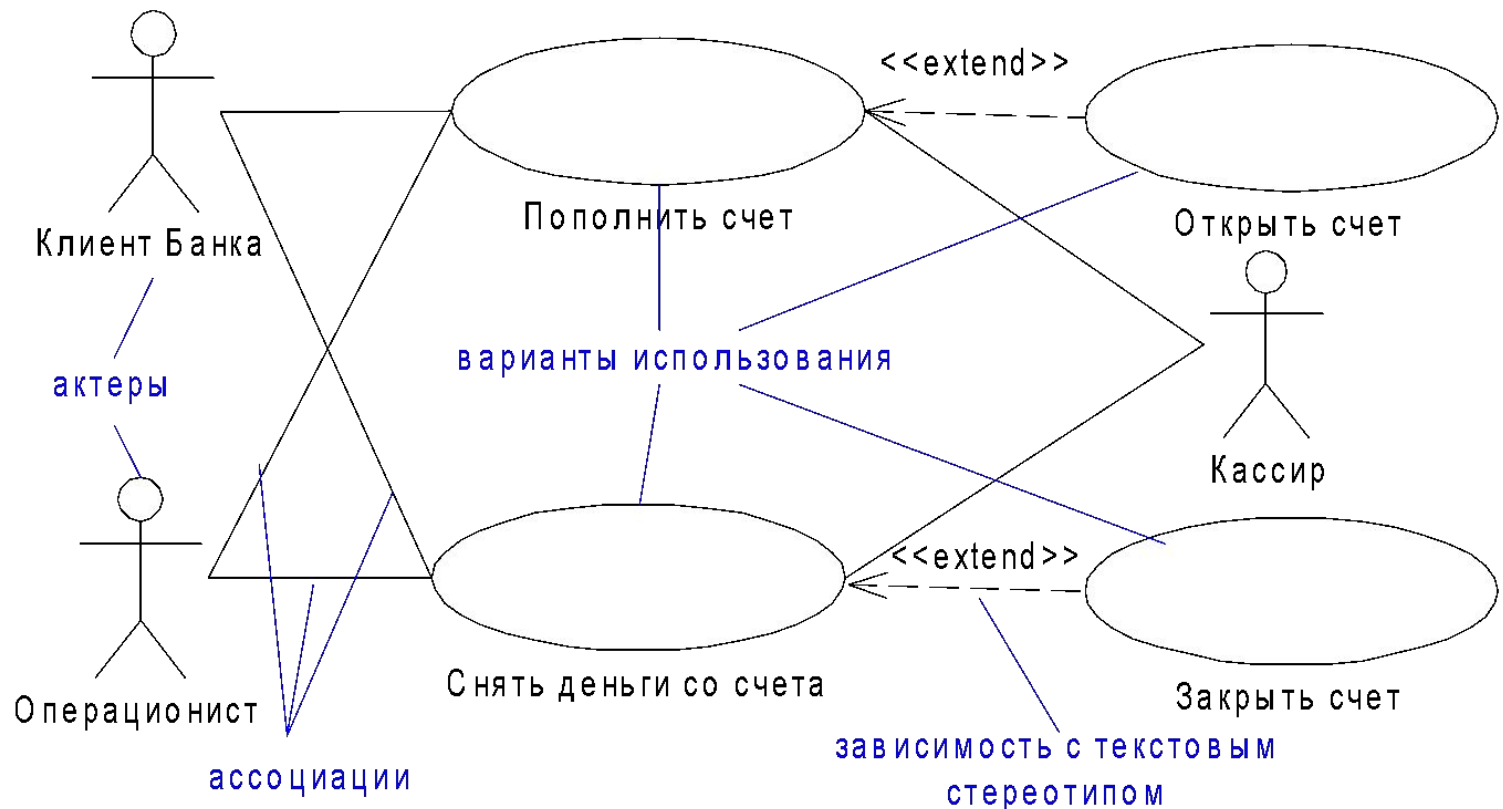


Диаграмма вариантов  
использования (use case  
diagram)

# Диаграмма вариантов использования (*use case diagram*)

- диаграмма, на которой изображаются варианты использования проектируемой системы, заключенные в границу системы, и внешние актеры, а также определенные отношения между актерами и вариантами использования



