# ЛЕКЦИЯ 5. СОЗДАНИЕ ПРОСТОЙ ДИАГРАММЫ ПРЕЦЕДЕНТОВ

*Диаграмма прецедентов* позволяет создать список операций, которые выполняет система. Часто этот вид диаграмм называют *диаграммой функций*, потому что на основе набора таких диаграмм создается список требований к системе и определяется множество выполняемых системой функций.

Каждая такая диаграмма или, как ее обычно называют, каждый Use case – это описание сценария поведения, которому следуют действующие лица (Actors).

Данный тип диаграмм используется при описании бизнес процессов автоматизируемой предметной области, определении требований к будущей программной системе. Отражает объекты как системы, так и предметной области и задачи, ими выполняемые.

5.1. Описание примера модели диаграммы прецедентов

Сценарий проекта таков: Сначала каждый преподаватель (профессор) университета заполняет специальную форму, в которой указывает, какие учебные курсы он намерен вести в следующем семестре. Данные из формы помещаются в университетский компьютер работником регистратуры.

После этого из полученных данных формируется каталог курсов, который раздается студентам. Студенты выбирают из каталога те курсы, на которых они собираются учиться, и подают заявки на обучение в регистратуру. Все эти данные также попадают в компьютер, где происходит их обработка и формирование списков курсов и студентов. В задачи создаваемой системы входит, в частности, такое комплектование учебных курсов, чтобы каждый курс посещало бы от трех до десяти студентов. Если на какой-то курс не набирается трех студентов, он отменяется.

После формирования курсов преподаватели получают списки студентов, которых им предстоит обучать, а каждый студент получает подтверждение о зачислении на курс и счет на оплату.

Первое, что требуется при построении модели, — определить действующие лица системы и сценарии поведения.

Действующих лиц в создаваемой системе четыре:

1. Преподаватель;
2. Студент;
3. Регистратор;
4. Биллинговая программа – система оплаты.

Первые три выбраны действующими лицами, поскольку они активно взаимодействуют с создаваемой системой. Биллинговая же программа чаще всего является отдельным программным продуктом, а в нашем случае она получает информацию для своей работы от создаваемой курсовой системы, поэтому может считаться самостоятельным действующим лицом.

5.2. Элементы диаграммы

При построении диаграммы прецедентов будем использовать пиктограммы типа "прецеденты" и "актеры". Помним , что термин "прецедент" это не очень удачный перевод на русский язык английского выражения use case (Computer Aided Software Engineering).

*Прецедент -* это описание множества последовательных событий, выполняемых компьютерной системой, которые приводят к наблюдаемому актером результату. Графически прецедент изображается в виде ограниченного непрерывной линией эллипса, обычно содержащего только имя прецедента.

*Актер* – это кто-то (или что-то) внешний по отношению к компьютерной системе, кто взаимодействует с ней. Графически актер изображается в виде пиктограммы, представляющей человека, поскольку актер это человек или группа людей, использующих данные, предоставляемые компьютерной системой.

5.3. Построение диаграммы

Теперь перенесем все это на диаграмму. Для этого нужно создать пустой проект, переключиться на папку Use Case View – для работы с диаграммами прецедентов (Use case) и открыть контекстное меню нажатием правой кнопки мыши. Если теперь выбрать пункт New•Actor (рис. 5.1), то вы получите действующее лицо;

