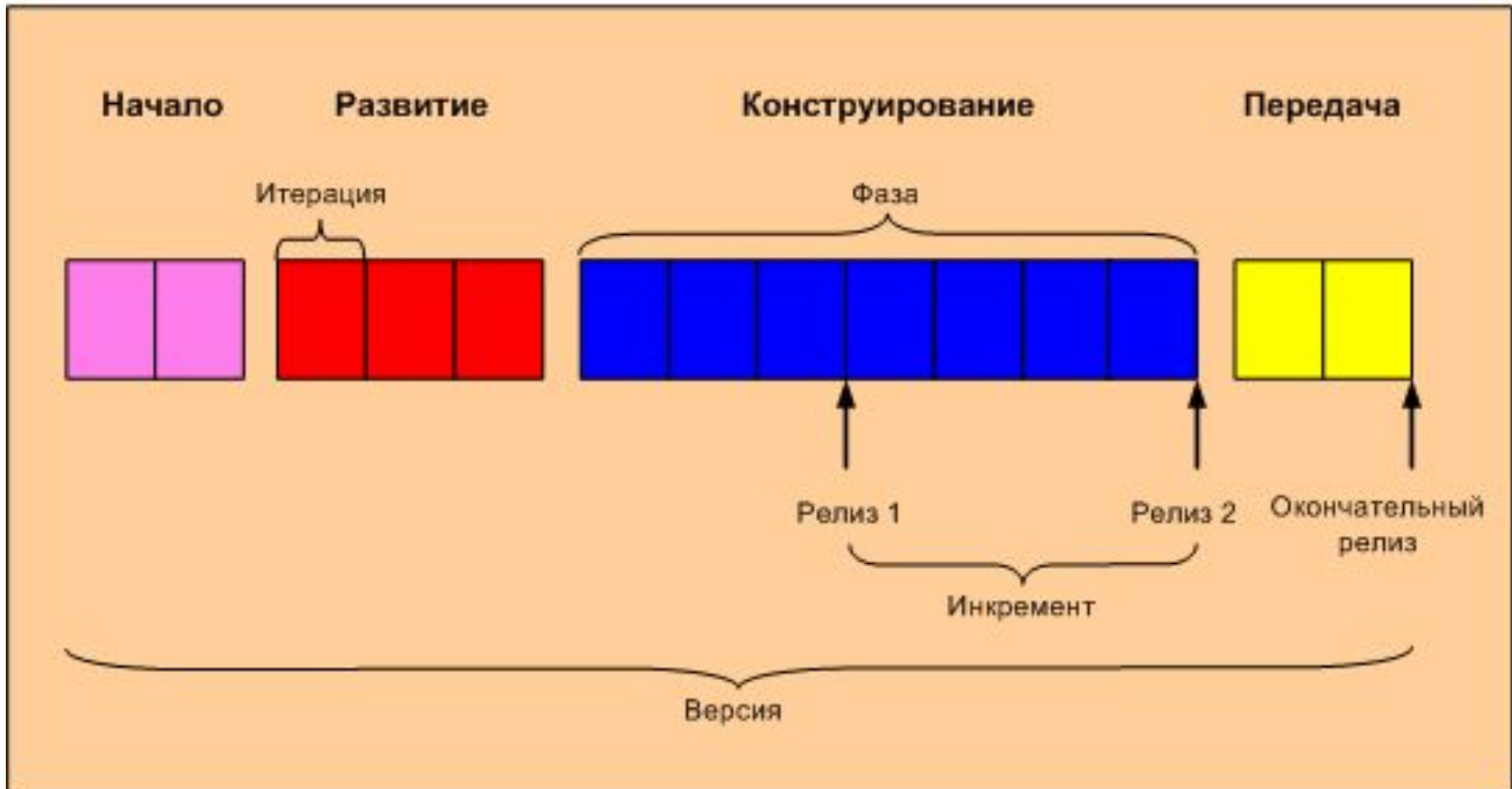


Начальная фаза проекта

Определение требований

Снова об объектных методах
Документы на разных стадиях RUP
О начальной фазе
О требованиях
Спецификация прецедентов
Что еще на начальной фазе?