# Назначение диаграммы вариантов использования

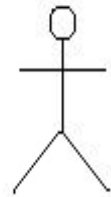
- Определить общие границы функциональности проектируемой системы в контексте моделируемой предметной области.
- Специфицировать требования к функциональному поведению проектируемой системы в форме вариантов использования.
- Разработать исходную концептуальную модель системы для ее последующей детализации в форме логических и физических моделей.
- Подготовить исходную документацию для взаимодействия разработчиков системы с ее заказчиками и пользователями

# Проектируемая система и ее окружение

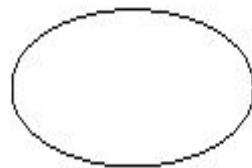


- *Субъект (subject)* – любой элемент модели, который обладает функциональным поведением

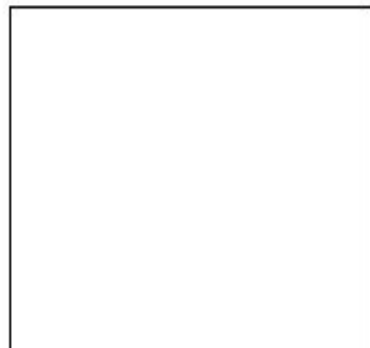
# Основные обозначения на диаграмме вариантов использования



actor



use case



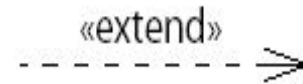
system boundary



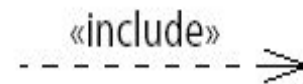
communication  
association



generalization



extend



include

# 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
- Отвечает на вопрос «Что должна выполнять система?», не отвечая на вопрос «Как она должна выполнять это?»
- Имена – отглагольное существительное или глагол в неопределенной форме

Проверка состояния  
текущего счета клиента

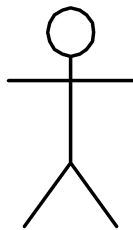
<<use case>>

Формирование отчета по  
выполненным заказам

Формирование отчета по  
выполненным заказам

# Актер (actor)

- – любая внешняя по отношению к проектируемой системе сущность, которая взаимодействует с системой и использует ее функциональные возможности для достижения определенных целей или решения частных задач
- *Примеры актеров:* кассир, клиент банка, банковский служащий, президент, продавец магазина, менеджер отдела продаж, пассажир авиарейса, водитель автомобиля, администратор гостиницы, сотовый телефон



Клиент банка







Удаленный  
пользователь

# Вопросы для идентификации актеров в системе

- Какие организации или лица будут использовать систему
- Кто будет получать пользу от использования системы
- Кто будет использовать информацию от системы
- Будет ли использовать система внешние ресурсы
- Может ли один пользователь играть несколько ролей при взаимодействии с системой
- Могут ли различные пользователи играть одну роль при взаимодействии с системой
- Будет ли система взаимодействовать с законодательными, исполнительными, налоговыми или другими органами

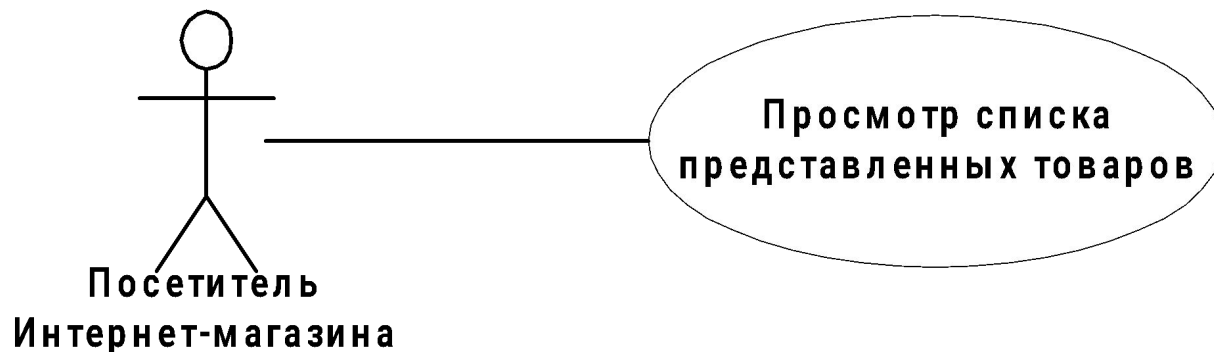


# Отношения на диаграмме вариантов использования

<i>Relationship</i>	<i>Function</i>	<i>Notation</i>
association	The communication path between an actor and a use case that it participates in	
include	The insertion of additional behavior into a base use case that explicitly describes the insertion	«include» 
extend	The insertion of additional behavior into a base use case that does not know about it	«extend» 
use case generalization	A relationship between a general use case and a more specific use case that inherits and adds features to it	

