

Прецеденты

Часть 1

Прецеденты

- Прецеденты – это повествовательные истории об использовании системы, которые широко используются для осмысления и формулировки требований. Они оказывают влияние на множество аспектов проекта, включая ООА/П.
- Прецеденты нужно продумывать детально. Основная идея состоит в исследовании и формулировке функциональных требований путем написания историй “из жизни системы”. Эти истории помогают сформулировать различные задачи и представляют собой сценарии использования системы. На первый взгляд, описать прецеденты не сложно, хотя зачастую достаточно сложно определить, что требуется от системы и описать это на нужном уровне детализации.

Исполнитель

- Исполнителем (actors) будем называть сущность, обладающую поведением, например, человека (идентифицируемого по роли, к примеру, кассира), компьютерную систему или организацию.

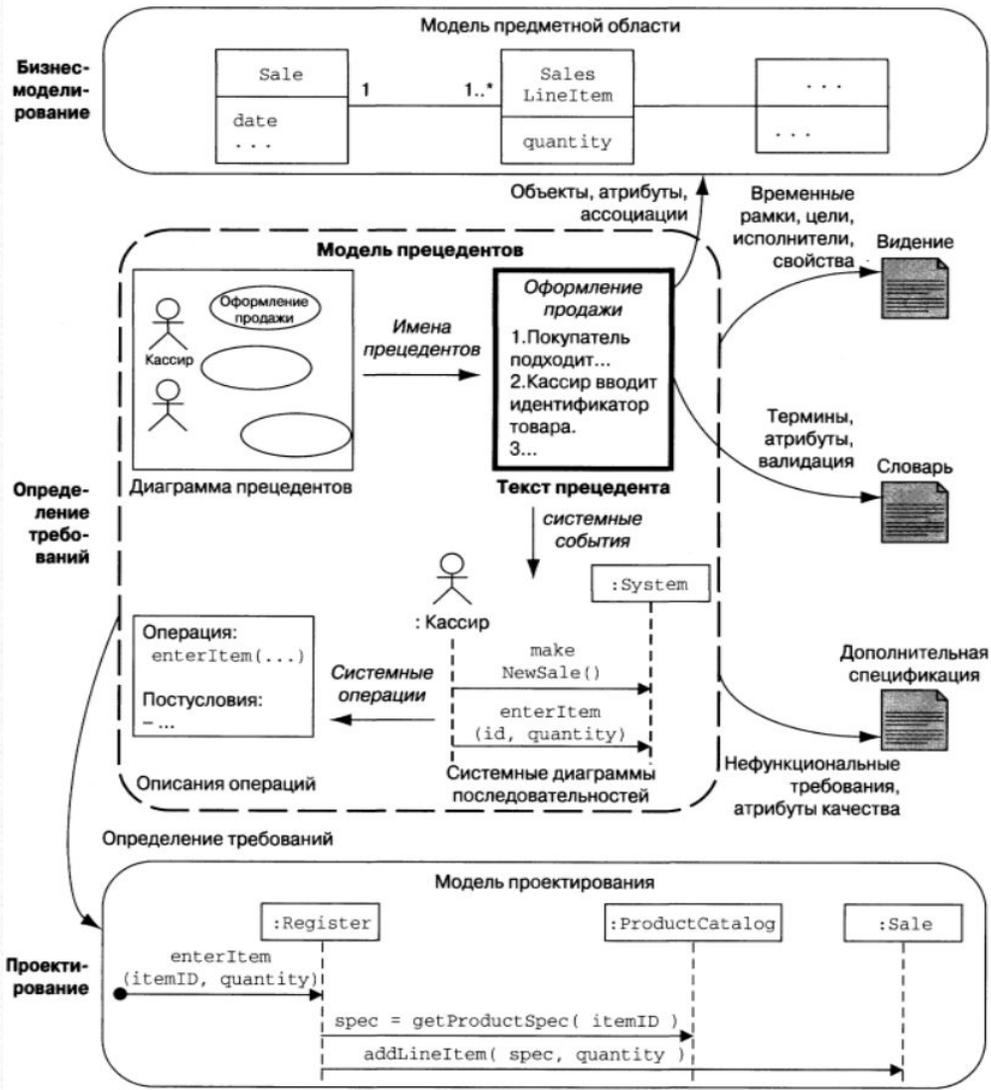
Сценарий

- Сценарий (scenario) – это специальная последовательность действий или взаимодействий между исполнителем и системой. Его иногда так же называют экземпляром прецедента (use case instance). Это один конкретный сценарий использования системы либо один проход прецедента, например, сценарий успешной покупки товара за наличный расчет либо сценарий неудачного завершения покупки из-за прерванной транзакции по обработке данных кредитной картой.

Прецедент

- Прецедент (use case) – это набор взаимосвязанных успешных и неудачных сценариев, описывающих использование системы исполнителем для решения одной из задач.

Пример взаимосвязи артефактов UP



Прецеденты и модель прецедентов

- В контексте UP модель прецедентов (Use-Case Model) относится к дисциплине “Требования”. Действительно, требования – это весь набор прецедентов, т.е. модель функционирования системы и ее окружения.
- Описание прецедентов – это текстовые документы, а не диаграммы. Моделирование прецедентов – это процесс написания текста, а не рисование.
- Модель прецедентов – это не единственный артефакт анализа требований в рамках UP. К данной стадии относятся также дополнительная спецификация, словарь терминов, видение системы и описание бизнес-правил.

