**ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ СРЕДНЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ МАРИЙ ЭЛ**

"**ЙОШКАР**-**ОЛИНСКИЙ** **ТЕХНОЛОГИЧЕСКИЙ** **КОЛЛЕДЖ**"

**ОТЧЁТ**

**О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ**

**«Диаграмма прецедентов по теме Космос»**

Студента 3 курса, А-31 группы

Специальность 09.02.07 – «Информационные системы и программирования»

Квалификация «Администратор БД»

Руководитель: Пинешкин Ю.С.

Разработал:Тимофеев М.А.

г. Йошкар-Ола, 2022

***Диаграмма вариантов использования (англ. use case diagram)*** в UML — диаграмма, отражающая отношения между акторами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне.

***Прецедент*** — возможность моделируемой системы (часть её функциональности), благодаря которой пользователь может получить конкретный, измеримый и нужный ему результат. Прецедент соответствует отдельному сервису системы, определяет один из вариантов её использования и описывает типичный способ взаимодействия пользователя с системой. Варианты использования обычно применяются для спецификации внешних требований к системе.

Назначение

***Основное назначение диаграммы*** — описание функциональности и поведения, позволяющее заказчику, конечному пользователю и разработчику совместно обсуждать проектируемую или существующую систему.

При моделировании системы с помощью диаграммы прецедентов системный аналитик стремится:

* чётко отделить систему от её окружения;
* определить действующих лиц (акторов), их взаимодействие с системой и ожидаемую функциональность системы;
* определить в глоссарии предметной области понятия, относящиеся к детальному описанию функциональности системы (то есть прецедентов).

Работа над диаграммой может начаться с текстового описания, полученного при работе с заказчиком. При этом нефункциональные требования (например, конкретный язык или система программирования) при составлении модели прецедентов опускаются (для них составляется другой документ).

Элементы

Для отражения модели прецедентов на диаграмме используются:

* ***рамки системы (англ. system boundary)*** — прямоугольник с названием в верхней части и эллипсами (прецедентами) внутри. Часто может быть опущен без потери полезной информации;
* ***актор (англ. actor)*** — стилизованный человечек, обозначающий набор ролей пользователя (понимается в широком смысле: человек, внешняя сущность, класс, другая система), взаимодействующего с некоторой сущностью (системой, подсистемой, классом). Акторы не могут быть связаны друг с другом (за исключением отношений обобщения/наследования);
* ***прецедент*** — эллипс с надписью, обозначающий выполняемые системой действия (могут включать возможные варианты), приводящие к наблюдаемым акторами результатам. Надпись может быть именем или описанием (с точки зрения актора) того, «что» делает система (а не «как»). Имя прецедента связано с непрерывным (атомарным) сценарием — конкретной последовательностью действий, иллюстрирующей поведение. В ходе сценария акторы обмениваются с системой сообщениями. Сценарий может быть приведён на диаграмме прецедентов в виде UML-комментария. С одним прецедентом может быть связано несколько различных сценариев.

***Отношения между прецедентами***

Часть дублирующейся информации в модели прецедентов можно устранить указанием связей между прецедентами:

* ***обобщение прецедента*** — стрелка с не закрашенным треугольником (треугольник ставится у более общего прецедента),
* ***включение прецедента*** — пунктирная стрелка со стереотипом «include»,
* ***расширение прецедента*** — пунктирная стрелка со стереотипом «extend» (стрелка входит в расширяемый прецедент, в дополнительном разделе которого может быть указана точка расширения и, возможно в виде комментария, условие расширения).

Правила

При работе с вариантами использования важно помнить несколько простых правил:

* каждый прецедент относится как минимум к одному действующему лицу;
* каждый прецедент имеет инициатора;
* каждый прецедент приводит к соответствующему результату.

Важность использования диаграммы прецедентов

Как уже упоминалось ранее, диаграммы прецедентов используются для сбора требований к использованию системы. В зависимости от ваших требований вы можете использовать эти данные различными способами. Ниже приведены несколько способов их использования.

