

Формализация функциональных
требований к системе с
помощью диаграммы вариантов
использования

Отдельные варианты использования могут применяться как для спецификации требований к проектируемой системе, так и для документирования процесса поведения имеющейся системы. Кроме этого, варианты использования неявно специфицируют требования, определяющие особенности взаимодействия пользователей с системой, которые специфицируют возможность корректной работы с предоставляемыми данной системой сервисами.

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

Применительно к программным системам предложена следующая *классификация требований*, которая получила название модели FURPS+, что соответствует первым буквам соответствующих категорий требований на английском языке:

- *функциональные требования* (Functionality)

- требования удобства использования (*Usability*)
- требования надежности (*Reliability*)
- требования производительности (*Performance*)
- требования возможности сопровождения (*Supportability*).

При этом символом "+" обозначены дополнительные условия, к которым относятся:

- проектные ограничения
- требования управления системой
- требования к графическому интерфейсу пользователя
- физические требования
- юридические требования.

Главными среди указанных требований являются **функциональные**, которые специфицируют особенности реализации отдельных бизнес-процессов моделируемой системы. **Функциональные требования** служат исходной информацией для построения диаграмм вариантов использования.

Графических средств языка *UML* на практике оказывается недостаточно для спецификации **функциональных требований**.

С этой целью рекомендуется дополнять этот тип диаграмм текстовыми **сценариями**, которые уточняют или детализируют последовательность действий, совершаемых системой при выполнении ее вариантов использования.

Сценарий (*scenario*) - определенная последовательность действий, которая описывает действия актеров и поведение моделируемой системы в форме обычного текста.

Сценарий используется для дополнительной иллюстрации взаимодействия актеров и вариантов использования. Предлагаются различные способы представления или написания подобных *сценариев*.

Один из таких **шаблонов** рассматривается ниже и может быть рекомендован для применения на начальных этапах концептуального моделирования (Таблица 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 варианта использования	Исключение № 1	Примечания № 1
Актеры		Исключение № 2	Примечания № 2
Цель	
Краткое описание			
Тип			

Ссылки на другие варианты использования		Исключение № N	Примечания № N

При написании *сценариев* вариантов использования текст *сценария* должен дополнять или уточнять диаграмму вариантов использования, но не заменять ее полностью. В противном случае будут потеряны достоинства визуального представления моделей.

Построение *диаграммы вариантов использования* - самый первый этап процесса *объектно-ориентированного анализа и проектирования*, цель которого - представить совокупность *функциональных требований* к поведению проектируемой системы. 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 и имеет свое специальное стандартное имя или стереотип <<useCaseModel>>.

Все заданные в этой модели требования допустимо представить в виде общей модели системы, которая может быть оформлена как отдельный пакет Система. Этот пакет в свою очередь может представлять собой иерархию пакетов, на самом верхнем уровне которых содержится множество классов, реализующих базовые варианты использования проектируемой системы. При этом пакет системы самого верхнего уровня может быть дополнительно помечен стереотипом <<topLevelPackage>>.

Особенности спецификации функциональных требований на диаграмме вариантов использования

(На примере на модели обычного банкомата)

Процесс функционирования этой системы хорошо знаком владельцам кредитных карточек, поэтому не требует дополнительного описания. Особенность отечественных банкоматов состоит в том, что в них отсутствует возможность перевода средств с одного счета на другой. Рассматриваемая система имеет двух актеров, один из которых является клиентом банкомата, а другой - Банком, который осуществляет выполнение соответствующих транзакций. Каждый из этих актеров взаимодействует с банкоматом, хотя главный *актер* Клиент, поскольку именно он инициирует функциональность банкомата.