RUP



Объектно-ориентированный анализ

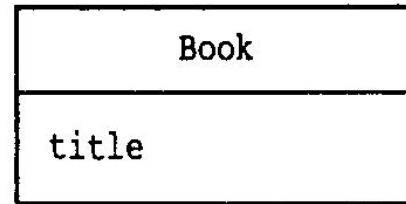
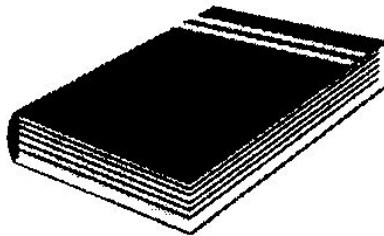
Библиотека:

- Книга
- Библиотека
- Читатель

Объектно-ориентированное проектирование

Понятие предметной области

Представление понятия предметной области



Представление на языке объектно-ориентированного программирования

```
public class Book
{
    private String title;

    public Chapter getChapter(int) {...}
}
```

Документы на разных стадиях

Бизнес-моделирование	Модель предметной области
Требования	Модель прецедентов
	Видение системы
	Дополнительная спецификация
	Словарь терминов
Проектирование	Модель проектирования
	Описание архитектуры
	Модель данных
Реализация	Модель реализации
Управление проектом	План разработки
Тестирование	Модель тестирования
Окружение	Набор документов

Начальная фаза

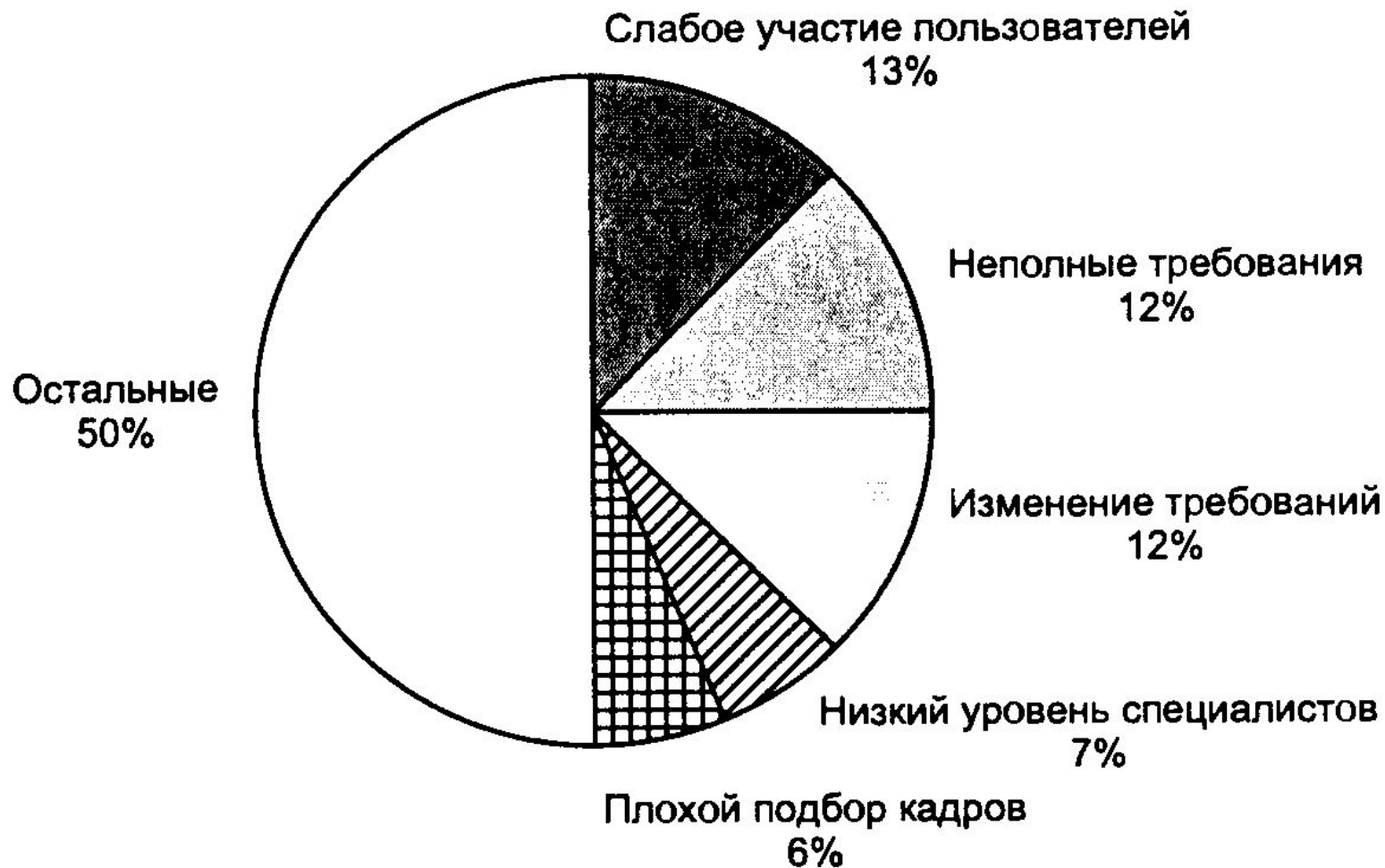
Что на начальной фазе?