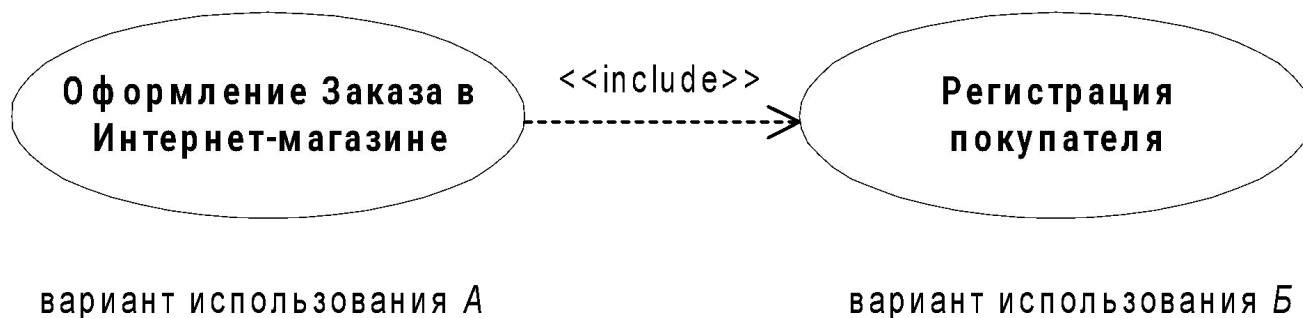
# Отношение ассоциации

- Ассоциация (association) является одним из фундаментальных понятий в языке UML 2.x 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.



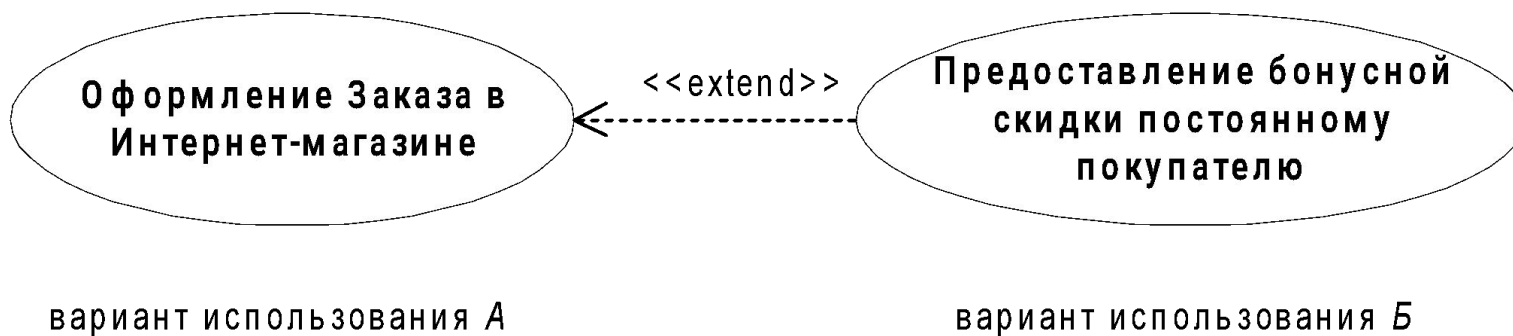
# Отношение включения

- 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*) специфицирует тот факт, что некоторый вариант использования содержит поведение, определенное в другом варианте использования