* ***Идентификация функций и как с ними взаимодействуют роли*** – основное назначение диаграмм сценариев использования.
* ***Для представления системы на высоком уровне*** – Особенно полезно при представленииее руководителям или заинтересованным сторонам. Вы можете выделить роли, которые взаимодействуют с системой, и функциональные возможности, предоставляемые системой, не углубляясь во внутреннюю работу системы.
* ***Идентификация внутренних и внешних факторов*** – Это может показаться простым, но в больших сложных проектах система может быть идентифицирована как внешняя роль в другом случае использования.

Объекты диаграммы прецедентов

Использовать диаграммы корпуса состоят из 4 объектов:

* ***Актер в использует диаграмму прецедентов*** – это **любая сущность, которая выполняет роль** в одной данной системе. Это может быть человек, организация или внешняя система и обычно рисуется как скелет, показанный ниже.
* Случай использования **представляет собой функцию или действие внутри системы.** Она нарисована как овал и названа функцией.
* Система используется для **определения сферы применения** и нарисована в виде прямоугольника. Это необязательный элемент, но полезный при визуализации больших систем. Например, вы можете создать все случаи использования, а затем использовать системный объект для определения области применения вашего проекта. Или вы даже можете использовать его, чтобы показать различные области, охваченные в разных релизах.
* Пакет является еще одним дополнительным элементом, который чрезвычайно полезен в сложных диаграммах. Подобно [диаграммам классов](https://creately.com/diagram-type/class-diagram), пакеты **используются для группировки случаев использования.** Они нарисованы, как показано на рисунке ниже.

Рекомендации по диаграммам прецедентов

Несмотря на то, что диаграммы использования могут быть использованы для различных целей, существуют некоторые общие рекомендации, которым необходимо следовать при рисовании примеров использования.

К ним относятся стандарты именования, направления стрелок, размещение вариантов использования, использование системных блоков, а также правильное использование отношений.

Мы подробно освещали эти рекомендации в отдельном посте блога. Так что продолжайте и ознакомьтесь с рекомендациями по диаграммам прецедентов.

Отношения в диаграммах вариантов использования

Существует пять типов отношений на диаграмме прецедентов:

* Ассоциация между актером и случаем использования
* Обобщение актера
* Расширить отношения между двумя случаями использования
* Включить взаимосвязь между двумя случаями использования
* Обобщение случая использования

Мы рассмотрели все эти отношения в отдельном посте блога, который содержит примеры с изображениями. Мы не будем вдаваться в подробности в этом посте, но вы можете проверить отношения на диаграммах прецедентов использования.

Как создавать диаграммы прецедентов использования

До сих пор вы узнали об объектах, отношениях и руководствах, которые критичны при рисовании диаграмм с примерами из практики. Я объясню различные процессы на примере банковской системы.

***Выявление актеров***

***Актеры*** – это внешние объекты, взаимодействующие с вашей системой. Это может быть человек, другая система или организация. В банковской системе наиболее очевидным действующим лицом является клиент. Другие актеры могут быть банковскими служащими или кассирами в зависимости от роли, которую вы пытаетесь показать в случае использования.

Примером внешней организации может служить налоговый орган или центральный банк. Кредитный процессор является хорошим примером внешней системы, связанной в качестве агента.

***Определение случаев использования***

Теперь пришло время идентифицировать случаи использования. Хороший способ сделать это – определить, что нужно участникам системы. В банковской системе клиенту необходимо будет открывать счета, вводить и выводить средства, запрашивать чековые книги и выполнять аналогичные функции. Так что все это можно рассматривать как случаи использования.

Случаи использования верхнего уровня всегда должны обеспечивать полную функцию, требуемую для агента. В зависимости от сложности системы вы можете расширить или включить случаи использования.

После того, как вы определили актёров и верхний уровень использования, у вас есть базовое представление о системе. Теперь вы можете точно настроить его и добавить к нему дополнительные слои деталей.

