

Лекция №1. Диаграммы вариантов использования UML

1.1 Унифицированный язык моделирования (UML)

UML (Unified Modeling Language – унифицированный язык моделирования) – стандартный язык для написания моделей анализа, проектирования и реализации объектно-ориентированных программных систем [1, 2]. UML может использоваться для визуализации, спецификации, конструирования и документирования результатов программных проектов. UML – это не визуальный язык программирования, но его модели прямо транслируются в текст на языках программирования (Java, C++, Visual Basic и т.п.) и даже в таблицы для реляционной БД.

Процесс разработки визуальных моделей в нотации языка UML обычно представляется в форме последовательной детализации модельных представлений от концептуальной модели к логической, а затем и к физической модели рассматриваемой программной системы. Этот процесс представляет собой построение ряда диаграмм.

Диаграмма в языке UML – это графическое представление множества элементов, наиболее часто изображается как связный граф из вершин (предметов) и дуг (отношений). Диаграммы рисуются для визуализации системы с разных точек зрения, затем они отображаются в систему.

Обычно диаграмма дает неполное представление элементов, которые составляют систему. Хотя один и тот же элемент может появляться во всех диаграммах, на практике он появляется только в некоторых диаграммах. Теоретически диаграмма может содержать любую комбинацию предметов и отношений, на практике ограничиваются малым количеством комбинаций. По этой причине UML включает девять видов диаграмм:

- 1) диаграммы классов;
- 2) диаграммы объектов;
- 3) диаграммы Use Case (диаграммы прецедентов);
- 4) диаграммы последовательности;

- 5) диаграммы сотрудничества (кооперации);
- 6) диаграммы схем состояний;
- 7) диаграммы деятельности;
- 8) компонентные диаграммы;
- 9) диаграммы размещения (развертывания).

1.2 Диаграммы Use Case

Диаграмма Use Case языка UML определяет поведение программной системы с точки зрения пользователя. Диаграмма Use Case рассматривается как главное средство для первичного моделирования динамики системы, используется для выяснения требований к разрабатываемой системе, фиксации этих требований в форме, которая позволит проводить дальнейшую разработку.

В русской литературе диаграммы Use Case часто называют *диаграммами прецедентов*, или *диаграммами вариантов использования*.

1.3 Основные графические элементы диаграммы вариантов использования

1. *Вариант использования (use case)* представляет собой общую спецификацию совокупности выполняемых системой действий с целью предоставления некоторого наблюдаемого результата, который имеет значение для одного или нескольких актеров.

Другими словами, вариант использования представляет собой законченный фрагмент поведения системы с точки зрения тех или иных заинтересованных лиц. Хотя каждый вариант использования предполагает реализацию системой некоторой последовательности действий, которые должны быть выполнены с целью предоставления соответствующего результата или сервиса, сами эти действия на диаграмме вариантов использования никак не изображаются.

Общепринятым графическим изображением варианта использования в

нотации языка UML является эллипс, внутри которого или ниже его записывается имя в форме строки текста. Примеры трех различных вариантов использования представлены на рис. 1.

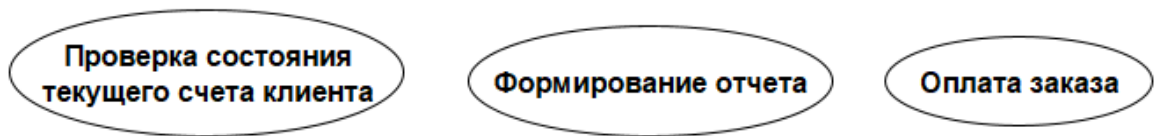


Рис. 1 Стандартное графическое обозначение варианта использования (три различных примера вариантов использования)

Примерами вариантов использования могут являться следующие фрагменты законченного функционального поведения системы: получение справки о состоянии счета клиента, оформление заказа на покупку товара, идентификация пользователя при входе в систему, отображение графической формы на экране монитора и другие поведения рассматриваемых систем.

2. Актер (actor) представляет собой внешнюю по отношению к проектируемой системе сущность, которая взаимодействует с системой и использует ее функциональные возможности для достижения определенных целей или решения частных задач.

Формально в контексте языка UML каждый актер специфицирует некоторую роль, которую играет пользователь или любая другая система, взаимодействующая с субъектом. Термин «роль» здесь используется неформально.

Общепринятым графическим изображением актера на диаграммах в нотации языка UML является фигурка «проволочного человечка», под которой записывается обязательное имя актера (на рис. 2 представлены три примера актеров).



Рис. 2 Символ графического обозначения актера (три различных примера актеров)

Примерами актеров могут быть: кассир, клиент банка, банковский служащий, президент, продавец магазина, менеджер отдела продаж, пассажир авиарейса, водитель автомобиля, администратор гостиницы, сотовый телефон, налоговая инспекция и другие сущности, имеющие отношение к концептуальной модели соответствующей предметной области.