- Модель прецедентов
- Видение проекта
- Дополнительная спецификация
- Словарь терминов
- Перечень рисков и план управления ими
- Прототипы и обоснование идеи
- План итерации
- План на следующую фазу и план разработки
- Перечень документов

О требованиях

- **Требования** (requirements) — это возможности или условия, которым должна соответствовать система или проект.

Насколько важна правильная постановка требований?



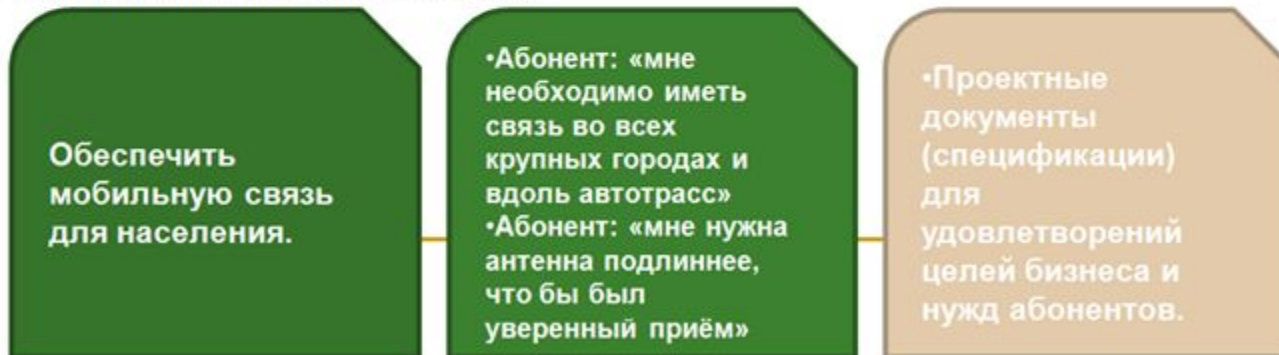
Типы требований

- **Функциональные требования**
- **Удобство**
- Надежность
- Производительность
- Возможность поддержки
- Реализация
- Интерфейс
- Операции
- Пакетирование
- Юридические вопросы

Превращения требований



Пример (компания мобильной связи)



Прецеденты

- **Исполнителем (actor)** будем называть сущность, обладающую поведением, например, человека (идентифицируемого по роли), компьютерную систему или организацию.
- **Сценарий (scenario)** — это специальная последовательность действий или взаимодействий между исполнителями и системой.

Прецеденты это требования?

Да,

НО НЕ ВСЕ

Прецедент типа «Черный ящик»

Стиль «черного ящика»: Система регистрирует покупку

Другой стиль: Система записывает сведения о покупке в базу данных

Самый плохой стиль: Система генерирует оператор SQL *INSERT* для данной продажи

Степень формализации

- **Сжатый** — аннотация в виде одного абзаца.
- **Свободный** — неформальный стиль описания. Описание прецедента занимает несколько абзацев и охватывает различные сценарии.
- **Развернутый** — наиболее подробный стиль описания. При таком подходе детально описываются все шаги и варианты развития сценария, а также предусловия и результаты.