***Ищите общие функциональные возможности для использования Включать***

Ищите общие функциональные возможности, которые могут быть повторно использованы в системе. Если вы найдете два или более случаев использования, которые имеют общую функциональность, вы можете извлечь общие функции и добавить его в отдельный случай использования. Затем вы можете подключить его через include relationship, чтобы показать, что он всегда вызывается, когда выполняется исходный сценарий использования.

***Возможно ли обобщение актеров и случаев использования***

Могут быть случаи, когда агенты ассоциируются с аналогичными случаями использования, в то время как запускается несколько случаев использования, уникальных только для них. В таких случаях можно обобщить агент, чтобы показать наследование функций. Аналогичную вещь можно сделать и в случае использования.

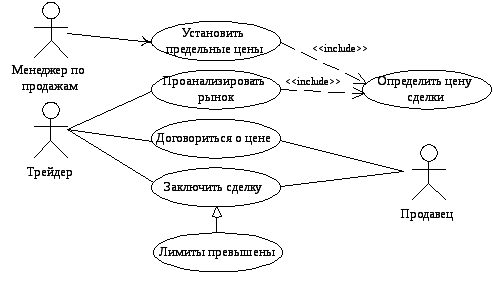
Одним из лучших примеров этого является случай использования “Оплатить” в платежной системе. Вы можете далее обобщить его до “Оплатить кредитной картой”, “Оплатить наличными”, “Оплатить чеком” и т.д. Все они имеют атрибуты и функциональность оплаты со специальными уникальными для них сценариями.

***Необязательные функции или дополнительные функции***

Есть некоторые функции, которые срабатывают опционально. В таких случаях можно использовать отношения расширения и прикрепить к ним правило расширения. В приведенном ниже примере банковской системы “Рассчитать бонус” является необязательным и срабатывает только при выполнении определенного условия.

Продление не всегда означает, что оно необязательно. Иногда вариант использования, связанный с удлинением, может дополнять вариант использования базы. Следует помнить, что базовый сценарий использования должен быть способен выполнять функцию самостоятельно, даже если сценарий использования расширения не вызывается.

### Шаблоны диаграмм прецедентов использования



*Шаблон варианта прецедентов для системы банкомата*

Мы пошли дальше и создали шаблоны диаграмм прецедентов использования для некоторых распространенных сценариев. Хотя ваша проблема или сценарий не будут в точности такими, вы можете использовать их в качестве отправной точки. Ознакомьтесь с нашими шаблонами [диаграмм прецедентов использования](https://creately.com/diagram-type/templates/use-case).

Особенности построения диаграмм прецедентов

При построении диаграмм прецедентов актеров нужно связывать с прецедентами, один актер может выполнять несколько прецедентов, и наоборот, у одного прецедента может быть несколько актеров, которые его выполняют. При построении диаграмм прецедентов нельзя использовать связи между двумя актерами. При построении диаграмм прецедентов актер всегда инициализирует тот или иной прецедент, иными словами стрелка всегда должна идти от актера к прецеденту.

Помимо связей между актерами и прецедентами, на диаграммах могут быть также представлены и отношения между прецедентами. Отношение включения(include) используется, когда в системе имеется фрагмент, который повторяется многократно в нескольких прецедентах, и вы не хотели бы, чтобы его описание копировалось в каждом из этих прецедентов (см. рис. 1.4). Отношение расширения (extend) используется тогда, когда имеется прецедент, который подобен другому прецеденту, но намного шире его, кроме того, при построении модели расширяющий прецедент может дополнять поведение базового прецедента, но для этого в базовом прецеденте должны быть определены так называемые «точки расширения.

В самом языке UML пакет Прецеденты является подпакетом пакета Элементы поведения. Последний специфицирует понятия, при помощи которых определяют функциональность моделируемых систем. Элементы пакета Прецеденты являются первичными по отношению к тем, с помощью которых могут быть описаны сущности, такие как системы и подсистемы. Однако внутренняя структура этих сущностей никак не описывается. Базовые элементы этого пакета — прецедент и актер.