3. Субъектом (*subject*) в контексте языка UML называется любой элемент модели, который обладает функциональным поведением.

Субъект вариантов использования может быть некоторой физической системой или любым другим элементом с поведением, таким как подсистема, класс или компонент. Графически субъект на диаграмме вариантов использования изображается в форме прямоугольника, внутри которого размещаются отдельные варианты использования. Каждый субъект имеет имя, которое должно соответствовать имени элемента моделируемой системы, владеющего данным поведением.

4. Комментарий (*comment*) в языке UML предназначен для включения в модель произвольной текстовой информации в форме примечания, которое может быть присоединено к одному или нескольким элементам разрабатываемой модели. В качестве такой информации могут быть, например, пояснения разработчика относительно назначения элементов диаграммы, рекомендации руководителя проекта по реализации модели, справочная информация об авторе и особенностях разработки отдельных элементов модели. Применительно к диаграммам вариантов использования комментарий может иметь уточняющую информацию, относящуюся к контексту тех или иных вариантов использования или актеров.

Графически комментарий на всех типах диаграмм изображается в форме прямоугольника с «загнутым» правым верхним уголком (рис. 3).



Рис. 3 Пример комментария в языке UML

1.4 Отношения на диаграмме вариантов использования

Под *отношением (relationship)* в языке UML понимается произвольная семантическая взаимосвязь между отдельными элементами модели.

Виды отношений на диаграммах вариантов использования:

1. Ассоциация (association) является одним из фундаментальных понятий в языке UML и может использоваться на различных канонических диаграммах при построении визуальных моделей. Применительно к диаграммам вариантов использования отношение ассоциации может служить только для обозначения взаимодействия актера с вариантом использования.

На диаграммах вариантов использования, так же как и на других диаграммах, отношение ассоциации обозначается сплошной линией, соединяющей актера и вариант использования. При этом следует различать *направленные* (рис. 4) и *ненаправленные* (рис. 5) ассоциации.

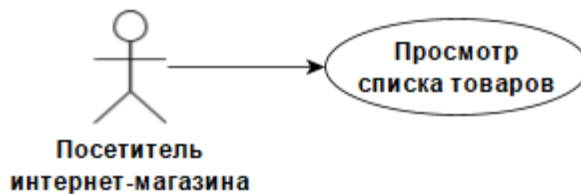


Рис. 4 Пример графического представления отношения направленной ассоциации



Рис. 5 Пример графического представления отношения ненаправленной ассоциации

В контексте диаграммы вариантов использования направленная ассоциация от актера к варианту использования может указывать на тот факт, что данный актер является инициатором выполнения соответствующего варианта использования. Направленная ассоциация от варианта использования к актеру может указывать также на тот факт, что данному актеру предоставляется справочная или отчетная информация о результатах функционирования моделируемой системы.

2. Отношение зависимости (dependency) определяется как форма взаимосвязи между двумя элементами модели, предназначенная для спецификации того обстоятельства, что изменение одного элемента модели приводит к изменению некоторого другого элемента.

Отношение включения (include) является частным случаем отношения зависимости и специфицирует тот факт, что некоторый вариант использования содержит поведение, определенное в другом варианте использования.

Графически данное отношение обозначается как отношение зависимости в форме пунктирной линии с «V»-образной стрелкой, направленной от зависимого варианта использования к независимому варианту использования (рис. 6). Зависимый вариант использования часто называют также *базовым* или *включающим*, а независимый – *включаемым* вариантом использования. При этом линия со стрелкой обязательно помечается ключевым словом `<<include>>`, записанным в форме стереотипа.

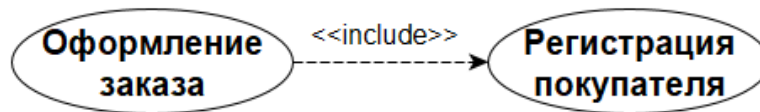


Рис. 6 Пример графического изображения отношения включения между вариантами использования

Отношение расширения (extend) является частным случаем отношения зависимости и определяет взаимосвязь одного варианта использования с некоторым другим вариантом использования, функциональность или поведение которого задействуется первым не всегда, а только при выполнении некоторых дополнительных условий.

Графически данное отношение обозначается как отношение зависимости в форме пунктирной линии с «V»-образной стрелкой, направленной от зависимого варианта использования к независимому варианту использования и соединенной с ним в так называемой *точке расширения (extension point)* (рис. 7). Зависимый вариант использования часто называют также *расширяющим*, а независимый – *базовым* или

расширяемым вариантом использования. При этом линия со стрелкой обязательно помечается ключевым словом <<extend>>, записанным в форме стереотипа.

