого Грейдера за работу обучающегося.

Название: Просмотр примера.

Акторы: Грейдер.

Описание: Грейдер не знает, как применить предложенные критерии при оценивании назначенной ему работы. Он запрашивает помощь в виде примера оценивания. Система открывает Грейдеру пример.

Название: Чтение личной итоговой оценки.

Акторы: Грейдер.

Описание: Грейдер читает свою итоговую оценку (за выполнение работы и парного оценивания) в журнале оценок LMS Moodle.

Продолжение рисунка 5.1

5.3.4 Классифицируем прецеденты по категориям:

– главные:

- 1) вход в систему;
- 2) выход из системы;
- 3) настройка критериев оценивания;
- 4) назначение грейдеров;
- 5) вычисление итоговых оценок всех обучающихся;
- 6) экспорт оценок;
- 7) отправка работы;
- 8) оценивание работы;

– второстепенные:

- 1) задание сроков отправки и оценивания работ;
- 2) чтение личной итоговой оценки;

– вспомогательный – просмотр примера.

Прецедент «Задание сроков отправки и оценивания работ» классифицирован как второстепенный в связи с тем, что PAsystem может быть использована обучающимися в демоверсии без ограничений деятельности во времени, чтобы сформировать у них навыки работы с системой.

5.3.5 Выделяем наиболее важные и рискованные прецеденты и записываем их в развернутой идеальной форме, как представлено на рисунке 5.2.

<p><i>Название:</i> Настройка критериев оценивания. <i>Акторы:</i> Инструктор. <i>Описание:</i> Инструктор создает элемент парного оценивания работ. Он настраивает критерии оценивания работ. <i>Триггер:</i> Инструктор создает элемент парного оценивания работ. <i>Основной поток:</i></p>	
Действия акторов	Отклик системы
1. Инструктор инициирует создание нового элемента парного оценивания работ	2. Система создает элемент парного оценивания работ, регистрирует его и информирует о его создании Moodle
3. Инструктор выбирает вариант оценивания с помощью рубрик	4. Система предоставляет в элементе парного оценивания работ элемент для создания рубрик
5. Инструктор создает рубрику и заполняет все критерии оценивания. При настройке атрибута критерия «Описание» выполняется подпоток S1. При настройке атрибута критерия «Балл» выполняется подпоток S2	6. Система прикрепляет созданную рубрику к элементу парного оценивания работ
<p><i>Подпотoki:</i> S1: <i>Описание критерия.</i> Инструктор заполняет для каждого критерия оценивания работы его описание. Система записывает описание критерия. Прецедент продолжается. S2: <i>Задание баллов оценивания.</i> Инструктор задает баллы оценивания для каждого критерия. Система записывает баллы критериев. Прецедент продолжается</p>	
<p><i>Название:</i> Назначение грейдеров. <i>Акторы:</i> Инструктор. <i>Описание:</i> Инструктор осуществляет распределение грейдеров. <i>Триггер:</i> Инструктор инициирует распределение грейдеров. <i>Основной поток:</i></p>	
Действия акторов	Отклик системы
1. Инструктор анализирует вид оцениваемой работы, выполненной обучающимися. При выполнении индивидуальных работ выполняется подпоток S3. При выполнении командных работ выполняется подпоток S4. Инструктор подтверждает распределение грейдеров	2. Система записывает распределение грейдеров этой работы

Рис. 5.2. Описание прецедентов в развернутой форме

Подпотoki:

S3: Случайное распределение. Инструктор указывает количество грейдеров, назначаемых каждому обучающемуся. Система по определенному алгоритму назначает уникальных грейдеров для обучающихся. Прецедент продолжается.

S4: Ручное распределение. Система отображает Инструктору список обучающихся и список потенциальных грейдеров. Инструктор выбирает обучающегося и назначает ему уникальных грейдеров. Прецедент продолжается

Название: Оценивание работы.

Акторы: Грейдер.

Описание: Грейдер выбирает работу обучающегося, назначенного ему для оценивания. Работа и критерии оценивания направляются этому Грейдеру. Грейдер изучает критерии оценивания работы. Грейдер читает работу обучающегося. Грейдер при помощи критериев оценки выставляет оценку данной работе по каждому из предложенных критериев и завершает оценивание. Система вычисляет оценку этого Грейдера за работу обучающегося и записывает ее.

Триггер: Грейдер выбирает работу обучающегося для оценивания.

Основной поток:

Действия акторов	Отклик системы
1. Грейдер выбирает работу обучающегося.	2. Система направляет Грейдеру работу и критерии оценивания.
3. Грейдер изучает критерии оценивания работы. Грейдер читает работу обучающегося. Грейдер при помощи критериев оценки выставляет оценку данной работе по каждому из предложенных критериев (E1) и завершает оценивание (E2)	4. Система вычисляет оценку Грейдера за работу обучающегося и записывает ее

Альтернативные потоки:

E1: Просмотр примера. При невозможности применить предложенные критерии при оценивании назначенной ему работы Грейдер запрашивает помощь в виде примера оценивания. Система направляет Грейдеру пример. Прецедент продолжается.

E2. Незавершенное оценивание. Грейдер не оценивает работу по каким-либо причинам. Прецедент завершается

Название: Вычисление итоговых оценок всех обучающихся.

Акторы: Инструктор.

Описание: Инструктор осуществляет вычисление итоговых оценок за оценивание работы. Система вычисляет оценку каждого обучающегося на основании оценок, выставленных ему грейдерами, и оценок, которые он выставил как грейдера. Система фиксирует итоговые оценки всех обучающихся.

Триггер: Инструктор инициирует вычисление итоговых оценок за оценивание работы

Продолжение рисунка 5.2

<i>Основной поток:</i>	
Действия акторов	Отклик системы
1. Инструктор анализирует состояние выполнения парного оценивания (ЕЗ). При наступлении даты окончания оценивания Инструктор инициирует вычисление итоговых оценок за парное оценивание	2. Система вычисляет оценку каждого обучающегося на основании оценок, выставленных ему грейдерами, и оценок, которые он выставил как греjder, и записывает результаты
3. Инструктор подтверждает завершение парного оценивания работы	4. Система фиксирует итоговые оценки всех обучающихся
<i>Альтернативные потоки:</i> ЕЗ: <i>Частичное оценивание.</i> Не все Грейдеры выполнили оценивание. Инструктор продлевает срок завершения оценивания. Прецедент завершается	

Продолжение рисунка 5.2

5.4 Домашнее задание № 2

5.4.1 Для предметной области выполнить описание модели функциональных требований к ИС.

5.4.2 Срок выполнения: 3-я неделя. Срок защиты: 4-я неделя. Максимальная оценка составляет 10 баллов, зачетная – 4,1 балла. Представление отчета через элемент «Отчет ЛР1+ДЗ2» курса «ОПРИС» на moodle.movs.psu.ru.

5.4.3 Содержание отчета:

- а) список Акторов,
- б) список прецедентов,
- в) описание всех прецедентов в высокоуровневой форме,
- г) классификация прецедентов по категориям,
- д) описание наиболее важных и рискованных прецедентов в развернутой идеальной форме.

5.5 Вопросы для самопроверки

1. Структура шаблона для описания высокоуровневого прецедента.
2. Определение понятия «Развернутый прецедент».
3. Структура шаблона для описания развернутого прецедента.
4. Деление прецедентов по степени важности.
5. Классификация прецедентов по учету технологий ввода/вывода.
6. Между какими прецедентами диаграммы, представленной на рисунке 5.3, должно быть установлено отношение обобщения?

7. Между какими прецедентами диаграммы, представленной на рисунке 5.3, должно быть установлено отношение включения?

8. Между какими прецедентами диаграммы, представленной на рисунке 5.3, должно быть установлено отношение расширения? Укажите точку расширения.



Рис. 5.3. Диаграмма прецедентов с отсутствующими отношениями и зависимостями

9. Как в описании развернутого прецедента отображается отношение обобщения между прецедентами?

10. Как в описании развернутого прецедента отображаются зависимости между прецедентами?

11. Как нефункциональные требования отображаются на диаграммах прецедентов и в их описании?

6 Практическое занятие № 3.

Объектно-ориентированный подход к моделированию: моделирование бизнес-процессов

6.1 Цель практического занятия № 3

Практическое занятие направлено на изучение основ моделирования бизнес-процессов при объектно-ориентированном подходе к моделированию и формированию соответствующей документации. Выполняется студентами очной формы обучения.

6.2 Методические указания к проведению практического занятия № 3

6.2.1 Диаграмма активностей [16] (*англ.* Activity Diagram, AD) или диаграмма деятельности [10]–[14]) – это диаграмма, на которой изображается граф деятельности. На графе деятельности изображается поток работ или работа процедуры.

6.2.2 Для проведения практического занятия № 3 необходимо предварительно изучить тему 6.3 [17].

6.3 Пример описания модели бизнес-процессов

6.3.1 Исходной является модель функциональных требований ИС RAsystem, построенная в п. 4.4.

6.3.2 При необходимости для описания бизнес-процессов могут быть использованы специальные обозначения объектов:

- пограничный объект, например, меню формы приложения (представлен на рисунке 6.1, а),
- объект-сущность, например, хранилище данных (представлен на рисунке 6.1, б),
- управляющий объект (представлен на рисунке 6.1, в).

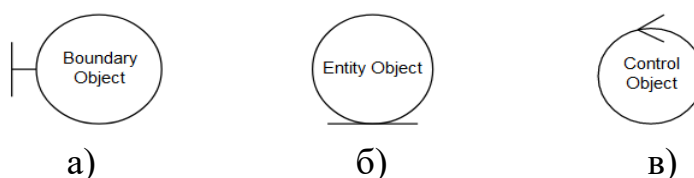


Рис. 6.1. Дополнительные объекты модели бизнес-процессов

6.3.2 Диаграмма активностей модели RAsystem представлена на рисунке 6.2.

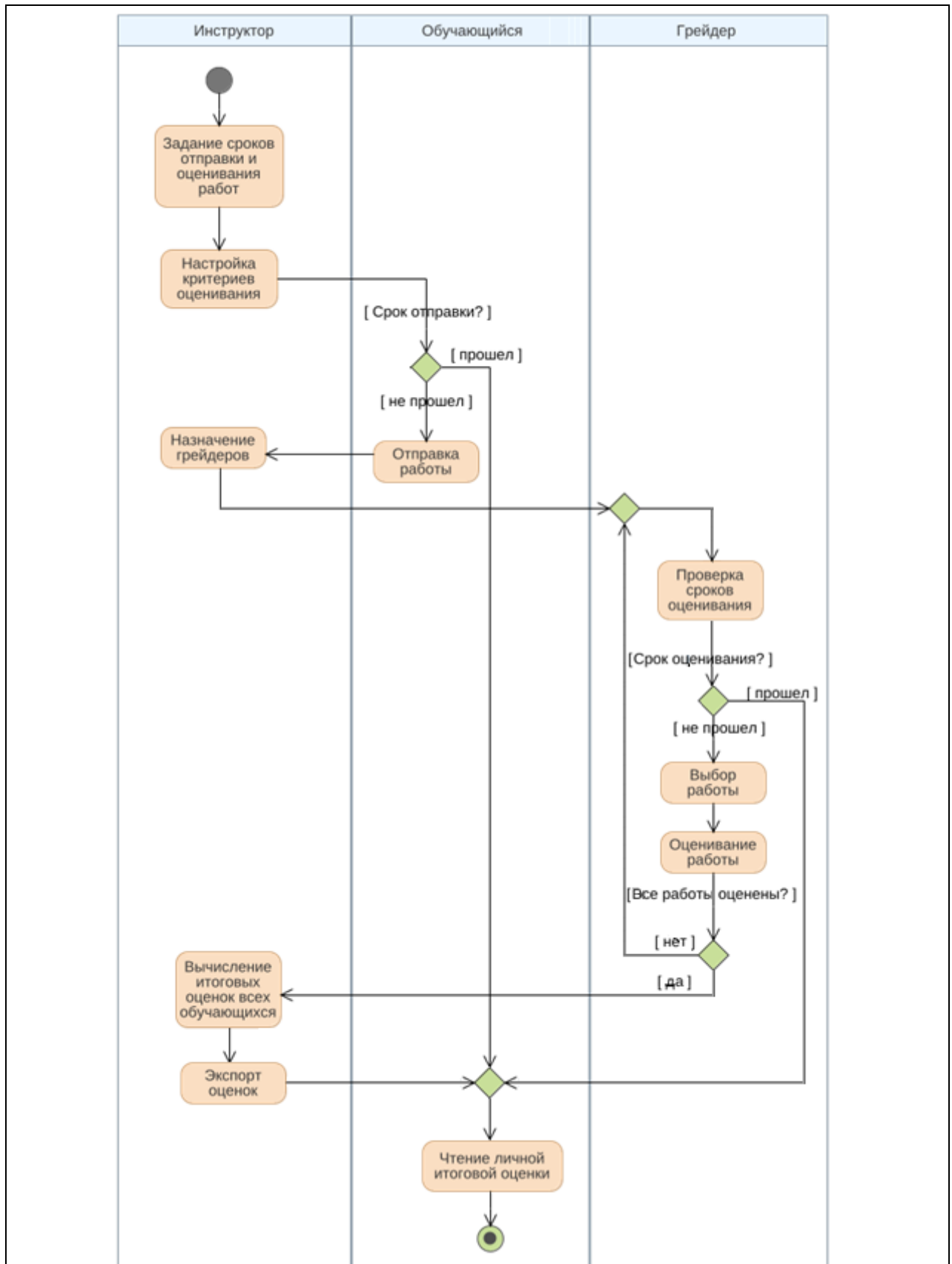


Рис. 6.2. Модель бизнес-процессов

6.3.3 При использовании инструмента [5] для создания диаграммы активностей нельзя использовать проект типа «BPMN Busines Process». Необходимо использовать проект типа «UML Software Architecture». Средства создания диаграммы активностей станут доступны после выполнения команды File => New Diagram => Activity Diagram.

6.3.4 Самой серьезной ошибкой при создании АД является непонимание сущности «бизнес-процесс» и представление его алгоритмом работы ИС. Список типичных ошибок, которые допускают студенты при создании АД, приведен в приложении Г.

6.4 Домашнее задание № 3

6.4.1 Для предметной области выполнить описание бизнес-процессов предметной области, которые автоматизирует информационная система, в нотации диаграмм активностей UML при объектно-ориентированном подходе к моделированию и формирование соответствующей документации.

6.4.2 Срок выполнения: 4-я неделя. Срок защиты: 5-я неделя. Максимальная оценка составляет 10 баллов, зачетная – 4,1 балла. Представление отчета через элемент «Отчет ДЗ3» курса «ОПРИС» на moodle.movs.psu.ru.

6.4.3 Содержание отчета:

- а) список Акторов, участвующих в бизнес-процессе,
- б) модель бизнес-процессов в виде диаграмм(-ы) активностей.

6.5 Вопросы для самопроверки

1. Роль моделирования бизнес-процессов при создании модели прецедентов.
2. Отличие активности от прецедента.
3. Отличие состояния с несколькими входами от синхронизатора.
4. Как в диаграмме активностей показать повторение бизнес-процесса?
5. Какие ошибки допущены в диаграмме активностей, представленной на рисунке 6.3?

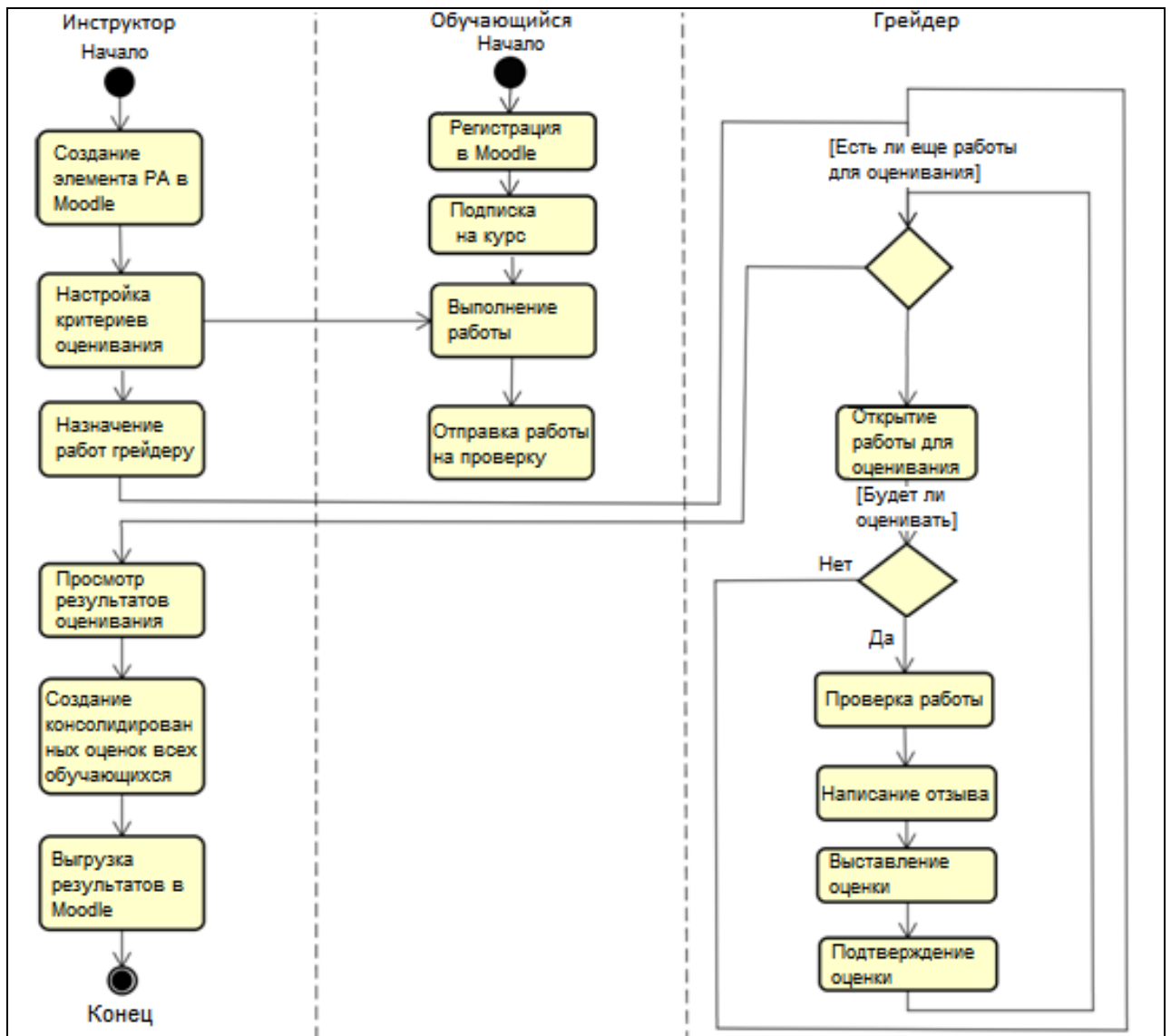


Рис. 6.3. Диаграмма активностей с ошибками

7 Практическое занятие № 4.

Объектно-ориентированный подход к моделированию: концептуальная модель

7.1 Цель практического занятия № 4

Практическое занятие направлено на изучение основ концептуального моделирования при объектно-ориентированном подходе к моделированию и формированию соответствующей документации. Выполняется студентами очной формы обучения.

7.2 Методические указания к проведению практического занятия № 4

Для проведения практического занятия № 4 необходимо предварительно изучить тему 6.4 [17].

7.3 Пример создания концептуальной модели

7.3.1 Исходной является модель функциональных требований ИС RAsystem, построенная в п. 4.4.

7.3.2 Выделение понятий. Исходными данными являются документы этапа анализа моделирования функциональности ИС: описание прецедентов и Глоссарий. В функциональных требованиях система парного оценивания была ограничена рамками ее применения только совместно с LMS Moodle. Поэтому в дальнейшем не будет рассматриваться вариант ее применения совместно с МООС-платформой, хотя такой вариант может быть задействован при дивергенции.

Применяем последовательно два подхода:

- 1) непосредственное выделение понятий,
- 2) поиск по списку стандартных категорий понятий.

При непосредственном выделении понятий выделяем претендентов на роль понятий – имена существительные или имена существительные с дополнениями:

а) Курс – учебное рабочее пространство Обучающегося и Инструктора по конкретной дисциплине при смешанном обучении, размещенное в LMS Moodle,

б) Moodle – платформа, на которой размещаются учебные или массовые открытые онлайн-курсы,

в) Задание – задача, которую нужно решить обучающимся, размещенная в курсе,

- г) Пользователь – персона, выполняющая учебную деятельность в Moodle и PAsystem,
- д) Обучающийся – персона, проходящая обучение на курсе,
- е) Грейдер – обучающийся, который оценивает задания, представленные другими обучающимися,
- ж) Инструктор – создатель курса или преподаватель, использующий курс,
- и) Работа – учебное задание, которое выполняет обучающийся и отправляет на оценивание грейдерами,
- к) Назначение – работа обучающегося, которая назначена грейдеру для оценивания,
- л) БД – хранилище критериев оценивания, работ обучающихся, списков грейдеров для каждой работы, результатов оценивания грейдерами каждой работы и итоговых оценок парного оценивания обучающихся,
- м) Оценка – количество баллов за назначение,
- н) Критерии оценивания – указания, которые нужны для пояснения требований, необходимых для оценивания работы,
- п) Список рубрик – множество критериев оценивания,
- р) Рубрика – описание критерия с указанием уровней соответствия этому критерию и количества баллов за каждый из этих уровней,
- с) Список уровней – множество уровней соответствия для конкретного критерия,
- т) Уровень – описание качества соответствия критерию и количество баллов, которые даются за него,
- у) Сроки крайние – даты отправки работы обучающимся и ее оценивания грейдерами,
- ф) Назначение грейдеров – распределение оценщикам работ других обучающихся,
- х) Отправка работы – загрузка файла выполненной работы в ИС,
- ц) Оценивание работы – определение уровня соответствия работы указанным критериям, которое выполняется грейдером; результат оценивания представляет собой консолидированную оценку по всем критериям,
- ш) Вычисление итоговых оценок – формирование итоговой оценки для каждого грейдера, объединяющей баллы, полученные грейдером от других обучающихся за его работу, и баллы, выставленные им за рецензию назначенных ему работ,
- щ) Итоговая оценка – баллы, полученные грейдером от других обучающихся за его работу, и баллы, выставленные им за рецензию назначенных ему работ,

э) Экспорт оценок – отправка итоговых оценок обучающихся за парное оценивание из БД PAsystem в Moodle,

ю) Журнал оценок – элемент курса Moodle, содержащий информацию о всех видах оцениваемой деятельности обучающихся на этом курсе (тесты, задания, парное оценивание и др.) и общую оценку за курс.

Проверяем список претендентов на роль понятий с помощью поиска по списку стандартных категорий понятий. Результат этого шага представлен в таблице 7.1. Понятия, которые выделены в категории «Транзакции», могут претендовать на роли ассоциаций. Поэтому окончательный список понятий будет сформирован после идентификации атрибутов понятий и выделения ассоциаций.

Таблица 7.1

Список понятий после поиска по списку стандартных категорий

Категория понятий	Имена претендентов на роль понятий
Спецификации, элементы дизайна или описания объекта	Сроки
Транзакции	Вычисление итоговых оценок, Назначение грейдеров, Отправка работы, Оценивание работы, Экспорт оценок
Элемент транзакции	Итоговая Оценка, Назначение, Оценка, Работа
Роли людей	Грейдер, Инструктор, Обучающийся, Пользователь
Контейнеры других объектов	Moodle, БД, Грейдер, Журнал оценок, Задание, Критерии оценивания, Курс, Назначение, Список рубрик, Список уровней
Содержимое контейнеров	Журнал оценок, Задание, Итоговая Оценка, Критерии оценивания, Курс, Оценка, Пользователь, Рубрика, Список назначений, Список рубрик, Список уровней, Сроки крайние
Другие системы, внешние по отношению к данной ИС	Moodle, БД
Записи финансовой, трудовой, юридической и др. деятельности	Журнал оценок

7.3.3 Выделение атрибутов понятий. Идентифицируем атрибуты понятий, выделенных в п. 7.3.2. Результат этого шага представлен в таблице 7.2.

Таблица 7.2

Список понятий и их атрибуты

Имена понятия	Имена атрибутов
Moodle	названиеСайта, количествоКурсов, множествоКурсов
Грейдер	спНазначений
Инструктор	-
Итоговая Оценка	idЗадания, idГрейдера, Балл
Журнал Оценок	общийБалл, спИтоговыхОценок, количествоИтоговыхОценок
Задание	idЗадания, название, сроки, описание, максБаллЗадания, критОцен, статус, спIdРабот, количествоIdРабот
Критерии Оценивания	названиеК, описаниеК, спРубрик
Курс	название, idКурса, журналОценок, спПользователей, количествоПользователей, множествоЗаданий, количествоЗаданий

Имена понятия	Имена атрибутов
Назначение	колIdГрейдеров, мнОценок
Обучающийся	группа
Оценка	idГрейдера, idНазначения, балл, максБалл
Пользователь	id, фИО, статус, idКурса
Работа	idРаботы, названиеРаботы, имяФайла, idОбучающегося
Рубрика	описаниеР, спУровней
СписокIdназначений	спIdНазначений, колIdНазначений
Список рубрик	Количество Рубрик, Множество Рубрик
Список уровней	множествоУровней, количествоУровней
Сроки крайние	датаОтправки, датаЗавершенияОценивания
Уровень	описаниеУ, баллУ

Для понятия «БД» выявление атрибутов не выполнялось в соответствии с п. 3.4.8. У претендентов на роль понятий ф)-ш), э) п. 7.3.2 атрибуты не были выявлены. Их имена являются отглагольными существительными, поэтому они будут претендовать на роли ассоциаций.

7.3.4 Выделение ассоциаций. Исходными данными являются документы этапа анализа моделирования функциональности ИС: описание прецедентов и Глоссарий. Применяем последовательно два подхода:

- 1) выделение ассоциаций по определению,
- 2) поиск по списку стандартных категорий ассоциаций.

При выделении ассоциаций по определению выделяем претендентов на роль ассоциаций – глаголы или глагольные формы. Ниже шрифтом Arial выделены имена ассоциаций, а в скобках записаны пары понятий, которые находятся в отношении ассоциации:

- а) Moodle содержит Курсы (Moodle – Курс),
- б) Курс содержит Задания (Курс – Задание),
- в) Курс содержит ЖурналОценок (Курс – ЖурналОценок),
- г) ЖурналОценок содержит ИтоговыеОценки (ЖурналОценок – ИтоговаяОценка),
- д) Задание содержит КритерииОценивания (Задание – КритерииОценивания),
- е) Задание содержит СрокиКрайние (Задание – СрокиКрайние),
- ж) КритерииОценивания содержат СписокРубрик (КритерииОценивания – СписокРубрик),
- и) СписокРубрик содержит Рубрику (СписокРубрик – Рубрика),
- к) Рубрика содержит СписокУровней (Рубрика – СписокУровней),
- л) СписокУровней содержит Уровень (СписокУровней – Уровень),
- м) Назначение содержит Оценки (Назначение – Оценка),
- н) Пользователь записан на Курс (Пользователь – Курс),

- п) Инструктор создает Курс (Инструктор – Курс),
- р) Инструктор добавляет Задание (Инструктор – Задание),
- с) Инструктор задает КритерииОценивания (Инструктор – Критери-иОценивания),
- т) Инструктор распределяет Грейдеров (Инструктор – Грейдер),
- у) Обучающийся отправляет Работу (Обучающийся – Работа),
- ф) Грейдер проверяет Назначение (Грейдер – Назначение),
- х) Оценка ставится за Назначение (Оценка – Назначение),
- ц) ИтоговаяОценка вычисляется по Оценкам (ИтоговаяОценка – Оценка),
- ш) ИтоговаяОценка экспортируется в ЖурналОценок (ИтоговаяОцен-ка – ЖурналОценок),
- щ) Работа является ответом на Задание (Работа – Задание),
- э) Грейдер имеет СписокIdНазначений (Грейдер – СписокIdНазначений),
- ю) Грейдер читает ИтоговуюОценку (Грейдер – ИтоговаяОценка).

Проверяем список претендентов на роль ассоциаций с помощью поиска по списку стандартных категорий ассоциаций. Результат этого шага пред-ставлен в таблице 7.3.

Таблица 7.3

Результат проверки ассоциаций по списку стандартных ассоциаций

Стандартная категория ассоциаций	Список ассоциаций (ассоциация из п. 7.3.4)
А известен / зарегистрирован / записан / включен в отчет / содержится в В	а)-н), ш), э)
А использует или управляет В	п)-т), ц), щ)
А взаимодействует с В	у)-х), ш), ю)

Далее в тексте этого раздела имена понятий выделены шрифтом Courier New.

7.3.5 Выделение ролей ассоциаций. Понятие Назначение выступает в различных ролях в двух ассоциациях, которые иллюстрирует рисунок 7.1:

- 1) в роли «непроверенное» в ассоциации ф) из п. 7.3.4,
- 2) в роли «проверенное» в ассоциации х) из п. 7.3.4.