Основные функциональные требования к банкомату заключаются в предоставлении клиенту возможности снятия наличных по кредитной карточке и получении справки о состоянии счета. Эти **функциональные требования** специфицируются отдельными вариантами использования, которые служат ключевыми элементами соответствующей концептуальной модели. Поскольку для выполнения этих вариантов использования необходимо аутентифицировать кредитную карточку, они оба обращаются к дополнительному сервису "Проверка ПИН-кода кредитной карточки". Как следует из существа выдвигаемых к банкомату **функциональных требований**, этот сервис может выступать в качестве отдельного варианта использования разрабатываемой диаграммы и соединяться с первыми двумя вариантами отношением включения. Соответствующая *диаграмма* вариантов использования может включать в себя только указанных двух актеров и три варианта использования (Рисунок 1).



Рисунок 1 - Диаграмма вариантов использования для модели банкомата

На следующем этапе разработки модели вариантов использования для рассматриваемой системы банкомата следует дополнить данную диаграмму текстовым *сценарием*, написанным на основе предложенного ранее *шаблона* (Таблица1). Этот *сценарий* будет дополнять диаграмму, раскрывая содержание и логическую последовательность отдельных действий, которые выполняются системой и актерами в процессе снятия наличных по кредитной карточке. В этом случае *сценарий* удобно представить в виде трех таблиц, каждая из которых описывает отдельный раздел *шаблона*.

В главном разделе *сценария* (Таблица2) указывается имя рассматриваемого варианта использования, имена взаимосвязанных с ним актеров, цель выполнения варианта, условный тип и ссылки на другие варианты использования.

Таблица 2 - Главный раздел сценария выполнения варианта использования "Снятие наличных по кредитной карточке"

Вариант использования	Снятие наличных по кредитной карточке
Актеры	Клиент, Банк
Цель	Получение требуемой суммы наличными
Краткое описание	Клиент запрашивает требуемую сумму. Банкомат обеспечивает доступ к счету клиента. Банкомат выдает клиенту наличные.

Тип	Базовый
Ссылки на другие варианты использования	Включает в себя ВИ: Проверка <u>ПИН-кода</u> кредитной карточки Идентифицировать кредитную карточку

В следующем разделе *сценария* (Таблица 3) описывается последовательность действий, приводящая к успешному выполнению рассматриваемого варианта использования. При этом инициатором действий должен выступать *актер* Клиент. Для удобства последующих ссылок каждое действие помечается порядковым номером в последовательности.

Таблица 3 - Раздел Типичный ход событий сценария выполнения варианта использования "Снятие наличных по кредитной карточке"

Действия актеров	Отклик системы
1. Клиент вставляет кредитную карточку в устройство чтения банкомата Исключение №1: Кредитная карточка недействительна	2. Банкомат проверяет кредитную карточку 3. Банкомат предлагает ввести <u>ПИН-код</u>

4. Клиент вводит персональный PIN-код	5. Банкомат проверяет <u>ПИН-код</u>
Исключение №2: Клиент вводит неверный <u>ПИН-код</u>	6. Банкомат отображает опции меню
7. Клиент выбирает снятие наличных со своего счета	8. Система делает запрос в Банк и выясняет текущее состояние счета клиента
	9. Банкомат предлагает ввести требуемую сумму
10. Клиент вводит требуемую сумму	12. Банкомат изменяет состояние счета клиента, выдает наличные и чек
11. Банк проверяет введенную сумму	
Исключение №3: Требуемая сумма превышает сумму на счете клиента	
13. Клиент получает наличные и чек	14. Банкомат предлагает клиенту забрать кредитную карточку
15. Клиент получает свою кредитную карточку	16. Банкомат отображает сообщение о готовности к работе

В третьем разделе *сценария* (Таблица4) описывается последовательность действий, выполняемых при возникновении исключительных ситуаций или исключений.