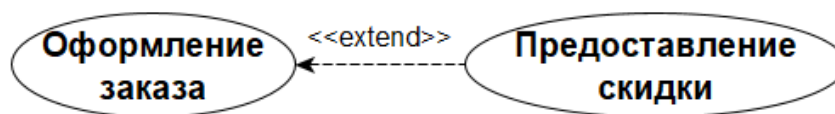


Рис. 7 Пример графического изображения отношения расширения между вариантами использования

3. Отношение обобщения (*generalization*) предназначено для спецификации того факта, что один элемент модели является специальным или частным случаем другого элемента модели.

Графически отношение обобщения обозначается сплошной линией со стрелкой в форме не закрашенного треугольника, которая указывает на общий элемент модели.

В изображенном на рис. 8 фрагменте диаграммы вариантов использования отношение обобщения специфицирует то, что вариант использования «Оплата товара по кредитной карте» является специальным случаем варианта использования «Оплата товара». При этом вариант использования «Оплата товара по кредитной карте», который является *потомком*, наследует все свойства поведения своего *родителя*, т.е. варианта использования «Оплата товара».

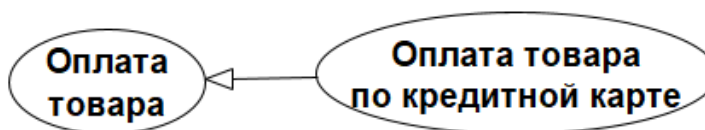


Рис. 8 Пример графического изображения отношения обобщения между вариантами использования

Пример графического изображения отношения обобщения между актерами показан на рис. 9.

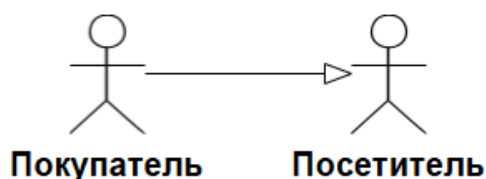


Рис. 9 Пример графического изображения отношения обобщения между актерами

1.5 Модель вариантов использования

Требование (*requirement*) – желательное свойство, характеристика или условие, которым должна удовлетворять программная система в процессе своей эксплуатации.

Спецификация требований к проектируемой системе в форме диаграммы вариантов использования и дополнительных сценариев может представлять собой самостоятельную модель, которая в языке UML получила название **модели вариантов использования**.

Содержание варианта использования может быть дополнительно специфицировано в форме пояснительного текста, который раскрывает смысл или семантику выполняемых действий при выполнении варианта использования. Такой пояснительный текст применительно к диаграммам вариантов использования получил специальное название текста-сценария или просто – сценария.

Сценарий (*scenario*) – специально написанный текст, который описывает поведение моделируемой системы в форме последовательности выполняемых действий актеров и самой системы.

Один из шаблонов для написания сценария отдельного варианта использования рассматривается в табл. 1.

Таблица 1. Шаблон для написания сценария отдельного варианта использования

Главный раздел	Раздел «Типичный ход событий»	Раздел «Исключения»	Раздел «Примечания»
Имя варианта использования	Типичный ход событий, приводящий к успешному выполнению данного варианта использования	Исключение № 1	Примечание № 1
Актеры		Исключение № 2	Примечание № 2
Цель	
Краткое описание		Исключение № <i>n</i>	Примечание № <i>n</i>
Тип			
Ссылка на другие варианты использования			

Для удобства записи и чтения сценария часто используют модификацию данного шаблона, располагая разделы последовательно сверху вниз. При этом написание сценариев модели начинают с базовых или основных вариантов использования, после чего рассматриваются сценарии второстепенных или включаемых вариантов использования. Сценарии расширяющих вариантов использования обычно помещаются в разделе исключений базового варианта использования.

1.6 Пример построения модели вариантов использования

Построим модель вариантов использования для системы продажи товаров в интернет-магазине.

Посетитель интернет-магазина может просматривать список товаров интернет-магазина, помещать товар в виртуальную корзину и изменять содержимое этой корзины.

Посетитель может стать покупателем, если он принимает решение об оформлении заказа на покупку выбранных им товаров.

Менеджер может изменять список товаров, а бухгалтер – принимать оплату за выбранный покупателем товар.

При оформлении заказа на покупку товара необходима регистрация покупателя.

При оформлении заказа постоянному покупателю может быть предоставлена специальная бонусная скидка.

В рамках рассматриваемой системы продажи товаров в интернет-магазине возможна оплата выбранных покупателем товаров, как наличными, так и по кредитной карточке.

1.7 Выделение актеров и вариантов использования

В соответствии с определением терминов «актер» и «вариант использования», для данной предметной области актерами будут люди, участвующие в работе интернет-магазина, а вариантами использования те

действия, которые они выполняют в интернет-магазине.