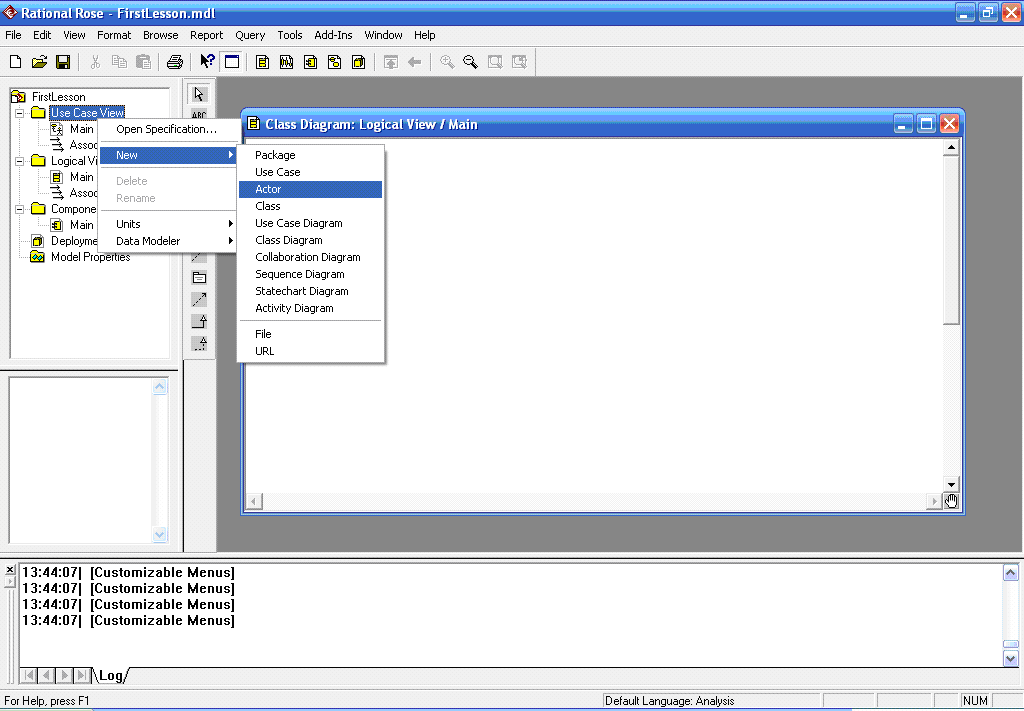


Рис. 5.1 Создание действующего лица

Теперь создадим всех действующих лиц (рис.5.2), рассмотренных в п. «Общее описание модели».

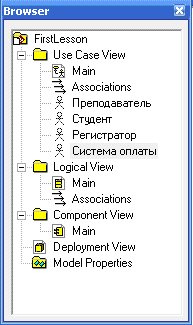


Рис. 5.2 Вид окна браузера проекта после создания всех действующих лиц («Актеров»)

Теперь выделим сценарии использования для нашего примера. Каждый из них описывает некоторое требование к функциям системы:

* Выбор курсов для преподавателя;
* Запрос расписания курсов;
* Регистрация на курсы;
* Создание каталогов ресурсов;
* Хранение информации о курсах;
* Хранение информации о преподавателях;
* Хранение информации о студентах.

Если выбрать New•Use Case, то будет создан сценарий поведения:

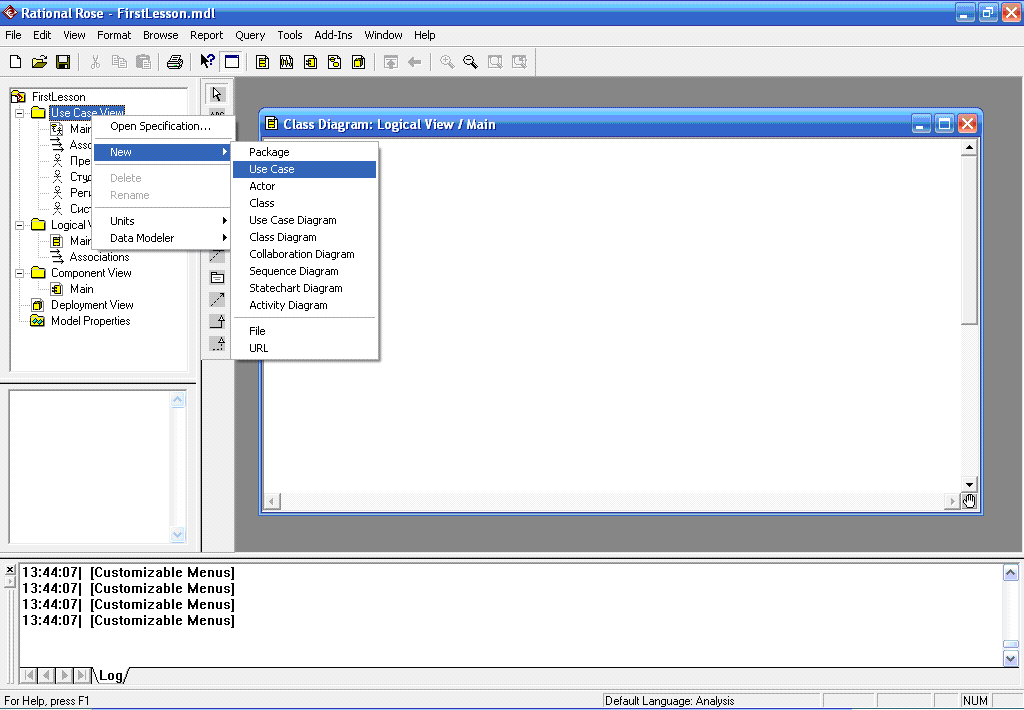


Рис. 5.3 Создание сценария поведения

Теперь создадим все описанные выше сценарии поведения:

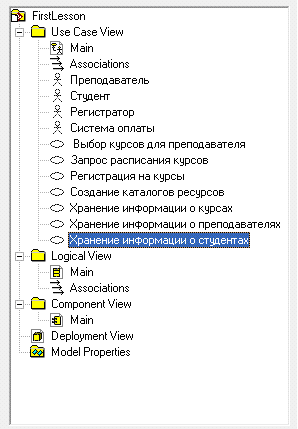


Рис. 5.4 Вид окна браузера проекта после создания всех  
сценариев поведения («Прецедентов»)

Все введенные нами действующие лица и сценарии поведения немедленно появляются в окне браузера, откуда вы можете перетаскивать их мышью на диаграммы.

Далее, как правило, строится *диаграмма сценариев поведения - прецедентов*. Для этого двойным щелчком на пиктограмме Main из папки Use Case View открывается *главная диаграмма сценариев.*

В нее из окна браузера перетаскиваются все действующие лица и сценарии поведения, которые были созданы в рамках модели:

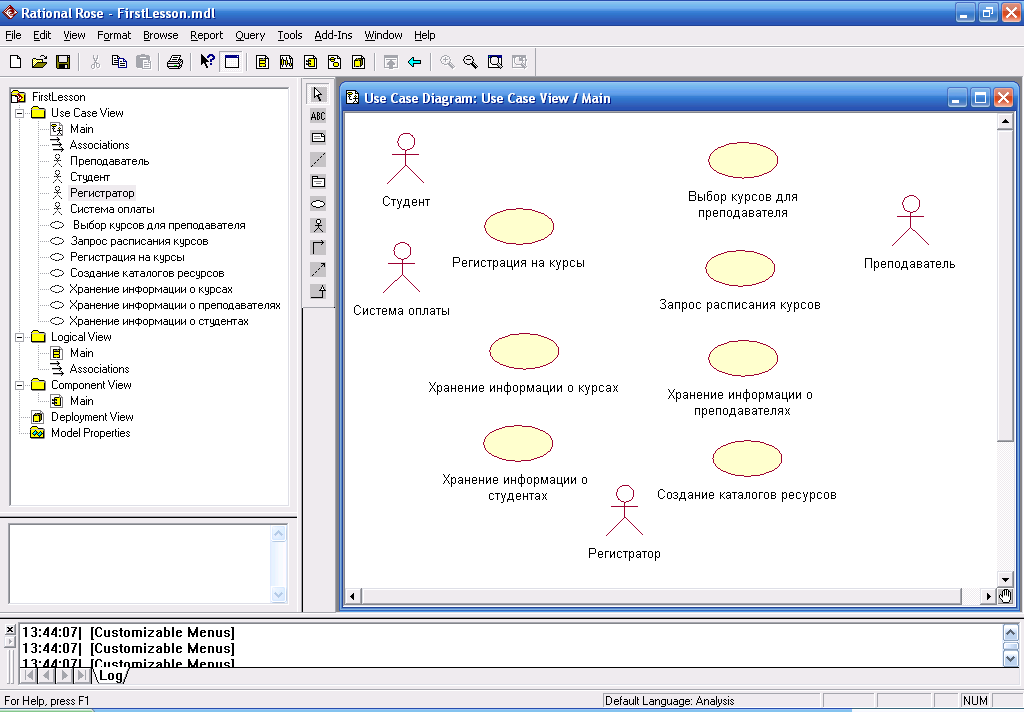


Рис. 5.5 Диаграмма сценариев поведения (пока без связей)

После размещения эти компоненты нужно связать между собой, чтобы отобразить взаимосвязи. В нашей модели наилучшим образом подойдут связи типа *«однонаправленная ассоциация» (Unidirectional Association).*

*Ассоциация* – это структурное двунаправленное отношение, описывающее совокупность взаимоотношений между объектами. По сути дела ассоциация является сверткой бинарных отношений между объектами. Эту свертку может мысленно выполнить в своем сознании специалист (например, программист), который видит пиктограмму ассоциации на UML диаграмме.