Спецификация прецедента

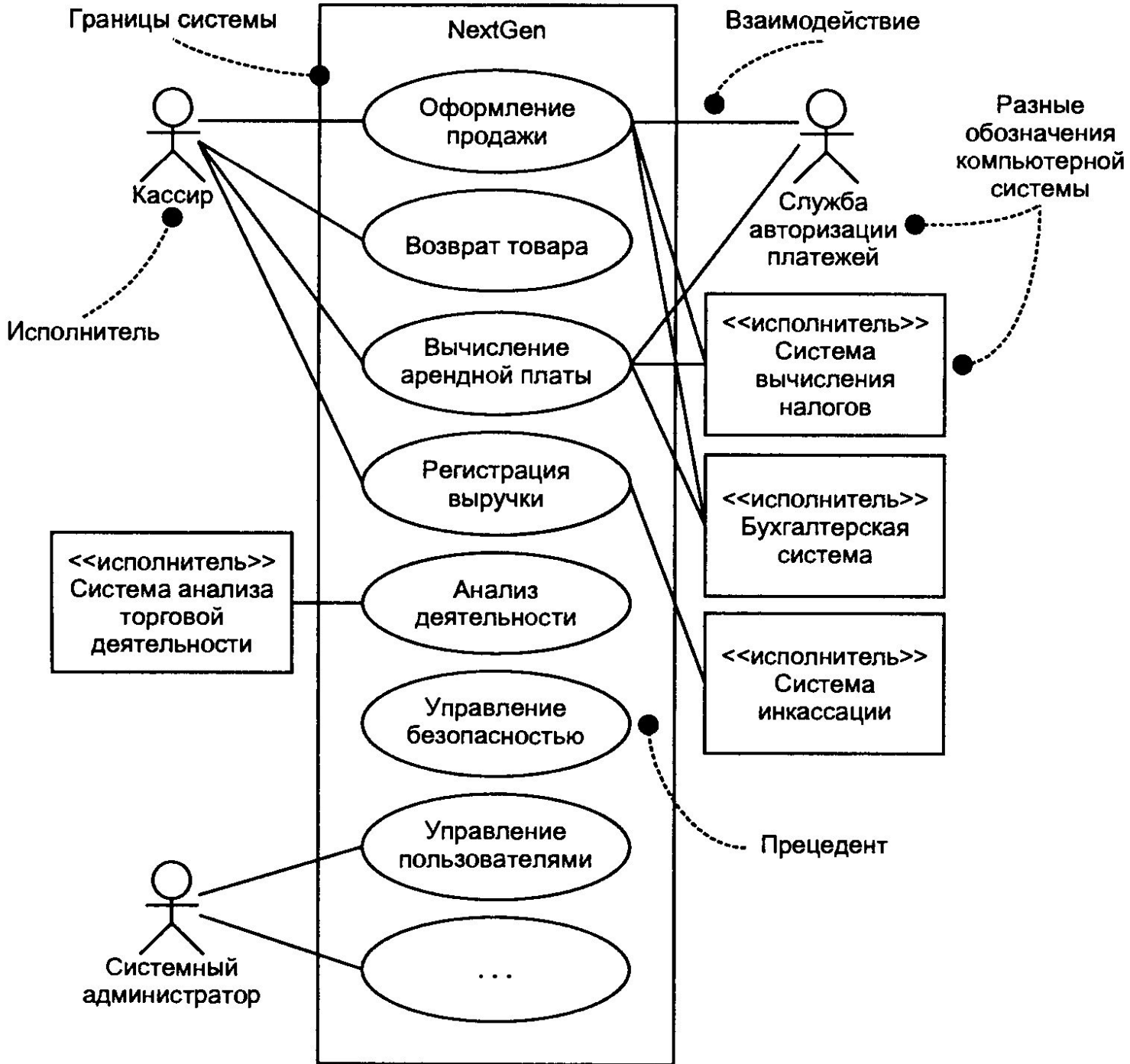
Use Case ID:			
Use Case Name:			
Created By:		Last Updated By:	
Date Created:		Date Last Updated:	

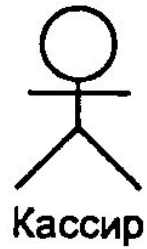
Actor:	
Description:	
Preconditions:	
Postconditions:	
Priority:	
Frequency of Use:	
Normal Course of Events:	
Alternative Courses:	
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

