

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Локтионова Оксана Геннадьевна

Должность: проректор по учебной работе

Дата подписания: 20.02.2017

Уникальный программный ключ:

0b817ca911e6668abb13a5d426d39e5f1c11eabb71e945014a4851fda56d089

## МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра информационных систем и технологий



" ДИАГРАММЫ ВАРИАНТОВ

ИСПОЛЬЗОВАНИЯ ЯЗЫКА UML "

Методические указания по выполнению  
практических работ для студентов направления  
подготовки бакалавров 09.03.02 Информационные  
системы 09.03.03 Прикладная информатика

Курск 2017

УДК 004.82 (075.8)

Составитель Т.И. Лапина

Рецензент

Доктор технических наук, профессор *Р.А. Томакова*

**Диаграммы вариантов использования языка UML:** методические указания по выполнению практических работ / Юго-Зап. гос. ун-т; сост. Т. И. Лапина. Курск, 2017. 36 с.: Библиогр.: с. 33.

Содержат краткие теоретические сведения о методах разработки функциональных моделей при проектировании информационных систем на основе использования нотаций языка UML.

Методические указания соответствуют требованиям программ по направлениям подготовки бакалавров: 09.03.02 Информационные системы, 09.03.03 Прикладная информатика.

Предназначены для студентов направления подготовки бакалавров 09.03.02 Информационные системы, 09.03.03, Прикладная информатика дневной и заочной форм обучения.

Текст печатается в авторской редакции

Подписано в печать 27.10.2017. Формат 60x84 1/16.  
Усл. печ. л. 2,1. Уч.-изд. л. 1,9. Тираж 100 экз. Заказ 1691. Бесплатно.  
Юго - Западный государственный университет.  
305040, г. Курск, ул. 50 лет Октября, 94.

## 1. Цель работы

Приобретение навыков использования инструментальных сред для разработки функциональных моделей при проектировании информационных систем на основе использования нотаций языка UML

## 2. Основные теоретические положения

*Унифицированный язык моделирования UML (Unified Modeling Language)* представляет собой язык для определения, представления, проектирования и документирования программных систем, организационно-экономических систем, технических систем и других систем различной природы. UML содержит стандартный набор диаграмм и нотаций самых разнообразных видов. Стандарт UML версии 2.1 предлагает следующий набор диаграмм для моделирования:

- *диаграммы вариантов использования (use case diagrams)* - для моделирования бизнес-процессов организации и требований к создаваемой системе;

- *диаграммы классов (class diagrams)* - для моделирования статической структуры классов объектов системы и связей между ними;

- *диаграммы поведения системы (behavior diagrams):*

  - диаграммы взаимодействия (interaction diagrams):*

    - диаграммы последовательности (sequence diagrams)* и *кооперативные диаграммы (collaboration diagrams)* - для моделирования процесса обмена сообщениями между объектами;

  - *диаграммы состояний (statechart diagrams)* - для моделирования поведения объектов системы при переходе из одного состояния в другое;

  - *диаграммы деятельностей (activity diagrams)* - для моделирования поведения системы в рамках различных вариантов использования, или моделирования деятельностей;

  - *диаграммы реализации (implementation diagrams):*

*диаграммы компонентов (component diagrams)* - для моделирования иерархии компонентов (подсистем) системы;

*диаграммы размещения (deployment diagrams)* — для моделирования физической архитектуры системы.

### 3. CASE-средство RATIONAL ROSE

Rational Rose - семейство объектно-ориентированных CASE-средств фирмы Rational Software Corporation предназначено для автоматизации процессов анализа и проектирования ПО, а также для генерации кодов на различных языках и выпуска проектной документации. Rational Rose использует метод объектно-ориентированного анализа и проектирования, основанный на языке UML. Текущая версия Rational Rose реализует генерацию кодов программ для C++, Visual C++, Visual Basic, Java, PowerBuilder, CORBA Interface Definition Language (IDL), генерацию описаний баз данных для ANSI SQL, Oracle, MS SQL Server, IBM DB2, Sybase, а также позволяет разрабатывать проектную документацию в виде диаграмм и спецификаций. Кроме того, Rational Rose содержит средства реверсного инжиниринга программ и баз данных, обеспечивающие повторное использование программных компонентов в новых проектах.

#### Структура и функции

В основе работы Rational Rose лежит построение диаграмм и спецификаций UML, определяющих архитектуру системы, ее статические и динамические аспекты. В составе Rational Rose можно выделить шесть основных структурных компонентов:

- репозиторий,
- графический интерфейс пользователя,
- средства просмотра проекта (браузер),
- средства контроля проекта,
- средства сбора статистики,
- генератор документов.

К ним добавляются генератор кодов (индивидуальный для каждого языка) и анализатор для C++, обеспечивающий реверсный инжиниринг.

*Репозиторий* представляет собой базу данных проекта.

**Браузер** обеспечивает «навигацию» по проекту, в том числе перемещение по иерархиям классов и подсистем, переключение от одного вида диаграмм к другому и т.д.

**Средства контроля и сбора статистики** дают возможность находить и устранять ошибки по мере развития проекта, а не после завершения его описания.

**Генератор отчетов** формирует тексты выходных документов на основе содержащейся в репозитории информации.

**Средства автоматической генерации кодов** программ на языке C++, используя информацию, содержащуюся в диаграммах классов и компонентов, формируют файлы заголовков и файлы описаний классов и объектов. Создаваемый таким образом «скелет» программы может быть уточнен путем прямого программирования на языке C++.

**Анализатор кодов C++** реализован в виде отдельного программного модуля. Его назначение - создавать модули проектов Rational Rose на основе информации, содержащейся в определяемых пользователем исходных текстах на C++. В процессе работы анализатор осуществляет контроль правильности исходных текстов и диагностику ошибок. Модель, полученная в результате его работы, может целиком или фрагментарно использоваться в различных проектах. Анализатор обладает широкими возможностями настройки по входу и выходу. Например, можно определить типы исходных файлов, базовый компилятор, задать, какая информация должна быть включена в формируемую модель, и какие элементы выходной модели следует выводить на экран. Таким образом, Rational Rose/C++ обеспечивает возможность повторного использования программных компонентов.

В результате разработки проекта с помощью CASE-средства Rational Rose формируются следующие документы:

- **диаграммы UML**, в совокупности представляющие собой модель разрабатываемой программной системы;
- **спецификации** классов, объектов, атрибутов и операций;
- заготовки **текстов программ**.

Тексты программ являются заготовками для последующей работы программистов. Состав информации, включаемой в программные файлы, определяется либо по умолчанию, либо по усмотрению пользователя. В дальнейшем эти исходные тексты развиваются программистами в полноценные программы.

## **Взаимодействие с другими средствами и организация групповой работы**

Для поддержки командной работы над проектом на каждой стадии жизненного цикла ПО имеется интегрированный набор продуктов *Rational Suite*. Rational Suite существует в следующих вариантах:

- Rational Suite AnalystStudio - предназначен для определения и управления полным набором требований к разрабатываемой системе;
- Rational Suite DevelopmentStudio - предназначен для проектирования и реализации ПО;
- Rational Suite TestStudio - представляет собой набор продуктов, предназначенных для автоматического тестирования приложений;
- Rational Suite Enterprise - обеспечивает поддержку полного жизненного цикла ПО и предназначен как для менеджеров проекта, так и отдельных разработчиков, исполняющих несколько функциональных ролей в команде разработчиков.

В состав Rational Suite, кроме Rational Rose, входят следующие компоненты:

- Rational Requisite Pro - средство управления требованиями, предназначенное для организации совместной работы группы разработчиков. Оно позволяет команде разработчиков создавать, структурировать, устанавливать приоритеты, отслеживать, контролировать изменения требований, возникающих на любом этапе разработки компонентов приложения;
- Rational ClearCase - средство управления конфигурацией ПО;
- Rational SoDA - средство автоматической генерации проектной документации;
- Rational ClearQuest - средство для управления изменениями и отслеживания дефектов в проекте на основе средств e-mail и Web;
- Rational TeamTest - средство автоматического обнаружения ошибок во время выполнения программы и генерации сценариев для проведения регрессионного тестирования;
- Rational Robot - средство для создания, модификации и автоматического запуска тестов;
- Rational Purify - средство для локализации трудно обнаруживаемых ошибок времени выполнения программы;
- Rational PureCoverage - средство идентификации участков кода, пропущенных при тестировании;

- Rational Quantify - средство количественного определения узких мест, влияющих на общую эффективность работы программы;
- Rational Suite PerformanceStudio - средство нагрузочного тестирования приложений «клиент-сервер» и Web-приложений.