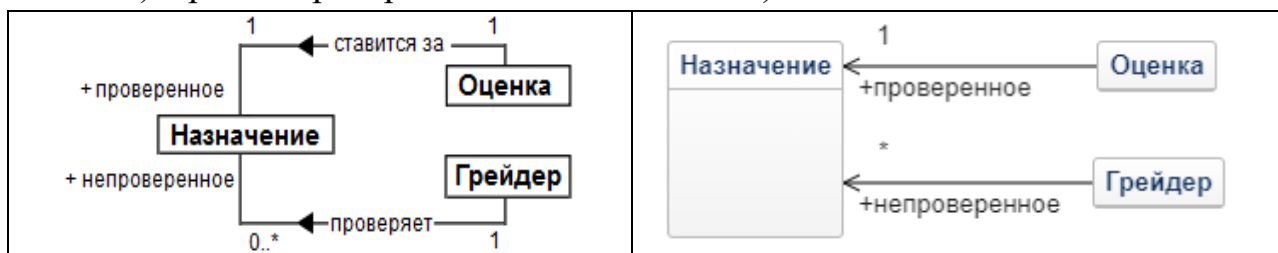


Рис. 7.1. Роли ассоциаций, связанных с понятием Назначение

Поэтому у перечисленных ассоциаций роли должны быть указаны явно. У остальных ассоциаций роли совпадают с именами ассоциаций. Кратности ролей ассоциаций приведены в таблице 7.4.

Таблица 7.4

Кратности ролей ассоциаций

Кратность ролей ассоциации		Ассоциации из п. 7.3.4
1	1	в), д), е), ж), к), р), с), у), х), щ), ю)
1	0..*	а), н), ф)
1	1..*	б), г), м), п), ц), ш)
1	2..*	и), л)
1	2..5	т), э)

7.3.6 Выявление ограничений. В ПрО существуют ограничения:

– связанные с понятиями:

1) у понятия СрокиКрайние значение атрибута «датаОтправки» должно быть установлено не позже «датаЗавершенияОценивания» хотя бы на три дня (для выполнения деятельности Грейдера);

2) у понятия Итоговая оценка значение атрибута «балл» не может быть установлено больше, чем значение атрибута «максБаллЗадания» понятия Задание;

3) у понятия Оценка значение атрибута «балл» не может быть установлено больше, чем значение его атрибута «максБалл»;

– связанные с ассоциациями:

1) ассоциация **отправляет** должна позволить отправить Обучающемуся на проверку Работу не позже наступления датаОтправки;

2) ассоциация **распределяет** должна позволить выполнить Инструктору распределение Грейдеров не раньше наступления датаОтправки;

3) ассоциация **проверяет** должна позволить выполнить Грейдеру оценивание Назначений не позже наступления датаЗавершенияОценивания;

4) ассоциация **экспортируется в** должна позволить отправить Итоговую оценку в Журнал оценок только после полного завершения оценивания Назначений оценщиками.

7.3.7 Выделение отношения «супертип» – «подтип». Этот тип отношений существует между понятиями:

– Пользователь:

1) Инструктор;

2) Обучающийся;

– Обучающийся – Грейдер;

– Работа – Назначение .

Проверяем правильность выявления отношения «супертип» – «подтип». Результат представлен в таблице 7.5.

Таблица 7.5

Правильность выполнения правил «100%» и «is_a»

Название правила	Выполнение правил для понятий			
	Пользователь – Инструктор	Пользователь – Обучающийся	Обучающийся – Грейдер	Работа – Назначение
1 Правило «100%»	+	+	+	+
2. Правило «is_a»	+	+	+	+

Так как правила «100%» и «is_a» выполняются для всех пар понятий, выявленных выше, значит, между понятиями этих пар существует отношение обобщения.

7.3.8 Выполнение многомерной множественной классификации. Дискриминатор «Роль» между подтипами Инструктор и Обучающийся и супертипом Пользователь определяет свойства разбиения:

– неполное, так как статусы «Инструктор» или «Обучающийся» совпадают с ролями пользователей LMS Moodle («Учитель» и «Студент» соответственно), а в Moodle допустимы еще роли «Управляющий» (с правами на создание курса и распределение ролей остальных пользователей) и «Ассистент» (с правами проверки деятельности студентов, но без права редактирования курса),

– с перекрытием, так как при необходимости Инструктор может переключаться на роль Обучающийся для мониторинга курса глазами студентов.

Поэтому разбиение по дискриминатору «Роль» для супертипа Пользователь и подтипов Инструктор и Обучающийся должно быть указано явно: Роль {incomplete, overlapping}.

Понятие Пользователь не может быть абстрактным, так как множество его подтипов (Инструктор и Обучающийся) не полностью покрывает множество Пользователь в соответствии с классификацией п. 7.3.8 а).

7.3.9 Выделение отношения агрегации. В ПрО существуют два вида агрегации (между понятиями категории «Контейнеры других объектов» и понятиями категории «Содержимое контейнеров»):

- 1) композитная – между парами понятий а)-м), э) из п. 7.3.4,
- 2) коллективная – между парой понятий н) из п. 7.3.4.

У каждого понятия категории «Контейнеры других объектов» среди атрибутов должны быть такие, которые обеспечивают связь с понятиями категории «Содержимое контейнеров». Поэтому понятию-контейнеру, участвующему в агрегации с кратностью:

- равной «1», нужен атрибут для единичного понятия-части;
- отличной от «1», нужны атрибуты:
 - 1) множество понятий-частей;
 - 2) количество элементов в таком множестве.

Для того чтобы в концептуальной модели продемонстрировать коллективную агрегацию, было введено понятие Курс. Понятие Задание явно не присутствовало в описании прецедентов. Оно было введено в концептуальную модель, чтобы показать связь LMS Moodle с PAsystem (отправка Работы представляет собой факт выполнения Задания).

7.3.10 При построении диаграмм понятий сгруппируем понятия семантически с учетом правила « 7 ± 2 »:

- Пользователи (Пользователь, Инструктор, Обучающийся, Грейдер, Курс, КритерииОценивания, Задание, СрокиКрайние, Работа, Назначение),
- Критерии оценивания (КритерииОценивания, СписокРубрик, Рубрика, СписокУровней, Уровень),
- Оценивание (Грейдер, Назначение, Работа, Оценка, ИтоговаяОценка, ЖурналОценок, Курс, Moodle).

Соответствующие диаграммы понятий представлены на рисунках 7.2–7.4.

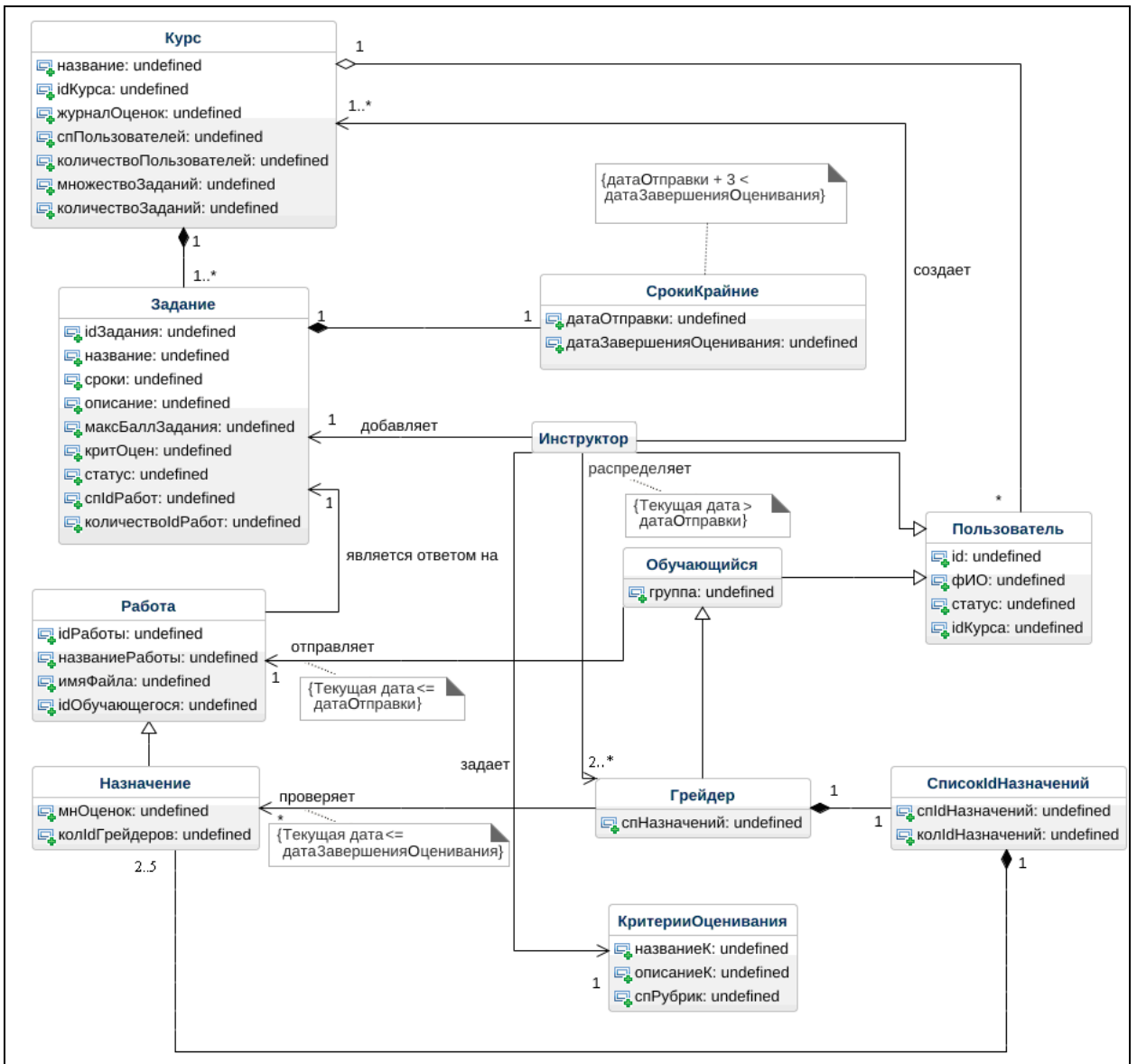


Рис. 7.2. Диаграмма понятий (Пользователи) PASystem

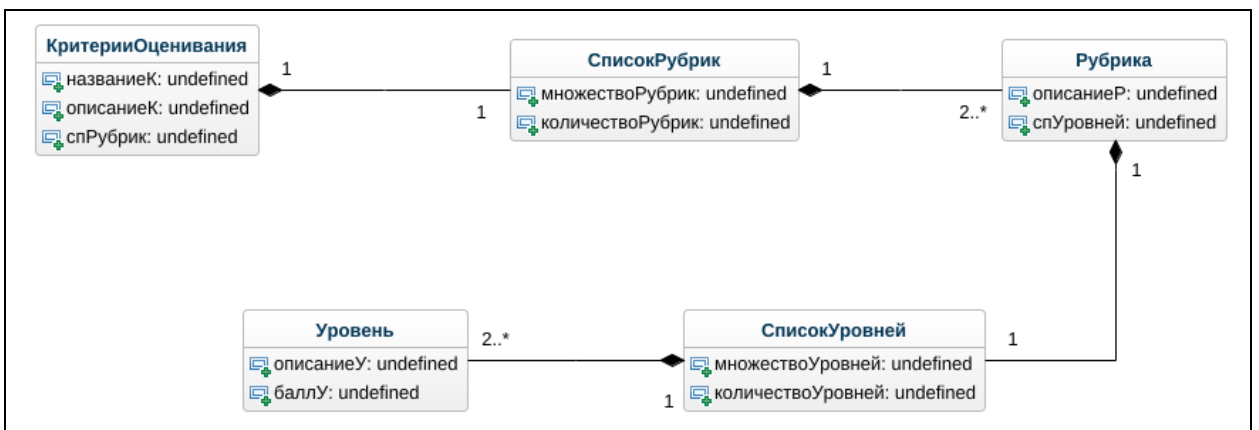


Рис. 7.3. Диаграмма понятий (Критерии оценивания) PASystem

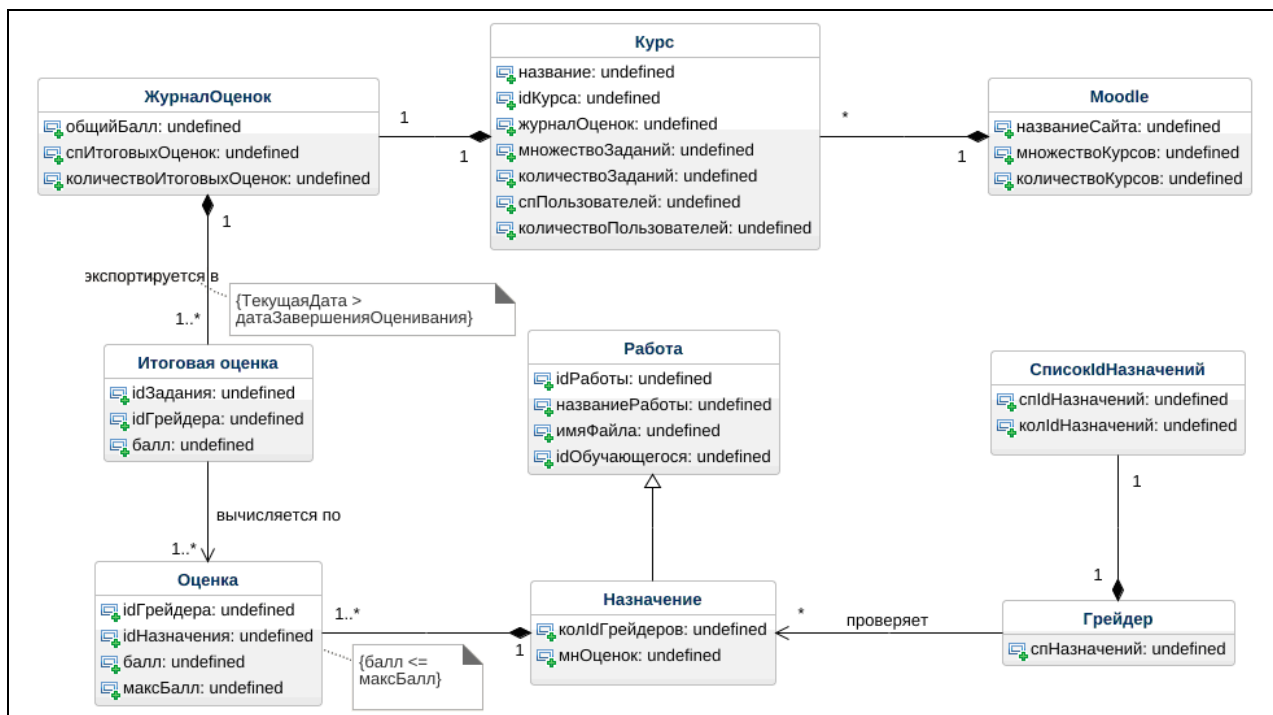


Рис. 7.4. Диаграмма понятий (Оценивание) PASystem

7.3.11 Общая диаграмма понятий представлена на рисунке 7.5.

7.4 Домашнее задание № 4

7.4.1. Для предметной области выполнить концептуальное моделирование при объектно-ориентированном подходе к моделированию и формированию соответствующей документации. На основе описания прецедентов построить концептуальную модель ИС. Выявить 5–7 понятий и отношения между ними. Выявить атрибуты понятий, роли отношений, множественную классификацию, а также ограничения. Предусмотреть использование отношений обобщения и агрегации.

7.4.2 Построить диаграмму(-ы) понятий, в которой(-ых) отразить концептуальную модель ПрО, построенную в п. 7.4.1:

- 7±2 понятий с указанием их атрибутов,
- ассоциации с указанием их ролей,
- ограничения, связанные с понятиями и ассоциациями,
- отношения обобщения,
- многомерную множественную классификацию и абстрактное(-ые) понятие(-я),
- отношения композитной агрегации,
- отношения коллективной агрегации.

7.4.3 При использовании инструмента [5] для создания диаграммы понятий используется вкладка «Class Diagram».

7.4.4 Срок выполнения: 5-я неделя. Срок защиты: 6-я неделя. Максимальная оценка составляет 5 баллов, зачетная – 2,25 балла. Представление отчета через элемент «Отчет ДЗ4+ЛР2» курса «ОПРИС» на moodle.movs.psu.ru.

7.4.5 Содержание отчета:

- а) выделение понятий (непосредственное выделение понятий и поиск по списку стандартных категорий понятий),
- б) выделение атрибутов понятий,
- в) выделение ассоциаций (по определению и по списку стандартных категорий ассоциаций),
- г) выделение ролей ассоциаций,
- д) выявление ограничений (связанных с понятиями и ассоциациями),
- е) выделение отношения обобщения,
- ж) выполнение многомерной множественной классификации и выявление абстрактных понятий,
- и) выделение отношения агрегации (композиционной и коллективной),
- к) семантическая группировка понятий с применением правила «7±2» (при необходимости),
- л) диаграмма(-ы) понятий, составленная(-ые) на основе элементов концептуальной модели ПрО, построенной в домашнем задании № 4.

7.5 Вопросы для самопроверки

1. Назначение концептуальной модели.
2. Почему при выделении отношения обобщения нужны оба правила: «100%» и «is_a»?
3. Сущность абстрактного понятия.
4. Отличие композиционной агрегации от коллективной.
5. Как используется многомерная множественная классификация в языках программирования?
6. Концептуальная модель диаграммы понятий.
7. Какие ошибки допущены в диаграмме понятий, представленной на рисунке 7.6?

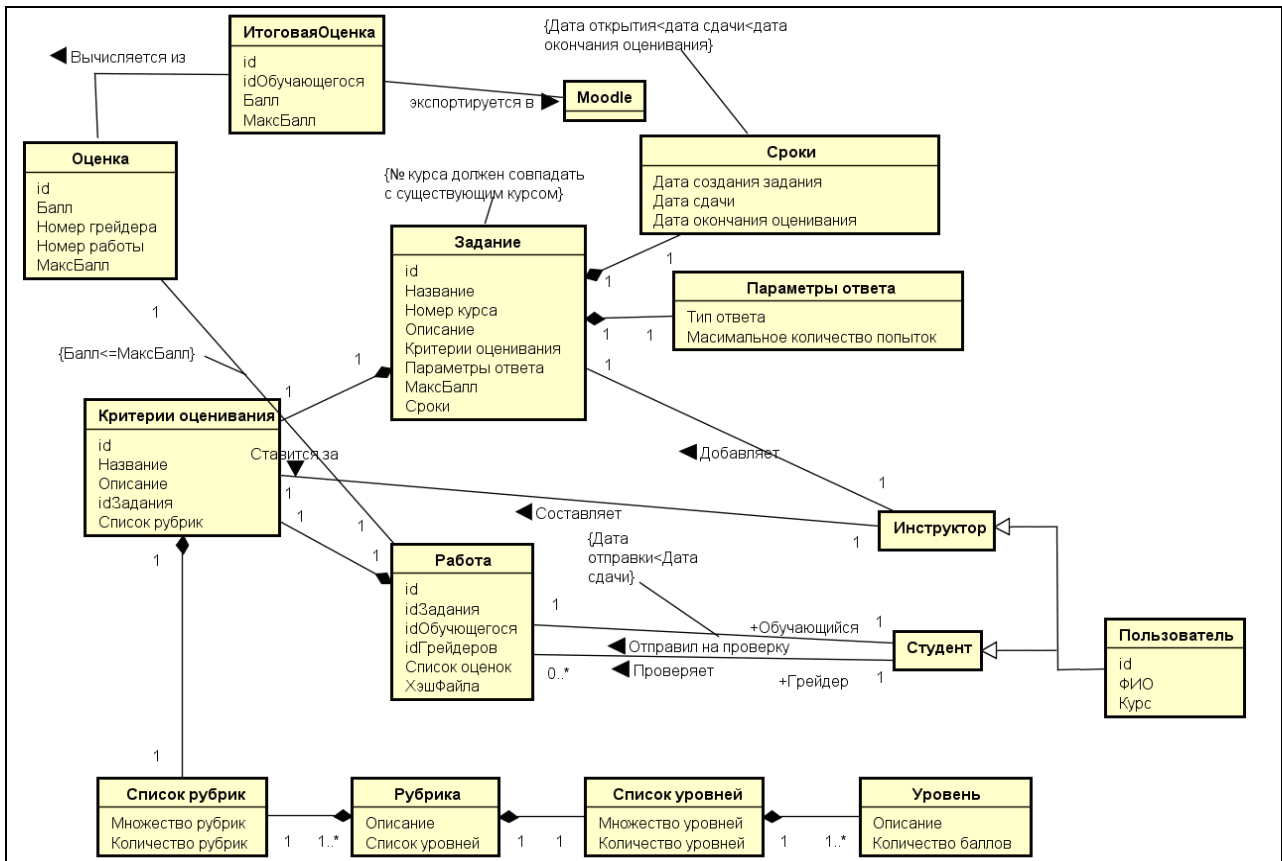


Рис. 7.6. Диаграмма понятий с ошибками

8 Лабораторная работа № 2. Моделирование на этапе анализа ИС при объектно-ориентированном подходе: моделирование поведения системы

8.1 Цель лабораторной работы № 2

Целью выполнения ЛР № 2 является изучение основных элементов диаграммы последовательностей, создание диаграммы последовательностей, а также получение навыков работы с инструментальными средствами создания Sequence Diagram. Выполняется студентами очной и заочной форм обучения.

Целью выполнения ЛР № 2 студентами очной формы обучения также является освоение методики описания системных операций, используемой для их определения на основе модели поведения системы.

8.2 Общие методические указания к выполнению ЛР № 2

8.2.1 Диаграмма последовательностей [10]–[18] (*англ.* Sequence Diagram, SD) – это диаграмма, на которой изображено упорядоченное во времени взаимодействие объектов. Основными элементами являются участвующие во взаимодействии объекты и последовательность сообщений, которыми они обмениваются.

8.2.2 Для выполнения ЛР № 2 необходимо предварительно изучить:

- стандарт ISO/IEC 19505-1:2012 [8]–[9],
- лексику, синтаксис, семантику и нотацию диаграмм последовательностей языка UML,
- инструмент для создания диаграмм последовательностей языка UML [5],
- тему 6.5 [17].

8.3 Сроки выполнения и представление отчета по ЛР № 2

Срок выполнения студентами очной формы обучения: 6-я и 7-я недели. Срок защиты: 8-я неделя. Представление отчета через элемент «Отчет ДЗ4+ЛР2» курса «ОПРИС» на moodle.movs.psu.ru.

Представление отчета студентами заочной формы обучения через элемент «Отчет ЛР2» курса «ОПРИС-3» на moodle.movs.psu.ru.

8.4 Пример выполнения ЛР № 2

8.4.1 Визуализация особенностей реализации каждого варианта использования выполняется на отдельной диаграмме последовательностей [15]. Достаточно построить ДП для:

- основных потоков,
- наиболее интересных альтернативных потоков и подпотоков.

В п. 3.4.8 были выделены восемь прецедентов ИС PASystem, По классификации, выполненной в п. 5.3.4, все они были отнесены к категории главных (далее будет использована их нумерация из п. 5.3.4, приведенная в таблице 8.1). Поэтому диаграммы последовательностей требуется построить для этих прецедентов, их основных потоков и подпотоков.

Таблица 8.1

Нумерация и имена прецедентов, для которых выполняется построения ДП

Номер главного прецедента	Имя главного прецедента
2	Настройка критериев оценивания
3	Отправка работы
4	Назначение грейдеров
5	Оценивание работы
6	Вычисление итоговых оценок всех обучающихся
7	Экспорт оценок

8.4.2 Требованиями к построению диаграмм последовательностей являются [15]:

а) выделение из всей совокупности классов только тех, объекты которых участвуют в моделируемом взаимодействии,

б) нанесение всех объектов в верхней горизонтальной части диаграммы с соблюдением порядка инициализации сообщений (допускается использование дополнительных объектов как в диаграмме активностей),

в) установление при необходимости различных статусов объектов – существующих постоянно или временно (только на период выполнения ими требуемых действий),

г) отображение линий жизни объектов, имеющих статус постоянных,

д) отображение при необходимости линий жизни объектов, имеющих статус временных только в пределах времени их существования в системе (с созданием и уничтожением),

е) спецификацию сообщений (выделение из текста описания прецедента всех генерируемых актором сообщений и их параметров),

ж) отображение сообщений на ДП в порядке генерации, их запись с учетом лексики сообщения ДП,

и) использование синхронных, асинхронных, ответных, альтернативных, рефлексивных, рекурсивных сообщений,

к) использование нотации MSC (Message Sequence Chart) для отображения составных шагов взаимодействия, подчиненных потоков, повторяющихся потоков,

л) создание отдельной ДП в сложных случаях для моделирования каждой ветви управления (в том числе, для подпотоков и/или альтернативных потоков прецедента).

8.4.3. Выделение типов сообщений. Примером асинхронного сообщения может служить прерывание операции при возникновении исключительной ситуации (альтернативный поток прецедента). В этом случае информация о такой ситуации передается вызывающему объекту для продолжения процесса дальнейшего взаимодействия. Рефлексивное сообщение характерно тем, что объект посылает сообщение самому себе. В ДП его используют, когда необходимо «...показать действие, выполняемое самим объектом (или внутри него), либо то, что объект сам себя вводит в некоторое состояние ...» [19].

8.4.4 Составной шаг взаимодействия. В ДП создается инструментом комбинированного фрагмента взаимодействия («Блок» или «Frame»). Для ветвлений используется имя «alt», а альтернативные последовательности сообщений изображаются внутри блока последовательно, друг под другом. Подпоследовательности отделяются пунктирной линией и для них указываются соответствующие сторожевые условия, напр., [else]. Это способ изображения альтернативных сообщений в нотации MSC, который был разработан и успешно применяется производителями встроенных систем [18].

8.4.5 Моделирование подчиненных потоков. Для моделирования подпотоков прецедентов допускается использование нотации MSC. Внутри каждого взаимодействия создаются ссылки на одноименные диаграммы последовательностей, которые оформляются с именем «gef». Каждый такой подпоток должен быть представлен самостоятельной ДП.

8.4.6 Моделирование повторяющихся потоков. Для моделирования допускается использование нотации MSC. При этом блоки с повторяющимися потоками оформляются с именем «loop».

8.4.7 Диаграммы последовательностей главных прецедентов PAsystem представлена на рисунках 8.1–8.10.

8.5 Пример определения системных операций

8.5.1 Исходной является модель поведения системы ИС PAsystem, построенная в п. 8.4.

8.5.2 Выделение списка системных операций. Список системных операций соответствует списку системных сообщений, представленных в диаграммах последовательностей п. 8.4.7: задатьКритерииОценивания, отправитьРаботу, назначитьГрейдеров, оценитьНазначение, вычислитьИтоговыеОценки, экспортироватьОценки.

8.5.3 Определение системных операций. Описываем СО в соответствии с рекомендуемым шаблоном. При описании изменений системы на уровне данных в результате выполнения системных операций используем действия по:

- созданию (уничтожению) объектов,
- формированию (разрыву) связей между объектами,
- модификации значений атрибутов объектов.

Пример описания СО приведен на рисунке 8.11.