Для реализации связей применяется метод перетаскивания. Сначала в палитре выбирается тип связи «однонаправленная ассоциация» (Uniderectional Association) , после чего нужно протянуть линию между действующим лицом и сценарием поведения. В результате на диаграмме возникнет стрелка. Аналогичным образом поступают со всеми компонентами диаграммы. Готовая диаграмма показана на Рис. 5.6.

Для создания границы («интерфейса») между актерами и прецендентами можно воспользовавшись пунктом меню «Tools/Create/Note Anchor» или выбрать среди пиктограмм инструмент «Anchor Note to Item». Выбрав инструмент необходимо щелкнуть один раз на середине одной из стрелок отношения далее во всех углах интерфейса и завершить границу в месте начала ее рисования. Если прямоугольник получился не очень ровый то это можно исправить выбрав в меню «Format/Line Style/Rectilinear»

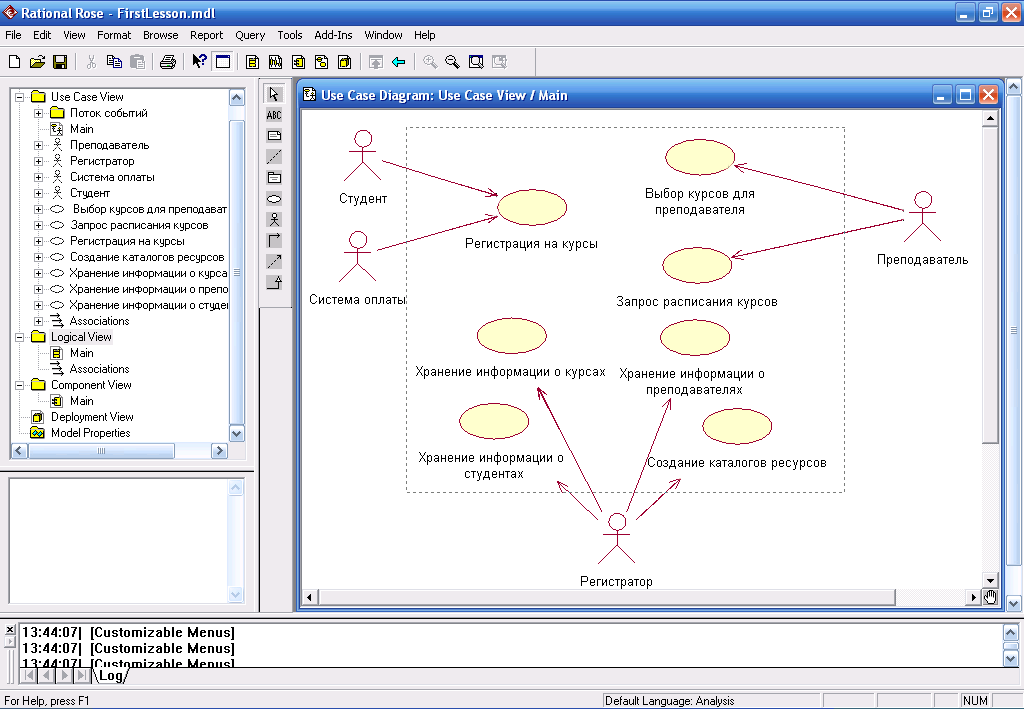
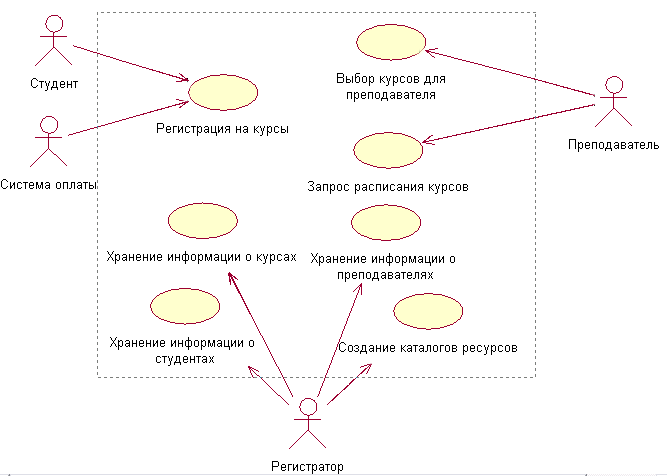


Рис. 5.6 Главная диаграмма прецедентов  
 системы регистрации учебных курсов Университета