Для организации групповой работы в Rational Rose возможно разбиение модели на управляемые подмодели. Каждая из них независимо сохраняется на диске или загружается в модель. В качестве подмодели может выступать пакет или подсистема.

### **Среда функционирования**

Rational Rose функционирует на различных платформах: IBM PC (Windows 95/98/NT), Sun SPARCstations (UNIX, Solaris, SunOS), Hewlett-Packard (HP UX), IBM RS/6000 (AIX).

## **4. РАБОТА В СРЕДЕ RATIONAL ROSE**

Элементы экрана интерфейса Rose - это браузер, окно документации, панели инструментов, окно диаграммы и журнал. Их назначение заключается в следующем:

- браузер (browser) - используется для быстрой навигации по модели;
- окно документации (documentation window) - применяется для работы с текстовым описанием элементов модели;
- панели инструментов (toolbars) - применяются для быстрого доступа к наиболее распространенным командам;
- окно диаграммы (diagram window) - используется для просмотра и редактирования одной или нескольких диаграмм UML;
- журнал (log) - применяется для просмотра ошибок и отчетов о выполнении различных команд.

На рис. 1 показаны различные части интерфейса Rose.

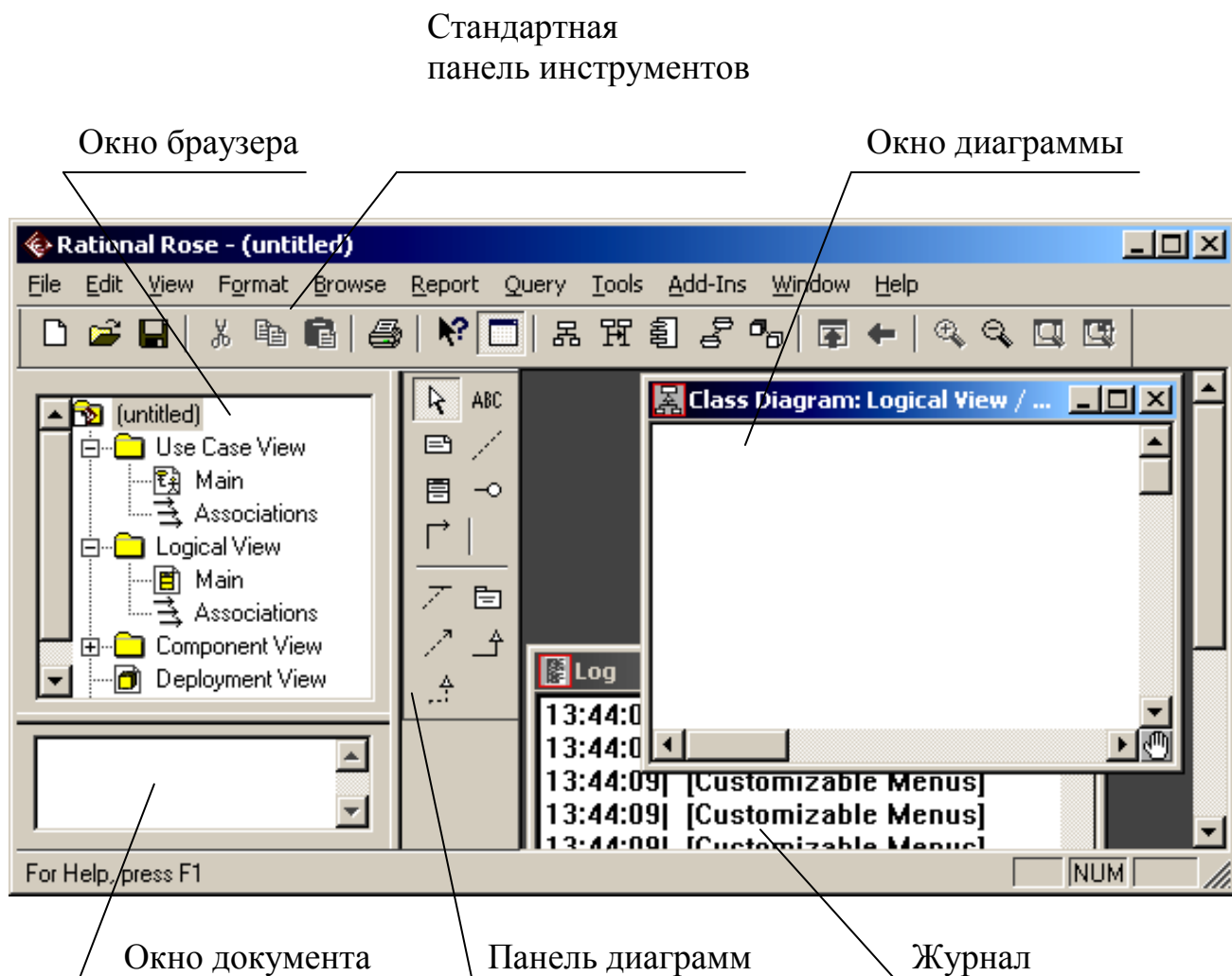


Рисунок 1– Интерфейс Rose

**Браузер** - это иерархическая структура, позволяющая осуществлять навигацию по модели. Все, что добавляется к ней - действующие лица, варианты использования, классы, компоненты, будет показано в окне браузера.

С помощью браузера можно:

- добавлять к модели элементы;
- просматривать существующие элементы модели;
- просматривать существующие связи между элементами модели;
- перемещать элементы модели;
- переименовывать эти элементы;
- добавлять элементы модели к диаграмме;
- связывать элемент с файлом или адресом Интернета;



- группировать элементы в пакеты;
- работать с детализированной спецификацией элемента;
- открывать диаграмму.

Браузер поддерживает четыре представления (view): представление вариантов использования, компонентов, размещения и логическое представление. Все они и содержащиеся в них элементы модели описаны ниже.

Организация браузера представляет собой древовидную структуру. Каждый элемент модели может содержать другие элементы, находящиеся ниже его в иерархии. Знак «-» около элемента означает, что его ветвь полностью раскрыта. Знак «+» - что его ветвь свернута.

**Окно документации.** С его помощью можно документировать элементы модели Rose. Например, можно сделать краткое описание каждого действующего лица. При документировании класса все, что будет написано в окне документации, появится затем как комментарий в сгенерированном коде, что избавляет от необходимости впоследствии вносить эти комментарии вручную. Документация будет выводиться также в отчетах, создаваемых в среде Rose.

**Панели инструментов** Rose обеспечивают быстрый доступ к наиболее распространенным командам. В этой среде существуют два типа панелей инструментов: стандартная панель и панель диаграммы. Стандартная панель видна всегда, ее кнопки соответствуют командам, которые могут использоваться для работы с любой диаграммой. Панель диаграммы своя для каждого типа диаграмм UML.

Все панели инструментов могут быть изменены и настроены пользователем. Для этого выберите пункт меню **Tools > Options**, затем вкладку **Toolbars**.

Показать или скрыть стандартную панель инструментов (или панель инструментов диаграммы) можно следующим образом.

1. Выберите пункт **Tools > Options**.
2. Выберите вкладку **Toolbars**.
3. Чтобы сделать видимой или невидимой стандартную панель инструментов (панель диаграммы), пометьте (или снимите пометку) контрольный переключатель **Show Standard ToolBar** (или **Show Diagram ToolBar**).

Для того чтобы увеличить размер кнопок на панели инструментов:

1. Щелкните правой кнопкой мыши по требуемой панели.
2. Выберите во всплывающем меню пункт *Use Large Buttons* (Использовать большие кнопки).

Для настройки панели инструментов:

1. Щелкните правой кнопкой мыши по требуемой панели.
2. Выберите пункт *Customize* (Настроить).
3. Чтобы добавить или удалить кнопки, выберите соответствующую кнопку и затем щелкните мышью по кнопке *Add* (Добавить) или *Remove* (Удалить), как показано на рис. 8.

**Окно диаграммы.** В нем видно, как выглядит одна или несколько диаграмм UML-модели. При внесении в элементы диаграммы изменений Rose автоматически обновит браузер. Аналогично при внесении изменений в элемент с помощью браузера Rose автоматически обновит соответствующие диаграммы. Это помогает поддерживать модель в непротиворечивом состоянии.