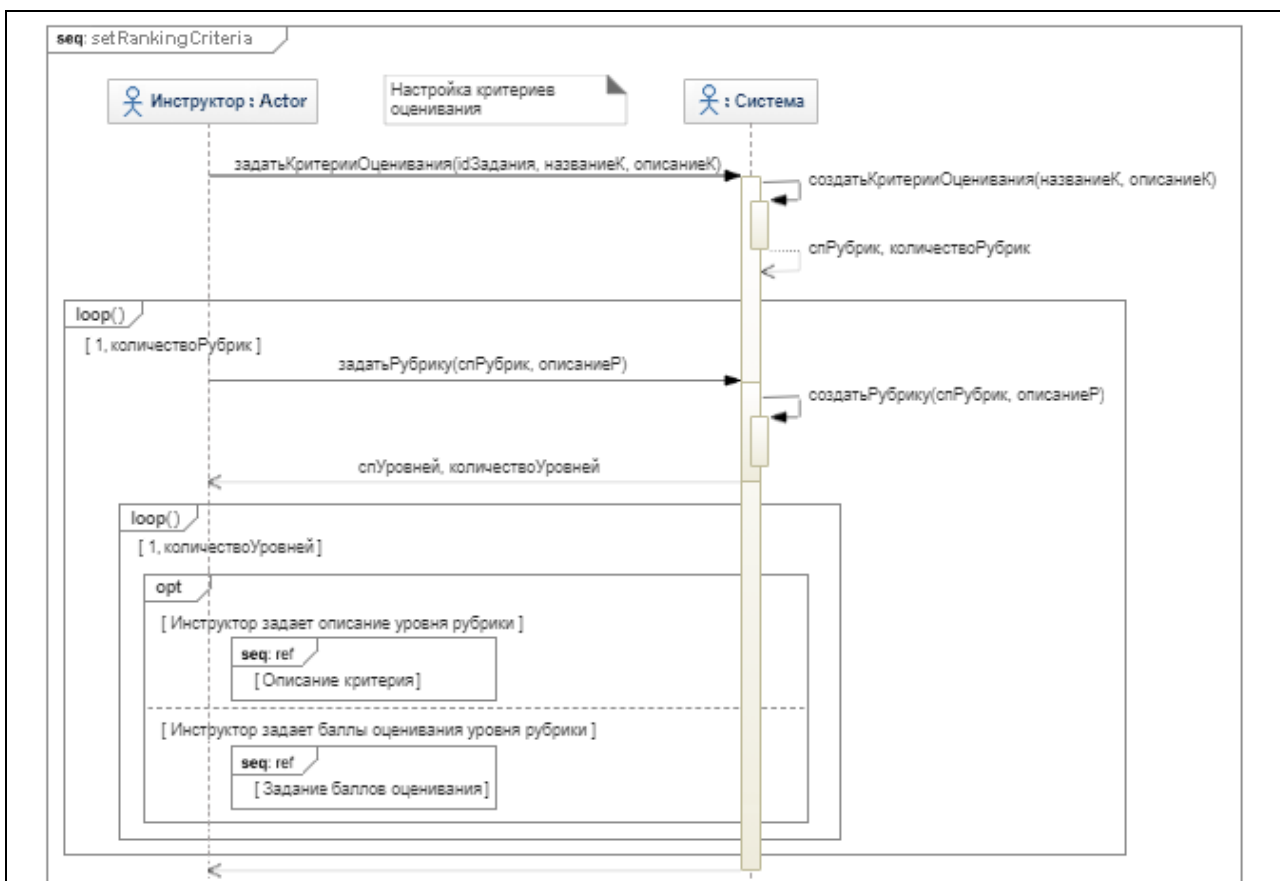


Рис. 8.1. Диаграмма последовательностей (прецедент 2, основной поток)

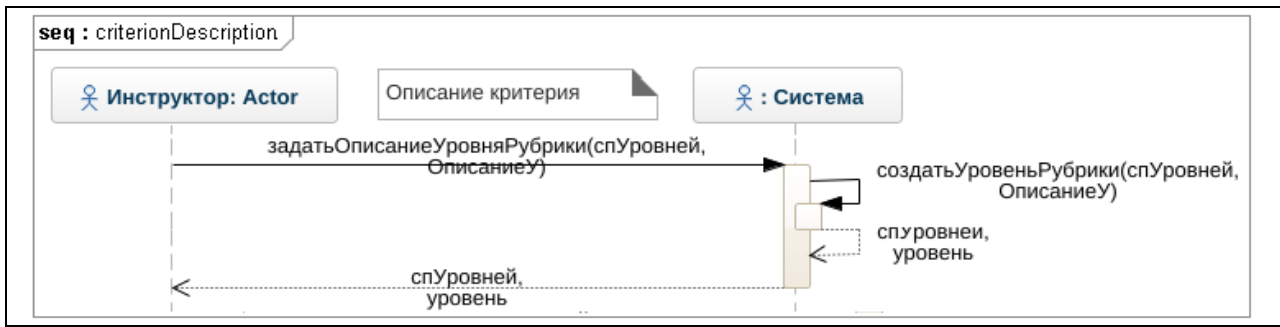


Рис. 8.2. Диаграмма последовательностей (прецедент 2, подпоток S1)

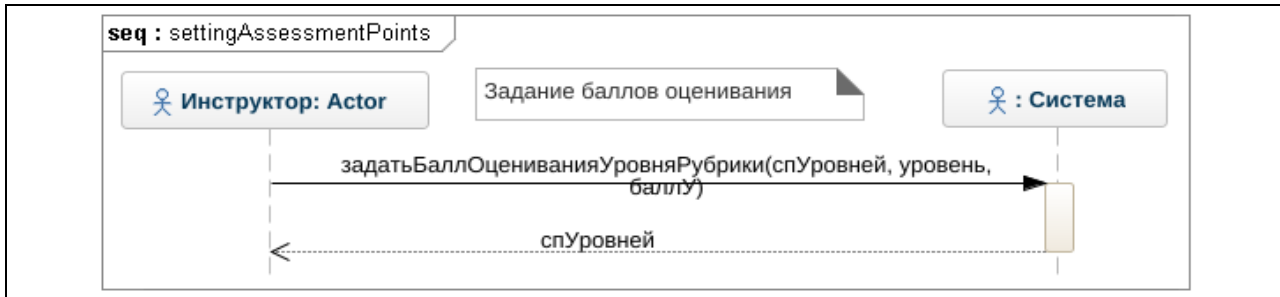


Рис. 8.3. Диаграмма последовательностей (прецедент 2, подпоток S2)



Рис. 8.4. Диаграмма последовательностей PASystem (прецедент 3, основной поток)

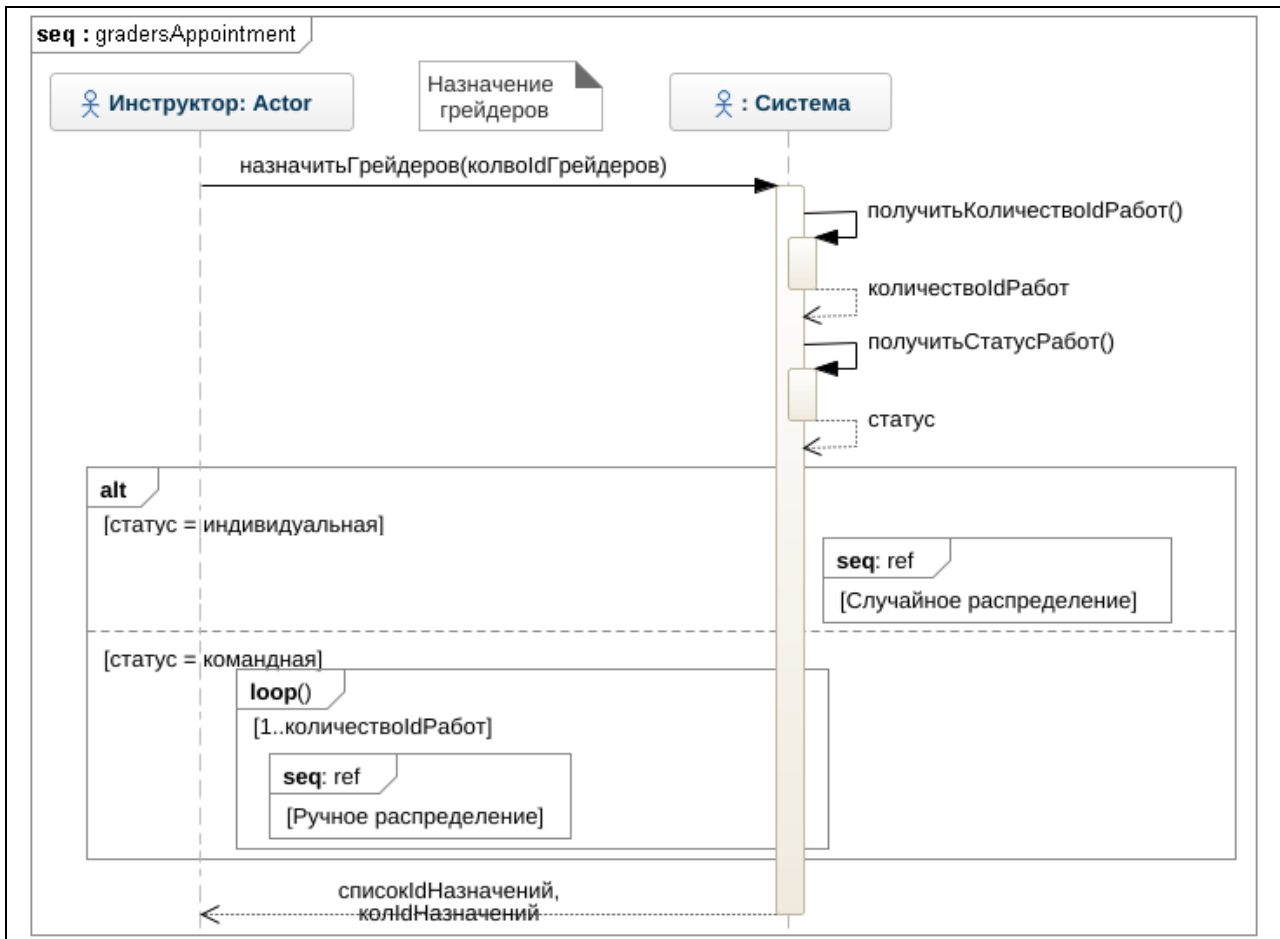


Рис. 8.5. Диаграмма последовательностей (прецедент 4, основной поток)

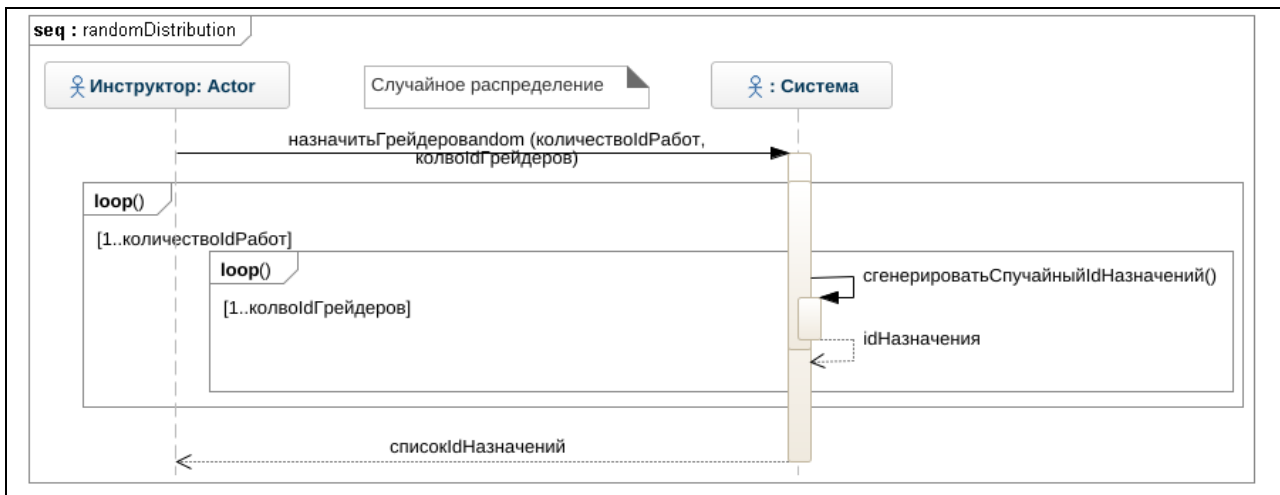


Рис. 8.6. Диаграмма последовательностей (прецедент 4, подпоток S3)

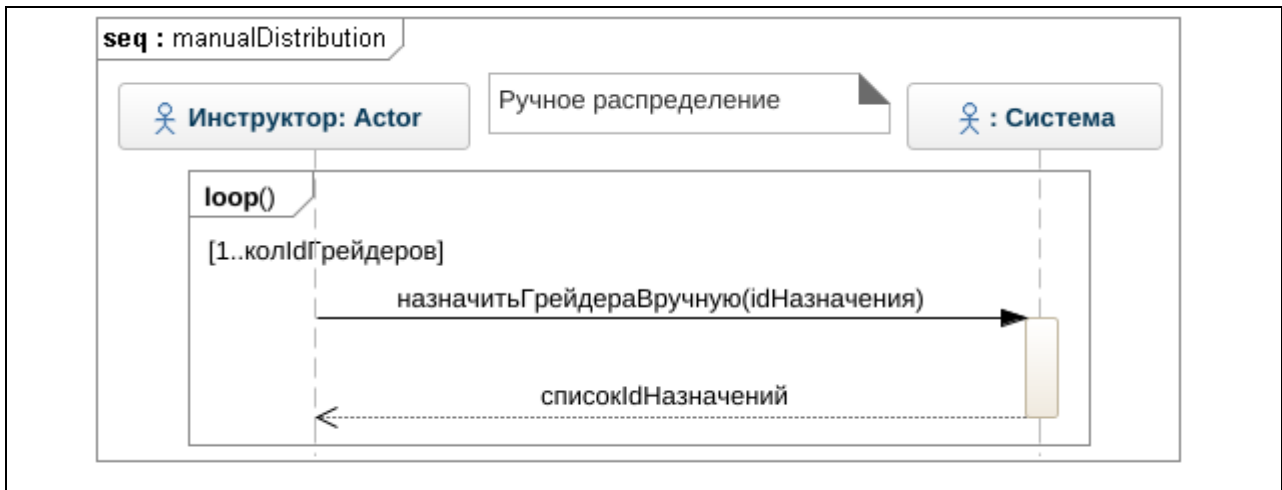


Рис. 8.7. Диаграмма последовательностей (прецедент 4, подпоток S4)

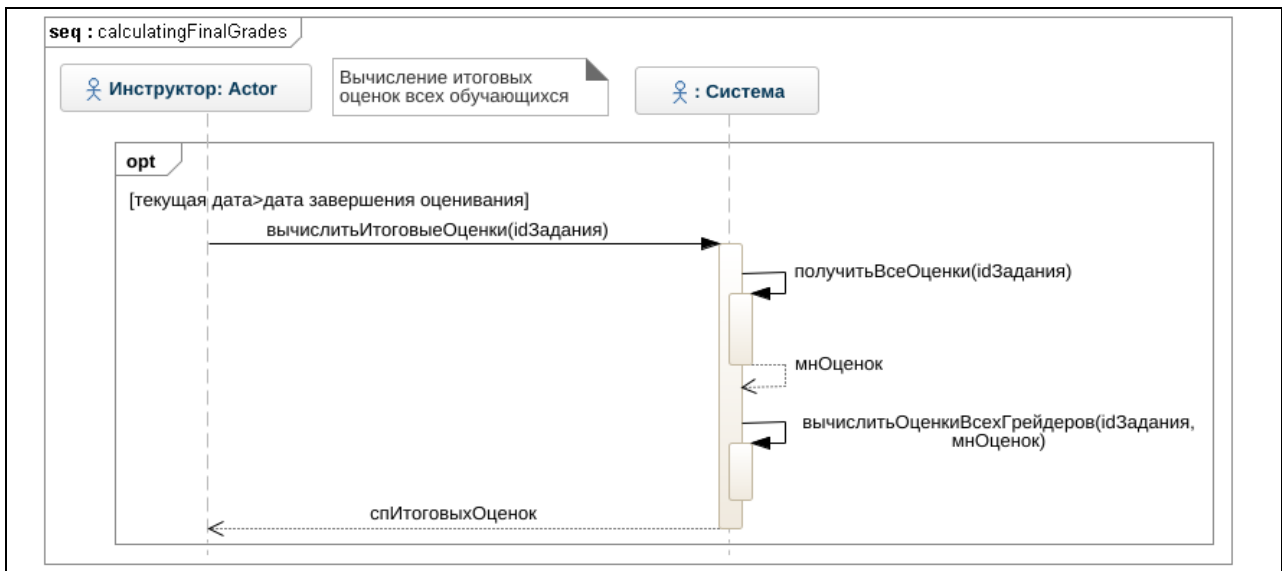


Рис. 8.8. Диаграмма последовательностей (прецедент 6, основной поток)

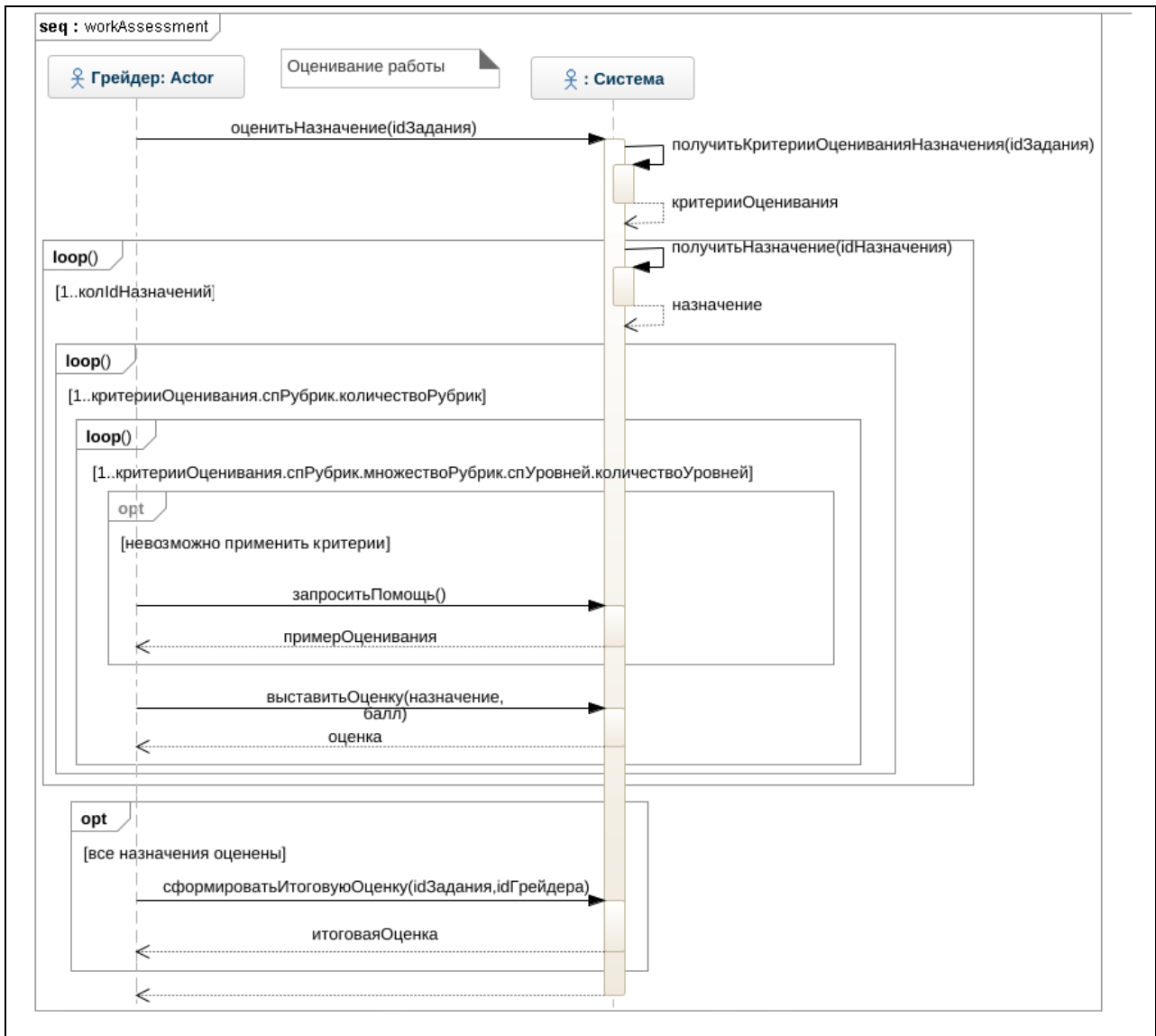


Рис. 8.9. Диаграмма последовательностей (прецедент 5, основной поток)

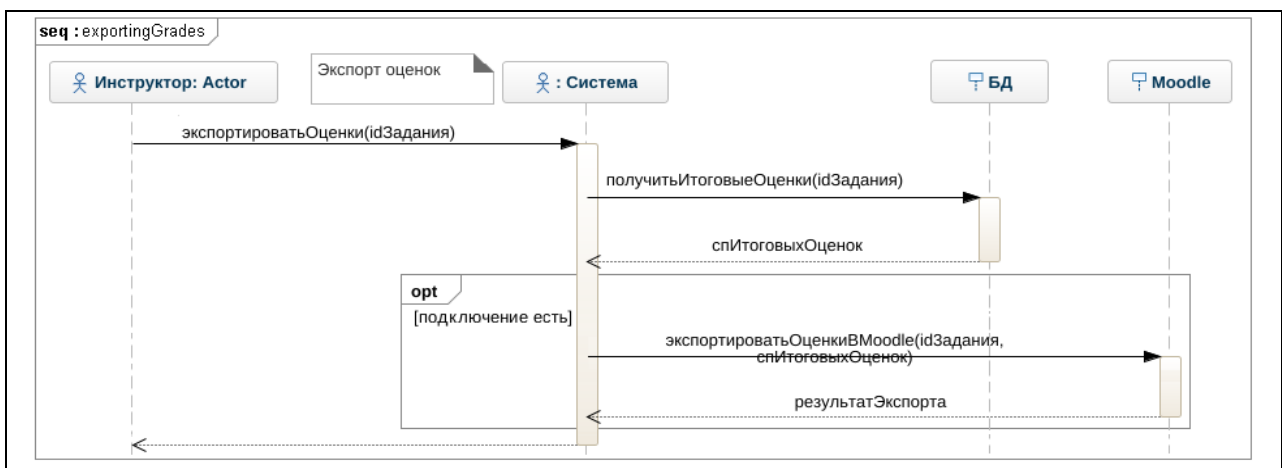


Рис. 8.10. Диаграмма последовательностей (прецедент 7, основной поток)

<p><u>Имя:</u> задатьКритерииОценивания(idЗадания, названиеК, описаниеК)</p> <p><u>Обязанности:</u> Предоставить возможность задания параметров парного оценивания для конкретного задания, в том числе, создания рубрик и уровней оценивания.</p> <p><u>Ссылки:</u> Прецедент «Настройка критериев оценивания»</p> <p><u>Примечание:</u> Использовать самый быстрый способ доступа к БД.</p> <p><u>Исключения:</u> нет.</p> <p><u>Предусловия:</u> В курсе LMS Moodle создано задание для парного оценивания со значением атрибута idЗадания, равным значению «idЗадания».</p> <p><u>Постусловия:</u> Для нового критерия оценивания создан объект :КритерииОценивания (создание экземпляра). Атрибут :КритерииОценивания.Название принял значение «названиеК» (модификация атрибута). Атрибут :КритерииОценивания.ОписаниеК принял значение «описаниеК» (модификация атрибута). Для нового списка рубрик создан объект :СписокРубрик (создание экземпляра). Для нового списка рубрик объект :СписокРубрик связан с объектом :КритерииОценивания (формирование связи). Создан объект :Рубрика (создание экземпляра). Объект :Рубрика связан с объектом :СписокРубрик (формирование связи). Атрибут : Рубрика.ОписаниеР принял значение «описаниеР» (модификация атрибута). Для нового списка уровней создан объект :СписокУровней (создание экземпляра). Для нового списка уровней объект :СписокУровней связан с объектом :Рубрика (формирование связи). Создан объект :Уровень (создание экземпляра). Объект :Уровень связан с объектом :СписокУровней (формирование связи). Атрибут :Уровень.ОписаниеУ принял значение «описаниеУ» (модификация атрибута). Атрибут :Уровень.БаллУ принял значение «балл» (модификация атрибута).</p> <p><u>Имя:</u> отправитьРаботу(idЗадания, idРаботы, названиеРаботы, idОбучающегося)</p> <p><u>Обязанности:</u> Отправить на оценивание работу конкретного обучающегося, которая является ответом на конкретное задание.</p> <p><u>Ссылки:</u> Прецедент «Отправка работы»</p> <p><u>Примечание:</u> Использовать самый быстрый способ доступа к БД; необходимы определенные права для чтения БД Moodle.</p> <p><u>Исключения:</u> нет.</p> <p><u>Предусловия:</u> Обучающийся загрузил файл ответа на задание в элемент парного оценивания Moodle, который хранится в БД Moodle. idРаботы хранится в атрибуте spidРабот этого задания.</p> <p><u>Постусловия:</u> Для новой работы создан объект :Работа (создание экземпляра). Атрибут :Работа.idРаботы принял значение «idРаботы» (модификация атрибута). Атрибут :Работа.названиеРаботы принял значение «названиеРаботы» (модификация атрибута). Атрибут :Работа.ИмяФайла принял значение «имяФайла» (модификация атрибута). Атрибут :Работа.IdОбучающегося принял значение «idОбучающегося» (модификация атрибута).</p>

Рис. 8.11. Описание системных операций PASystem

Имя: назначитьГрейдеров(колIdГрейдеров)
Обязанности: Назначить каждой отправленной обучающимися работе указанное количество грейдеров для оценивания.
Ссылки: Прецедент «Назначение грейдеров»
Примечание: Использовать самый быстрый способ доступа к БД; необходимы определенные права для чтения БД Moodle.
Исключения: нет.
Предусловия: Все обучающиеся отправили работы на проверку; список id_Работ и их количество хранится в атрибутах спIdРабот и количествоIdРабот в соответствующем задании в БД Moodle.
Постусловия:
Для нового назначения создан объект :Назначение (создание экземпляра).
Атрибут :Назначение.КолIdГрейдеров принял значение «колIdГрейдеров» (модификация атрибута).
Атрибут :Назначение.мнОценок принял значение «пусто» (модификация атрибута).
Атрибут :Грейдер.спНазначений.КолIdНазначений принял значение «КолIdГрейдеров» (модификация атрибута).
Атрибут :Грейдер.спНазначений.спIdНазначений принял значение «списокIdНазначений» (модификация атрибута).

Имя: оценитьНазначение(idЗадания)
Обязанности: Сохранить оценку, которую выставил грейдер за работу обучающегося, и в случае, если грейдер проверил все назначенные ему работы, вычислить ту долю итоговой оценки, которая формируется из оценок, выставляемых грейдером.
Ссылки: Прецедент «Оценивание работы»
Примечание: Использовать самый быстрый способ доступа к БД; необходимы определенные права для чтения БД Moodle.
Исключения: При возникновении трудностей при оценивании грейдер запрашивает пример оценивания работы.
Предусловия: У грейдера есть непроверенные работы, назначенные ему для оценивания.
Постусловия:
Для нового критерия оценивания создан объект :КритерииОценивания со значениями атрибутов, полученными из БД (создание экземпляра).
Для новой оценки создан объект :Оценка (создание экземпляра).
Для новой оценки объект :Оценка связан с объектом :Назначение (формирование связи).
Атрибут :Оценка.idГрейдера принял значение «idГрейдера» (модификация атрибута).
Атрибут :Оценка.idНазначения принял значение «idНазначения» (модификация атрибута).
Атрибут :Оценка.максБалл принял значение «БаллУ» текущего уровня текущей рубрики критериев оценивания (модификация атрибута).
Атрибут :Оценка.балл принял значение «балл» (модификация атрибута).
Для новой итоговой оценки создан объект :Итоговая оценка (создание экземпляра).
Атрибут :ИтоговаяОценка.idЗадания принял значение «idЗадания» (модификация атрибута).
Атрибут :ИтоговаяОценка.idГрейдера принял значение «idГрейдера» (модификация атрибута).
Атрибут :ИтоговаяОценка.балл принял значение, вычисленное по значениям «:Назначение.мнОценок» (модификация атрибута).

Продолжение рисунка 8.11

<p><u>Имя:</u> вычислитьИтоговыеОценки(idЗадания)</p> <p><u>Обязанности:</u> Вычислить консолидированные оценки для всех обучающихся (на основе баллов, полученных от грейдеров, и выставленных самими обучающимися другим участникам парного оценивания) и сохранить их в БД системы.</p> <p><u>Ссылки:</u> Прецедент «Вычисление итоговых оценок всех обучающихся»</p> <p><u>Примечание:</u> Использовать самый быстрый способ доступа к БД.</p> <p><u>Исключения:</u> Если не все грейдеры выставили оценки, алгоритм вычисления корректирует консолидированные оценки.</p> <p><u>Предусловия:</u> Дата завершения оценивания прошла. Грейдеры выполнили проверку назначенных им работ и выставили баллы в соответствии с критериями оценивания.</p> <p><u>Постусловия:</u> Атрибут <u>:Назначение.мнОценок</u> принял значение, вычисленное по значениям «:Оценка.балл», выставленным всеми назначенными грейдерами (модификация атрибута). Атрибут <u>:ИтоговаяОценка.балл</u> принял значение, вычисленное по значениям «:Назначение.мнОценок» (модификация атрибута).</p> <p><u>Имя:</u> экспортироватьОценки(idЗадания)</p> <p><u>Обязанности:</u> Экспортировать консолидированные оценки за парное оценивание конкретного задания в журнал оценок Moodle.</p> <p><u>Ссылки:</u> Прецедент «Экспорт оценок»</p> <p><u>Примечание:</u> Использовать самый быстрый способ доступа к БД, необходимы определенные права для записи в БД Moodle.</p> <p><u>Исключения:</u> Не удалось подключиться к LMS Moodle. Реакция: отмена операции экспорта.</p> <p><u>Предусловия:</u> Вычислены итоговые оценки всех грейдеров.</p> <p><u>Постусловия:</u> нет (это системное сообщение не приводит к изменениям внутри системы, а только к изменениям в другой ИС, внешней по отношению к PASystem, т.е. реализуется межсистемное взаимодействие).</p>
--

Продолжение рисунка 8.11

8.6 Требования, указания и задания к ЛР № 2

Для студентов очной и заочной форм обучения: построить диаграммы последовательностей для главных прецедентов в соответствии с требованиями п. 8.4.2 а)-л).

Для студентов очной формы обучения:

- на основе диаграмм последовательностей выделить список системных операций,
- описать системные операции в терминах изменений на уровне данных.

8.7 Содержание отчета по ЛР № 2

Для студентов очной и заочной форм обучения: отчет по ЛР № 2 должен содержать диаграммы последовательностей для главных прецедентов в соответствии с п. 5.4.3, д).

Для студентов очной формы обучения: отчет по ЛР № 2 должен содержать выделение системных операций и описание системных операций.

8.8 Вопросы для самопроверки

1. Понятие системного сообщения.
2. Понятие системной операции.
3. Связь между системным сообщением и системной операцией.
4. Отличие системной операции от системного сообщения.
5. Концептуальная модель диаграммы последовательностей.
6. Какие ошибки допущены в описании СО на рисунке 8.12?

<p><u>Имя:</u> Грейдер оценил работу обучающегося</p> <p><u>Обязанности:</u> Она сохраняет оценку, которую выставил грейдер обучающемуся, и в случае, если грейдер проверил все назначенные ему работы, начисляет вознаграждение.</p> <p><u>Ссылки:</u> Прецедент «Грейдер оценил работу обучающегося»</p> <p><u>Примечание:</u> Использовать самый быстрый способ доступа к БД.</p> <p><u>Исключения:</u></p> <p><u>Предусловия:</u> У грейдера есть не оцененные назначенные работы для проверки.</p> <p><u>Постусловия:</u></p> <p>Для подтвержденной грейдером оценки создается объект :<u>Оценка</u> (создание экземпляра). Атрибут :<u>Оценка.idРаботы</u> принял значение «idРаботы» (модификация атрибута). Атрибут :<u>Оценка.idГрейдера</u> принял значение «idГрейдера» (модификация атрибута). Атрибут :<u>Оценка.Балл</u> принял значение «Оценка» (модификация атрибута). Объект :<u>Оценка</u> добавляется в объект :<u>СписокОценок</u> (формирование связи). Для вознаграждения грейдеру создается объект :<u>Вознаграждение</u> (создание экземпляра). Атрибут idГрейдера принимает значение «idГрейдера» (модификация).</p>
--

Рис. 8.12. Описание СО с ошибками

8.9 Оценивание ЛР № 2

Оценивание выполнения ЛР № 2 приведено в таблицах 8.2–8.3. Максимальная оценка составляет у студентов очной формы обучения 10 баллов, зачетная – 4,1 балла. Максимальная оценка составляет у студентов заочной формы обучения 20 баллов, зачетная – 8,5 балла.