Таблица 4 - Раздел Исключения сценария выполнения варианта использования "Снятие наличных по кредитной карточке"

Исключение №1. Кредитная карточка недействительна или неверно вставлена	
Действия актера	Отклик системы
15. Клиент получает свою кредитную карточку	3. Банкомат отображает информацию о неверно вставленной кредитной карточке 14. Банкомат возвращает клиенту его кредитную карточку
Исключение №2. Клиент вводит неверный <u>ПИН-код</u>	
4. Клиент вводит новый <u>ПИН-код</u>	6. Банкомат отображает информацию о неверном <u>ПИН-коде</u>
Исключение №3. Требуемая сумма превышает сумму на счете клиента	
10. Клиент вводит новую требуемую сумму	12. Банкомат отображает информацию о превышении кредита

Можно дополнить данный *сценарий*, аналогичным образом описав не только варианты использования "Получение справки о состоянии счета" и "Проверка Пин-кода кредитной карточки", но и рассмотрев другие исключения, например отказ клиента от получения наличных после проверки ПИН-кода и т.п. При этом *полнота сценариев* и модели вариантов использования будут определяться теми *функциональными требованиями*, которые сформулированы в рамках конкретного проекта разработки соответствующего банкомата.

Отдельные небольшие по своему объему *сценарии* могут быть размещены на диаграмме в форме примечаний.

Примечание (note) предназначено для включения в модель произвольной текстовой информации, имеющей непосредственное отношение к контексту разрабатываемого проекта.

Это могут быть комментарии разработчика (например, дата и версия разработки диаграммы или ее отдельных компонентов), ограничения (например, на значения отдельных связей или экземпляры сущностей) и *помеченные значения*. Графически примечания на всех типах диаграмм обозначаются прямоугольником с "загнутым" верхним правым уголком (См.рисунок 2).

Текст примечания размещается внутри этого прямоугольника. Примечание может относиться к любому элементу диаграммы, в этом случае их соединяет пунктирная линия. Если примечание относится к нескольким элементам, то от него проводятся соответственно, несколько линий.



Рисунок 2 - Примеры примечаний на диаграммах вариантов использования

Если в примечании указывается *ключевое слово* <<**constraint**>>, то данное примечание является ограничением, налагаемым на соответствующий элемент модели, но не на саму диаграмму. При этом *запись* ограничения заключается в *фигурные скобки*.

Рекомендации по разработке

Общее количество актеров в модели не должно превышать 20, а вариантов использования - 50. В противном случае модель теряет свою наглядность и заменяет собой одну из некоторых других диаграмм.

- Определить главных или первичных и второстепенных актеров
- Определить цели главных актеров по отношению к системе
- Сформулировать основные варианты использования, которые специфицируют *функциональные требования* к системе

- Упорядочить варианты использования по степени убывания риска их реализации
- Рассмотреть все базовые варианты использования в порядке убывания их степени риска
- Выделить участников, интересы, предусловия и постусловия выполнения выбранного варианта использования
- Написать успешный *сценарий* реализации выбранного варианта использования
- Определить исключения или неуспех в выполнении *сценария* варианта использования
- Написать *сценарии* для всех исключений
- Выделить общие варианты использования и изобразить их взаимосвязи с базовыми со стереотипом <<include>>
- Выделить варианты использования для исключений и изобразить их взаимосвязи с базовыми со стереотипом <<extend>>
- Проверить диаграмму на отсутствие дублирования вариантов использования и актеров

Отдельный экземпляр варианта использования по своему содержанию является выполнением последовательности действий, которая инициализируется посредством экземпляра сообщения от экземпляра актера. В качестве отклика или ответной реакции на сообщение актера выполняется последовательность действий, установленная для данного варианта использования. При этом актеры могут генерировать новые сообщения для инициирования вариантов использования.

Подобное взаимодействие будет продолжаться до тех пор, пока не закончится выполнение требуемой последовательности действий экземпляром варианта использования, и указанный в модели экземпляр актера не получит требуемый экземпляр сервиса. Окончание взаимодействия означает отсутствие инициализации сообщений от актеров для базовых вариантов использования.

Варианты использования могут быть дополнительно специфицированы примечаниями с текстом, которые в последующем могут стать прототипами операций и методов совместно с атрибутами. Дальнейшая разработка моделей связана с реализацией вариантов использования в виде графа деятельности, посредством конечного автомата или любого другого механизма логического представления поведения, включающего предусловия и постусловия. Взаимодействие между вариантами использования и актерами может уточняться на диаграмме кооперации, когда описываются взаимосвязи между системой, содержащей эти варианты использования, и окружением или внешней средой этой системы.