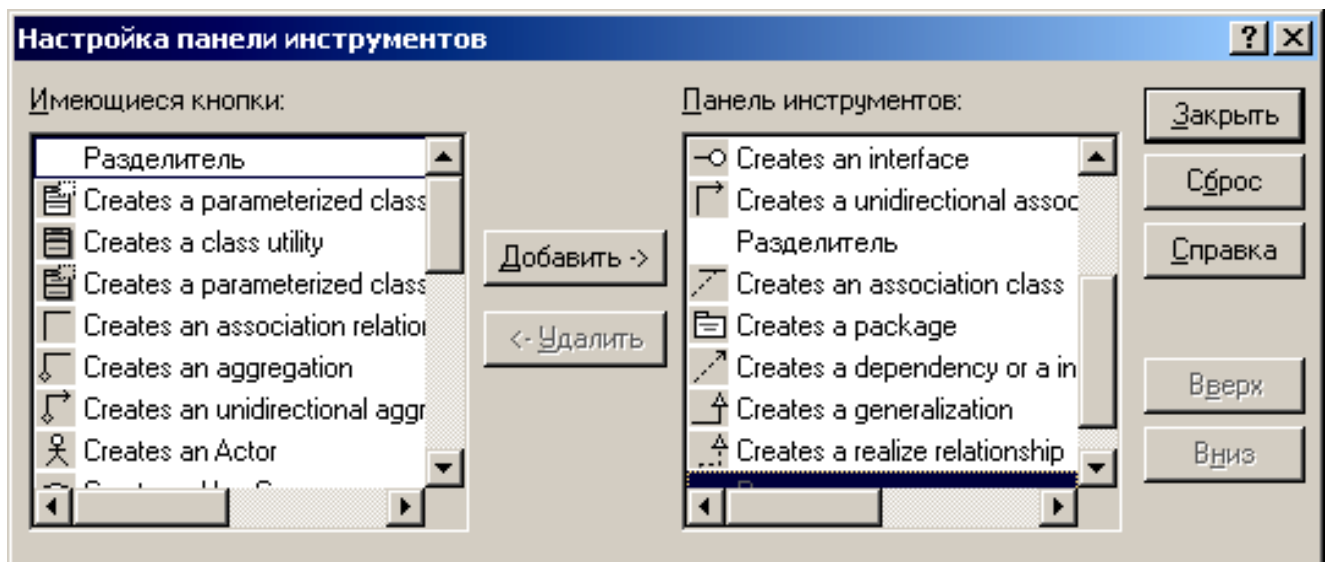


Рисунок 2– Настройка стандартной панели инструментов

**Журнал.** По мере работы над вашей моделью определенная информация будет направляться в окно журнала. Например, туда помещаются сообщения об ошибках, возникающих при генерации кода. Не существует способа закрыть журнал совсем, но его окно может быть минимизировано.

### Четыре представления модели Rose

В модели Rose поддерживаются четыре представления - это представление вариантов использования, логическое представление, пред-

ставление компонентов и представление размещения. Каждое из них предназначено для своих целей и для соответствующей аудитории. В этой практической работе мы кратко рассмотрим первое представление.

***Представление вариантов использования (Use Case View)*** содержит:

- Действующих лиц.
- Варианты использования.
- Документацию по вариантам использования, детализирующую происходящие в них процессы (потоки событий), включая обработку ошибок. Эта пиктограмма соответствует внешнему файлу, прикрепленному к модели Rose. Вид пиктограммы зависит от приложения, используемого для документирования потока событий.
- Диаграммы вариантов использования. Обычно у системы бывает несколько таких диаграмм, каждая из которых показывает подмножество действующих лиц и/или вариантов использования.
- Пакеты, являющиеся группами вариантов использования и/или действующих лиц.

Представление вариантов использования может также содержать некоторые диаграммы последовательности и кооперативные диаграммы.

На рис. 3 изображено представление вариантов использования в браузере Rose, где:

- Main*** – главная диаграмма вариантов использования;
- Bank Officer, Credit System, Customer*** – действующие лица;
- Change PIN, Deposit Funds, Make Payment, Transfer Funds, View Balance, Withdraw Money*** – варианты использования;
- Associations*** – связи.

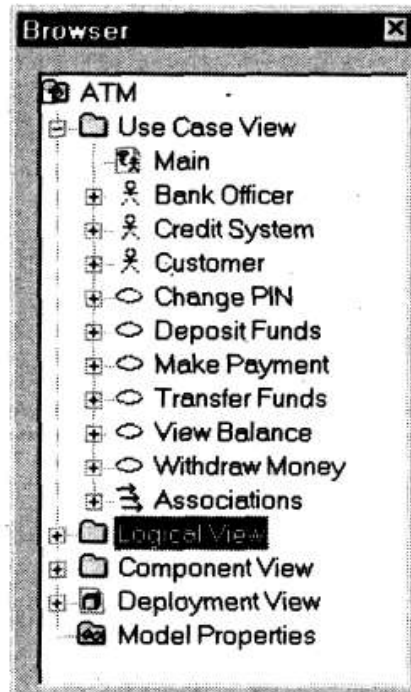


Рис. 3. Представление вариантов использования

## 5. Порядок выполнения задания

### 5.1 Постановка задачи создания системы

Перед руководителем информационной службы университета ставится задача разработки новой клиент-серверной системы регистрации студентов взамен старой системы на мейнфрейме. Новая система должна позволять студентам регистрироваться на курсы и просматривать свои таблицы успеваемости с персональных компьютеров, подключенных к локальной сети университета. Профессора должны иметь доступ к онлайн-системе, чтобы указать курсы, которые они будут читать, и проставить оценки за курсы.

Из-за недостатка средств университет не в состоянии заменить сразу всю существующую систему. База данных, содержащая всю информацию о курсах (каталог курсов), остается функционировать в прежнем виде. Эта база данных поддерживается реляционной СУБД. Новая система будет работать с существующей БД в режиме доступа, без обновления.

В начале каждого семестра студенты могут запросить каталог курсов, содержащий список курсов, предлагаемых в данном семестре. Информация о каждом курсе должна включать имя профессора, наимено-

вание кафедры и требования к предварительному уровню подготовки (прослушанным курсам).

Новая система должна позволять студентам выбирать 4 курса в предстоящем семестре. В дополнение каждый студент может указать 2 альтернативных курса на тот случай, если какой-либо из выбранных им курсов окажется уже заполненным или отмененным. На каждый курс может записаться не более 10 и не менее 3 студентов (если менее 3, то курс будет отменен). В каждом семестре существует период, когда студенты могут изменить свои планы. В это время студенты должны иметь доступ к системе, чтобы добавить или удалить выбранные курсы. После того как процесс регистрации некоторого студента завершен, система регистрации направляет информацию в расчетную систему, чтобы студент мог внести плату за семестр. Если курс окажется заполненным в процессе регистрации, студент должен быть извещен об этом до окончательного формирования его личного учебного плана.

В конце семестра студенты должны иметь доступ к системе для просмотра своих электронных таблиц успеваемости. Поскольку эта информация конфиденциальная, система должна обеспечивать ее защиту от несанкционированного доступа.

Профессора должны иметь доступ к онлайн-системе, чтобы указать курсы, которые они будут читать, и просмотреть список студентов, записавшихся на их курсы. Кроме того, профессора должны иметь возможность проставить оценки за курсы.

## 5.2. Составление глоссария проекта

Глоссарий предназначен для описания терминологии предметной области. Он может быть использован как неформальный словарь данных системы.

Ниже приведены термины и их значения.

<b>Термин</b>	<b>Значение</b>
Курс (Course)	Учебный курс, предлагаемый университетом
Конкретный курс (Course Offering)	Конкретное чтение данного курса в конкретном семестре (один и тот же курс может вестись в нескольких параллельных сессиях). Включает точные дни недели и время
Каталог курсов (Course Catalog)	Полный каталог всех курсов, предлагаемых университетом

Расчетная система (Billing System)	Система обработки информации об оплате курсов
Оценка	Оценка, полученная студентом за конкретный курс
Профессор (Professor)	Преподаватель университета
Табель успеваемости (Report Card)	Все оценки за все курсы, полученные студентом в данном семестре
Список курса (Roster)	Список всех студентов, записавшихся на конкретный курс
Студент (Student)	Личность, проходящая обучение в университете
Учебный график (Schedule)	Курсы, выбранные студентом в текущем семестре

### 5.3. Описание дополнительных спецификаций

Назначение дополнительных спецификаций - определить требования к системе регистрации курсов, которые не охватывает модель вариантов использования. Вместе они образуют полный набор требований к системе.

Дополнительные спецификации определяют нефункциональные требования к системе, такие, как надежность, удобство использования, производительность, сопровождаемость, а также ряд функциональных требований, являющихся общими для нескольких вариантов использования.

#### Функциональные возможности

Система должна обеспечивать многопользовательский режим работы.

Если конкретный курс оказывается заполненным в то время, когда студент формирует свой учебный график, включающий данный курс, то система должна известить его об этом.