Таблица 8.2

Оценивание выполнения ЛР № 2 (очная форма обучения)

Баллы	Выполнение требований к ЛР № 2
1	2
Общий 10, в том числе 9	Выполнены требования 8.4.2 а)-л); ошибок лексики и синтаксиса диаграммы последовательностей нет, ошибок лексики и синтаксиса СО нет
Общий 9,9–8, в том числе 9,9–7	Выполнены требования 8.4.2 а)-ж); имеются ошибки лексики и синтаксиса диаграммы последовательностей, имеются ошибки лексики и синтаксиса описания СО
Общий 7,9–6, в том числе 7,9–5	Выполнены требования 8.4.2 а)-г), и)-л); ошибок лексики и синтаксиса последовательностей понятий нет, имеются ошибки лексики и синтаксиса описания СО
Общий 4,9–4,1, в том числе 4,9–3,1	Выполнены требования 8.4.2 а)-г), и)-л); имеются ошибки лексики и синтаксиса диаграммы последовательностей, имеются ошибки лексики и синтаксиса описания СО

Таблица 8.3

Оценивание выполнения ЛР № 2 (заочная форма обучения)

Баллы	Выполнение требований к ЛР № 2
1	2
Общий 20, в том числе 19	Выполнены требования 8.4.2 а)-л); ошибок лексики и синтаксиса диаграммы последовательностей нет
Общий 19,9–16, в том числе 19,9–15	Выполнены требования 8.4.2 а)-ж); имеются ошибки лексики и синтаксиса диаграммы последовательностей
Общий 15,9–12, в том числе 15,9–11	Выполнены требования 8.4.2 а)-г), и)-л); ошибок лексики и синтаксиса последовательностей понятий нет
Общий 10,9–8,5, в том числе 10,9–7,5	Выполнены требования 8.4.2 а)-г), и)-л); имеются ошибки лексики и синтаксиса диаграммы последовательностей

Примечание: обязательным является выполнение п. 2.1 (0,1 балла), 2.2 (0,1 балла), 2.3 (0,1 балла), 2.4 (0,2 балла) и 8.4.1 (0,5 балла); общая сумма – 1 балл.

9 Практическое занятие № 5.

Этап проектирования ИС при объектно-ориентированном подходе: моделирование поведения системы

9.1 Цель практического занятия № 5

Практическое занятие направлено на изучение основ моделирования поведения системы на этапе проектирования при объектно-ориентированном подходе к моделированию и формированию соответствующей документации в виде диаграммы сотрудничества. Выполняется студентами очной формы обучения.

9.2 Методические указания к проведению практического занятия № 5

9.2.1 Диаграмма сотрудничества [17] (*англ.* Colaboration Diagram) (диаграмма кооперации [14]–[15], диаграмма коммуникации [16]) – это диаграмма, на которой показано взаимодействие объектов внутри системы.

9.2.2 Для проведения практического занятия № 5 необходимо предварительно изучить [14]–[16].

9.2.3 Для описания времени жизни объектов во взаимодействии на диаграмме сотрудничества необходимо использовать ключевые слова, представленные в таблице 9.1 [18].

Таблица 9.1

Ключевые слова для описания времени жизни объектов

Ключевое слово	Способ использования	Описание
create	стереотип операции в сообщении	Операция создает объект, т. е. данное сообщение является вызовом конструктора
destroy	стереотип операции в сообщении	Операция уничтожает объект, т. е. данное сообщение является вызовом деструктора
destroyed	ограничение роли классификатора	Объект уничтожается в процессе описываемого взаимодействия
new	ограничение роли классификатора	Объект создается в процессе описываемого взаимодействия
transient	ограничение роли классификатора	Объект создается и уничтожается в процессе описываемого взаимодействия. Такой объект называется временным. Данное ограничение эквивалентно одновременному указанию ограничений new и destroyed

9.2.4 Для описания стереотипа полюса роли ассоциации на диаграмме сотрудничества рекомендуются значения, представленные в таблице 9.2 [13, 18].

9.2.5 Для указания того факта, что в процессе взаимодействия объект меняет свое состояние, на диаграмме сотрудничества необходимо использовать

зависимости с соответствующим стереотипом, представленные в таблице 9.3 [18]. Для зависимости можно указать номер (как у сообщения), чтобы показать, при выполнении какой именно операции происходит изменение состояния объекта [18].

9.2.6 Часто на диаграмме сотрудничества специфицируют процедурный (вложенный) поток управления (синхронные сообщения), изображая его в виде сплошной закрашенной стрелки. В общем случае, чтобы смоделировать последовательность управления шаг за шагом (асинхронные сообщения), на диаграмме кооперации необходимо использовать плоский поток управления с представлением в виде заостренной стрелки [13].

9.2.7 Для моделирования понятий-контейнеров на диаграмме сотрудничества необходимо использовать коллекции объектов или мультиобъекты.

Таблица 9.2

Стереотипы полюса роли ассоциации

Стереотип	Описание
«association» («ассоц»)	Объект на полюсе роли ассоциации связан с объектом на противоположном полюсе фактической связью, реализующей ассоциацию [18] (ссылка указывает, что соответствующий объект виден ассоциации) [13]
«global» («глоб»)	Объект на полюсе роли ассоциации имеет глобальную область определения относительно объекта на противоположном полюсе [18] (ссылка указывает, что соответствующий объект виден, поскольку он находится в объемлющей области действия) [13]
«local» («лок»)	Объект на полюсе роли ассоциации имеет локальную область определения относительно объекта на противоположном полюсе, т. е. является временным объектом для выполнения операции [18] (ссылка указывает, что соответствующий объект виден, поскольку он находится в локальной области действия) [13]
«parameter» («парам»)	Объект на полюсе роли ассоциации является параметром операции объекта на противоположном полюсе [18] (ссылка указывает, что соответствующий объект виден, потому что он является параметром) [13]
«self» («сам»)	Роль ассоциации является фиктивной связью, введенной для моделирования вызова операции данного объекта [18] (указывает, что соответствующий объект виден, поскольку он является диспетчером операции) [13]

Таблица 9.3

Стереотипы зависимостей

Стереотип	Описание
«become»	Зависимость проводится от объекта в исходном состоянии к тому же самому объекту, но в измененном состоянии, имя которого указывают в квадратных скобках
«copy»	Означает, что создается новый объект – копия исходного; после создания копия объекта существует как новый независимый объект

9.3 Пример модели поведения системы на этапе проектирования

9.3.1 Исходной является модель поведения системы ИС PAsystem, построенная в п. 8.4.

9.3.2 Диаграммы сотрудничества некоторых прецедентов PAsystem представлены на рисунках 9.1–9.3.

9.4 Домашнее задание № 5

9.4.1 Построить диаграммы сотрудничества для главных прецедентов в соответствии с рекомендациями п. 9.2.

9.4.2 Представить в диаграммах объекты, коллекции объектов/мультиобъекты и классы, установить сотрудничество между ними, отобразить взаимодействие с помощью синхронных и асинхронных сообщений, обеспечить видимость между объектами в контексте сотрудничества.

9.4.3 Для построения диаграммы сотрудничества необходимо использовать инструмент draw.io [26].

9.4.4 Срок выполнения: 9-я неделя. Срок защиты: 9-я неделя. Максимальная оценка составляет 5 баллов, зачетная – 2,25 балла. Представление отчета через элемент «Отчет Д35» курса «ОПРИС» на moodle.movs.psu.ru.

9.4.5 Содержание отчета:

- диаграммы сотрудничества для главных прецедентов,
- графический(-ие) файл(-ы) диаграмм сотрудничества.

9.5 Вопросы для самопроверки

1. Назначение диаграммы сотрудничества.
2. Отличие диаграммы сотрудничества от диаграммы последовательностей.
3. Описание времени жизни объектов во взаимодействии.
4. Способы видимости объектов во взаимодействии.
5. Описание изменения состояния объекта во взаимодействии.
6. Отличие коллекции объектов от мультиобъекта.
7. Концептуальная модель диаграммы сотрудничества.
6. Какие ошибки допущены в диаграмме сотрудничества, представленной на рисунке 9.4?

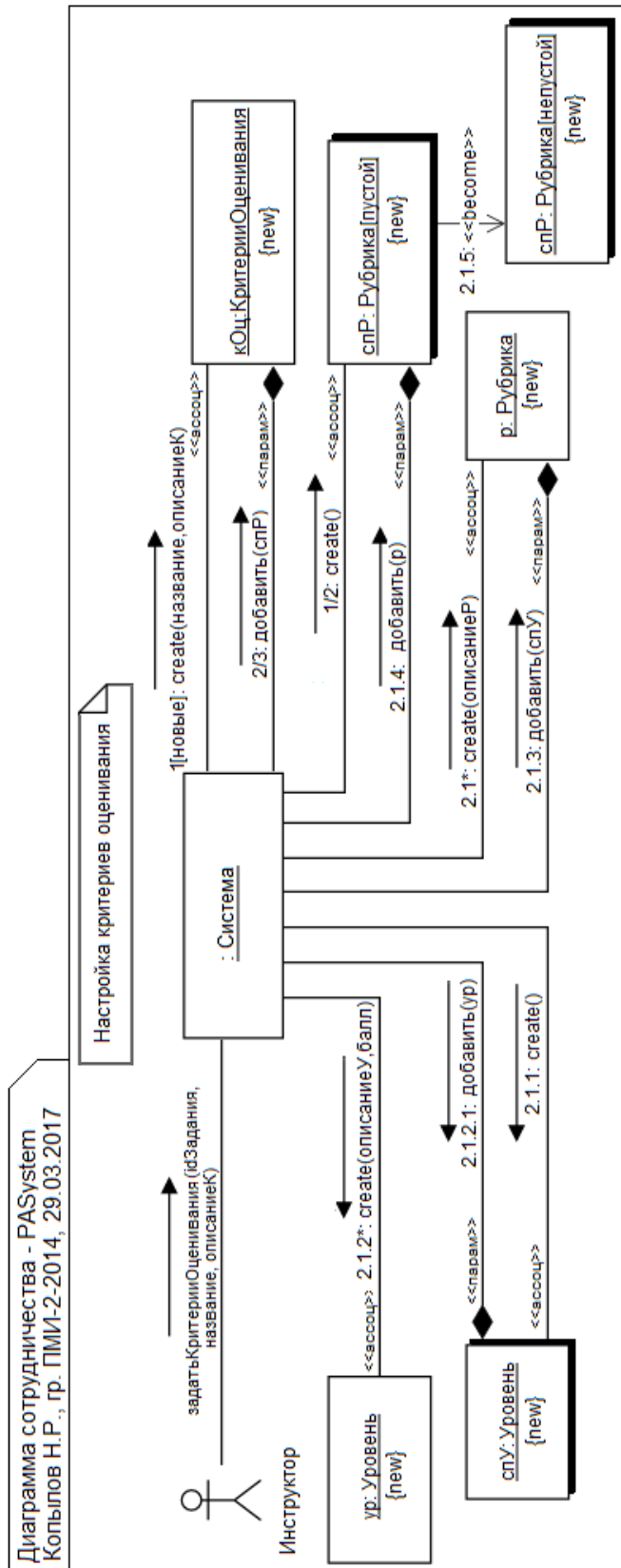


Рис. 9.1. Диаграмма сотрудничества PASystem (прецедент 2, сценарий основного потока с подпотоками)

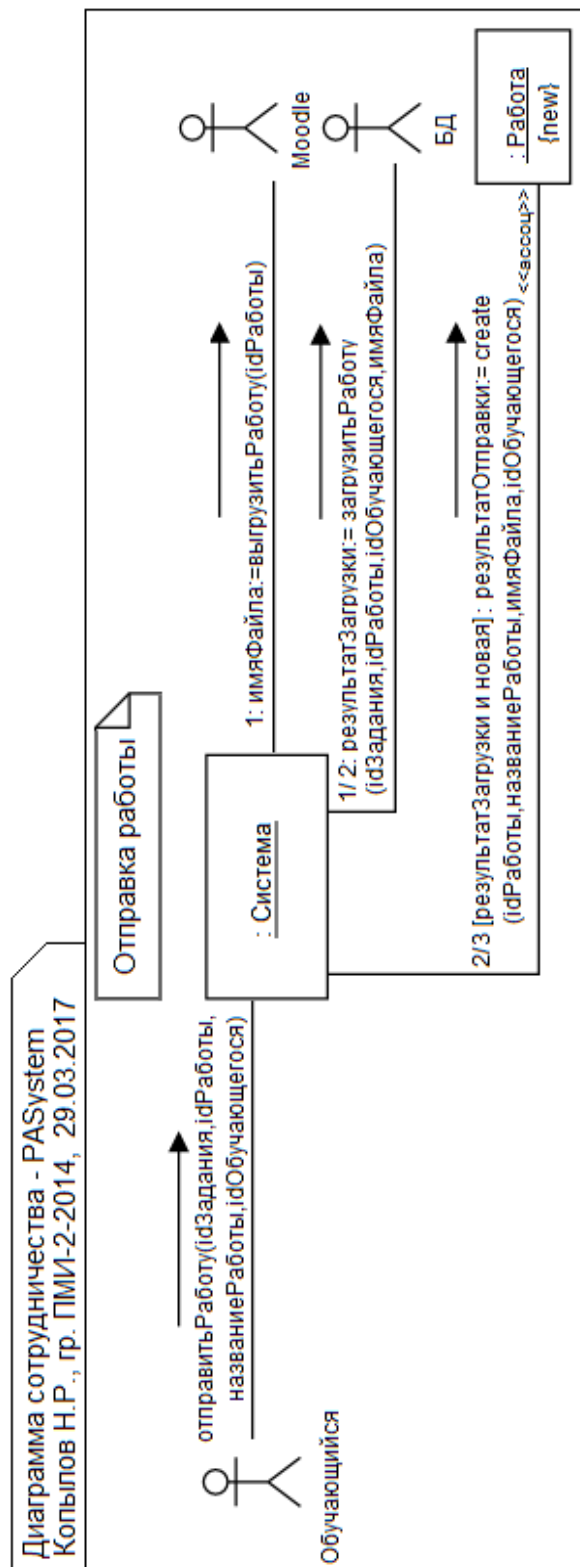


Рис. 9.2. Диаграмма сотрудничества RASystem (прецедент 3, сценарий основного потока)

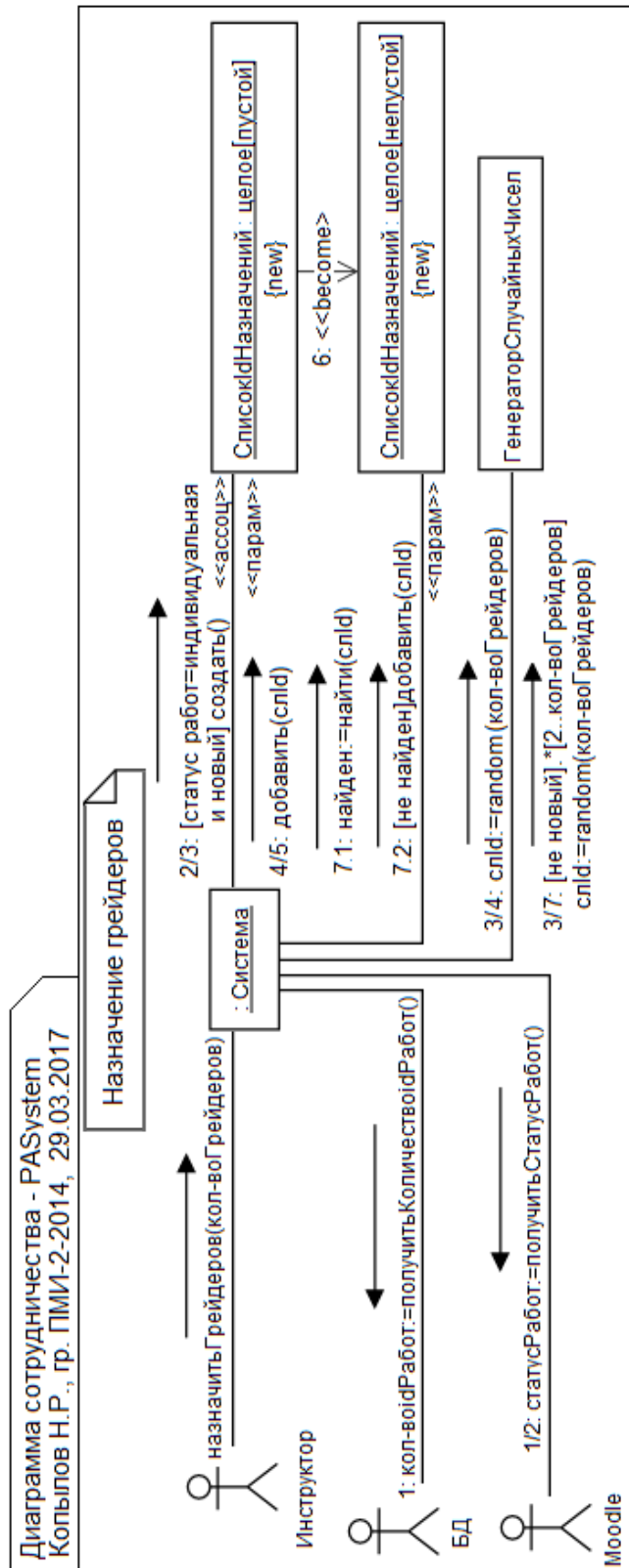


Рис. 9.3. Диаграмма сотрудничества PASystem (прецедент 4, сценарий основного потока и подпотока S3)

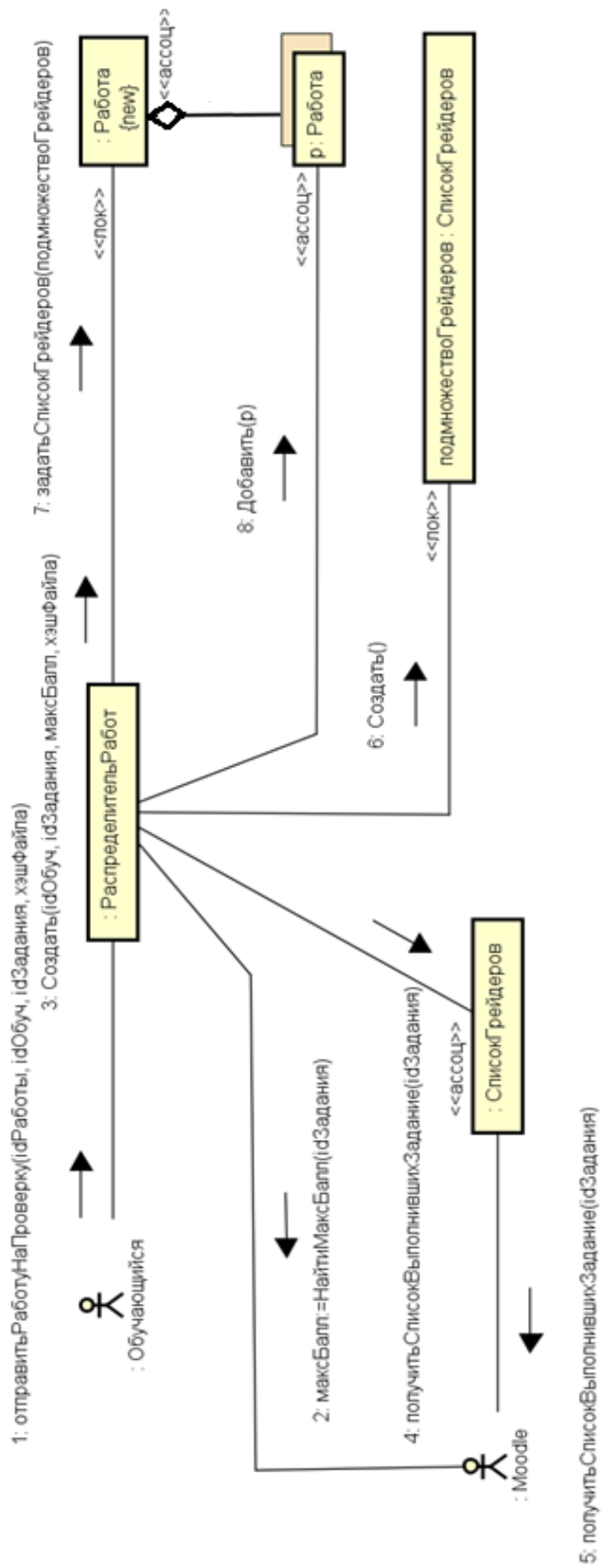


Рис. 9.4. Диаграмма сотрудничества с ошибками

10 Лабораторная работа № 3.

Этап проектирования ИС при объектно-ориентированном подходе: моделирование структуры системы

10.1 Цель лабораторной работы № 3

Целью выполнения ЛР № 3 является изучение основных элементов диаграммы классов, создание диаграммы классов, а также получение навыков работы с инструментальными средствами создания Class Diagram. Выполняется студентами очной и заочной форм обучения.

10.2 Общие методические указания к выполнению ЛР № 3

10.2.1 Диаграмма классов [11]–[18] (*англ.* Class Diagram, CD) – это графическое представление статической модели, в которой собраны декларативные (статические) элементы, такие как классы, типы, а также их содержимое и отношения. Содержит некоторые конкретные элементы поведения (напр., операции), однако их динамика в этой диаграмме не отображается.

10.2.2 Для выполнения ЛР № 3 необходимо предварительно изучить:

- стандарт ISO/IEC 19505-1:2012 [8]–[9],
- лексику, синтаксис, семантику и нотацию диаграммы классов языка UML,
- инструмент для создания диаграмм классов языка UML [5].

10.2.3 Прямое проектирование – это создание программного кода из модели [14]. При создании диаграммы классов учитывается синтаксис целевого языка ООП, в который может быть выполнена с помощью CASE-средств генерация программы.

10.2.4 Зависимости в диаграмме классов отображают способы видимости классов, отличные от видимости посредством ассоциации [18]. Зависимость Вызов (<<call>>) означает, что метод одного класса вызывает операцию другого класса. Наиболее общей зависимостью является Использование (<<use>>), которая означает, что одному элементу для правильного функционирования необходимы услуги другого элемента (сюда входят вызов, конкретизация, параметр, отправка, другие зависимости) [13].

10.3 Сроки выполнения и представление отчета по ЛР № 3

Срок выполнения студентами очной формы обучения: 10-я неделя. Срок защиты: 11-я неделя. Представление отчета через элемент «Отчет ЛР3» курса «ОПРИС» на moodle.movs.psu.ru.

Представление отчета студентами заочной формы обучения через элемент «Отчет ЛР3» курса «ОПРИС-3» на moodle.movs.psu.ru.

10.4 Пример выполнения ЛР № 3

10.4.1 Выделение классов. Далее в тексте этого раздела имена классов выделены шрифтом Courier New. Выделение классов производим на основе:

а) диаграмм сотрудничества, подготовленных в п. 9.3:

- 1) PAsystem (Система);
- 2) КритерииОценивания;
- 3) Рубрика;
- 4) Уровень;
- 5) Работа;
- 6) СписокИдНазначений;

б) диаграммы понятий, подготовленной в п. 7.3.11:

- 1) СписокРубрик;
- 2) СписокУровней;
- 3) Назначение;
- 4) Оценка;
- 5) ИтоговаяОценка;

в) рекомендуемых шаблонов проектирования GRAPS [17, 27]:

- 1) СервисСозданияКрОцен;
- 2) СервисРабот;
- 3) СервисВычИтогОценки.

Абстрактного класса в диаграмме классов нет.

10.4.2 Диаграмма классов PAsystem представлена на рисунке 10.1.

10.4.3 Распределение обязанностей между некоторыми классами в соответствии с шаблонами проектирования GRAPS:

а) классам СервисВычИтогОценки, *Итоговая Оценка*, *Оценка* вменяется обязанность знать итоговую оценку обучающегося (шаблон «Expert»),

б) классам СписокУровней, Рубрика, СписокРубрик, КритерииОценивания, СервисРабот, СервисВычИтогОценки вменяется обязанность создавать объекты классов Уровень, СписокУровней, Рубрика, СписокРубрик, СписокИдНазначений, Оценка, ИтоговаяОценка соответственно, т.к. между ними установлено отношение агрегации (шаблоны «Creator» и «Low Coupling»),

в) классы СервисСозданияКрОцен, СервисРабот, СервисВычИтогОценки были дополнительно введены, чтобы разгрузить класс PAsystem, распределив между этими сервисами обязанности по функциональному принципу (шаблон «High Cohesion»),

г) классам PAsystem, СервисСозданияКрОцен, СервисРабот, СервисВычИтогОценки вменяется обязанность отвечать за обработку системных сообщений (шаблон «Controller»).

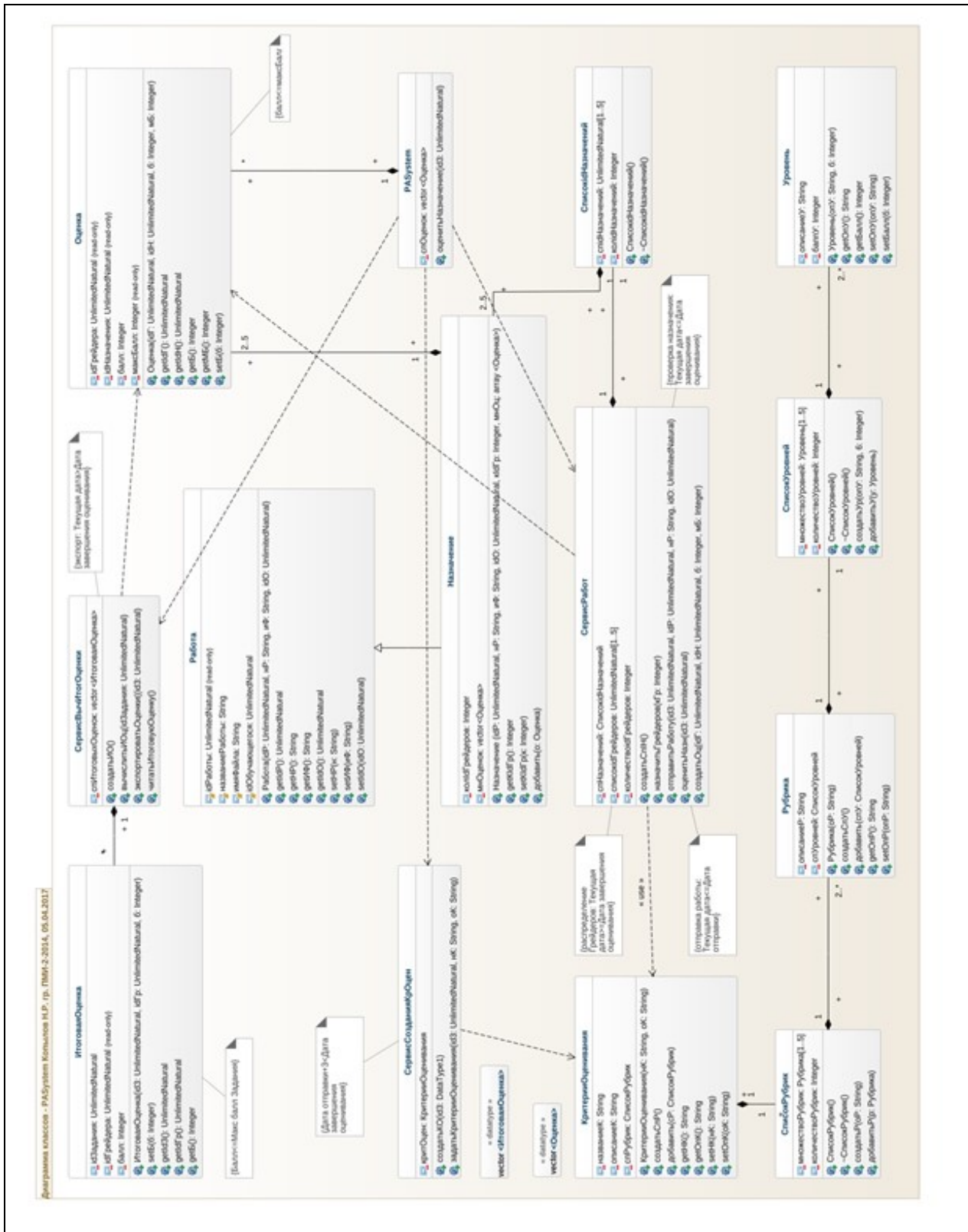


Рис. 10.1. Диаграмма классов PASystem

10.4.4 Атрибуты классов, информация о типах данных и об области видимости.

Необходимо предварительно выбрать целевой язык ООП для прямого проектирования. Система PAsystem будет реализована на языке C++. Поэтому для целочисленных данных в диаграмме классов использован тип «int», для строковых – «char *», для обозначения пустого значения – «void».

Атрибуты классов п. 10.4.1 а) и б) совпадают с атрибутами одноименных понятий концептуальной модели. Атрибутами классов п. 10.4.1 в) являются:

- а) критОцен для класса СервисСозданияКрОцен;
- б) спНазначений и списокIdГрейдеров для класса СервисРабот;
- в) спИтоговыхОценок для класса СервисВычИтогОценки.

Атрибуты классов-контейнеров, предназначенные для хранения элементов других классов, моделируются двумя способами:

- 1) массивами языка C++ (классы СписокУровней, СписокРубрик, СервисРабот, СписокidНазначений,
- 2) последовательными контейнерами C++ (array <тип>) (классы PAsystem, Назначение, СервисВычИтогОценки).