www.itq.ch/tools/use_case_template.doc

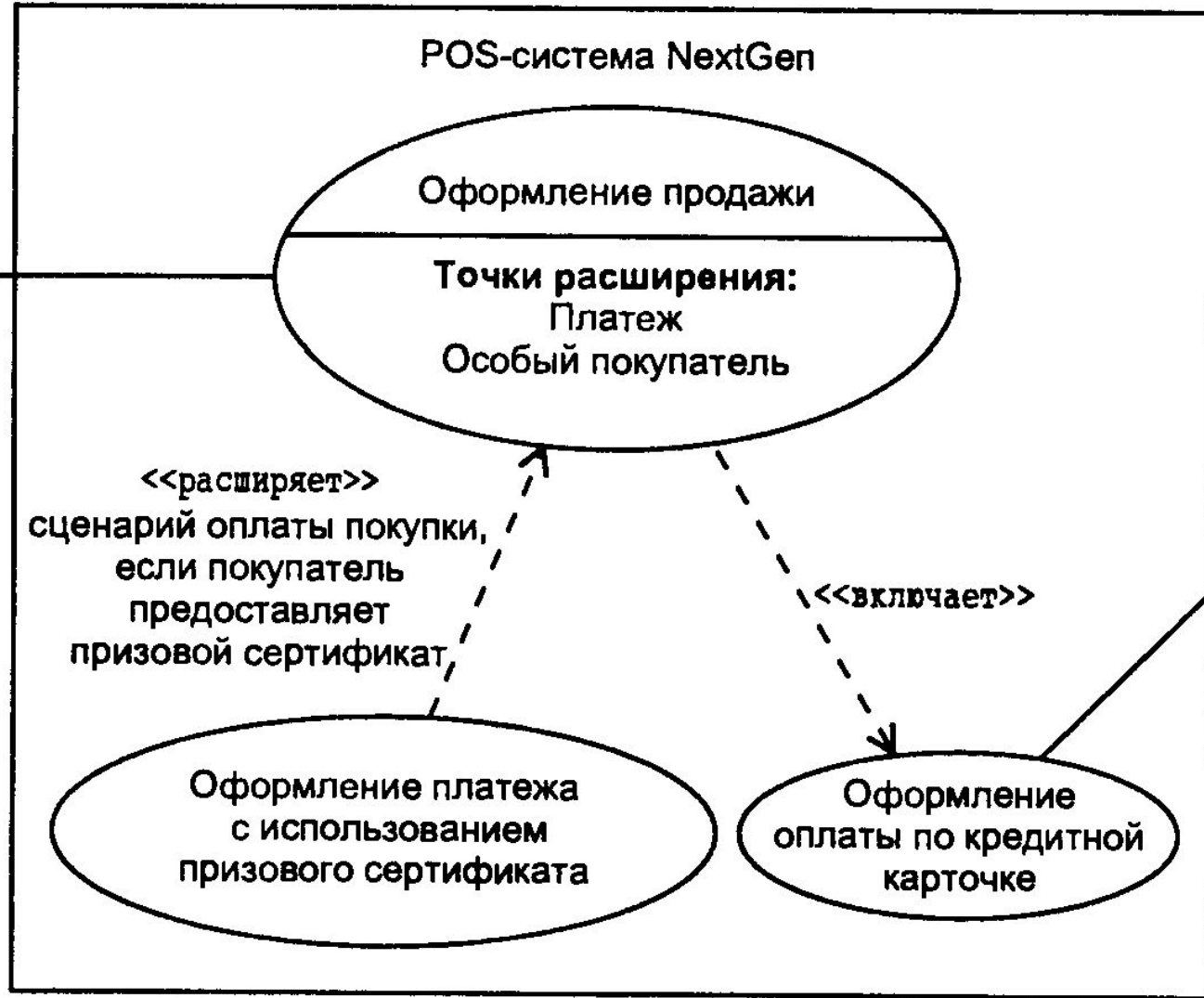
Пример развернутого описания «Оформить продажу»

К. Ларман, Введение в объектно-ориентированный анализ, проектирование и унифицированный процесс UP, стр. 77





Кассир



<<исполнитель>>
Система
авторизации
кредитных
карточек

Примеры артефактов UP

Бизнес-моделирование

Требования



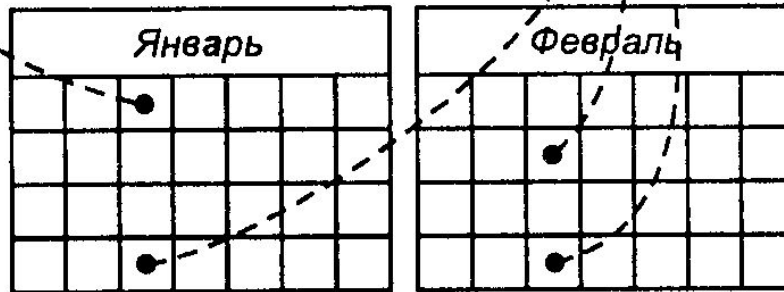
Когда

Один раз на каждой итерации. Придерживайтесь принципа краткости; не старайтесь определить или уточнить все требования