#### Удобство использования

Пользовательский интерфейс должен быть Windows 95/98-совместимым.

#### Надежность

Система должна быть в работоспособном состоянии 24 часа в день 7 дней в неделю, время простоя - не более 10%.

## **Производительность**

Система должна поддерживать до 2000 пользователей, одновременно работающих с центральной базой данных пользователей, и до 500 пользователей, одновременно работающих с локальными серверами.

## **Безопасность**

Система не должна позволять студентам изменять любые учебные графики, кроме своих собственных, а также не должна позволять профессорам модифицировать конкретные курсы, выбранные другими профессорами.

Только профессора имеют право ставить студентам оценки.

Только регистратор может изменять любую информацию о студентах.

## **Проектные ограничения**

Система должна быть интегрирована с существующей системой каталога курсов, функционирующей на основе реляционной СУБД.

### **5.4. Создание модели вариантов использования**

Действующие лица:

- *Student* (Студент) - записывается на курсы.
- *Professor* (Профессор) - выбирает курсы для преподавания.
- *Registrar* (Регистратор) - формирует учебный план и каталог курсов, ведет все данные о курсах, профессорах и студентах.
- *Billing System* (Расчетная система) - получает от данной системы информацию по оплате курсов.
- *Course Catalog* (Каталог курсов) - передает в систему информацию из каталога курсов, предлагаемых университетом.

### **Упражнение 1. Создание действующих лиц в среде Rational Rose**

Для того чтобы поместить действующее лицо в браузер:

1. Щелкните правой кнопкой мыши по пакету представления вариантов использования в браузере.
2. Выберите пункт *New > Actor* в открывшемся меню.
3. В браузере появится новое действующее лицо под названием

*NewClass*. Слева от его имени вы увидите пиктограмму действующего лица UML.

4. Выделив новое действующее лицо, введите его имя.
5. После создания действующих лиц сохраните модель под именем *coursereg(analysis)* с помощью пункта меню *File > Save*.

Результат выполнения упражнения показан на рис. 4.

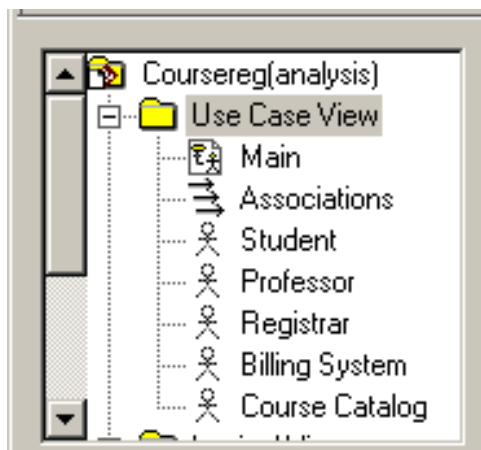


Рисунок 4– Представление действующих лиц в браузере

### Варианты использования

Исходя из потребностей действующих лиц выделяются следующие варианты использования:

- *Login* (Войти в систему).
- *Register for Courses* (Зарегистрироваться на курсы).
- *View Report Card* (Просмотреть таблицу успеваемости).
- *Select Courses to Teach* (Выбрать курсы для преподавания).
- *Submit Grades* (Проставить оценки).
- *Maintain Professor Information* (Вести информацию о профессорах).
- *Maintain Student Information* (Вести информацию о студентах).
- *Close Registration* (Закрывать регистрацию).

### Упражнение 2. Создание вариантов использования в среде Rational Rose

Для того чтобы поместить вариант использования в браузер:

1. Щелкните правой кнопкой мыши по пакету представления вариантов использования в браузере.
2. Выберите в появившемся меню пункт *New > Use Case*.



3. Новый вариант использования под названием *NewUseCase* появится в браузере. Слева от него будет видна пиктограмма варианта использования UML.
  4. Выделив новый вариант использования, введите его название.
- Результат выполнения упражнения показан на рис.5.

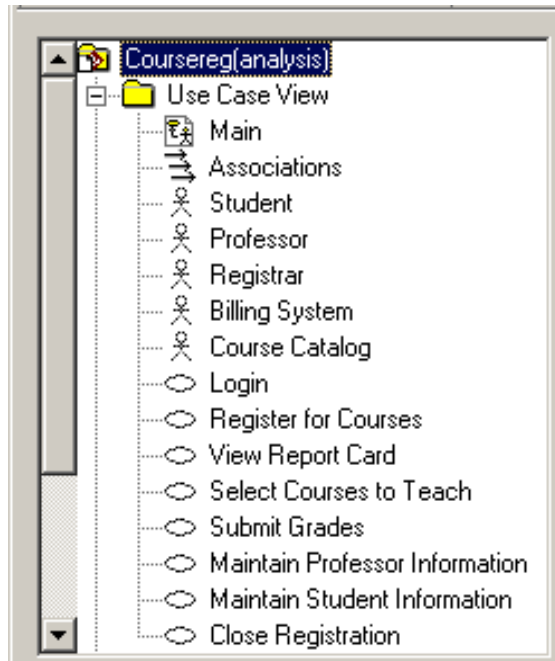


Рисунок 5—Представление вариантов использования в браузере

### **Диаграмма вариантов использования**

Создайте диаграмму вариантов использования для системы регистрации. Требуемые для этого действия подробно перечислены далее. Готовая диаграмма вариантов использования изображена на рис. 6.

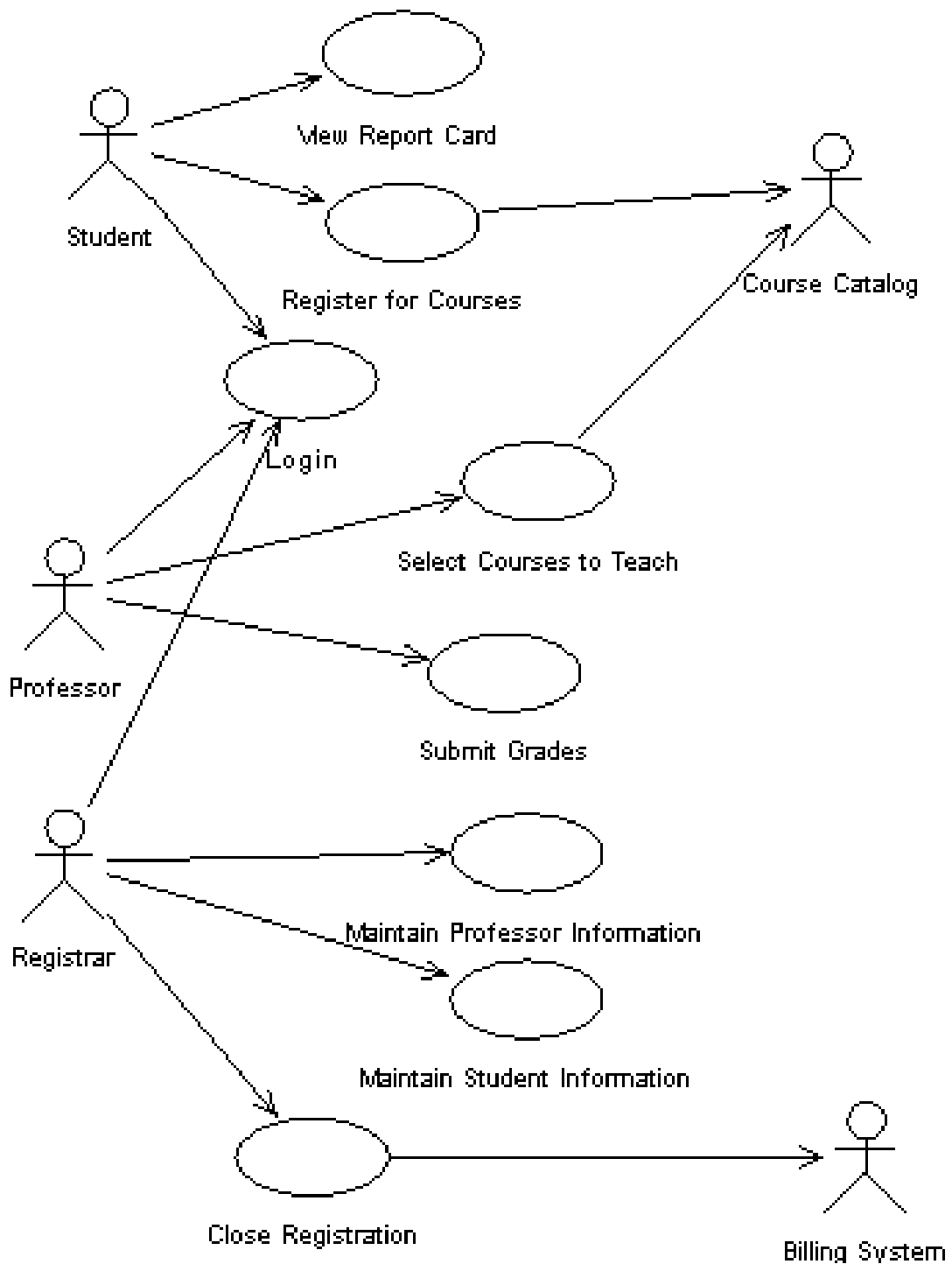


Рисунок 6—Диаграмма вариантов использования для системы регистрации

В среде Rose диаграммы вариантов использования создаются в представлении вариантов использования. Главная диаграмма (*Main*) предлагается по умолчанию. Для моделирования системы можно затем разработать необходимое количество дополнительных диаграмм.