В первом случае кроме атрибута-массива классы имеют целочисленный атрибут для хранения текущего значения количества элементов в нем.

В UML по умолчанию атрибут имеет область видимости «+» (public). Атрибуты всех классов, кроме класса Работа, размещены в части класса private (область видимости «-»). Атрибуты класса Работа, который является родительским по отношению к классу Назначение, размещены в части класса protected (область видимости «#»). Это позволяет реализовать отношение обобщения средствами механизма наследования классов в целевом языке ООП.

Вычисляемых атрибутов в диаграмме классов нет.

10.4.5 Дополнительная информация об атрибутах классов [16].

Множественность не является обязательной и записывается после типа данных в виде диапазона целочисленных значений, напр., «ФИО: char *» или «ФИО: char [1..30]».

У атрибутов могут быть указаны свойства:

- readOnly (значение атрибута не может быть изменено),
- ordered (у атрибута с множественностью, большей единицы, значения упорядочены),
- unique (у атрибута с множественностью, большей единицы, дубликаты значений запрещены).

По умолчанию для атрибута установлено свойство «changeable», свидетельствующее об отсутствии ограничений на модификацию значения.

10.4.6 Операции классов.

Операции-действия классов п. 10.4.1 а) соответствуют сообщениям диаграмм сотрудничества. Операции-модификаторы, имеющие в имени префикс «get», указаны для всех атрибутов классов. Операции-селекторы, имеющие в имени префикс «set», указаны только для тех атрибутов классов, которые будут изменяться после установки начальных значений конструктором этого класса.

Операции-действия классов п. 10.4.1 в) соответствуют п. 10.4.3.

Большинство классов п. 10.4.1 а) и б) имеют параметризованные конструкторы. Классы СписокРубрик и СписокУровней имеют конструктор по умолчанию. Так как класс Назначение является производным от класса Работа, его конструктор вызывает явно конструктор базового класса для инициализации унаследованных атрибутов. Для классов п. 10.4.1 в) конструкторы не указаны.

Деструкторы в диаграмме классов указаны явно только для классов СписокidНазначений, СписокРубрик и СписокУровней, у которых такие деструкторы должны явно освобождать память элементов массива соответствующего класса-контейнера. У остальных классов деструкторы не указаны, так как предполагается наличие у каждого класса деструктора по умолчанию.

Операции всех классов размещены в части класса public (область видимости «+»). Это позволяет использовать их другим классам.

10.4.7 Дополнительная информация об аргументах операций [16].

Направление аргумента операции указывает, что аргумент является входным («in»), выходным («out») или обновляемым («in/out»). В модели системы PASystem все операции имеют только входные аргументы.

10.4.8 Свойства операций. У операций могут быть указаны свойства [16]:

- isLeaf (не может быть полиморфной / переопределяться при наследовании),
- isQuery (ее выполнение не изменяет состояния объекта).

Свойство «isQuery» может установлено для get-операций классов.

10.4.9 Отношения между классами. В диаграмме классов присутствуют отношения обобщения, агрегации, ассоциации и зависимости.

10.4.10 Обобщение между понятиями «Работа» и «Назначение» выделено в п. 7.3.7 и сохраняется между одноименными классами в диаграмме классов.

10.4.11 Композитная агрегация между понятиями, выделенная в п. 7.3.4 ж)–м), и кратности ролей сохраняются между одноименными классами в диаграмме классов. Дополнительно это отношение установлено между такими парами классов:

- а) СервисРабот – СписокidНазначений (кратности ролей 1:1),
- б) СервисВычиТоГОценки – ИтоговаяОценка (кратности ролей 1:0..*),
- в) PASystem – Оценка (кратности ролей 1:0..*).

10.4.12 Ассоциации связывают классы п. 10.4.1 в) с остальными элементами диаграммы классов (кратности ролей 1:1):

- а) PASystem – СервисСозданияКроцен,
- б) PASystem – СервисРабот,
- в) PASystem – СервисВычИтогОценки,
- г) СервисСозданияКроцен – КритерииОценивания,
- д) СервисРабот – Оценка,
- е) СервисВычИтогОценки – Оценка.

10.4.13 Зависимость Использование установлена между классами СервисРабот и КритерииОценивания, т.к. при оценивании работ обучающихся Грейдеры используют критерии оценивания задания.

10.4.14 Направление навигации указано явно между классами п. 10.4.12.

10.4.15 Ограничения совпадают с выявленными в п. 7.3.6.

10.4.16 При использовании целевого языка ООП, который не поддерживает локализацию имен переменных / констант / классов / объектов и т. п., следует использовать общие рекомендации для имен:

- имена классов, следовательно, и имена конструкторов / деструкторов, начинать с латинского символа «С», напр., CRubric, ~CRubricList,
- имена аргументов, имеющих направление «in» и инициализирующих значения атрибутов, начинать с префикса «i» («initial») или «a» (начальная буква алфавита), напр., «+setFN (iFileName: char *): void» или «+setScore (aScore: int): void».

10.4.17 При использовании инструмента [5] следует учитывать такие ограничения:

- имена базовых типов данных фиксированные и необязательно совпадают с набором типов данных в целевом языке ООП,
- для создания пользовательских типов данных и структур данных необходимо создавать новые типы данных с помощью «Data Type»,
- нет полного соответствия между видом модели в среде редактора и экспортированном png-файле (напр., после удаления имен ролей агрегации; при указании множественности атрибутов классов; при указании отсутствия значения, возвращаемого операцией, и т.п.),
- отсутствует возможность графической идентификации вычислимого атрибута (хотя установить это значение можно в свойстве «Derived»),
- часть дополнительной информации об аргументах операций, заданной средствами редактора модели, присутствует в модели, но не отображается на диаграмме (кроме свойств «readOnly» и «static»),
- для операций предусмотрено их размещение по умолчанию в открытой части класса, при необходимости другую часть класса следует указать для операции в ее свойстве «Visibility»,

– часть дополнительной информации об аргументах операций, заданной средствами редактора модели, присутствует в модели, но не отображается на диаграмме (кроме множественности),

– у аргументов операций не предусмотрена возможность задания значений по умолчанию,

– часть свойств операций, заданных средствами редактора модели, присутствует в модели, но не отображается на диаграмме (кроме свойств «readOnly» и «static»),

– использование объекта диаграмм «Comment» для указания ограничений позволяет связать их с помощью объекта «Comment link» с классами или с ролями ассоциаций.

10.4.18 Список типичных ошибок, которые допускают студенты при создании CD, приведен в приложении Д.

10.5 Пример описания некоторых классов

10.5.1 Класс СервисСозданияКрОцен. Предназначен для того, чтобы инструктор мог создавать критерии оценивания задания. Является контроллером прецедента «Настройка критериев оценивания». Является создателем объектов класса КритерииОценивания.

Имеет атрибут «критОцен» (объект класса КритерииОценивания).

Имеет операции:

– «создатьКО (id3: int) : void» (создает критерии оценивания задания с идентификатором id3),

– «здатьКритерииОценивания (id3: int, нК: char *, оК: char *): void» (здает название и описание критерия оценивания задания с идентификатором id3).

10.5.2 Класс КритерииОценивания. Предназначен для создания критерия оценивания задания и его параметров. Является контейнером и создателем объектов класса СписокРубрик.

Имеет атрибуты:

– «названиеК» (строка названия критерия оценивания),

– «описаниеК» (строка описания критерия оценивания),

– «спРубрик» (объект класса СписокРубрик).

Имеет операции:

– «КритерииОценивания (нК: char *, оК: char *)» (параметризованный конструктор, срабатывающий при создании объекта этого класса и устанавливающий начальные значения атрибутов «названиеК» и «описаниеК»),

– «создатьСпР (): void» (создает объект класса СписокРубрик),

- «добавить (сР: СписокРубрик) :void» (добавляет/обновляет значение атрибута «СпРубрик),
- «getНК (): char *» (возвращает значение атрибута «названиеК»),
- «getОпК (): char *» (возвращает значение атрибута «описаниеК»),
- «setНК (нк: char *): void» (устанавливает значение атрибута «названиеК»),
- «setОпК (ок: char *): void» (устанавливает значение атрибута «описаниеК»).

10.5.3 Класс СписокРубрик. Предназначен для создания списка рубрик критерия оценивания и его параметра. Является контейнером и создателем объектов класса *Рубрика*.

Имеет атрибуты:

- «множествоРубрик» (массив объектов класса *Рубрика*),
- «количествоРубрик» (текущее количество объектов атрибута «множествоРубрик»).

Имеет операции:

- «СписокРубрик ()» (конструктор по умолчанию, срабатывающий при создании объекта этого класса и устанавливающий пустое значение атрибута «множествоРубрик» и нулевое значение атрибута «количествоРубрик»),
- «~СписокРубрик ()» (деструктор, который уничтожает все объекты, содержащиеся в атрибуте «множествоРубрик», освобождая занятую ими память, и удаляет сам объект класса СписокРубрик),
- «создатьР (о: char *): void» (создает объект класса *Рубрика* с указанным начальным значением его атрибута «описаниеР»),
- «добавить (Р: Рубрика): void» (добавляет объект *Р* класса «*Рубрика*» в атрибут «множествоРубрик» и увеличивает значение атрибута «количествоРубрик»).

10.5.4 Класс *Рубрика*. Предназначен для создания рубрики критерия оценивания задания и ее параметров. Является контейнером и создателем объектов класса *СписокУровней*.

Имеет атрибуты:

- «описаниеР» (строка описания рубрики оценивания),
- «спУровней» (объект класса *СписокУровней*).

Имеет операции:

- *Рубрика* (оР: char *) (параметризованный конструктор, срабатывающий при создании объекта этого класса и устанавливающий начальное значение атрибута «ОписаниеР»),
- создатьСпУ (): void (создает объект класса *СписокУровней*),
- добавитьСпУ (спУ: СписокУровней) (добавляет/обновляет значение атрибута «СпУровней»),

- getOP (): char * (возвращает значение атрибута «описаниеР»),
- setOpУ (опР: char *): void (устанавливает значение атрибута «описаниеР»).

10.6 Указания и задания к ЛР № 3

10.6.1 Построить диаграмму классов (5–7 классов), описывающую реализуемую систему.

10.6.2 Использовать различные типы отношений между классами (ассоциация, агрегация, композиция, обобщение, зависимости). Отобразить роли и кратности ассоциаций, абстрактный класс, операции, области видимости, вычисляемые атрибуты, направления навигации, зависимости, ограничения.

10.6.3 Выбрать язык объектно-ориентированного программирования для прямого проектирования (целевой язык). Использовать в диаграмме классов типы данных выбранного языка. Проиллюстрировать использование дополнительной информации об атрибутах класса, аргументах операций и свойств операций, указав их полный набор в соответствующих таблицах отчета.

10.6.4 Определить классы на целевом языке ООП. Дать комментарии к диаграмме классов в виде краткого описания классов, их атрибутов и операций. Аргументировать выбор структур данных.

10.7 Содержание отчета по ЛР № 3

Содержание отчета по ЛР № 3:

- диаграмма классов,
- комментарии к диаграмме классов,
- определение классов на целевом языке ООП.

10.8 Оценивание ЛР № 3

Оценивание выполнения ЛР № 3 приведено в таблицах 10.1–10.2. Максимальная оценка составляет у студентов очной формы обучения 5 баллов, зачетная – 2 балла. Максимальная оценка составляет у студентов заочной формы обучения 15 баллов, зачетная – 6,5 баллов.

Таблица 10.1

Оценивание выполнения ЛР № 3 (очная форма обучения)

Баллы	Выполнение требований к ЛР № 3
Общий 5, в том числе 4	Выполнены требования 10.6.1–10.6.4; ошибок лексики и синтаксиса диаграммы классов нет
Общий 4,9–4, в том числе 3,9–3	Выполнены требования 10.6.1–10.6.3; имеются ошибки лексики и синтаксиса диаграммы классов
Общий 3,9–3, в том числе 2,9–2	Выполнены требования 10.6.1–10.6.2; ошибок лексики и синтаксиса классов понятий нет

Общий 2,9–2, в том числе 1,9–1	Выполнены требования 10.6.1–10.6.2; имеются ошибки лексики и синтаксиса диаграммы классов
-----------------------------------	--

Таблица 10.2

Оценивание выполнения ЛР № 3 (заочная форма обучения)

Баллы	Выполнение требований к ЛР № 3
Общий 15, в том числе 14	Выполнены требования 10.6.1–10.6.4; ошибок лексики и синтаксиса диаграммы классов нет
Общий 14,9–12, в том числе 13,9– 11,9	Выполнены требования 10.6.1–10.6.3; имеются ошибки лексики и синтаксиса диаграммы классов
Общий 11,9–9, в том числе 10,9– 8,9	Выполнены требования 10.6.1–10.6.2; ошибок лексики и синтаксиса классов понятий нет
Общий 8,9–6,5, в том числе 7,9–5,5	Выполнены требования 10.6.1–10.6.2; имеются ошибки лексики и синтаксиса диаграммы классов

Примечание: обязательным является выполнение п. 2.1 (0,1 балла), 2.2 (0,5 балла) и 2.3 (0,4 балла); общая сумма – 1 балл.

11 Лабораторная работа № 4.

Этап реализации ИС при объектно-ориентированном подходе

11.1 Цель лабораторной работы № 4

Целью выполнения ЛР № 4 является изучение основных элементов диаграммы компонентов, создание диаграммы компонентов, а также получение навыков работы с инструментальными средствами создания Component Diagram. Выполняется студентами очной и заочной форм обучения.

Целью выполнения ЛР № 4 студентами очной формы обучения также является изучение основных элементов диаграммы развертывания, создание диаграммы развертывания, а также получение навыков работы с инструментальными средствами создания Deployment Diagram.

11.2 Общие методические указания к выполнению ЛР № 4

11.2.1 Диаграмма компонентов [11]–[18] (*англ.* Component Diagram) – это диаграмма, на которой изображены организация типов компонентов и зависимости между ними. Эта диаграмма показывает структуру и возможные способы доставки сообщений компонента. Последовательность сообщений в компоненте она не отражает.

Существует два вида таких диаграмм [12]:

1) набор (множество, «библиотека») компонентов и их зависимости, представляет собой детали, из которых можно сконструировать ИС,

2) сконфигурированная система и те компоненты, которые были использованы для ее построения (выбранные из всего набора компонентов): каждый компонент связан с другими компонентами, чьими услугами он пользуется, все связи должны быть полностью согласованы с точки зрения соответствия интерфейсов компонент.

11.2.2 Для выполнения ЛР № 4 необходимо предварительно изучить:

– стандарт ISO/IEC 19505-1:2012 [8]–[9],

– лексику, синтаксис, семантику и нотацию диаграммы компонентов языка UML,

– инструмент для создания диаграмм компонентов языка UML [5].

Студентам очной формы обучения необходимо предварительно изучить:

– лексику, синтаксис, семантику и нотацию диаграммы развертывания языка UML,

– инструмент для создания диаграмм развертывания языка UML [5].

11.2.3 При создании моделей ИС и реализации ее ПО следует придерживаться разделения интерфейса и реализации. Интерфейс определяет соглашение, а реализация представляет собой его конкретное воплощение и обязуется

точно следовать полной семантике интерфейса [13]. Средства операционных систем, основанные на компонентах (напр., COM+ , CORBA и Enterprise Java Beans) используют интерфейсы в качестве элементов, связывающих компоненты друг с другом [13]. Язык UML позволяет моделировать как интерфейсы, так и их реализации.

11.2.4 Компонент. Представляет собой часть системы, редко используемую самостоятельно. Соответствует набору интерфейсов и обеспечивает его реализацию, что позволяет его замещать (можно заменять другим компонентом, который соответствует тем же интерфейсам).

Компонент может быть построен из других. Допускается организация компонентов путем установления между ними связей зависимости, ассоциации, обобщения, агрегации. Компонент может быть использован повторно.

11.2.5 Связи между компонентами. Между компонентами могут быть установлены связи ассоциации, обобщения, агрегации и зависимости.

11.2.6 Интерфейс. Представляет собой набор операций, используемый для спецификации сервиса класса или компонента.

Интерфейс, который реализован компонентом, называется предоставляемым (данный компонент предоставляет интерфейс в виде сервиса другим компонентам). Компонент может декларировать множество предоставляемых интерфейсов.

Интерфейс, который компонент использует, называется требуемым: ему соответствует данный компонент, когда запрашивает сервисы от других компонентов.

Имя интерфейса рекомендуется начинать с заглавной буквы «I».

Компонент может соответствовать множеству требуемых интерфейсов. Компонент может одновременно и предоставлять, и требовать интерфейсы.

11.2.7 Связь между компонентом и его интерфейсами выражается одним из двух способов:

1) интерфейс изображается в пиктографической форме (предоставляемый интерфейс выглядит как «шарик», требуемый интерфейс – как «гнездо», оба соединены линией с компонентом; этот способ употребляется чаще),

2) интерфейс изображается в расширенной форме в виде прямоугольника с секциями как понятие или класс.

11.2.8 Порты. Представляют собой часть компонента. Используются для более полного контроля над реализацией в противоположность интерфейсам, которые удобны для описания общего поведения компонента. Если компонент имеет порты, то взаимодействие с ним на входе и на выходе происходит через них. Выражаемое внешне поведение такого компонента представляет собой сумму его портов. Порт наделен уникальностью: один компонент может взаимодействовать с другим через определенный порт.

На диаграмме компонентов порт представляется квадратом на боковой грани компонента. Как предоставляемый, так и требуемый интерфейс может быть соединен с символом порта. Предоставляемый интерфейс изображает сервис, который может быть запрошен извне через данный порт, а требуемый интерфейс – сервис, который порт должен получить от какого-либо другого компонента. У каждого порта есть имя, он может быть идентифицирован по компоненту и имени. Имя компонента вместе с именем порта идентифицирует порт для использования его другими компонентами [12].

11.2.9 Внутренняя структура. Компонент может быть реализован как единый фрагмент кода, но в больших системах желательно иметь возможность строить крупные компоненты из мелких. Внутренняя структура компонента содержит части, которые вместе с соединениями между ними составляют его реализацию. Часть – это единица реализации компонента, которой присвоены имя и тип, напр., частями компонента «Компилятор» являются «Лексический анализатор», «Синтаксический анализатор», «Генератор объектного кода» и «Оптимизатор кода». Каждая часть идентифицируется по ее имени.

Часть имеет множественность в пределах компонента: в экземпляре компонента содержится один или по несколько экземпляров каждой части определенного типа, напр., в компиляторе таких частей как «Оптимизатор кода» может быть больше одной.

Следует помнить, что часть – это не то же самое, что класс, напр., полная версия компилятора может быть сконфигурирована с разными уровнями оптимизации, а необходимый оптимизатор кода может выбираться во время исполнения.

11.2.10 Коннектор. Представляет собой связь коммуникации между двумя частями или портами в контексте компонента компонентами [12]. В экземпляре охватывающего компонента коннектор может быть:

- ссылкой (link), которая является экземпляром обычной ассоциации,
- временной ссылкой (transient link), которая является связью использования между двумя компонентами; вместо обычной ассоциации она может быть обеспечена параметром процедуры или локальной переменной, которая служит результатом операции.

Коннекторы изображаются одним из двух способов:

- 1) линией, если два компонента явно связаны друг с другом (либо напрямую, либо через порты),
- 2) интерфейсом в «шарик» – «гнездо», если два компонента подключены друг к другу, потому что имеют совместимые интерфейсы.

Можно связать внутренние порты с внешними портами объемлющего компонента. В этом случае коннектор является делегирующим, поскольку сообщения из внешнего порта делегируются внутреннему. Эта связь изображает-

ся стрелкой, направленной от внутреннего порта к внешнему компонентами [12].

11.2.11 Стереотип. Представляет собой новый вид элемента модели, который определяется на основе уже существующего элемента. По содержанию и форме совпадает с этим существующим элементом, но может иметь иное значение и использоваться по-другому [12]. Напр., использование стереотипа <<database>> позволяет не указывать интерфейсы, реализуемые этим компонентом, т.к. они поддерживаются всеми базами данных. Список predefined стереотипов компонентов языка UML представлен в таблице 11.1 [12].

Стереотип является расширением языка UML и, в общем случае, не поддерживается инструментами моделирования при прямом проектировании.

Таблица 11.1

Предопределенные стереотипы компонентов языка UML

Ключевое слово	Описание
<<document>>	Стереотипизированный компонент, представляющий собой документ
<<entity>>	Класс-сущность содержит информацию, которая должна храниться постоянно и не уничтожается с уничтожением объектов данного класса или прекращением работы моделируемой системы, связанные с выключением системы или завершением программы. Как правило, этот класс соответствует отдельной таблице базы данных. В этом случае его атрибуты являются полями таблицы, а операции – присоединенными или хранимыми процедурами. Этот класс пассивный и лишь принимает сообщения от других классов модели
<<executable>>	Стереотипизированный компонент, указывающий на программу, которая может выполняться на узле
<<file>>	Стереотипизированный компонент, представляющий собой документ, в котором содержится исходный код программы или данные
<<library>>	Стереотипизированный компонент, представляющий собой статическую или динамическую библиотеку
<<source>>	Разновидность компонента-файла, представляющего собой файл с исходным текстом программы, который после компиляции может быть преобразован в исполнимый файл
<<table>>	Стереотипизированный компонент, представляющий собой таблицу базы данных

11.2.12 Диаграмма развертывания [15]–[17] (*англ.* Deployment Diagram) (диаграмма размещения [12]–[14]) – это диаграмма, на которой описывается конфигурация для работающих узлов и экземпляров компонентов, а также объектов, которые на них существуют, соединенных маршрутами коммуникационных ассоциаций.

Существует два вида таких диаграмм [12]:

1) описательная диаграмма развертывания, на которой изображены типы узлов системы и те компоненты, которые на них содержатся,

2) диаграмма развертывания в реальном мире, на которой показаны отдельные узлы и их связи в конкретной версии системы.

11.2.13 Узел. Представляет собой вычислительный ресурс (физический объект), используемый во время выполнения программы ИС, обладающий, по крайней мере, памятью, а зачастую и процессором. К узлам относятся не только вычислительные устройства, но и человеческие ресурсы или ресурсы механической обработки данных [12]. Для узла можно задать атрибуты и операции, напр., указать атрибуты «скорость процессора» и «объем памяти», а также операции «включить», «выключить», «приостановить» [13].

11.2.14 Связи между узлами. Между узлами могут быть установлены связи ассоциации, обобщения, агрегации и зависимости. Ассоциация между двумя узлами указывает на существование между ними коммуникационных маршрутов. У ассоциации может быть стереотип, который показывает сущность маршрута коммуникаций (напр., вид канала или сети) [12]. Ассоциации можно использовать для моделирования не прямых соединений типа спутниковой линии связи между двумя удаленными процессорами [13].

11.2.15 Артефакт (компонент). Представляет собой материализацию множества логических элементов, таких как классы и кооперации. Узлы исполняют артефакты, артефакты работают на узлах. Выделяют три вида артефактов [13]: артефакты размещения (необходимы и достаточны для формирования исполняемой системы, напр., DLL- или EXE-файлы), артефакты рабочих продуктов (результаты процесса разработки, напр., файлы исходного кода и файлы данных, из которых создаются артефакты размещения), артефакты исполнения (создаются в результате работы системы (напр., объект .NET, который создается из DLL). Класс может быть реализован одним или несколькими артефактами, а артефакт, в свою очередь, размещен в одном или нескольких узлах [13]. Внутри символов узлов могут находиться экземпляры компонентов. Это указывает на то, что компонент существует или выполняется на данном узле. Символы компонентов, в свою очередь, могут содержать в себе символы объектов. Это обозначает, что объект является частью компонента [12]. Компоненты представляют собой выполнимые блоки программного кода. Компоненты, которые не существуют как единицы времени выполнения, на диаграмме развертывания не указываются [12].

11.2.16 Зависимости. Компоненты связаны между собой пунктирными стрелками зависимостей, напр., через интерфейсы. Такие связи обозначают, что один компонент использует услуги другого. Когда на диаграмме нужно точно указать вид зависимости, используют стереотип [12].

11.2.17 Миграция компонентов. Миграцию компонентов между узлами или объектов между компонентами можно изобразить с помощью стереотипа «become» зависимости. В этом случае компонент или объект является резидентом на своем узле или в компоненте только в течение определенного периода времени [12].

11.3 Сроки выполнения и представление отчета по ЛР № 4

Срок выполнения студентами очной формы обучения: 11-я неделя. Срок защиты: 12-я неделя. Представление отчета через элемент «Отчет ЛР4» курса «ОПРИС» на moodle.movs.psu.ru.

Представление отчета студентами заочной формы обучения через элемент «Отчет ЛР4» курса «ОПРИС-3» на moodle.movs.psu.ru.

11.4 Пример выполнения ЛР № 4 (диаграмма компонентов)

11.4.1 Исходной является модель структуры системы ИС PAsystem, построенная в п. 10.4.

11.4.2 Декомпозиция ИС. В результате проектирования статической структуры ИС PAsystem были выделены части системы:

- а) система парного оценивания,
- б) сервис создания критериев оценивания,
- в) сервис по отправке работ обучающихся и назначения им грейдеров,
- г) сервис вычисления итоговой оценки.

11.4.3 Специфицируем интерфейсы, которые представляют основные соединения части п. 11.4.2, а:

- ICrservInterface – с частью п. 11.4.2, б,
- IAssservInterface – с частью п. 11.4.2, в,
- IFinAssInterface – с частью п. 11.4.2, г.

11.4.4 Определяем соответствующие компоненты, реализующие указанные интерфейсы:

- а) CreateEvCrService,
- б) AssignmentService,
- в) CalcFinAssService,
- г) PAsystem.

На стороне компонента PAsystem определены требуемые интерфейсы, на стороне остальных компонент – предоставляемые.

11.4.5 Все компоненты п. 11.4.4 имеют стереотип «executable», так как любой транслятор целевого языка ООП С++ является компилятором и создает исполнимый код. Эти компоненты зависят от:

- компонента исходного файла интерфейса класса (*.hpp),
- компонента исходного файла реализации класса (*.cpp).

11.4.6 Внутренняя структура и зависимости. Компоненты исходных файлов реализации классов (п. 11.4.4, а–в) зависят от компонентов, реализующих классы-части и производные классы соответственно.

Компонент PASystem также зависит от компонента БД.

11.4.7 Связи обобщения и агрегации между компонентами соответствуют указанным отношениям между классами ИС.

11.4.8 Порты и коннекторы компонентов не используются.

11.4.9 Диаграмма компонентов ИС PASystem приведена на рисунке 11.1.

11.4.10 На рисунке 11.2 представлена диаграмма компонентов системы парного оценивания, интегрированной с LMS Moodle, разработанная с учетом применения подхода DDD к созданию архитектуры системы. Таблицы БД ИС совпадают с частью таблиц LMS Moodle. В результате PASystem представляет собой три программных модуля:

- 1) Domain.dll,
- 2) InfrastructureDb.dll,
- 3) WebApp.

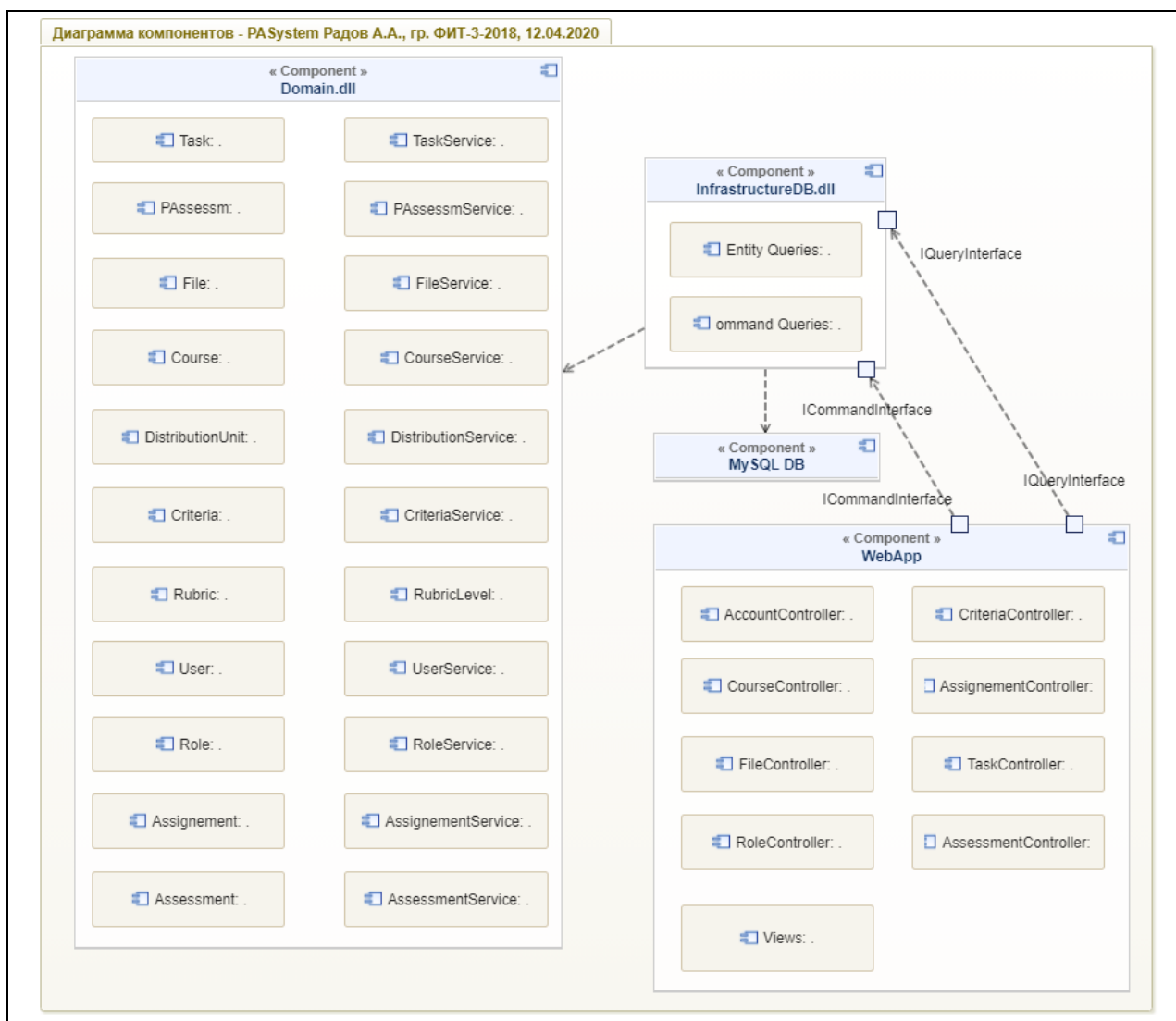


Рис. 11.2. Диаграмма компонентов PASystem (подход DDD)

11.4.11 Модуль Domain.dll представляет собой библиотеку классов, в которой хранится структура всей предметной области, все сущности, отношения между ними. Кроме того, там хранятся сервисы, которые обеспечивают операции над сущностями: добавление, чтение, удаление из БД, а также различные специфические запросы к БД (напр., выгрузить по идентификатору пользователя задания или выгрузить оценки, выставленные этим пользователем). Данный модуль является ядром приложения. Все остальные модули системы обращаются к нему. Он не зависит от других модулей и является независимой от фреймворка частью программы, поэтому при смене СУБД не требуется вносить никаких изменений в данный модуль.