Прецеденты и модель прецедентов

- Модель прецедентов может включать диаграммы прецедентов на UML, отражающие имена прецедентов и исполнителей, а также их взаимоотношения. Диаграммы прецедентов позволяют оценить рамки системы и ее окружение, а также обеспечивают простой способ перечисления имен прецедентов.

Зачем нужны описания прецедентов

- У потребителей и конечных пользователей есть свои задачи, решение которых должна обеспечивать компьютерная система. Эти задачи могут варьироваться от регистрации торговых операций до оценивания объемов добычи нефти из перспективных месторождений. Существует несколько способов выделения этих задач или системных требований. Наилучшие из них достаточно просты и доступны, поскольку это облегчает участие конечных пользователей в определении требований к системе. А участие в этом процессе потребителей снижает риск провала проекта.

Зачем нужны описания прецедентов

- Одной из главных причин провала проектов по разработке программных систем является недостаточное привлечение пользователей к процессу разработки. Поэтому важность участия пользователей в процессе анализа системы трудно переоценить.
- Важной особенностью описания прецедентов является их ориентация на цели и задачи пользователя. В процессе описания необходимо задавать вопросы: “Кто является пользователем системы? Каковы типичные сценарии использования системы? Каковы цели и задачи пользователей?”. Такой взгляд на систему гораздо больше ориентирован на пользователя, чем обычный список системных требований.

Прецеденты и функциональные требования

- Прецеденты — это требования. В основном, это функциональные требования, указывающие на то, что должна делать система. В контексте типов требований, определяемых моделью FURPS+, основное внимание уделяется функциональным требованиям. Однако остальные типы требований тоже могут быть связаны с прецедентами. В рамках UP и большинства других современных методов, прецеденты являются основным механизмом, рекомендуемым для их определения и исследования.

Три типа исполнителей

- Исполнитель (actor) — это сущность, обладающая поведением. К числу исполнителей может относиться и сама рассматриваемая система, если она вызывает службы других систем. В прецеденте могут участвовать основные и вспомогательные (второстепенные) исполнители. Исполнителями являются не только люди, но и организации, машины и программы.

Основной исполнитель

- Основной исполнитель (primary actor) — его задачи выполняются с использованием системы. Примером основного исполнителя является кассир.
- Зачем его идентифицировать? Чтобы определить цели пользователя, на основе которых формулируются прецеденты.

Вспомогательный исполнитель

- Вспомогательный исполнитель (supporting actor) — обслуживает систему (например, предоставляет информацию). Примером вспомогательного исполнителя является служба авторизации платежей.
- Зачем его идентифицировать? Чтобы определить внешние интерфейсы и протоколы.

Закулисный исполнитель

- Закулисный исполнитель (offstage actor) — заинтересован в реализации прецедента, но не является основным или вспомогательным исполнителем. Примером закулисного исполнителя является налоговая служба.
- Зачем его идентифицировать? Чтобы удостовериться, что все интересы определены и удовлетворены. Интересы закулисных исполнителей обычно не очевидны и их легко упустить из виду, если не идентифицировать их в явной форме.

Основные форматы прецедентов

- Прецеденты описываются в различных форматах. Выделяют несколько уровней формализации описания прецедентов:
- Сжатый
- Свободный
- Развернутый

Сжатый

- Сжатый — аннотация в виде одного абзаца. Обычно она описывает только главный успешный сценарий. Пример такого описания приведен выше для прецедента Оформление продажи (Process Sale).
- Этот формат применяется на стадии анализа требований для быстрого определения задач и масштабов системы. Для подобного описания требуется несколько минут.

Свободный

- Свободный — неформальный стиль описания. Описание прецедента занимает несколько абзацев и охватывает различные сценарии. Примером такого описания является рассмотренный выше прецедент Возврат товара.
- Этот формат применяют в тех же случаях, что и предыдущий.

Развернутый

- Развернутый— наиболее подробный стиль описания. При таком подходе детально описываются все шаги и варианты развития сценария, а также предусловия и результаты.
- Этот формат применяют после определения основных задач системы, когда множество прецедентов уже описаны в сжатом формате. В развернутом формате представляют порядка 10% наиболее важных для приложения прецедентов.

Развернутое описание прецедента

- Развернутые описания прецедентов структурированы и содержат большое количество деталей.
- При итеративной и эволюционной разработке на стадии анализа требований в этом формате представляют порядка 10% жизненно важных прецедентов. Обычно такое описание формируется на первом семинаре по анализу требований, после чего сразу же начинается реализация этих прецедентов в системе.

Раздел описания прецедента	Комментарий
Название прецедента	Осмысленное название, определяющее основную функцию прецедента
Рамки	Разрабатываемая система
Уровень	“Задача, определенная пользователем” или “вспомогательная функция”
Основной исполнитель	Лицо, инициирующее и реализующее работу сценария
Заинтересованные лица и их требования	Кто заинтересован в реализации этого сценария и чего он хочет
Предусловия	Какие условия должны быть выполнены для успешной реализации данного сценария
Результат	Что гарантируется при успешном завершении сценария
Основной успешный сценарий	Типичный ход событий, который приводит к успешному завершению сценария
Расширения	Альтернативные успешные или неуспешные сценарии
Специальные требования	Нефункциональные требования, связанные с данными прецедентом
Список технологий и типов данных	Методы ввода-вывода и форматы данных
Частота использования	Определяет сроки реализации и тестирования
Открытые вопросы	Вопросы, не решаемые реализацией данного прецедента