1. В качестве основного актера описываемой системы можно рассматривать актера «Посетитель». В соответствии с теми действиями, которые может выполнять посетитель интернет-магазина, можно выделить два варианта использования, связанных с актером «Посетитель» отношением ассоциации: «Просмотр списка товаров» и «Изменение содержимого корзины».

2. Т.к. посетитель интернет-магазина может стать покупателем, то можно выделить актера «Покупатель», связанного отношением обобщения с актером «Посетитель», при этом в качестве родителя будет выступать актер «Посетитель», а в качестве потомка «Покупатель», т.к. последний наследует все свойства поведения посетителя интернет-магазина, но и обладает собственными. Вариантами использования, связанными отношением ассоциации с актером «Покупатель», будут «Оформление заказа на покупку товаров» и «Оплата выбранного товара».

3. В качестве других актеров рассматриваемой системы могут выступать «Менеджер» и «Бухгалтер».

При этом «Менеджер» будет связан отношением ассоциации с вариантом использования «Изменение списка товаров», а бухгалтер – с вариантом использования «Оплата выбранного товара».

4. Поскольку при оформлении заказа на покупку товара необходима регистрация покупателя, и эта функциональность выполняется всегда, она может быть выделена в отдельный вариант использования «Регистрация покупателя», который будет связан с базовым отношением включения. С другой стороны, при оформлении заказа постоянному покупателю может быть предоставлена специальная бонусная скидка. Это требование может быть также представлено в качестве отдельного варианта использования «Предоставление бонусной скидки», который будет связан с базовым отношением расширения.

5. Дальнейшая детализация модели может быть выполнена на основе

установления дополнительного отношения обобщения для варианта использования «Оплата выбранного товара». Если в рамках рассматриваемой системы продажи товаров в интернет-магазине возможна оплата выбранных покупателем товаров, как наличными, так и по кредитной карточке, то в этом случае диаграмма может быть дополнена соответствующими вариантами использования. При этом варианты использования «Оплата товара наличными» и «Оплата товара по кредитной карте» будут связаны с вариантом использования «Оплата выбранного товара» отношением обобщения.

В результате разрабатываемая диаграмма должна содержать 9 вариантов использования и 4-х актеров, между которыми установлены соответствующие отношения ассоциации, включения, расширения и обобщения. Все варианты использования желательно заключить в прямоугольник, который служит для обозначения субъекта проектируемой системы.

Одним из средств построения диаграмм вариантов использования UML является бесплатная программа yEd Graph Editor (<https://www.yworks.com/products/yed>).

В итоге должна получиться диаграмма, изображенная на рис. 10.



Рис. 10 Диаграмма вариантов использования для системы продажи товаров в интернет-магазине

1.8 Написание текстового сценария

Дополним разработанную диаграмму вариантов использования текстовым сценарием, написанным на основе предложенного шаблона.

В качестве примера выберем вариант использования «Просмотр списка товаров», для которого соответствующий сценарий удобно представить в виде двух отдельных таблиц (таблица 2 и таблица 3), каждая из которых описывает отдельный раздел шаблона.

Таблица 2. Главный раздел сценария выполнения варианта использования «Просмотр списка товаров»

Вариант использования	Просмотр списка товаров
Актеры	Посетитель интернет-магазина
Цель	Получение требуемой информации о товарах, представленных в интернет-магазине

Краткое описание	Посетитель интернет-магазина просматривает информацию о товарах. Система обеспечивает доступ к любому товару и удобную навигацию по различным категориям товаров
Тип	Базовый
Ссылки на другие варианты использования	Отсутствуют

Таблица 3. Раздел типичного хода событий сценария выполнения варианта использования «Просмотр списка товаров»

Действия актеров	Отклик системы
1. Посетитель загружает исходную страницу интернет-магазина в браузер	2. Система отображает исходную страницу интернет-магазина
3. Посетитель интернет-магазина выбирает категорию интересующих товаров	4. Система отображает информацию о выбранной категории товаров
5. Посетитель интернет-магазина выбирает интересующий товар	6. Система отображает общую информацию о выбранном товаре
7. Посетитель интернет-магазина выбирает просмотр детальной информации об интересующем товаре	8. Система отображает детальную информацию о выбранном товаре
9. Посетитель интернет-магазина может пожелать вернуться на исходную страницу интернет-магазина	10. Система отображает исходную страницу интернет-магазина

Поскольку для рассматриваемого варианта использования отсутствуют расширяющие варианты использования, секции исключений останутся пустыми.

Список литературы

1. Орлов С.А. Программная инженерия. Учебник для вузов. 5-е издание обновленное и дополненное. Стандарт третьего поколения. СПб.: Питер, 2016. 640 с.

2. Леоненков А.В. Самоучитель UML – 2-е изд., перераб. и доп. СПб.: БХВ-Петербург, 2006. 432 с.