Модуль InfrastructureDb.dll представляет собой библиотеку классов, которая отвечает за взаимодействие с БД. Хранит все запросы к БД. Эта часть зависит от используемой БД, поэтому при смене СУБД придется изменять запросы для другой СУБД. Взаимодействие с данным модулем и веб-частью разрабаты-

ваемого приложения происходит через интерфейсы ICommandBuilder и IQueryBuilder. Данные интерфейсы предназначены для создания команд и запросов к БД соответственно.

Модуль WebApp является веб-приложением. Он отвечает за отображение интерфейса и логику, связанную с отображением интерфейса. Этот модуль содержит шаблоны веб-страниц и контроллеры, которые обращаются к InfastructureDb.dll и Domain.dll для выгрузки сущностей из БД и использование различных сервисов соответственно.

11.5 Пример выполнения ЛР № 4 (диаграмма развертывания)

11.5.1 Исходной является модель структуры системы ИС PAsystem, построенная в п. 11.4.

11.5.2 Диаграмма развертывания ИС PAsystem приведена на рисунке 11.3.

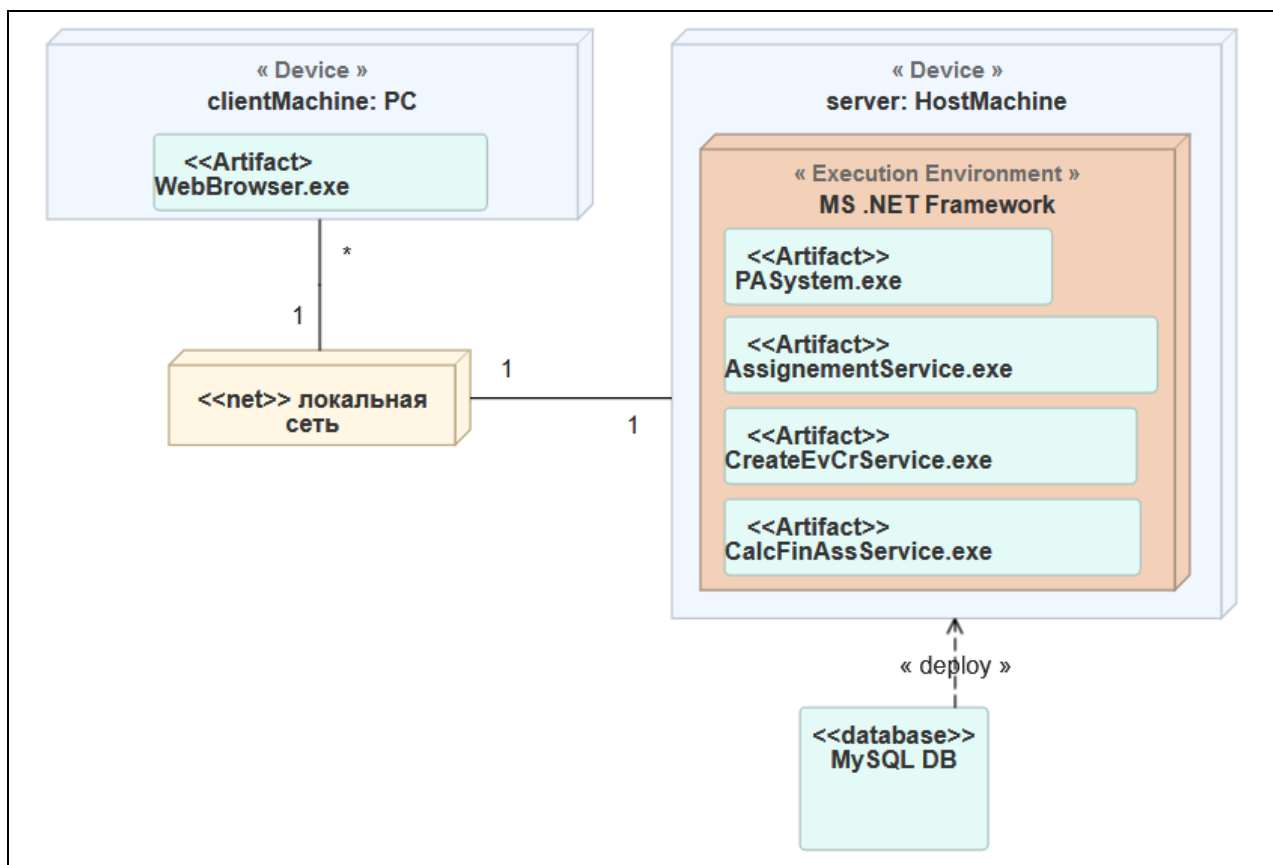


Рис. 11.3. Диаграмма развертывания PAsystem

11.5.3 На рисунке 11.4 представлена диаграмма развертывания PAsystem, разработанная с учетом применения подхода DDD на основе п. 11.3.10. Передача данных между ПК пользователя и обоими серверами осуществляется по протоколу HTTP, между InfastructureDb.dll и БД Moodle_DB – по протоколу TCP. На сервере 195.133.146.151 компонент WebApp зависит от библиотеки Domail.dll и

БД App_DB, библиотека InfrastructureDb.dll – от баз данных App_DB и Moodle_DB. WebApp отправляет HTTP-запросы к сервису moodle_mobile_auth.

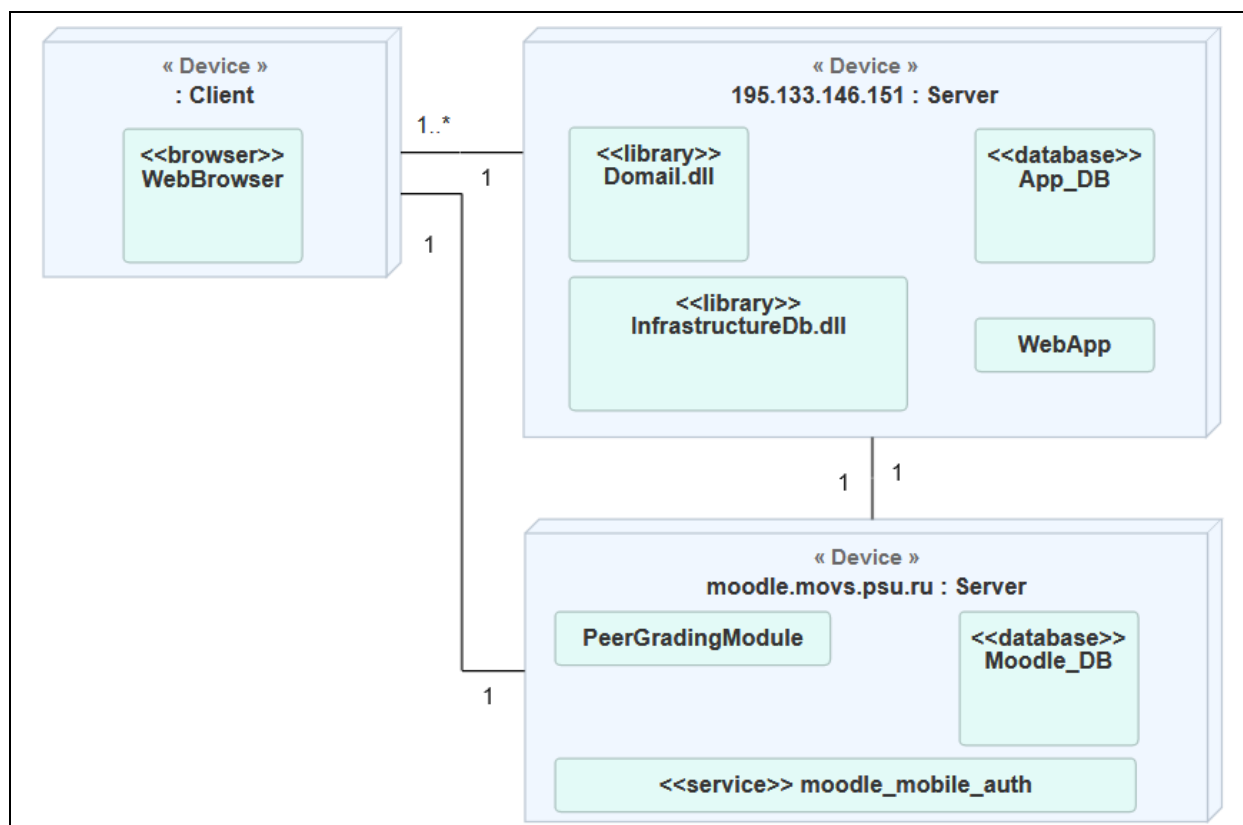


Рис. 11.4. Диаграмма развертывания PASystem (подход DDD)

11.6 Содержание отчета по ЛР № 4

11.6.1 Для студентов очной и заочной форм обучения: отчет по ЛР № 4 должен содержать диаграмму компонентов, в которой представлены:

- а) интерфейсы и их внутренняя структура,
- б) компоненты и их внутренняя структура,
- в) связь между компонентами и их интерфейсами,
- г) отношения между компонентами,
- д) порты и коннекторы компонентов.

11.6.2 Для студентов очной формы обучения: отчет по ЛР № 4 должен содержать диаграмму развертывания, в которой представлены:

- а) узлы,
- б) связи между узлами,
- в) артефакты,
- г) зависимости,
- д) миграция компонентов.

11.7 Оценивание ЛР № 4

Оценивание выполнения ЛР № 4 приведено в таблицах 11.2–11.3. Максимальная оценка составляет у студентов очной формы обучения 5 баллов, зачетная – 2,1 балла. Максимальная оценка составляет у студентов заочной формы обучения 5 баллов, зачетная – 2,5 баллов.

Таблица 11.2

Оценивание выполнения ЛР № 4 (очная форма обучения)

Баллы	Выполнение требований к ЛР № 4
Общий 5, в том числе 4	Выполнены: все требования п. 11.6.1–11.6.2
Общий 4,9–4, в том числе 3,9–3	Выполнены: до 80% требований п. 11.6.1–11.6.2 (ошибок лексики и синтаксиса диаграмм компонентов и диаграмм развертывания нет)
Общий 3,9–3, в том числе 2,9–2	Выполнены: до 60% требований п. 11.6.1–11.6.2 (имеются ошибки лексики и синтаксиса диаграмм компонентов и диаграмм развертывания)
Общий 2,9–2,1 в том числе 1,9–1,1	Выполнены: до 40% требований п. 11.6.1–11.6.2 (имеются серьезные ошибки лексики и синтаксиса диаграмм компонентов и диаграмм развертывания)

Таблица 11.3

Оценивание выполнения ЛР № 4 (заочная форма обучения)

Баллы	Выполнение требований к ЛР № 4
Общий 5, в том числе 4	Выполнены: все требования п. 11.6.1
Общий 4,9–4, в том числе 3,9–3	Выполнены: до 80% требований п. 11.6.1 (ошибок лексики и синтаксиса диаграмм компонентов нет)
Общий 3,9–3, в том числе 2,9–2	Выполнены: до 60% требований п. 11.6.1 (имеются ошибки лексики и синтаксиса диаграмм компонентов)
Общий 2,9–2,5 в том числе 1,9–1,5	Выполнены: до 40% требований п. 11.6.1 (имеются серьезные ошибки лексики и синтаксиса диаграмм компонентов)

Примечание: обязательным является выполнение п. 2.1 (0,1 балла), 2.2 (0,5 балла) и 2.3 (0,4 балла); общая сумма – 1 балл.

12 Варианты описания предметных областей индивидуальных заданий

1. ИС-регистратор договоров залога в ломбарде. О каждом договоре залога известно: дата открытия договора (обязательное поле), дата фиксации договора (когда договор подписан, его фиксируют и после этого изменить уже нельзя, дата фиксации обязательно позже даты открытия), дата закрытия договора (обязательное поле, дата закрытия обязательно позже даты фиксации, после закрытия за пользование кредитом начисляется пеня, к закрытию – стандартные проценты в соответствии с выбранной процентной ставкой), количество денег, выданных на руки (обязательное поле), оценочная стоимость предметов залога (обязательное поле, может быть больше, но не может быть меньше денег, выданных на руки); номер договора (обязательное уникальное поле), ФИО клиента (обязательное поле), процентная ставка (напр., «стандарт», «для постоянных клиентов», «для друзей» и т.п., обязательное поле). ИС работает в режиме администратора и в режиме кассира. Администратор может просматривать список и редактировать отдельные записи. В режиме кассира ИС обрабатывает ситуацию, когда клиент хочет выкупить свои вещи. При выкупе клиент должен платить проценты по договору, пеню (в случае, если он просрочил договор) и стоимость выкупаемых вещей. Кассир ищет договор по его номеру, ИС рассчитывает стоимость пользования кредитом и пеню (пеня начисляется, если клиент просрочил оплату процентов), помогает кассиру правильно посчитать сдачу. Должны быть выведены на печать акт возврата предметов залога и фискальный чек (сколько денег взято, сколько денег удержано как проценты, сколько взято за пеню).

2. Активация программного продукта. ИС может обслуживать несколько отделений предприятия. При этом каждое отделение имеет свой уникальный номер. Также существует сервер в Интернете, на который передаются данные со всех филиалов предприятия. При этом должны быть запрещены несколько инсталляций программы с одинаковым номером отделения. Для начала активации пользователь должен ввести лицензионный код и номер отделения. При этом ИС сгенерирует случайный пароль отделения. Чтобы защитить от повторных инсталляций с одинаковым номером отделения, пользователь должен сообщить этот пароль администратору сервера, чтобы администратор прописал этот пароль на сервере для отделения с этим номером. Когда администратор прописал сгенерированный пароль на сервере, активация может быть завершена. Активацию можно провести по телефону или через Интернет. При активации по телефону пользователь сообщает администратору пароль отделения и ждет, когда администратор продиктует пользователю по телефону тайный от-

вет сервера. При активации через Интернет пользователь просто ждет, когда администратор пропишет пароль на сервере: пароль и сервер передаются в обоих направлениях автоматически.

3. Центр управления паролями. ИС предназначена для обмена секретными сообщениями и хранения паролей (пароль можно хранить внутри тайного сообщения). Тайные сообщения могут быть доступны нескольким сотрудникам. Доступ к сообщениям возможен, только если сотрудник указал свой пароль доступа к тайным сообщениям. О каждом тайном сообщении известны: заголовок, текст секретного сообщения, список получателей. У каждого сотрудника есть собственная архивная копия каждого тайного сообщения. Управлять архивной копией может только сам сотрудник. Управлять основной копией может любой получатель этого секретного сообщения, при этом все изменения, внесенные в основную копию, видны всем сотрудникам. При просмотре списка тайных сообщений сотрудник видит, какие сообщения были изменены после создания архивной копии. При просмотре сообщения, если текущая копия не совпадает с архивной, сотрудник видит, что именно изменилось в тайном сообщении. При удалении сообщения удаляется его основная копия, но архивные копии отдельных сотрудников остаются.

4. Интернет-портал с информацией о предприятиях. О каждом предприятии известно: срок оплаты (информация размещается на портале на платной основе, срок может быть не указан, если выбран тестовый пакет), название предприятия, адрес, номер телефона, факс, e-mail, имя контактного лица, адрес сайта, описание предприятия, логотип, пакет (тестовый, стандартный, VIP – определяет количество дополнительных услуг для данного предприятия), список рубрик, в которых расположено предприятие, логин пользователя, который управляет этим предприятием. Работа в режиме администратора. Администратор может просматривать список предприятий, редактировать и удалять существующие предприятия.

5. Интернет-портал с информацией о предприятиях. О каждом предприятии известно: срок оплаты (информация размещается на портале на платной основе, срок может быть не указан, если выбран тестовый пакет), название предприятия, адрес, номер телефона, факс, e-mail, имя контактного лица, адрес сайта, описание предприятия, логотип, пакет (тестовый, стандартный, VIP – определяет количество дополнительных услуг для данного предприятия), список рубрик, в которых расположено предприятие, логин пользователя, который управляет этим предприятием. Работа в режиме тестовой регистрации. Посетитель может разместить на портале новое объявление с тестовым пакетом. При этом посетитель должен пройти CAPTCHA-тест, система должна проверить, что посетитель является владельцем введенного адреса e-mail. После размещения объявления оно может быть отредактировано владельцем, но доступно посети-

телям станет только после подтверждения администратором. Владелец предприятия видит, подтверждено предприятие или нет.

6. ИС управления заданиями предприятия. О каждом задании известно: название задания, дата запуска (может быть не задана, если задание еще не запущено), дата остановки (дата остановки обязательно позже даты запуска и не может быть установлена, если дата запуска не задана), приоритет (целое число, влияет на порядок сортировки заданий), имя сотрудника, название проекта, описание задания. Администратор может просматривать списки заданий, создавать новые задания, удалять и редактировать существующие задания. Сотрудник может видеть только свои задания, может создавать задания для себя и редактировать свои задания. Сотрудник может запустить или остановить свое задание. Сотрудник может работать над заданиями из большого количества различных проектов.

7. ИС обмена заметками. Любой сотрудник может оставить другому сотруднику или группе сотрудников заметку. При этом получатель заметки должен поставить отметку о прочтении. Каждому сотруднику доступен список непрочитанных заметок, но он может также просмотреть и архив заметок. Когда сотрудник ставит отметку о прочтении, заметка удаляется из списка непрочитанных заметок сотрудника, но остается в архиве. О каждой заметке известны заголовок, тело заметки, список получателей (получатели не могут повторяться, нельзя отправить заметку себе), для каждого получателя – дата одобрения заметки (при создании заметки это поле пустое, при одобрении заполняется текущей датой).

8. ИС-регистратор договоров залога в ломбарде. О каждом договоре залога известно: дата открытия договора (обязательное поле), дата фиксации договора (когда договор подписан, его фиксируют и после этого изменить уже нельзя, дата фиксации обязательно позже даты открытия), дата закрытия договора (обязательное поле, дата закрытия обязательно позже даты фиксации, после закрытия за пользование кредитом начисляется пеня, к закрытию – стандартные проценты в соответствии с выбранной процентной ставкой), количество денег, выданных на руки (обязательное поле), оценочная стоимость предметов залога (обязательное поле, может быть больше, но не может быть меньше денег, выданных на руки); номер договора (обязательное уникальное поле), ФИО клиента (обязательное поле), процентная ставка (напр., «стандарт», «для постоянных клиентов», «для друзей» и т.п., обязательное поле). ИС работает в режиме администратора и в режиме кассира. Администратор может просматривать список и редактировать отдельные записи.

9. ИС-регистратор договоров залога в ломбарде. О каждом договоре залога известно дата открытия договора (обязательное поле), дата фиксации договора (когда договор подписан, его фиксируют и после этого изменить уже нельзя, дата фиксации обязательно позже даты открытия), дата закрытия дого-

вора (обязательное поле, дата закрытия обязательно позже даты фиксации, после закрытия за пользование кредитом начисляется пеня, к закрытию – стандартные проценты в соответствии с выбранной процентной ставкой), количество денег, выданных на руки (обязательное поле), оценочная стоимость предметов залога (обязательное поле, может быть больше, но не может быть меньше денег, выданных на руки); номер договора (обязательное уникальное поле), ФИО клиента (обязательное поле), процентная ставка (напр., «стандарт», «для постоянных клиентов», «для друзей» и т.п., обязательное поле). ИС работает в режиме администратора и в режиме кассира. Администратор может просматривать список и редактировать отдельные записи. В режиме кассира ИС обрабатывает ситуацию, когда клиент хочет продлить договор. При продлении договора клиент может оплатить проценты наличными или увеличить количество денег, выданных на руки, и уплатить проценты с этой разницы. При этом увеличить количество денег, выданных на руки, можно только, если это количество меньше оценочной стоимости. Кассир ищет договор по его номеру, ИС рассчитывает стоимость пользования кредитом и пенью (пеня насчитывается, если клиент просрочил оплату процентов), помогает кассиру правильно посчитать сдачу. Должны быть выведены на печать договор и фискальный чек (сколько денег выдано, сколько денег удержано как проценты, сколько взято за пенью).

10. ИС настройки системы антиспама. Пользователь может настраивать список адресов, от которых система должна пропускать все письма («белый» список); параметры распознавания спама (применять анализ изображений (да / нет); применять анализ заголовков (да / нет); фактор спама (считать спамом сообщение с фактором, большим, чем указанное число от 0 до 100), список запрещенных фраз).

11. Журнал нагрузки сотрудника предприятия. О каждом элементе журнала нагрузки известно: дата и время начала работы, дата и время окончания работы (должна быть обязательно больше даты начала, но разница между датами не может быть больше 8 часов), проводилась работа в офисе или дома (да / нет), имя сотрудника, название проекта, название задачи, комментарий. Сотрудник может просматривать свой журнал нагрузки, редактировать, добавлять и удалять записи из него. Поскольку сотрудник может редактировать очень старые записи из журнала нагрузки, то он должен иметь возможность выбрать любое задание, над которым когда-либо работал, а таких задач может быть тысячи.

12. ИС для учета движения денежных средств на счетах предприятия. О каждом счете известны его название (уникальное поле), остаток средств. На счете (поле заполняется автоматически, при создании нового счета равно нулю), в каком банке открыт счет, номер счета в банк (16-значное десятичное число). По каждому счету хранится список всех изменений по счету (транзак-

ции). О каждой транзакции известно: название счета, дата, сумма транзакции, остаток на счете после проведения платежа, назначение платежа, направление транзакции (приход или расход). ИС работает в режиме администратора и в режиме кассира. Администратор может просматривать список счетов и редактировать их. При работе в режиме кассира он может просматривать список транзакций по выбранному счету и редактировать их. При редактировании автоматически изменяется остаток на счете.

13. ИС для учета движения денежных средств на счетах предприятия. О каждом счете известны его название (уникальное поле), остаток средств. На счете (поле заполняется автоматически, при создании нового счета равно нулю), в каком банке открыт счет, номер счета в банке (16-значное десятичное число). По каждому счету хранится список всех изменений по счету (транзакции). О каждой транзакции известно: название счета, дата, сумма транзакции, остаток на счете после проведения платежа, назначение платежа, направление транзакции (приход или расход). ИС работает в режиме администратора и в режиме контроллера-кассира. Администратор может просматривать список счетов и редактировать их. При работе в режиме контроллера-кассира он может просматривать список всех изменений (кто, когда, на каком счете провел изменения), которые вносятся на счета, и детали изменений (по каждому полю транзакции выводится значение до изменения и после изменения).

14. ИС-фискальный регистратор. ИС регистрирует фискальные чеки. О каждом чеке известно: список товаров, имя покупателя, имя кассира, дата, общая стоимость, НДС. О каждом товаре в чеке известно: название товара, цена за единицу, количество приобретенного товара, общая стоимость, НДС. ИС работает в режиме администратора. Администратор может настраивать стандартный заголовок и стандартное окончание чека, может устанавливать ставку НДС, может настраивать список товаров своего магазина (при этом указывается артикул, название, цена за единицу, учитывается ли НДС для этого товара). Артикул в чеке не указывается, поскольку артикул предназначен только для внутреннего использования.

15. ИС-фискальный регистратор. ИС регистрирует фискальные чеки. О каждом чеке известно: список товаров, имя покупателя, имя кассира, дата, общая стоимость, НДС. О каждом товаре в чеке известно: название товара, цена за единицу, количество приобретенного товара, общая стоимость, НДС. ИС работает в режиме кассира. Кассир вводит свое имя, имя клиента, для каждого товара – артикул и количество товара, а ИС формирует фискальный чек, подставляя вместо артикула название и стоимость товара. На чеке должны быть написаны стандартный заголовок, список товаров, имя кассира, имя клиента, сумма, НДС, стандартное окончание. ИС помогает кассиру правильно посчитать сдачу.

13 Критерии парного оценивания моделей, созданных студентами

Критерии парного оценивания моделей, созданных студентами в рамках выполнения лабораторных работ и практических занятий, приведены в таблице 13.1. Они использованы в элементе «Парное оценивание работ студентов» сайта moodle.movs.psu.ru.

Таблица 13.1

Критерии парного оценивания моделей

Номер критерия	Описание критерия	Требования	Балл за выполнение требований		Способ оценивания / максимальная оценка
			4	5	
1	Границы системы, акторы и прецеденты	3 а) границы системы определены, б) акторы идентифицированы, в) прецеденты идентифицированы не выполнено одно или больше требований а)-в)	1	5	выбор варианта балла / 1
			0		
			2		
2	Запись прецедентов в высокоуровневой форме	2 все прецеденты записаны в высокоуровневой форме часть прецедентов записана в высокоуровневой форме прецеденты не записаны в высокоуровневой форме	1	2	выбор варианта балла / 2
			0		
			2		
3	Классификация прецедентов по категориям: главные, второстепенные и дополнительные	1 выполнена не выполнена	1	1	выбор варианта балла / 1
			0		
4	Запись прецедентов в развернутой идеальной форме	2 наиболее важные и рискованные прецеденты выделены и записаны не только наиболее важные и рискованные прецеденты выделены и записаны отсутствует	2	2	выбор варианта балла / 2
			1		
			0		

1	2	3	4	5
5	Отношения между акторами	определены все виды отношений между акторами и проиллюстрированы на диаграмме прецедентов не определены отношения между акторами и/или не проиллюстрированы на диаграмме прецедентов	1 0	выбор варианта балла / 1
6	Определены такие виды отношений между прецедентами и проиллюстрированы на диаграмме	ассоциация обобщение включение расширение	1 2 1 1	суммирование баллов / 5
7, 9, 22, 25, 30, 43, 45, 47	Лексика и синтаксис диаграммы прецедентов / активностей / понятий / последовательностей / сотрудничества / классов / компонентов / разветвления	удовлетворяют стандарту UML не удовлетворяют стандарту UML диаграмма (-ы) отсутствует (-ют)	2 1 0	выбор варианта балла / 2
8	Диаграмма активностей отражает бизнес-процесс	полностью частично не отражает или отсутствует	2 1 0	выбор варианта балла / 2
10, 11, 12	Процесс выявления понятий / атрибутов понятий / ассоциаций при создании концептуальной модели	отражен не отражен	1 0	выбор варианта балла / 1
13	Роли ассоциаций в концептуальной модели	присутствуют отсутствуют	1 0	выбор варианта балла / 1
14	Кратность ролей ассоциации в концептуальной модели	указана для всех ролей указана для части ролей не указана	2 1 0	выбор варианта балла / 2
15, 16	Ограничения для понятий / ассоциаций в концептуальной модели	присутствуют отсутствуют	1 0	выбор варианта балла / 1

1	2	3	4	5
17	Разбиение понятий на подтипы и выделение супертипов в концептуальной модели	присутствует	1	выбор варианта балла / 1
18	Выявленные обобщения правилам «100%» и «is_a» в концептуальной модели	отсутствует	0	
19	Выявленные обобщения правилам «100%» и «is_a» в концептуальной модели	удовлетворяют	1	выбор варианта балла / 1
20	Абстрактное понятие в концептуальной модели	не удовлетворяют	0	
21	Многомерная множественная классификация в концептуальной модели	идентифицировано	1	выбор варианта балла / 1
22	Отношения агрегации в концептуальной модели	не идентифицировано	0	
23	Содержательными являются системные сообщения в диаграмме (-ах) последовательностей	выполнена	1	выбор варианта балла / 1
24	Диаграмма (-ы) последовательностей описывает (-ют) поведение системы на этапе анализа	не выполнена	0	
25	Шаблону и требованиям объектно-ориентированного подхода	выделены все	2	
26	Шаблону и требованиям объектно-ориентированного подхода	выделены все	2	
27	Работа с коллекциями на диаграмме сотрудничества	выделены частично	1	выбор варианта балла / 2
28	Содержательными являются системные сообщения в диаграмме (-ах) последовательностей	выделены частично	1	выбор варианта балла / 2
29	Содержательными являются системные сообщения в диаграмме (-ах) последовательностей	не выделены	0	
30	Содержательными являются системные сообщения в диаграмме (-ах) последовательностей	все	2	
31	Содержательными являются системные сообщения в диаграмме (-ах) последовательностей	часть	1	выбор варианта балла / 2
32	Содержательными являются системные сообщения в диаграмме (-ах) последовательностей	содержательные системные сообщения отсутствуют	0	
33	Содержательными являются системные сообщения в диаграмме (-ах) последовательностей	полностью	2	
34	Содержательными являются системные сообщения в диаграмме (-ах) последовательностей	частично	1	выбор варианта балла / 2
35	Содержательными являются системные сообщения в диаграмме (-ах) последовательностей	не описывает (-ют)	0	
36	Содержательными являются системные сообщения в диаграмме (-ах) последовательностей	удовлетворяет описание всех системных операций	3	
37	Содержательными являются системные сообщения в диаграмме (-ах) последовательностей	удовлетворяет описание части системных операций	2	выбор варианта балла / 3
38	Содержательными являются системные сообщения в диаграмме (-ах) последовательностей	не удовлетворяет описание системных операций	1	
39	Содержательными являются системные сообщения в диаграмме (-ах) последовательностей	описание системных операций отсутствует	0	
40	Содержательными являются системные сообщения в диаграмме (-ах) последовательностей	корректно отображена	2	
41	Содержательными являются системные сообщения в диаграмме (-ах) последовательностей	не корректно отображена	1	выбор варианта балла / 2
42	Содержательными являются системные сообщения в диаграмме (-ах) последовательностей	отсутствует	0	

1	2	3	4	5
28	Сообщение классу на диаграмме сотрудничества	предусмотрено	1	выбор варианта балла / 1
29	Видимость объектов на диаграмме сотрудничества	не предусмотрено	0	выбор варианта балла / 1
31, 32, 34, 35	Ассоциации / роли ассоциаций / ограничения для ассоциаций на диаграмме классов	отображена явно	1	выбор варианта балла / 1
33	Типы данных в диаграмме классов	не отображена явно	0	выбор варианта балла / 1
36, 37	Обобщение / агрегация в диаграмме классов	присутствуют	1	выбор варианта балла / 1
		отсутствуют	0	выбор варианта балла / 1
		указаны корректно для всех агрибутов	2	выбор варианта балла / 2
		указаны корректно для части агрибутов	1	выбор варианта балла / 2
		указаны не корректно или отсутствуют	0	выбор варианта балла / 2
		корректно отображено с помощью агрибутов и отношений	2	выбор варианта балла / 2
		отображено только с помощью отношений	1	выбор варианта балла / 2
		не отображено	0	выбор варианта балла / 2
38	Набор операций в диаграмме классов	обеспечивает корректное функционирование всех классов	2	выбор варианта балла / 2
		корректное функционирование части классов	1	выбор варианта балла / 2
		отсутствует	0	выбор варианта балла / 2
39	Информация об области видимости в диаграмме классов принципу инкапсуляции и отношениям между классами	соответствует	1	выбор варианта балла / 2
		не соответствует	0	выбор варианта балла / 2
40	Вычисляемые агрибуты в диаграмме классов	соответствуют семантике Про	2	выбор варианта балла / 2
		не соответствуют семантике Про	1	выбор варианта балла / 2
		отсутствуют, но должны быть по семантике Про, или не предусмотрены	0	выбор варианта балла / 2

1	2	3	4	5
41	Направления навигации в диаграмме классов	указаны корректно у всех ассоциаций указаны корректно у части ассоциаций не указаны у ассоциаций	2 1 0	выбор варианта балла / 2
42	Необходимые зависимости в диаграмме классов	указаны корректно указаны не корректно не указаны	2 1 0	выбор варианта балла / 2
44	Диаграмма компонентов диаграмме классов	соответствует не соответствует	1 0	выбор варианта балла / 1
46	Диаграмма разветвления компонентов диаграмме	соответствует не соответствует	1 0	выбор варианта балла / 1

14 Выписки из учебных планов

В таблице 14.1 представлены выписки из учебных планов.