Для того чтобы открыть уже созданную диаграмму вариантов использования (Main):

1. Откройте данное представление, щелкнув по значку «+» рядом с представлением вариантов использования в браузере.
2. Откройте главную диаграмму *Main*, дважды щелкнув мышью. Строка заголовка изменится, включив фразу [*Use Case Diagram: Use Case view / Main*].

Для создания новой диаграммы вариантов использования:

1. Щелкните правой кнопкой мыши по пакету представления вариантов использования *UseCaseView* в браузере.
2. Выберите пункт *New > Use Case Diagram* из локального меню.
3. Выделив новую диаграмму, введите ее имя.
4. Дважды щелкните по названию этой диаграммы в браузере, чтобы открыть ее.

### Упражнение 3. Построение диаграммы вариантов использования

1. Откройте диаграмму вариантов использования *Main*.
2. Перетащите действующее лицо или вариант использования мышью из браузера на диаграмму вариантов использования.
3. С помощью кнопки *Unidirectional Association* (Однонаправленная ассоциация) панели инструментов нарисуйте ассоциации между действующими лицами и вариантами использования.

Наличие общего варианта использования *Login* для трех действующих лиц позволяет обобщить их поведение и ввести новое действующее лицо *Any User*. Эти лица связываются с *Any User* связью *Generalization* (Обобщение). Модифицированная диаграмма вариантов использования показана на рис. 13.

### Упражнение 4. Добавление описаний к вариантам использования

1. Выделите в браузере вариант использования *Register for Courses*.
2. В окне документации введите следующее описание к этому варианту использования: «This use case allows a student to register for courses in the current semester» («Этот вариант использования дает студенту возможность зарегистрироваться на курсы в текущем семестре»).
3. Создайте с помощью MS Word три текстовых файла с описаниями, приведенными ниже, вариантов использования *Login* (Войти в систему), *Register for Courses* (Зарегистрироваться на курсы) и *Close Registration* (Закреть регистрацию).

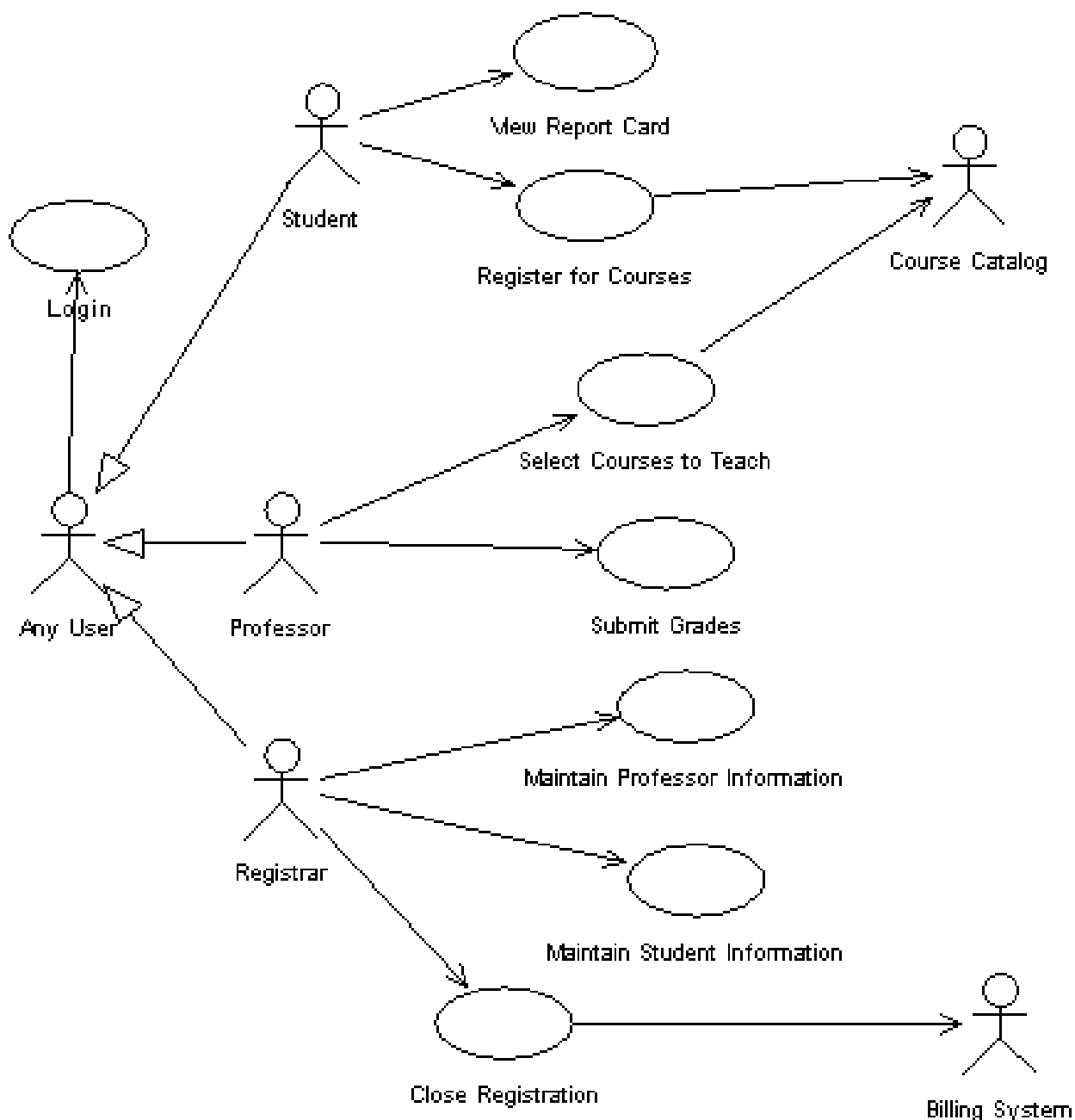


Рисунок 7– Модифицированная диаграмма вариантов использования

### **Вариант использования Login**

**Краткое описание.** Данный вариант использования описывает вход пользователя в систему регистрации курсов.

#### **Основной поток событий**

Данный вариант использования начинает выполняться, когда пользователь хочет войти в систему регистрации курсов.

Действия актера	Действия (Отклик) системы
1. Вариант использования начинается после запуска пользователем	2. Система запрашивает имя пользователя и пароль.
3. Пользователь вводит имя и пароль.	4. Система проверяет имя и пароль, после чего открывается доступ в систему.
<u>Альтернативный поток событий A1.</u> Неправильное имя/пароль.	

### **Альтернативные потоки**

#### ***Альтернативный поток событий A1. Неправильное имя/пароль.***

Если во время выполнения Основного потока обнаружится, что пользователь ввел неправильное имя и/или пароль, система выводит сообщение об ошибке. Пользователь может вернуться к началу Основного потока или отказаться от входа в систему, при этом выполнение варианта использования завершается.

### **Предусловия**

Отсутствуют.

### **Постусловия**

Если вариант использования выполнен успешно, пользователь входит в систему. В противном случае состояние системы не изменяется.

### **Вариант использования Register for Courses**

**Краткое описание.** Данный вариант использования позволяет студенту зарегистрироваться на конкретные курсы в текущем семестре. Студент может изменить свой выбор (обновить или удалить курсы), если изменение выполняется в установленное время в начале семестра. Система каталога курсов предоставляет список всех конкретных курсов текущего семестра.

### **Основной поток событий**

Данный вариант использования начинает выполняться, когда студент хочет зарегистрироваться на конкретные курсы или изменить свой график курсов.