Поток событий

Внешняя граница системы регистрации учебных курсов

Рис. 5.7 Главная диаграмма прецедентов

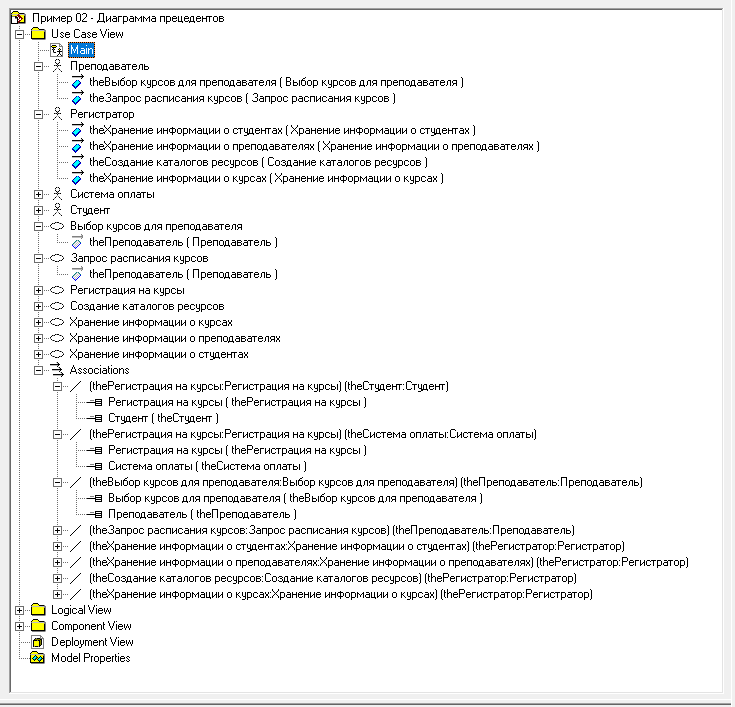


Рис. 5.8 Структура диаграммы прецедентов

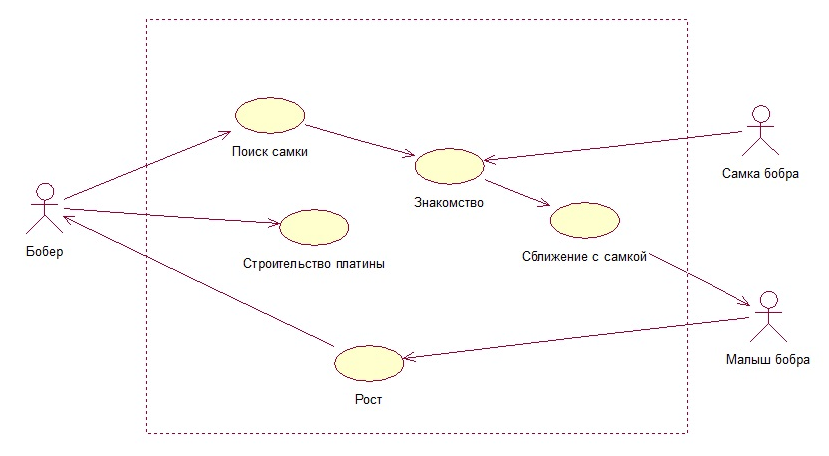


Рис. 5.9 Пример диаграммы прецедентов

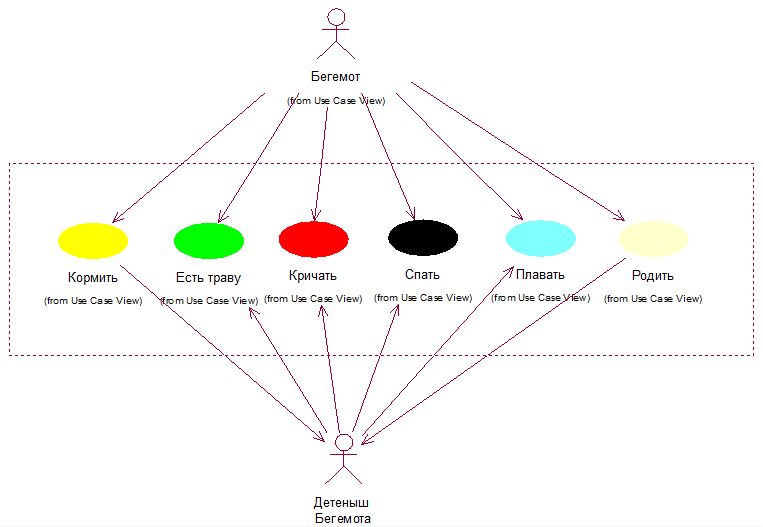


Рис. 5.10 Пример диаграммы прецедентов