Таблица 14.1

Выписки из учебных планов

Наименование	Кол-во семестров	Лекции	Практические	Самостоятельная работа	Лабораторные работы	Кредиты	Форма отчетности, триместр
Наименование							
02.03.02 Фундаментальная информатика и информационные технологии							
УП 9999 Базовая	1	14	28	88	14	4	Экзамен, 8
УП 10003 СУОС-ЗАО	3	4	8	128	4	4	Экзамен, 11
УП 11163 Базовая	1	28	14	66		3	Зачет, 10
УП 11192 Заочная	2	6	4	98		3	Зачет, 12
Наименование							
09.03.02 Информационные системы и технологии							
Специализация Безопасность информационных систем							
УП 11158 Базовая	1	28	14	66		3	Зачет, 10

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Эванс Э. Предметно-ориентированное проектирование (DDD). Структуризация сложных программных систем. М.: Вильямс, 2017. 448 с.
2. Moodle. URL: <https://moodle.org/> (дата обращения: 15.06.2021).
3. ГОСТ 7.32-2017. Система стандартов по информации, библиотечному и издательскому делу. Отчет о научно-исследовательской работе. Структура и правила оформления: межгосударственный стандарт. М.: Стандартинформ, 2017. 32 с. URL: http://www.consultant.ru/document/cons_doc_LAW_292293/ (дата обращения: 15.06.2021).
4. ГОСТ Р 7.0.5-2008 Национальный стандарт Российской Федерации. Система стандартов по информации, библиотечному и издательскому делу. Библиографическая ссылка. Общие требования и правила составления. М.: Стандартинформ, 2008. 44 с. URL: <http://docs.cntd.ru/document/gost-r-7-0-5-2008> (дата обращения: 15.06.2021).
5. GenMyModel. URL: <https://www.genmymodel.com/> (дата обращения: 15.06.2021).
6. ГОСТ 19. 201-78. Межгосударственный стандарт. Единая система программной документации. Техническое задание. Требования к содержанию и оформлению. М.: Стандартинформ, 2010. с. 59-60. URL: <https://meganorm.ru/Data2/1/4294850/4294850109.pdf> (дата обращения: 15.06.2021).
7. ГОСТ 34. 602-89. Межгосударственный стандарт Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы. М.: Стандартинформ, 2010. URL: <http://docs.cntd.ru/document/gost-34-602-89> (дата обращения: 15.06.2021).
8. ISO/IEC 19505-1:2012. Information technology – Object Management Group Unified Modeling Language (OMG UML) Part 1: Infrastructure. URL: <https://www.iso.org/standard/32624.html> (дата обращения: 15.06.2021).
9. ISO/IEC 19505-1:2012. Information technology - Object Management Group Unified Modeling Language (OMG UML) Part 2: Superstructure. URL: <https://www.iso.org/standard/52854.html> (дата обращения: 15.06.2021).
10. About the Unified Modeling Language Specification Version 2.4. URL: <https://www.omg.org/spec/UML/2.4/> (дата обращения: 15.06.2021).
11. Якобсон А., Буч Г., Рамбо Дж. Унифицированный процесс разработки программного обеспечения. СПб: Питер, 2002. 496 с.
12. Рамбо Дж., Якобсон А., Буч Г. UML: Специальный справочник. СПб.: Питер, 2002. 656 с.
13. Буч Г., Рамбо Д., Якобсон И. Язык UML. Руководство пользователя. М.: ДМК Пресс, 2006. 496 с.

14. *Вендров А.М.* Проектирование программного обеспечения экономических информационных систем: учебник. М.: Финансы и статистика, 2006. 544 с.
15. *Леоненков А.В.* Объектно-ориентированный анализ и проектирование с использованием UML и IBM RATIONAL ROSE: учеб. пособие. М.: Интернет-Университет информационных технологий: БИНОМ. Лаборатория знаний, 2006. 320 с.
16. *Орлов С.А., Цилькер Б.Я.* Технологии разработки программного обеспечения. Современный курс по программной инженерии: учебник для вузов. СПб.: Питер, 2012. 608 с.
17. *Шаврин С.М., Лядова Л.Н., Чуприна С.И.* Моделирование и проектирование информационных систем: учеб.-метод. пособие. Пермь: Изд-во Перм. гос. ун-та, 2007. 152 с.
18. *Новиков Ф.А.* Учебно-методическое пособие по дисциплине «Анализ и проектирование на UML» СПб.: Санкт-Петербургский государственный университет информационных технологий, механики и оптики, 2007. 286 с. URL: <https://books.ifmo.ru/file/pdf/424.pdf> (дата обращения: 15.06.2021).
19. *Введение в UML.* Лекция 7: Диаграммы прецедентов: крупным планом. URL: http://www.intuit.ru/studies/professional_retraining/941/courses/229/lecture/5962?page=2 (дата обращения: 15.06.2021).
20. *Национальный Открытый Университет «ИНТУИТ».* URL: <https://www.intuit.ru/> (дата обращения: 15.06.2021).
21. *Coursera.* URL: <https://www.coursera.org/> (дата обращения: 15.06.2021)
22. *edX.* URL: <https://www.edx.org/> (дата обращения: 15.06.2021).
23. *Udacity.* URL: <https://www.udacity.com/> (дата обращения: 15.06.2021).
24. *Saylor Academy.* URL: <https://www.saylor.org/> (дата обращения: 15.06.2021).
25. ГОСТ Р ИСО 9241-161-2016. Эргономика взаимодействия человек-система. Часть 161. Элементы графического пользовательского интерфейса. URL: <http://docs.cntd.ru/document/1200141126> (дата обращения: 15.06.2021).
26. *draw.io.* URL: <https://app.diagrams.net/> (дата обращения: 15.06.2021).
27. *Ларман К.* Применение UML и шаблонов проектирования. Введение в объектно-ориентированный анализ, проектирование и унифицированный процесс UP. М.: Вильямс, 2004. 624 с.
28. *OMG Unified Modeling Language™ (OMG UML), Superstructure. Version 2.2.* URL: <https://www.omg.org/spec/UML/2.2/Superstructure/PDF> (дата обращения: 15.06.2021).

ПРИЛОЖЕНИЕ А

Пример оформления титульного листа отчета по лабораторной работе в соответствии с ГОСТ 7.32

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ Федеральное государственное автономное образовательное учреждение высшего образования «ПЕРМСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»	ОТЧЕТ по работе №1 "Моделирование на этапе анализа ИС: функциональность ИС" по дисциплине "Основы проектирования и реализации информационных систем"
МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ Федеральное государственное автономное образовательное учреждение высшего образования «ПЕРМСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»	Работу выполнил студент гр. _____ (шифр группы) Петров С.В. _____ (подпись) «__» _____ 202_ _
ОТЧЕТ по работе №1 "Моделирование на этапе анализа ИС: функциональность ИС" по дисциплине "Основы проектирования и реализации информационных систем"	Проверил _____ кафедры МОВС (доц., ст. преп., асс.) ФИО _____ (подпись) «__» _____ 202_ _
Пермь 202_ _	

Рис. А.1 – Подготовка титульного листа отчета в текстовом процессоре

ПРИЛОЖЕНИЕ Б

Инструкция по экспорту файлов модели из инструмента GenMyModel после создания UML диаграмм

1. В результате экспорта модели Вы должны получить файл модели (файл *.xmi) и ее графическое представление (файл *.png), которые необходимо отправлять на проверку работы.
2. Исходный внешний вид проекта представлен на рисунке Б.1.

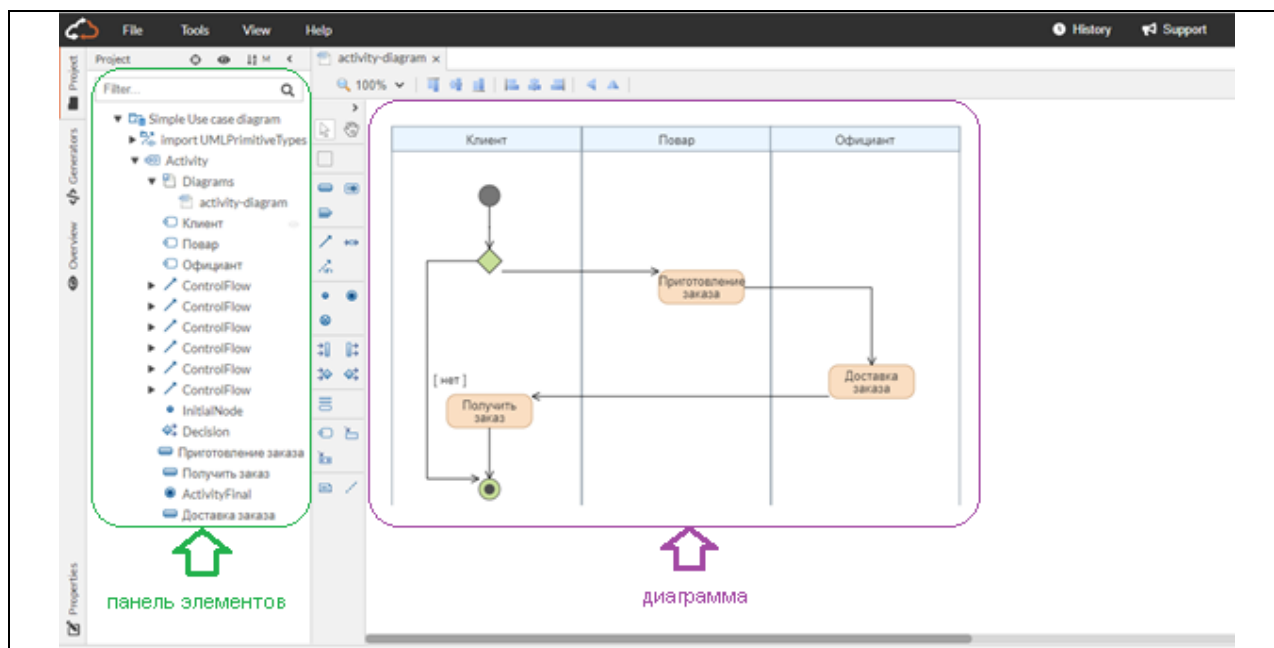


Рис. Б.1. Вид проекта и диаграммы в окне редактора диаграмм инструмента GenMyModel

3. Убедитесь, что элементы, представленные слева на панели элементов, и элементы диаграммы совпадают. Для удаления элемента с диаграммы используйте один из вариантов:

- а) нажмите на элемент правой кнопкой мыши и выберите опцию **Delete from Model** в контекстном меню, как показано на рисунке Б.2,
- б) выберите элемент и воспользуйтесь сочетанием **Shift+Del**.

При использовании только **Del** элемент не будет удален из списка элементов панели, а только с диаграммы. Это приводит к тому, что в файле модели остаются все элементы, которые на диаграмме не отображаются. В некоторых случаях размер xmi-файла может достигать нескольких мега байтов, что не соответствует представлению модели в png-файле.

4. Для экспорта файла модели выполните команду **File→ Export as XMI with Diagrams** как на рисунке Б.3, а) и сохраните полученный xmi-файл.

5. Для экспорта графического файла визуального представления модели выполните команду **File → Export as Image** как на рисунке Б.3, б).

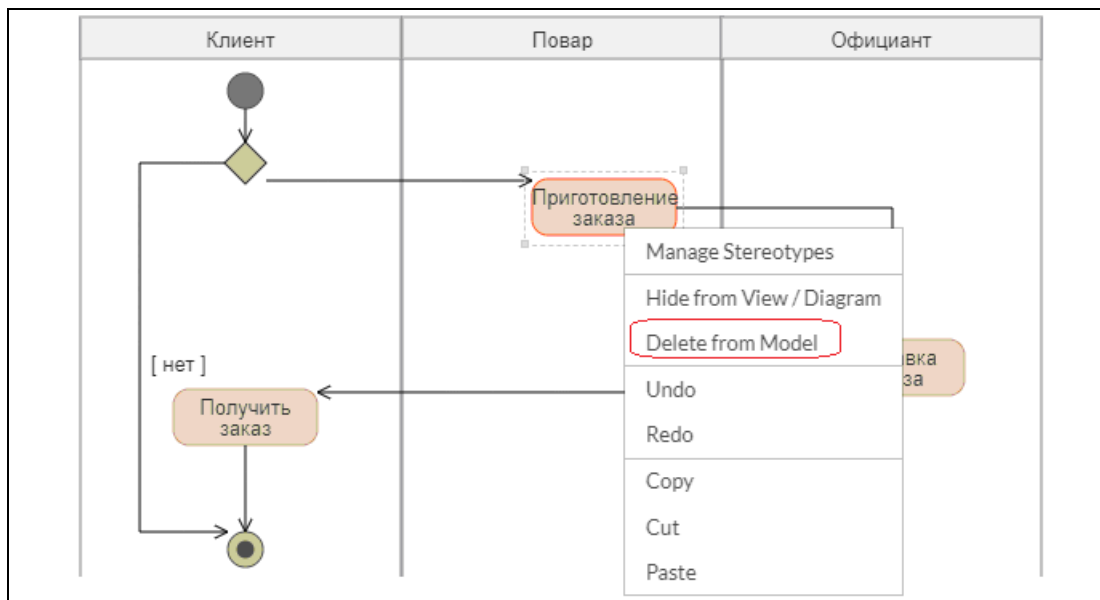


Рис. Б.2 – Один из вариантов удаления элемента из модели в окне редактора диаграмм инструмента GenMyModel

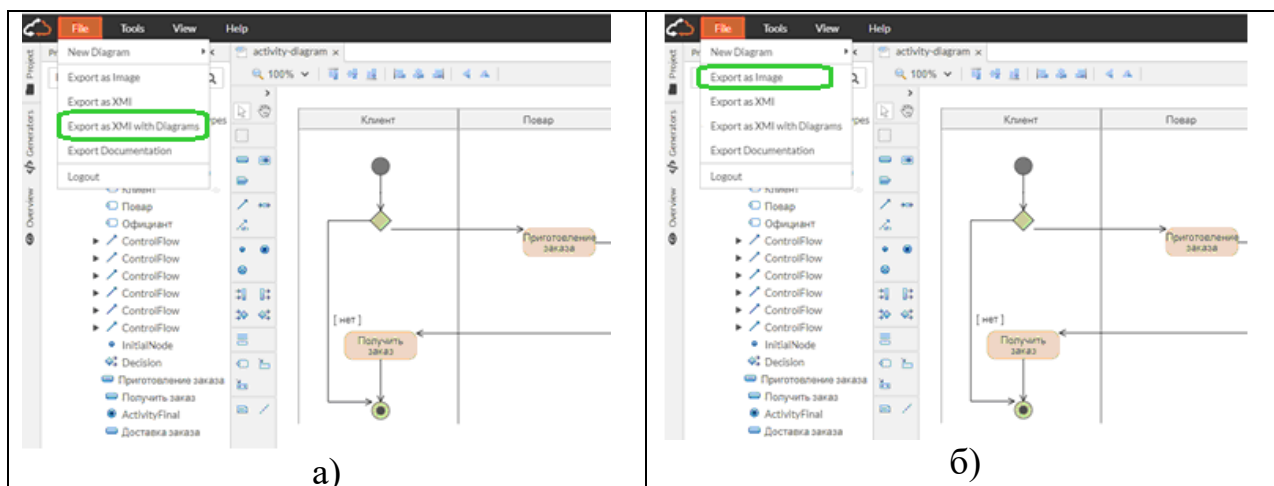


Рис. Б.3. Экспорт модели (xml-файла) и графического файла модели в окне редактора диаграмм инструмента GenMyModel

6. Выберите формат файла (png) как на рисунке Б.4, и сохраните полученный png-файл.

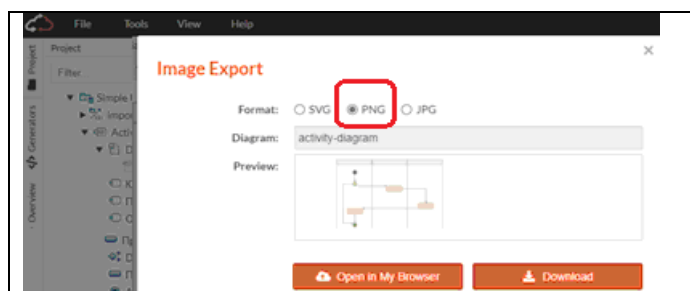


Рис. Б.4. Выбор формата графического файла модели при экспорте в окне редактора диаграмм инструмента GenMyModel



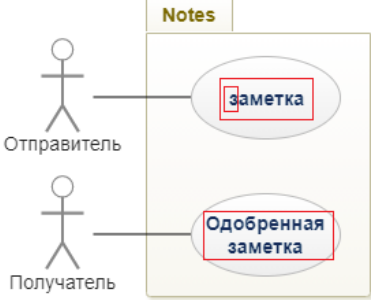
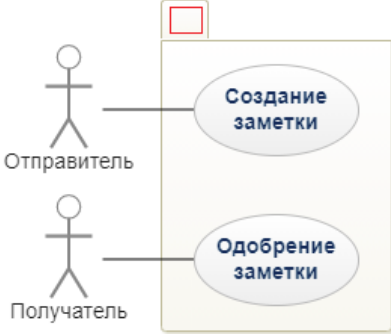
ПРИЛОЖЕНИЕ В

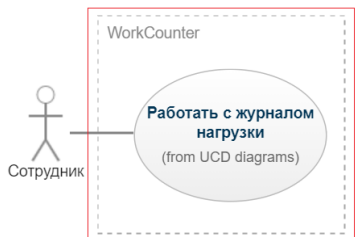
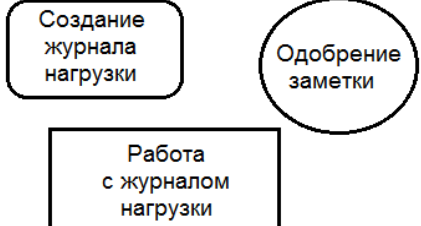

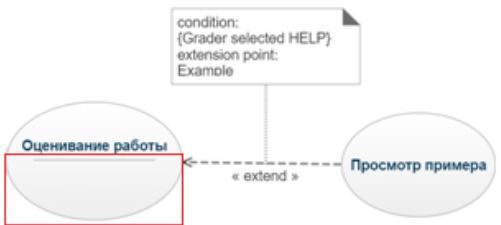
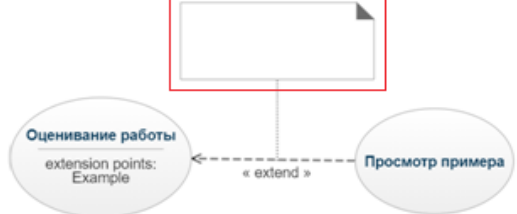
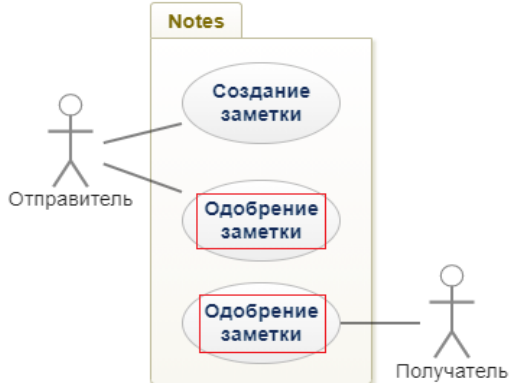
Типичные ошибки в UCD студентов



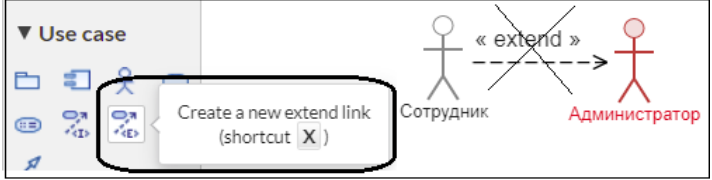



В таблице В.1 представлены типичные ошибки, допускаемые студентами при создании диаграмм прецедентов.

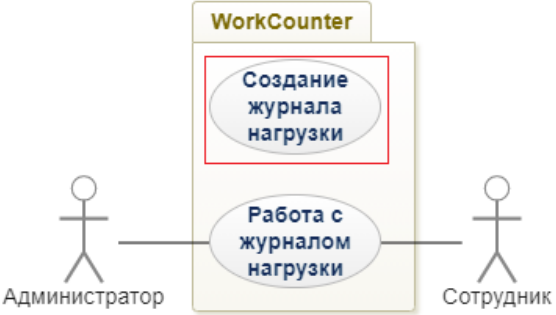
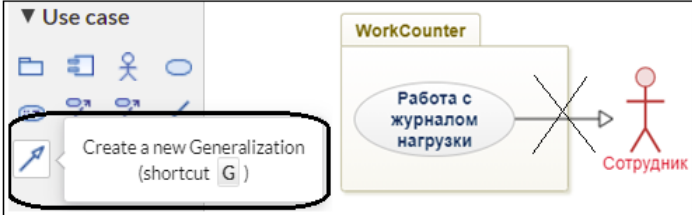

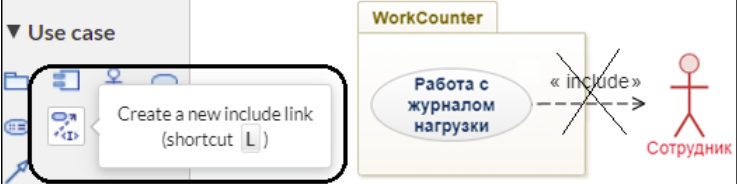
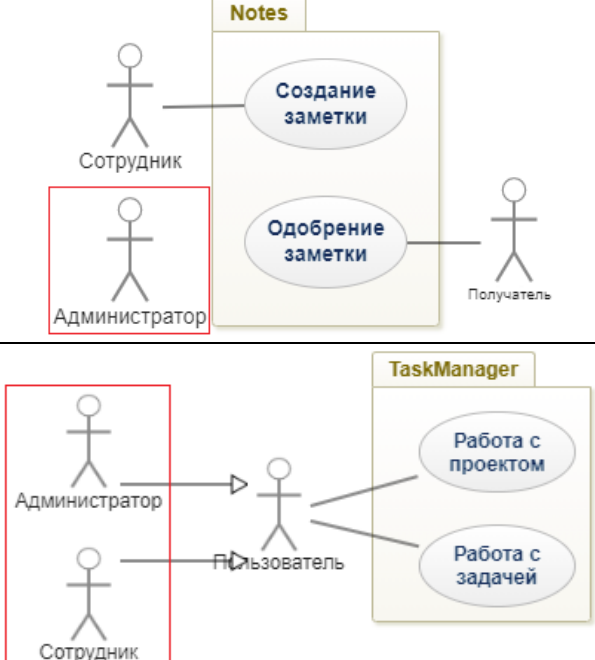
Таблица В.1

Типичные ошибки, допускаемые студентами
при создании диаграмм прецедентов

Ошибка	Пример с ошибкой
<p style="text-align: center;">1</p> <p>Отсутствует граница системы</p>	<p style="text-align: center;">2</p> 
<p>Некорректное имя актора, должно быть представлено именем существительным, начинаться с большой буквы</p>	
<p>Некорректное название прецедента, должно быть представлено в виде действия (глаголом или именем существительным отглагольной формы) с большой буквы</p>	
<p>Отсутствует название системы (имя субъекта)</p>	

1	2
<p>Использование элементов, не входящих в список, разрешенных стандартом UML для UCD</p>	
<p>Прецеденты должны быть представлены соответствующим элементом UCD. GenMyModel не позволит создать такие элементы при использовании панели «Use case diagram»</p>	
<p>Отсутствие точки расширения у прецедента с зависимостью расширения</p>	
<p>Отсутствует текст в точке расширения прецедента</p>	
<p>Отсутствует текст в условии расширения прецедента</p>	
<p>Имена прецедентов должны быть уникальными</p>	

1	2
<p>Имена акторов должны быть уникальными</p>	
<p>Использование отношения включения между акторами. GenMyModel не позволит создать такую модель</p>	
<p>Использование отношения расширения между акторами. GenMyModel не позволит создать такую модель</p>	
<p>Размещение актора внутри системы</p>	
<p>Использование отношения ассоциации между прецедентами</p>	
<p>Размещение прецедента вне системы</p>	

1	2
<p>Прецедент не имеет отношений с другими элементами диаграммы</p>	
<p>Использование отношения обобщения не между двумя акторами или прецедентами. GenMyModel не позволит создать такую модель</p>	
<p>Использование зависимости расширения не между двумя прецедентами. GenMyModel не позволит создать такую модель</p>	
<p>Использование зависимости включения не между двумя прецедентами. GenMyModel не позволит создать такую модель</p>	
<p>Актор не имеет ни одного отношения ассоциации с прецедентами</p>	

1	2
<p>Родительский прецедент, имеющий зависимости включения или расширения, не имеет отношения ассоциации</p>	
<p>Прецедент с зависимостью включения, включает всего один прецедент</p>	
<p>Два и более актора с одинаковыми наборами ассоциированных прецедентов</p>	
<p>Зависимости включения / расширения направлены не в ту сторону</p>	

1	2
<p>Злоупотребление зависимостью включения</p>	
<p>Неиерархическая структура прецедентов</p>	
<p>Зависимый прецедент включения ассоциирован с актором</p>	
<p>Зависимый прецедент расширения ассоциирован с актором</p>	

ПРИЛОЖЕНИЕ Г

Типичные ошибки в AD студентов

На рисунке Г.1 представлен результат неверного использования инструмента GenMyModel для создания диаграммы активностей.

В таблице Г.1 представлены типичные ошибки, допускаемые студентами при создании диаграмм активностей.