Действия актера	Действия (Отклик) системы
	1. Система запрашивает требуемое действие (создать, обновить, удалить график).
2. Студент указывает действие, после чего выполняется один из подчиненных потоков (создать, обновить, удалить или принять график).	
<u>Подчиненный поток событий П1. Создать график</u>	
<u>Подчиненный поток событий П2. Обновить график</u>	
<u>Подчиненный поток событий П3. Удалить график</u>	
<u>Подчиненный поток событий П4. Принять график</u>	
<u>Альтернативный поток событий А1. Сохранить график (вместо принятия графика)</u>	
<u>Альтернативный поток событий А2. Не выполнены предварительные требования, курс заполнен или имеют место конфликты графика (вместо принятия графика)</u>	
<u>Альтернативный поток событий А3. График не найден (вместо «Обновить график» или «Удалить график»)</u>	
<u>Альтернативный поток событий А4. Система каталога курсов недоступна.</u>	
<u>Альтернативный поток событий А5. Регистрация на курсы закончена.</u>	
<u>Альтернативный поток событий А6. Удаление отменено.</u>	

### **Подчиненные потоки**

#### ***Подчиненный поток событий П1. Создать график***

Действия актера	Действия (Отклик) системы
	1. Система выполняет поиск в каталоге курсов доступных конкретных курсов и выводит их список.
2. Студент выбирает из списка 4 основных и 2 альтернативных кур-	3. После выбора система создает график студента.

са.	
4. Выполняется подчиненный поток П4 «Принять график».	

***Подчиненный поток событий П2. Обновить график***

<b>Действия актера</b>	<b>Действия (Отклик) системы</b>
	1. Система выводит текущий график студента.
	2. Система выполняет поиск в каталоге курсов доступных конкретных курсов и выводит их список.
3. Студент может обновить свой выбор курсов, удаляя или добавляя конкретные курсы.	4. После выбора система обновляет график.
5. Выполняется подчиненный поток П4 «Принять график».	

***Подчиненный поток событий П3. Удалить график***

<b>Действия актера</b>	<b>Действия (Отклик) системы</b>
	1. Система выводит текущий график студента.
	2. Система запрашивает у студента подтверждения удаления графика.
3. Студент подтверждает удаление.	4. Система удаляет график. Если график включает конкретные курсы, на которые записался студент, он должен быть удален из списков этих курсов.

***Подчиненный поток событий П4. Принять график***

Действия актера	Действия (Отклик) системы
	1. Система проверяет выполнение студентом предварительных требований (прохождение определенных курсов), факт открытия конкретного курса и отсутствие конфликтов графика.
	2. Система вносит данные о студенте в список выбранного конкретного курса.
	3. Курс фиксируется в графике, и график сохраняется в системе.

### **Альтернативные потоки**

#### ***Альтернативный поток событий А1. Сохранить график***

В любой момент студент может вместо принятия графика сохранить его. В этом случае шаг «Принять график» заменяется на следующий:

Действия актера	Действия (Отклик) системы
	1. «Незафиксированные» конкретные курсы помечаются в графике как «выбранные».
	2. Система сохраняет график
	3. Курс фиксируется в графике, и график сохраняется в системе.

***Альтернативный поток событий А2. Не выполнены предварительные требования, курс заполнен или имеют место конфликты графика.***

Если во время выполнения подчиненного потока «Принять график» система обнаружит, что студент не выполнил необходимые предварительные требования, или выбранный им конкретный курс заполнен, или имеют место конфликты графика, то выдается сообщение об ошибке. Студент может либо выбрать другой конкретный курс и продолжить выполнение варианта использования, либо сохранить график, либо отменить операцию, после чего основной поток начнется с начала.



### ***Альтернативный поток событий А3. График не найден.***

Если во время выполнения подчиненных потоков «Обновить график» или «Удалить график» система не может найти график студента, то выдается сообщение об ошибке. После того как студент подтвердит это сообщение, основной поток начнется с начала.

### ***Альтернативный поток событий А4. Система каталога курсов недоступна.***

Если окажется, что невозможно установить связь с системой каталога курсов, то будет выдано сообщение об ошибке. После того как студент подтвердит это сообщение, вариант использования завершится.

### ***Альтернативный поток событий А5. Регистрация на курсы закончена.***

Если в самом начале выполнения варианта использования окажется, что регистрация на текущий семестр закончена, будет выдано сообщение, и вариант использования завершится.

### ***Альтернативный поток событий А6. Удаление отменено.***

Если во время выполнения подчиненного потока «Удалить график» студент решит не удалять его, удаление отменяется, и основной поток начнется сначала.

## **Предусловия**

Перед началом выполнения данного варианта использования студент должен войти в систему.

## **Постусловия**

Если вариант использования завершится успешно, график студента будет создан, обновлен или удален. В противном случае состояние системы не изменится.

## **Вариант использования Close Registration**

**Краткое описание.** Данный вариант использования позволяет регистратору закрывать процесс регистрации. Конкретные курсы, на которые не записалось достаточного количества студентов, отменяются. В расчетную систему передается информация о каждом студенте по каждому конкретному курсу, чтобы студенты могли оплатить курсы.

## Основной поток событий

Данный вариант использования начинает выполняться, когда регистратор запрашивает прекращение регистрации.

Действия актера	Действия (Отклик) системы
	1. Система проверяет состояние процесса регистрации. Если регистрация еще выполняется, выдается сообщение, и вариант использования завершается.
	2. Для каждого конкретного курса система проверяет, ведет ли его какой-либо профессор и записалось ли на него не менее трех студентов. Если эти условия выполняются, система фиксирует конкретный курс в каждом графике, который включает данный курс.
	3. Для каждого студенческого графика проверяется наличие в нем максимального количества основных курсов; если их недостаточно, система пытается дополнить альтернативными курсами из списка данного графика. Выбирается первый доступный альтернативный курс. Если таких курсов нет, то никакое дополнение не происходит.
	4. Система закрывает все конкретные курсы. Если в каком-либо конкретном курсе оказывается менее трех студентов (с учетом добавлений, сделанных в п.3), система отменяет его и исключает из каждого содержащего его графика.
	5. Система рассчитывает плату за обучение для каждого студента в текущем семестре и направляет информацию в расчетную систему. Расчетная система посылает студентам счета для оплаты с копией их окончательных графиков.
Альтернативный поток событий A1_ Конкретный курс никто не ведет.	
Альтернативный поток событий A2_ Расчетная система недоступна.	

## **Альтернативные потоки**

***Альтернативный поток событий A1. Конкретный курс никто не ведет.***

Если во время выполнения основного потока обнаруживается, что некоторый конкретный курс не ведется никаким профессором, то этот курс отменяется. Система исключает данный курс из каждого содержащего его графика.

***Альтернативный поток событий A2. Расчетная система недоступна.***

Если невозможно установить связь с расчетной системой, спустя некоторое установленное время система вновь попытается связаться с ней. Попытки будут повторяться до тех пор, пока связь не установится.

### **Предусловия**

Перед началом выполнения данного варианта использования регистратор должен войти в систему.

### **Постусловия**

Если вариант использования завершится успешно, регистрация закрывается. В противном случае состояние системы не изменится.

## **Упражнение 5. Прикрепление файла к варианту использования**

1. Щелкните правой кнопкой мыши по варианту использования.
2. В открывшемся меню выберите пункт ***Open Specification***.
3. Перейдите на вкладку файлов.
4. Щелкните правой кнопкой мыши по белому полю и из открывшегося меню выберите пункт ***Insert File***.
5. Укажите созданный ранее файл и нажмите на кнопку ***Open*** (***Открыть***), чтобы прикрепить файл к варианту использования.

В результате представление вариантов использования в браузере примет следующий вид (рис.8).