Несколько раз в течение этапа развития

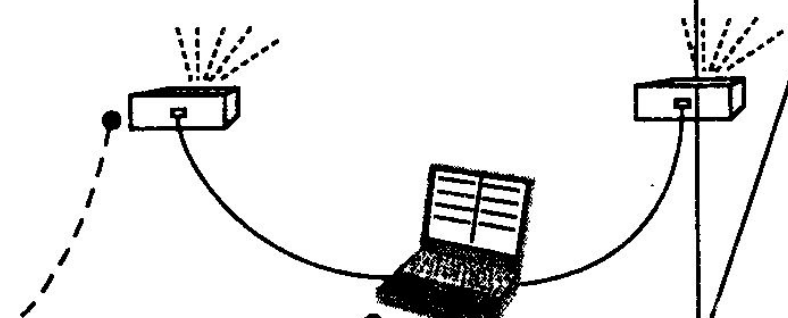
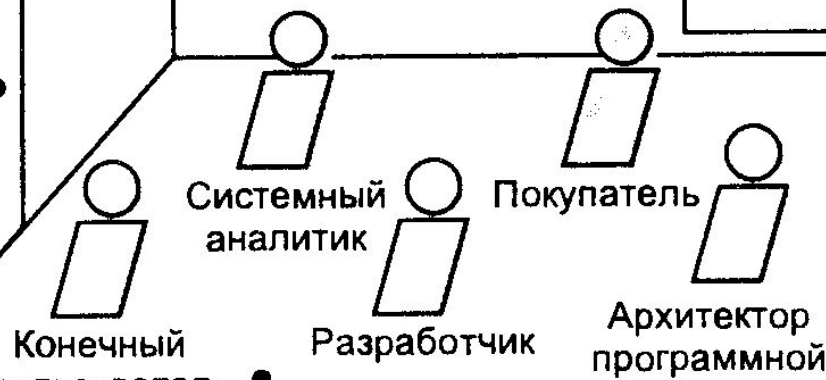
Где

На семинаре по определению требований



Две смежные проекции

Прецедент: Оформление продажи ... Основной успешный сценарий: 1. ... 2. ... 3. ... Расширения:	Прецедент: Возврат товара ... Основной успешный сценарий: 1. ... 2. ... 3. ... Расширения:
--	--



Что нужно кроме
прецедентов?

Другие требования

- Дополнительная спецификация
- Словарь терминов
- Видение (View)

Дополнительная спецификация

Функциональность (общая для многих прецедентов):

- Регистрация событий и обработка ошибок
- Подключаемые бизнес-правила
- Безопасность

Удобство использования:

- Человеческие факторы

Дополнительная спецификация

Надежность:

- Возможность восстановления информации

Производительность

Возможности поддержки:

- Адаптация
- Конфигурирование

Дополнительная спецификация

Дополнительные ограничения

Приобретаемые компоненты

Бесплатные компоненты

Интерфейсы:

- Необходимые аппаратные средства
- Программные интерфейсы

Вопросы законодательства

Видение

Введение

Позиционирование:

- Экономические предпосылки
- Формулировка проблемы
- Место системы

Заинтересованные лица:

- Демографические особенности рынка
- Заинтересованные лица, не являющиеся пользователями системы
- Пользователи системы
- Задачи высокого уровня
- Задачи уровня пользователя
- Окружение

Видение

Обзор:

Перспективы продукта

Преимущества системы

Предположения и зависимости

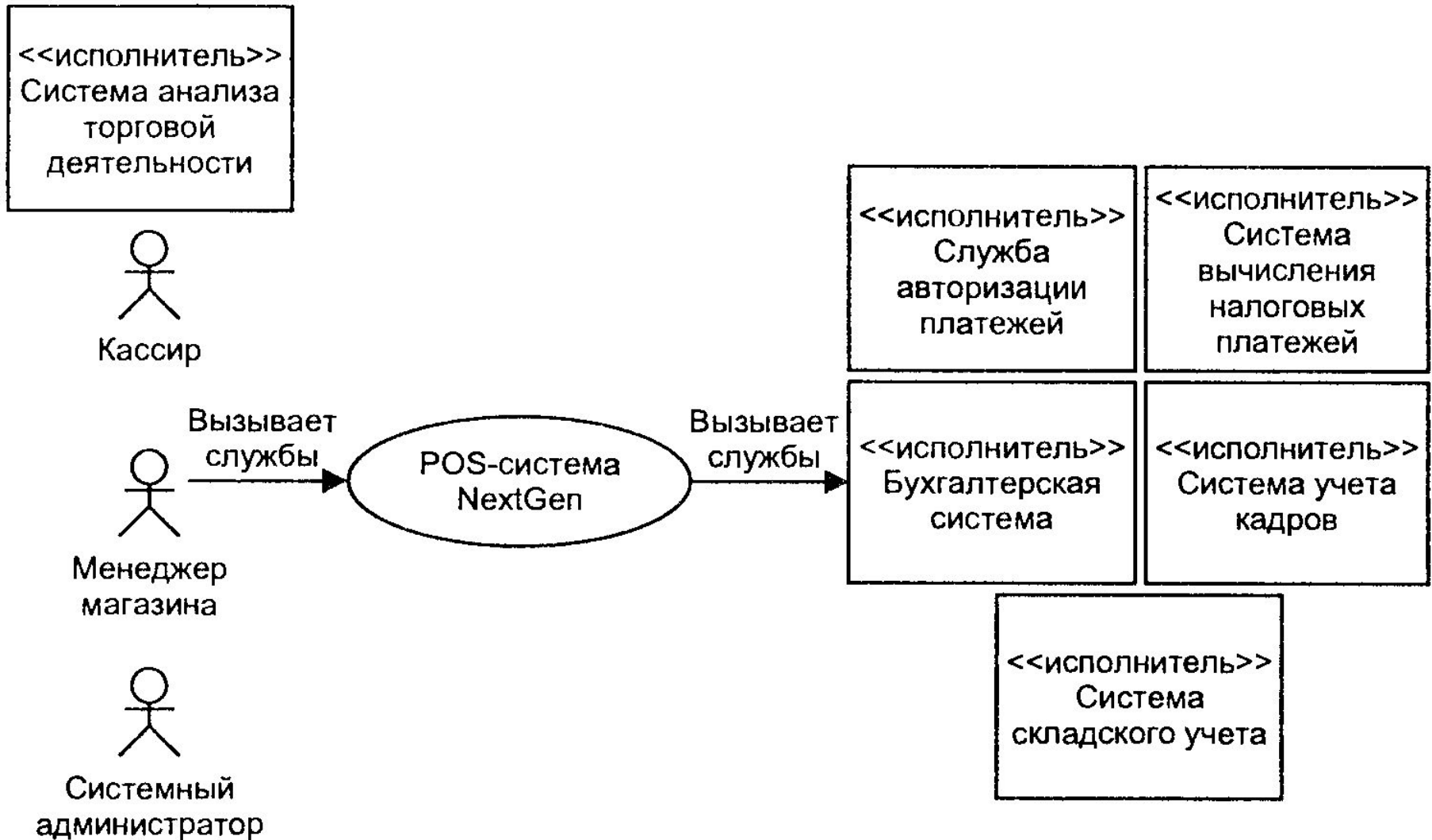
Стоимость и ценообразование

Лицензирование и установка

Основные свойства системы

Другие требования и ограничения

Видение: контекст системы



Не слишком ли много UML на начальной стадии проекта?

Разрабатываемые документы в основном являются текстовыми. Большая часть диаграмм приходится на следующую фазу — фазу развития.

К. Ларман, «Вы не поняли, что такое начальная фаза, если:

- Для большинства проектов она занимает более нескольких недель.
- На этом этапе вы стараетесь определить все требования.
- Надеетесь, что оценки и планы будут реалистичными.
- Занимаетесь определением архитектуры, хотя это надо делать на стадии развития.
- Считаете правильной следующую последовательность действий:
 - 1) определение требований;
 - 2) проектирование архитектуры;
 - 3) реализация.
- У вас отсутствует артефакт “Видение”.
- Имена большинства прецедентов и исполнителей не определены.
- Все прецеденты описываются во всех подробностях.
- Ни один из прецедентов не описан в деталях (на самом деле, 10-20% прецедентов должны быть подробно описаны, чтобы оценить масштаб проблемы)»