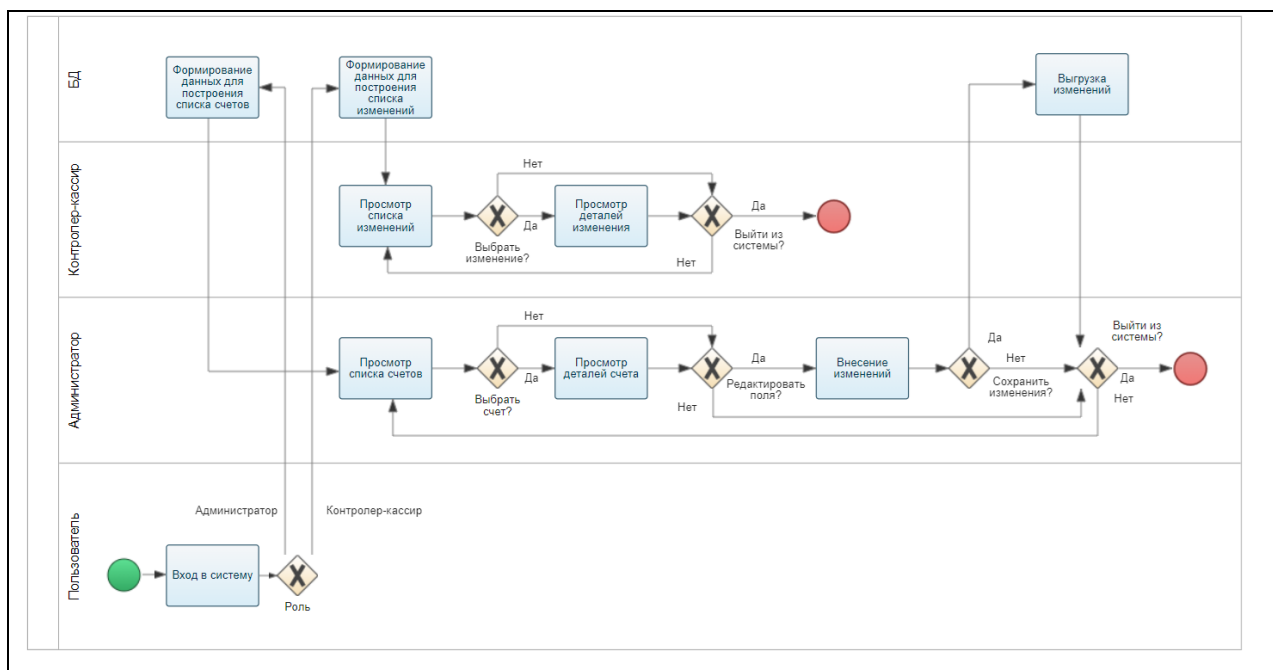


Рис. Г.1. Результат создания AD как проекта типа «BPMN Busines Process»

Таблица Г.1

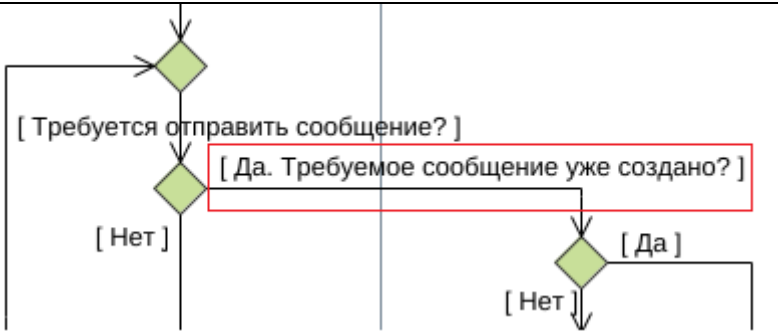
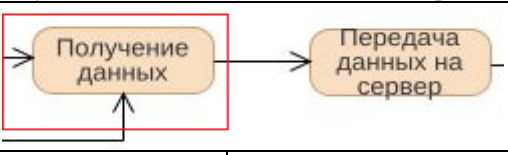
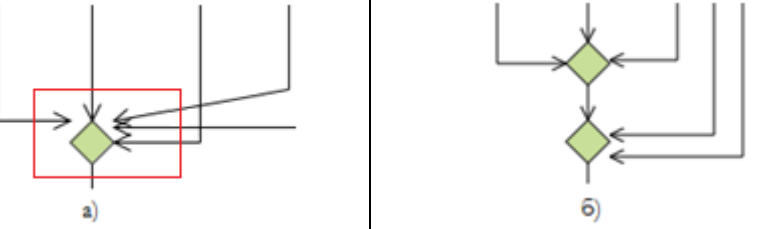
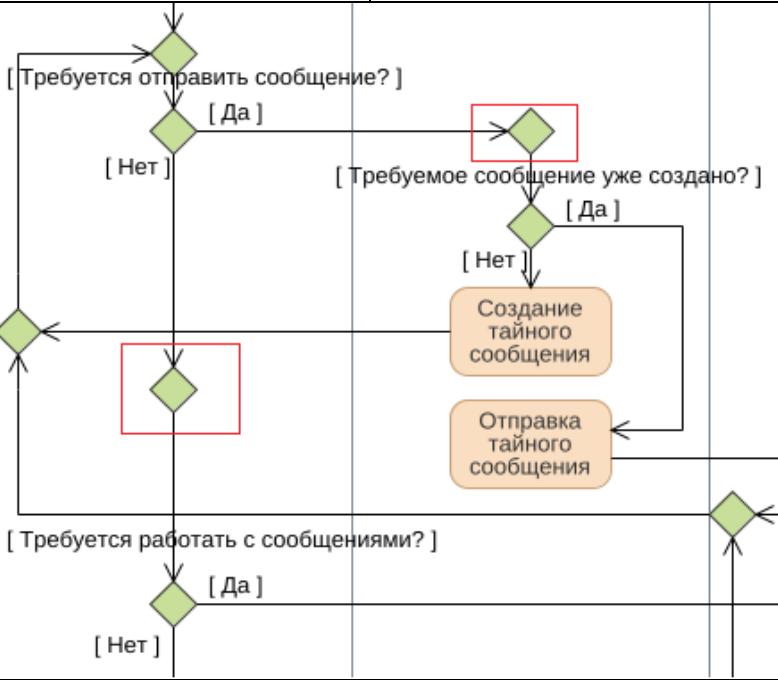
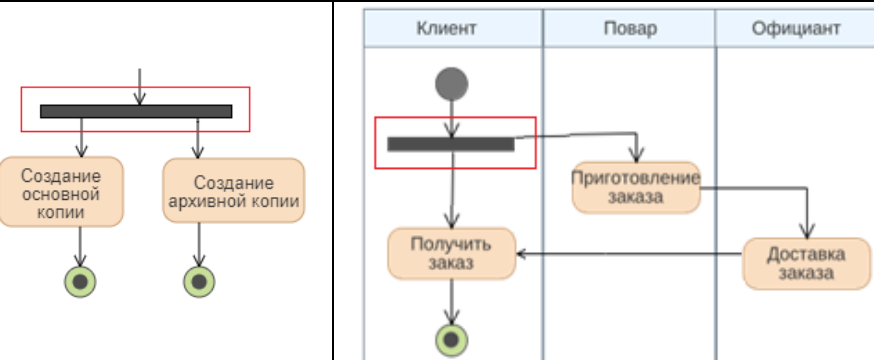
Типичные ошибки, допускаемые студентами при создании диаграмм активностей

Ошибка	Пример с ошибкой
1	2
Использование неверного элемента для дорожки пользователя	

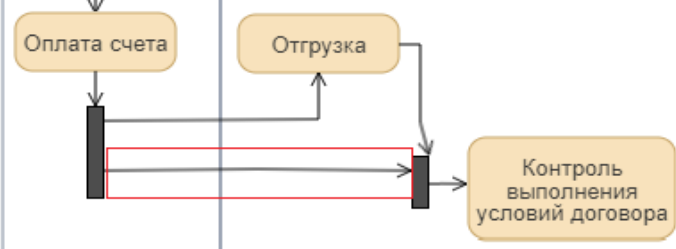
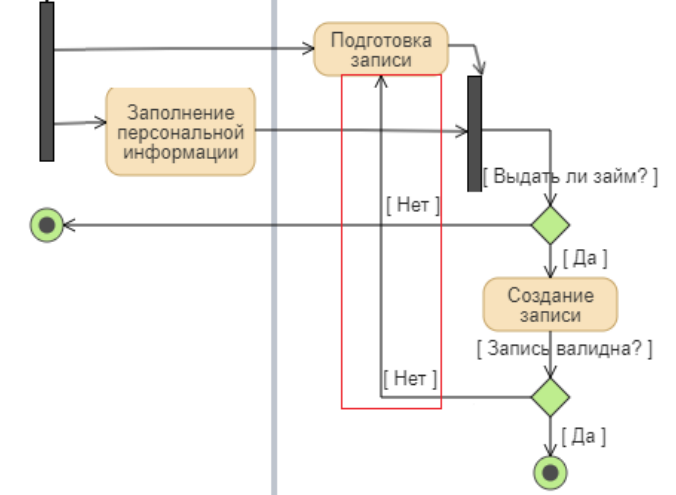
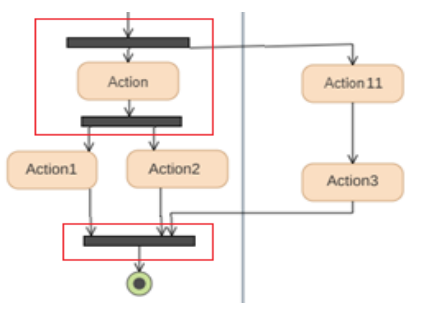
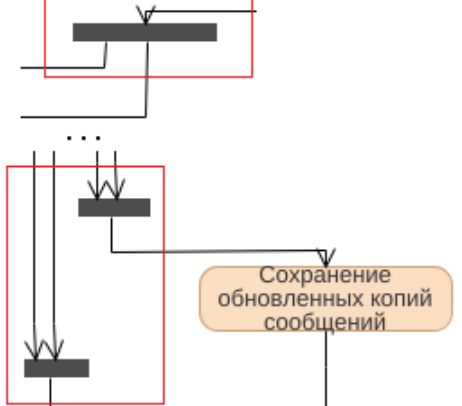
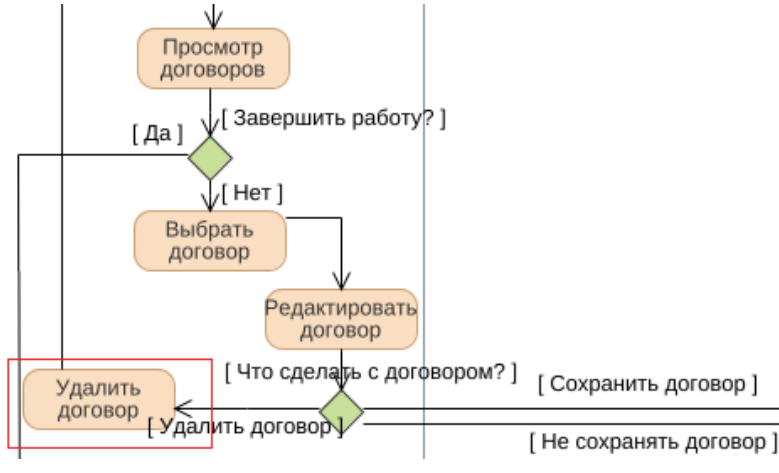
1	2
<p>Создание пустой дорожки участника (не удалена дорожка участника после удаления элементов из нее)</p>	
<p>Более одного начального узла</p>	
<p>Использование неверного элемента для конечного состояния деятельности</p>	
<p>Активности должны быть представлены соответствующим элементом AD. GenMyModel не позволит создать такие элементы при использовании панели «Activity diagram»</p>	
<p>Некорректное имя объекта (в имени участника используются символы, отличные от букв, цифр или символа подчеркивания)</p>	

1	2	
Неверное имя активности (начинается с глагола)		
Имена участников или названия активностей начинаются с маленькой буквы		
Отсутствие точки инициализации или конечного состояния (вместо начального или конечного состояния используется активность)		
Дублирование активностей		
Несоответствие количества входящих и выходящих переходов (активность имеет больше одного выходящего перехода)		
Отсутствие выходящего перехода из элемента (активность не может выступать в роли конечного состояния)		

1	2
Использование нескольких условных переходов подряд	
Использование комментария вместо условия	
Использование активности вместо условия	
Отсутствие знака вопроса в конце текста условия	
Альтернатива условного перехода не имеет подписи	
Неверный условный переход (условный переход имеет только одну альтернативу)	

1	2		
<p>Соединение подписи альтернативы и условия</p>			
<p>Отсутствие элемента слияния (использование активности вместо слияния)</p>			
<p>Неверное использование элемента слияния: а) ошибка, б) вариант исправления ошибки</p>			
<p>Ошибочное использование элемента слияния между условными переходами (использование элемента без смысловой нагрузки между условными переходами)</p>			
<p>Использование разветвителя без синхронизатора</p>			

1	2
<p>Использование синхронизатора без разветвителя</p>	
<p>Возможна повторная активация уже активной активности</p>	
<p>Альтернатива условного перехода выходит за пределы зоны между парными разветвителем и синхронизатором</p>	
<p>Синхронизатор никогда не будет активирован. Семантическая ошибка: продолжение бизнес-процесса возможно только при приобретении пакетов ВСЕХ трех видов, хотя предполагается выбор пакета только одного вида</p>	

1	2	
<p>Бизнес-процесс после разветвителя, не имеющий активностей</p>		
<p>Переход в активность, размещенную внутри пары синхронизатор-разветвитель</p>		
<p>Не согласовано количество разветвителей и синхронизаторов</p>		
<p>Несуществующая активность (удаление договора участником не предусмотрено, но присутствует в бизнес-процессе). Противоречие в бизнес-процессе (удалить договор можно только после его редактирования)</p>		

1	2	
<p>Подмена понятия «бизнес-процесс» понятием «алгоритм работы с ИС»:</p> <ol style="list-style-type: none"> 1. Использование лишней активности «Вход в систему». 2. Использование лишней активности «Выход в системы» как конечного состояния бизнес-процесса. 3. Использование БД как участника 		
<p>Ошибка размещения элементов. Такие ошибки затрудняют чтение и восприятие диаграммы.</p> <ol style="list-style-type: none"> 1. Наложение перехода в элемент слияния и перехода от разветвителя 		
<ol style="list-style-type: none"> 2. Наложение перехода в условный переход и его альтернативы. 3. Сдвинута подпись альтернативы 		

ПРИЛОЖЕНИЕ Д


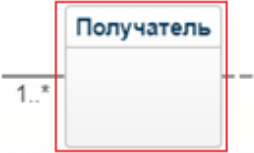
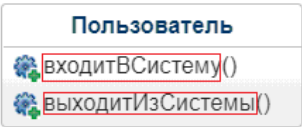
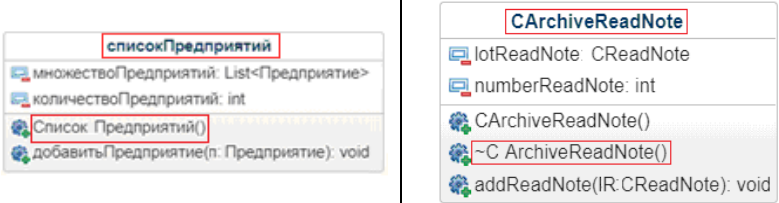
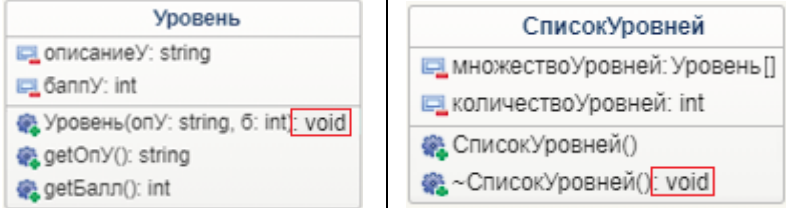
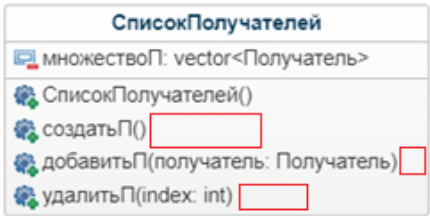
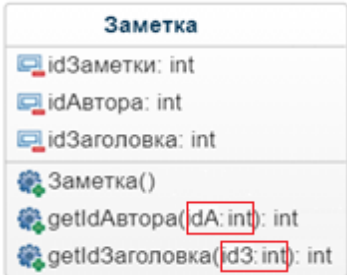
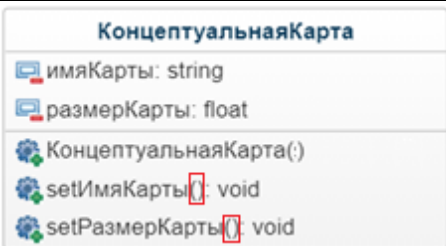
Типичные ошибки в CD студентов

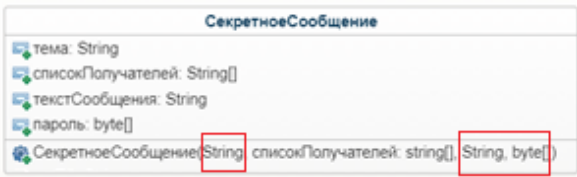
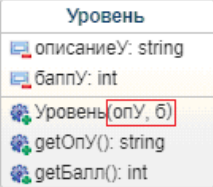
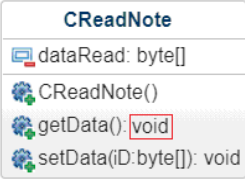
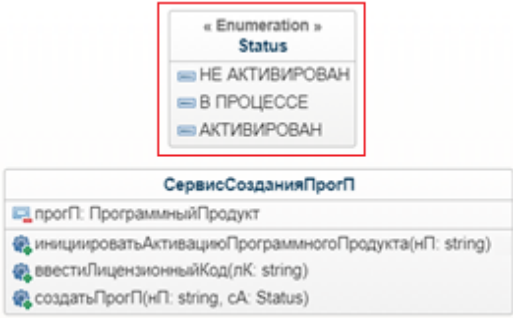


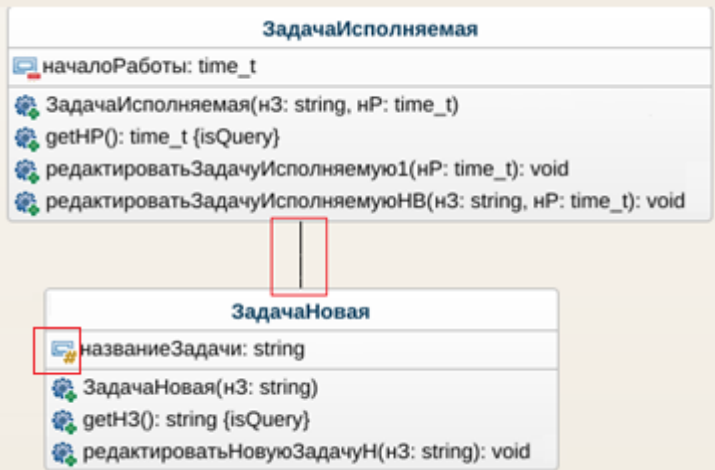
В таблице Д.1 представлены типичные ошибки, допускаемые студентами при создании диаграмм классов.

Таблица Д.1

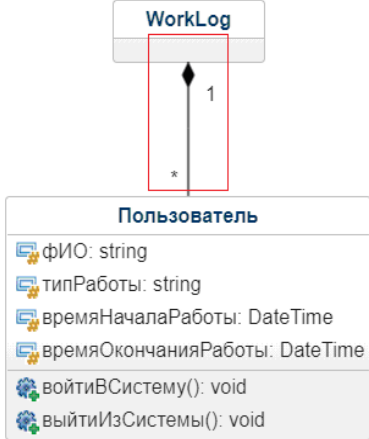
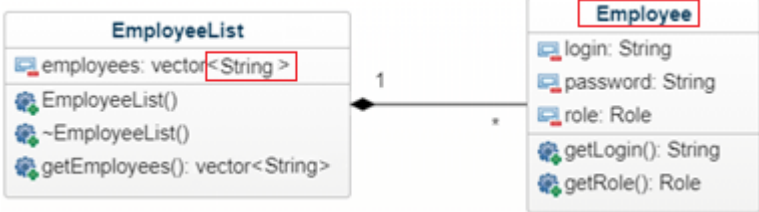

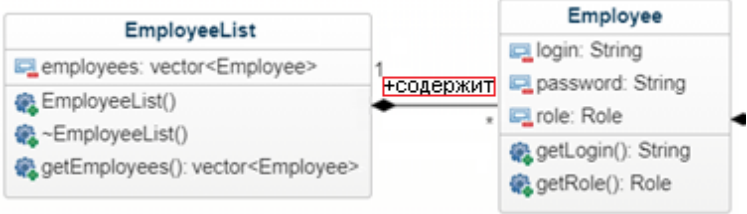
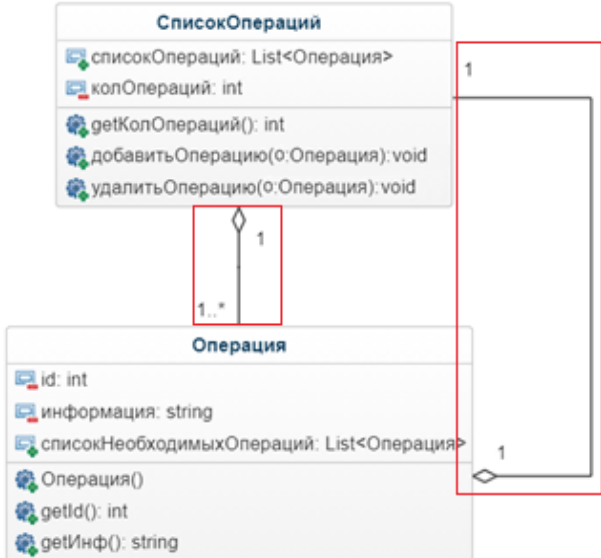
Типичные ошибки, допускаемые студентами при создании диаграмм активностей

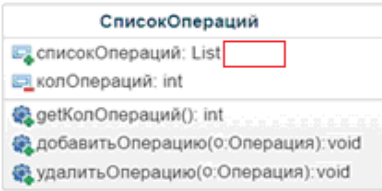



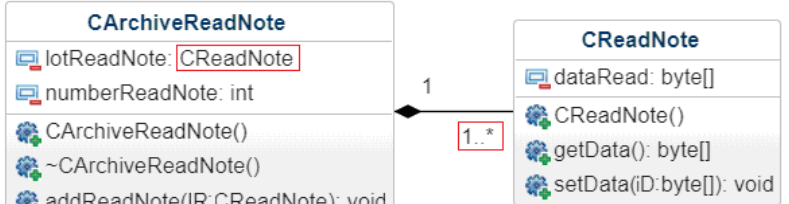
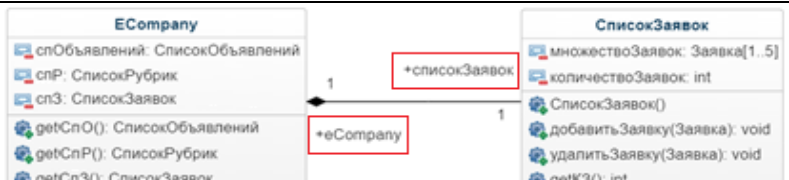
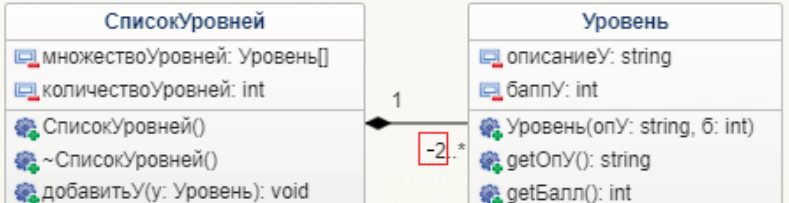
Ошибка	Пример с ошибкой
1	2
Элементы диаграммы должны быть представлены соответствующим элементом CD	
Некорректное имя класса (имена классов, интерфейсов, перечислений должны начинаться с большой буквы и не должны содержать пробелы)	
Некорректное имя атрибута или операции (имена атрибутов и операций должны начинаться с маленькой буквы (кроме конструкторов и деструкторов) и не должны содержать пробелы)	
Некорректное имя типа данных атрибута, аргумента операции, результата операции (имена типов данных должны соответствовать именам, используемым в целевом языке ООП, или именам других классов модели)	


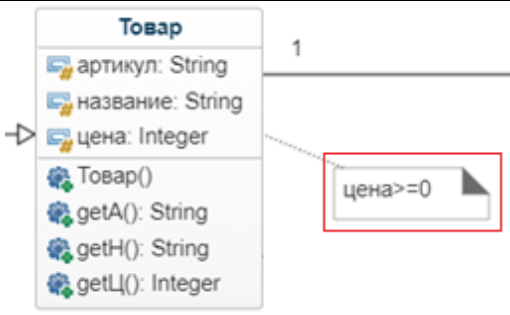
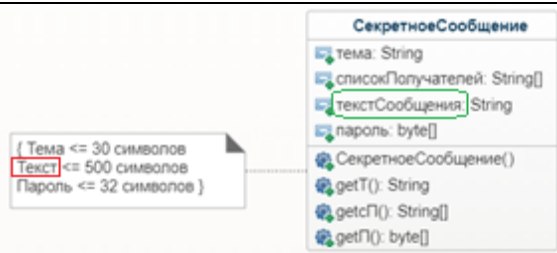
1	2
Используемый тип данных не соответствует контексту предметной области	 <pre> classDiagram class Project { name: char Project(n: char) Project() } </pre>
Класс (кроме участвующих в обобщении) или интерфейс не содержит хотя бы один атрибут или операцию	 <pre> classDiagram class Получатель { } classDiagram class "1..*" Получатель </pre>
Имя операции выражает действие (не глагол в повелительном наклонении)	 <pre> classDiagram class Пользователь { входитВСистему() выходитИзСистемы() } </pre>
Имя конструктора/деструктора не совпадает с именем класса	 <pre> classDiagram class списокПредприятий { множествоПредприятий: List<Предприятие> количествоПредприятий: int Список Предприятий() добавитьПредприятие(п: Предприятие): void } class CArchiveReadNote { lotReadNote: CReadNote numberReadNote: int CArchiveReadNote() ~CArchiveReadNote() addReadNote(IR: CReadNote): void } </pre>
У конструкторов или деструкторов указан тип результата	 <pre> classDiagram class Уровень { описаниеУ: string баллУ: int Уровень(онУ: string, б: int): void getОнУ(): string getБалл(): int } class СписокУровней { множествоУровней: Уровень [] количествоУровней: int СписокУровней() ~СписокУровней(): void } </pre>
У операции (кроме конструкторов и деструкторов) не указан тип результата	 <pre> classDiagram class СписокПолучателей { множествоП: vector<Получатель> СписокПолучателей() создатьП() добавитьП(получатель: Получатель) удалитьП(index: int) } </pre>
Лишний параметр у операции get	 <pre> classDiagram class Заметка { idЗаметки: int idАвтора: int idЗаголовка: int Заметка() getIdАвтора(idA: int): int getIdЗаголовка(id3: int): int } </pre>
Отсутствие параметров у операции set	 <pre> classDiagram class КонцептуальнаяКарта { имяКарты: string размерКарты: float КонцептуальнаяКарта() setИмяКарты() setРазмерКарты() } </pre>

1	2
<p>Не указано имя аргумента в описании операции</p>	
<p>Не указан тип данных аргумента в описании операции</p>	
<p>Неверный тип результата у операции get (get... должна возвращать значение с типом данных соответствующего атрибута)</p>	
<p>Класс, интерфейс и перечисление не имеет никаких связей</p>	
<p>Неверный тип связи между элементами «класс» и «перечисление» (должна быть типа «зависимость»)</p>	
<p>Избыточное имя роли ассоциации</p>	
<p>Класс, содержащий члены или операции типа protected, не связан со своими потомками обобщением</p>	

1	2
<p>Неверное направление у обобщения</p>	
<p>Дублирование атрибутов у классов-потомков</p>	
<p>Дублирование операций у классов-потомков</p>	
<p>Лишняя ассоциация между предком и потомком</p>	
<p>Лишняя ассоциация между классами-потомками</p>	

1	2
<p>Отсутствует атрибут для реализации агрегации/композиции</p>	
<p>При композиции или агрегации в классе-целое ни у одного из атрибутов в типе не содержится имя типа класса-части</p>	
<p>Неверное направление у агрегации</p>	
<p>Лишнее именование агрегации как ассоциации (используется имя роли)</p>	
<p>«Цикл» в агрегациях/композициях</p>	

1	2
<p>Не указан тип данных элементов атрибута, имеющего тип данных стандартного контейнера языка ООП</p>	
<p>Отсутствие операций (кроме конструктор/деструктор) с типом атрибута для объектной части, отличного от стандартного контейнера языка ООП</p>	
<p>Нет кратности у роли агрегации/композиции</p>	
<p>Неверная кратность агрегации (должна быть «один-многим»)</p>	
<p>Неверная кратность агрегации (должна быть «один-к-одному»)</p>	
<p>Избыточное имя роли ассоциации</p>	
<p>При указании кратности роли не все числа целые неотрицательные</p>	

1	2
<p>При указании кратности роли в виде промежутка он не начинается с меньшего значения</p>	
<p>Текст ограничения не заключен в фигурные скобки</p>	
<p>Ограничения для класса не соответствуют атрибутам этого класса</p>	

Учебное издание

Дацун Наталья Николаевна
Отинов Андрей Валерьевич
Гашева Татьяна Сергеевна
Власов Дмитрий Игоревич

Основы проектирования и реализации информационных систем

Учебно-методическое пособие

Редактор *Е. В. Шумилова*
Корректор *В. Е. Пирожкова*
Компьютерная верстка: *Н. Н. Дацун*

Объем данных 5,78 Мб
Подписано к использованию 13.10.2021

Размещено в открытом доступе
на сайте www.psu.ru
в разделе НАУКА / Электронные публикации
и в электронной мультимедийной библиотеке ELiS

Издательский центр
Пермского государственного
национального исследовательского университета
614990, г. Пермь, ул. Букирева, 15