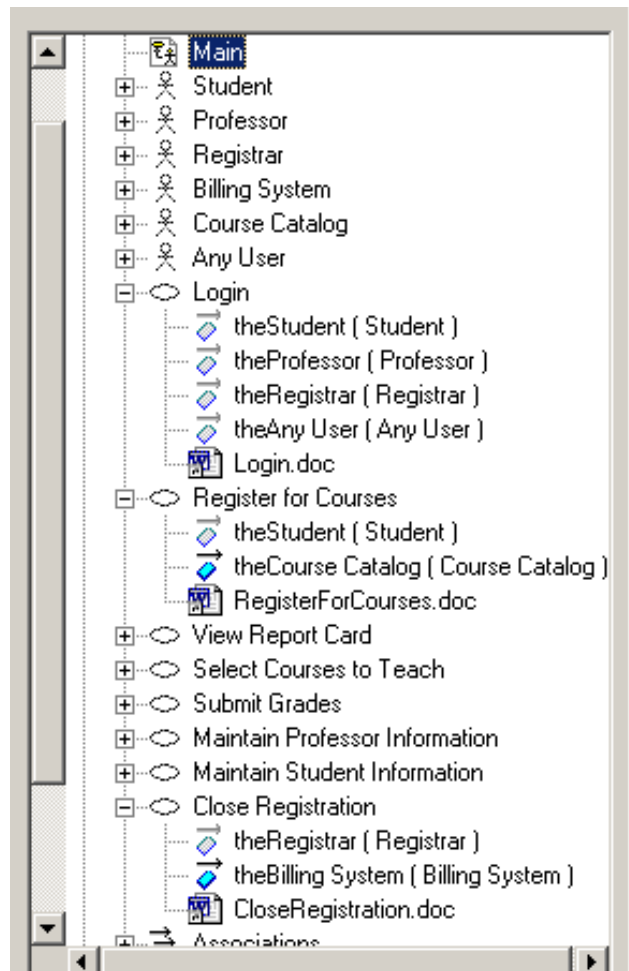


Рисунок 8– Представление вариантов использования в браузере

### Удаление вариантов использования и действующих лиц.

Существуют два способа удалить элемент модели - из одной диаграммы или из всей модели. Для удаления элемента модели из диаграммы:

1. Выделите элемент на диаграмме.
2. Нажмите на клавишу **Delete**.
3. Обратите внимание, что хотя элемент и удален с диаграммы, он остался в браузере и на других диаграммах системы.

Для удаления элемента из модели:

1. Выделите элемент на диаграмме.
2. Выберите пункт меню **Edit > Delete from Model** или нажмите сочетание клавиш **<CTRL> + <D>**.

## Выполнение работы в Rational Rose

1. Изучить раздел 1 и 2.
2. Изучить раздел 3. Выполнить упражнения 1-5.
3. Создать концептуальную модель системы в соответствии с вариантом задания.
  - a. Выполнить постановку задачи. Для этого:
    - ознакомится с предметной областью по теме работы;
    - выяснить субъективные цели и задачи (желаемый результат) у заказчика (его роль выполняет преподаватель);
    - провести анализ полученных данных;
    - создать содержательное описание системы.
  - При этом должны быть выполнены следующие требования:
    - беседа с заказчиком проводится в форме «вопрос-ответ»;
    - исполнитель работы считается экспертом в предметной области.
  - Методические указания:
    - Перед выполнением беседы с заказчиком заранее подготовить вопросы о проектируемой информационной системе.
    - Вопросы следует задавать в доступной непрофессионалу в области информационных технологий форме, избегая специальных и технических терминов.
    - При проведении беседы предпочтительно вести некоторые пометки или записи.
    - Протокол беседы необходимо в полном виде представить в отчете.
  - b. Разработать глоссарий проекта.
  - c. Описать дополнительные спецификации.
  - d. Создать модель вариантов использования. Для этого разработать:
    - список действующих лиц;
    - список вариантов использования;
    - диаграмму вариантов использования;
    - текстовые сценарии – основные и альтернативные потоки событий вариантов использования.
4. Создать разработанную в п. 3 модель вариантов использования в среде Rational Rose.

## Выполнение работы в MS Visio

Создание диаграммы использования в MS Visio:

1. Запустите MS Visio. (Кнопка "Пуск"/ "Программы" / MS Visio).
2. Появится окно, в котором необходимо выбрать папку **Software/UML Model Diagram**.
3. В открывшемся списке форм (Shapes) для построения физической диаграммы следует выбрать пункт **UML Use Case**.
4. В результате проделанных действий на экране появится окно, в левой части которого будет отображен набор графических символов, а в правой части - лист для рисования диаграммы:

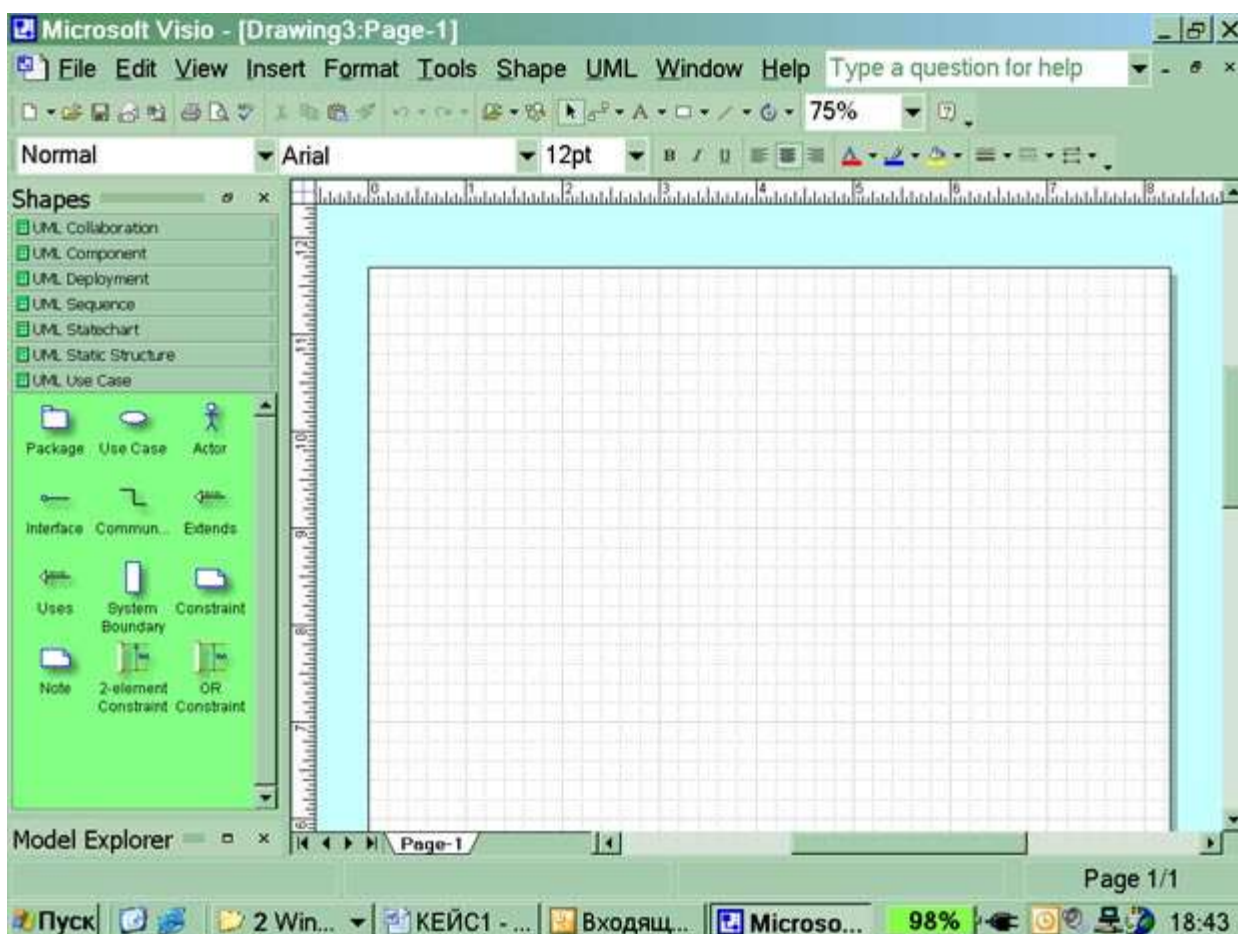



Рисунок. 9 – Общий вид окна MS Visio

5. Для изображения прямоугольника на панели инструментов "Стандартная" найдите и зафиксируйте щелчком мыши пиктограмму с изображением прямоугольника. Затем, при нажатой правой клавиши мыши вы сможете нарисовать произвольного размера прямоугольник.

6. Для изображения на диаграмме контрагентов следует воспользоваться графическим символом с изображением человечка . Графический символ переносится на рабочее поле мышкой при нажатой правой клавише.

**Примечание.** Для последующего перемещения графических символов по рабочему полю необходимо зафиксировать пиктограмму Pointer Tool с изображением стрелки, размещенную на панели инструментов "Форматирование". Только после этого графический символ будет доступен для перемещения его мышкой.

5. Соедините линиями изображение каждого контрагента с прямоугольником. Для этого на панели инструментов "Стандартная" щелчком мыши зафиксируйте пиктограмму с изображением линии Line Tool и при нажатой левой клавише мышки осуществите соединение фигур.

6. Внесите наименования контрагентов "Покупатели (аптеки)", "Покупатели (дистрибьюторы)", "Поставщики (Россия)", "Поставщики (импорт)", "Транспортные компании". Для того чтобы внести надписи на диаграмме, необходимо на панели инструментов "Форматирование" зафиксировать пиктограмму Text Tool (символ буквы "А"). Щелкните мышкой на изображении человечка, курсор установится на поле с надписью Actor. Введите в это поле наименование контрагента.

7. Введите наименование компании "МЕД" в нарисованный прямоугольник, щелкнув мышкой по прямоугольнику. Обратите внимание на то, что при этом должна быть активна пиктограмма Text Tool (символ буквы "А").

8. Аналогичным образом внесите надписи к линиям соединения фирмы и контрагентов.

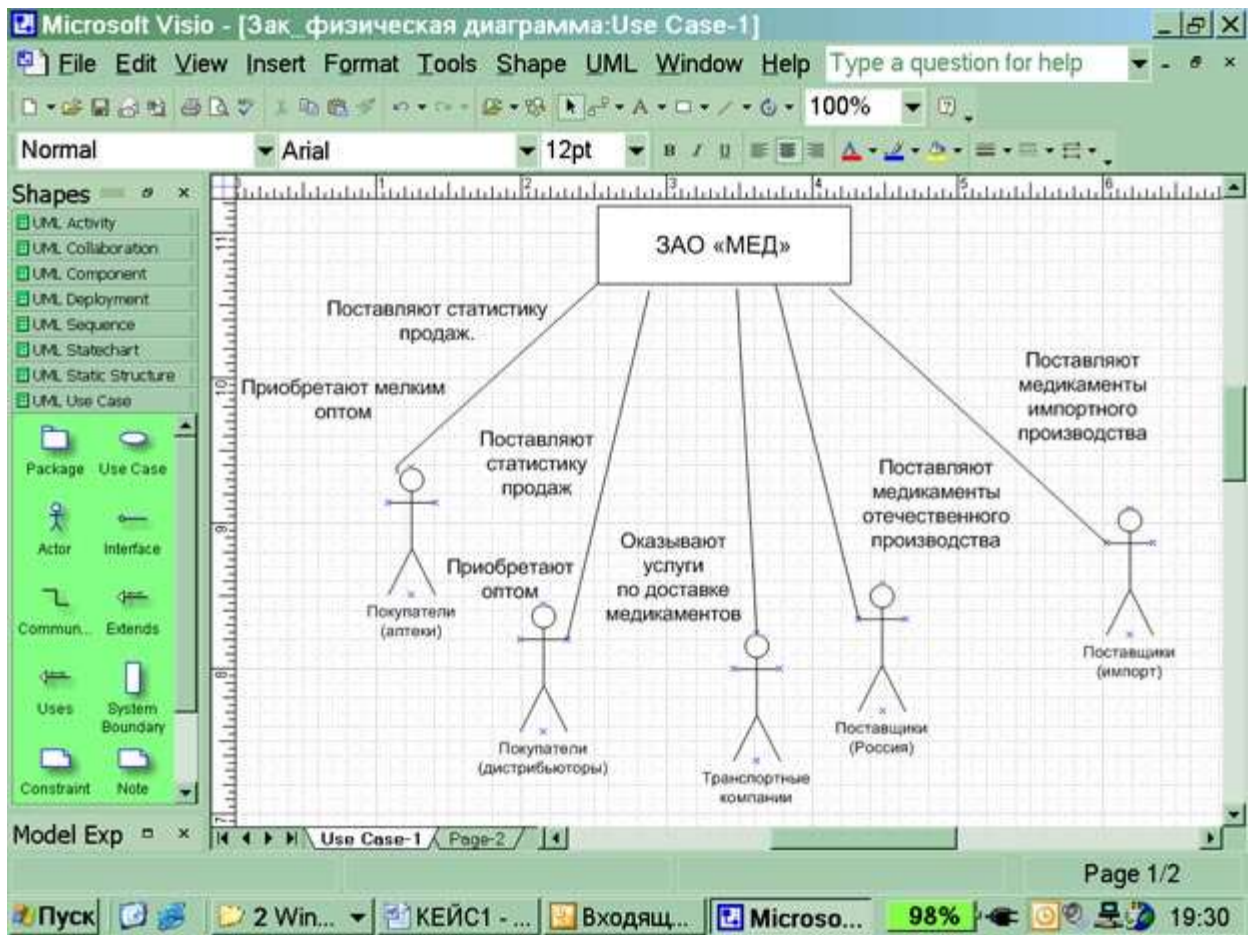


Рисунок 10 – Пример диаграммы использования ЗАО "МЕД"



## **4 Содержание отчета**

В качестве отчета о выполненной работе предъявите преподавателю:

1. на экране в среде Rational Rose: упражнения 1-5 (раздел 3) и модель вариантов использования по индивидуальному заданию;
2. отчет в печатном виде содержащий:
  - задание,
  - постановку задачи (протокол беседы с заказчиком, содержательное описание системы),
  - глоссарий проекта,
  - дополнительные спецификации,
  - модель вариантов использования (список действующих лиц, список вариантов использования, диаграмма вариантов использования, текстовые сценарии).
  -

### **Контрольные вопросы**

1. Для чего используется язык UML?
2. Какие диаграммы входят в состав языка UML?
3. В чем смысл варианта использования?
4. Каково назначение диаграмм вариантов использования.
5. Назовите основные свойства вариантов использования.
6. Назовите основные компоненты диаграмм вариантов использования.
7. Что такое «действующее лицо»?
8. Какую роль могут играть действующие лица по отношению к варианту использования?
9. Каким образом анализ внешних событий позволяет определить варианты использования системы?

### **Библиографический список**

1. Вендров, А. М. Проектирование программного обеспечения экономических информационных систем [Текст]: учебник для студ. вуз. / А. М. Вендров. - М.: Финансы и статистика, 2000. - 352 с.
2. Леоненков, А.В. Самоучитель UML [Текст] / А. Леоненков. - СПб. : БХВ-Петербург, 2001. - 304 с. : ил. - ISBN 5-94157-008-2.

3. Смирнова, Г. Н. Проектирование экономических информационных систем [Текст] : учебник / А. А. Сорокин, Ю. Ф. Тельнов. - М. : Финансы и статистика, 2003. - 512 с. - ISBN 5-279-02295-0.
4. Торрес, Р. Дж. Практическое руководство по проектированию и разработке пользовательского интерфейса [Текст] / Р. Дж. Торрес. - М. : Вильямс, 2002. - 400 с. - ISBN 5-8459-0367-X.
5. Меняев, М. Ф. Управление проектами MS Project [Текст] : учебное пособие / М. Ф. Меняев. - М. : Омега-Л, 2005. - 276 с. с. : ил. - ISBN 5-98119-367-0.

## Перечень вариантов заданий

№№ п.п.	Название предметной об- ласти	№№ п.п.	Название предметной области
1.	Автовокзал	2.	Музей
3.	Автопредприятие	4.	Музыкальная школа
5.	Аптека	6.	Налоговая инспекция
7.	Армия	8.	НИИ
9.	Ателье одежды	10.	Общежитие
11.	АЭС	12.	Оптовый склад
13.	Аэропорт	14.	Отдел кадров
15.	Банк	16.	Отдел снабжения
17.	Библиотека	18.	Отдел сбыта
19.	Больница	20.	Пенсионный фонд
21.	Бюджет города	22.	Подбор персонала (Кадровое агентство)
23.	Видеопрокат	24.	Пожарная служба
25.	Виноделие	26.	Поликлиника
27.	Военкомат	28.	Производство
29.	Гараж (автостоянка)	30.	Производство автомобилей
31.	География	32.	Радиостанция
33.	ГИБДД	34.	Речной порт
35.	Государственная дума	36.	Санаторий
37.	Дворец пионеров	38.	Служба доставки
39.	Детский сад	40.	Социальная работа
41.	Дискография	42.	Спортивный клуб
43.	Документооборот	44.	Строительство
45.	Досье КГБ	46.	Судопроизводство
47.	ДТП	48.	Торговая фирма
49.	ЖД станция	50.	Торговля недвижимостью
51.	ЗАГС	52.	Турфирма
53.	Расписание	54.	Тюрьма
55.	Заповедник	56.	Учебный план (Кафедра)
57.	Зоопарк	58.	Учебный процесс (Факультет)
59.	Интернет	60.	Фирма
61.	Картинная галерея	62.	Фотоателье
63.	Киноиндустрия	64.	Цветочные магазины
65.	Киоски	66.	Церковь
67.	Коллекции	68.	Цирк

№№ п.п.	Название предметной об- ласти	№№ п.п.	Название предметной области
69.	Колхоз	70.	Чемпионат
71.	Компьютер и комплек- тующие	72.	Школа
73.	Лесничество	74.	Ювелирный магазин
75.	Магазины	76.	Юридические лица
77.	Морг	78.	Юридическая консультация