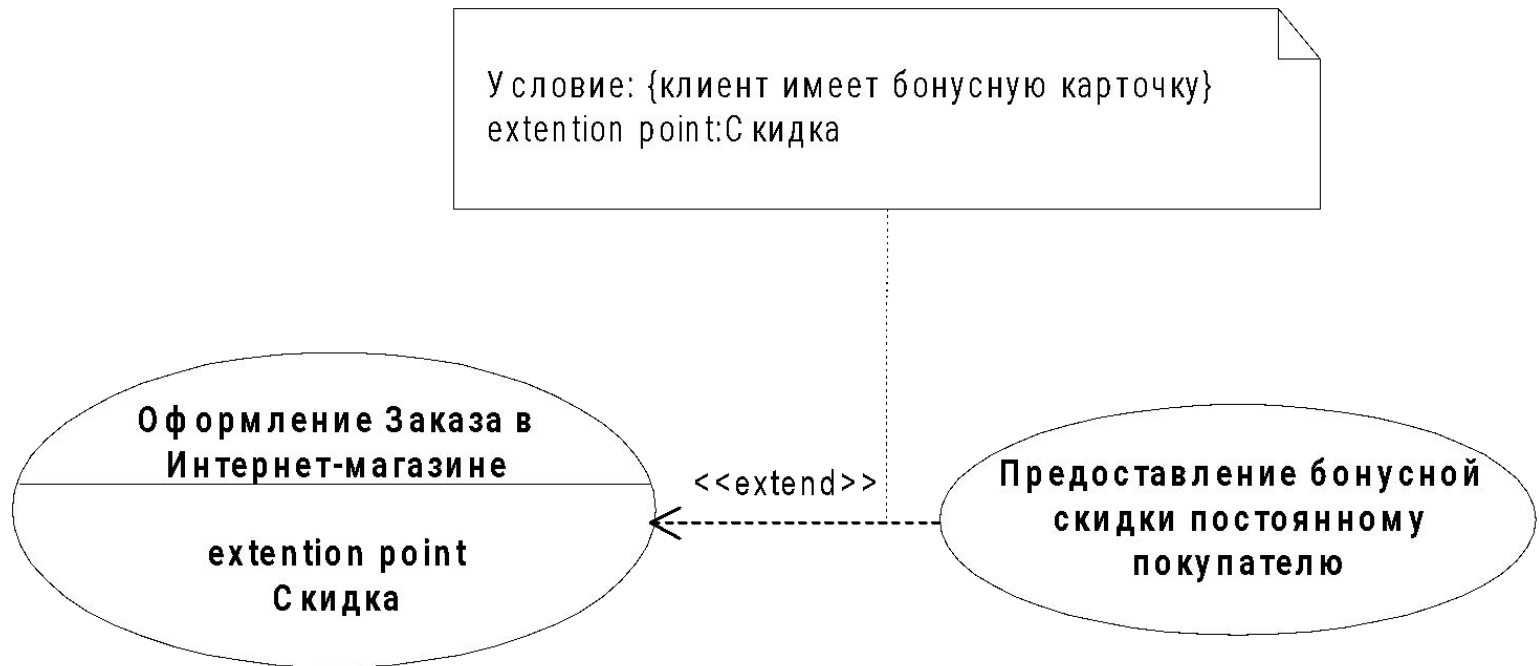


# Отношение расширения

- Отношение *расширения* (*extend*) 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.



# Изображение отношения расширения с условием выполнения



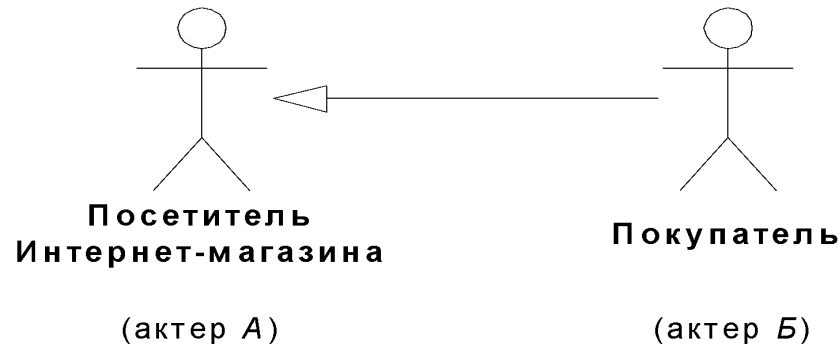
# Отношение обобщения

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



вариант использования А

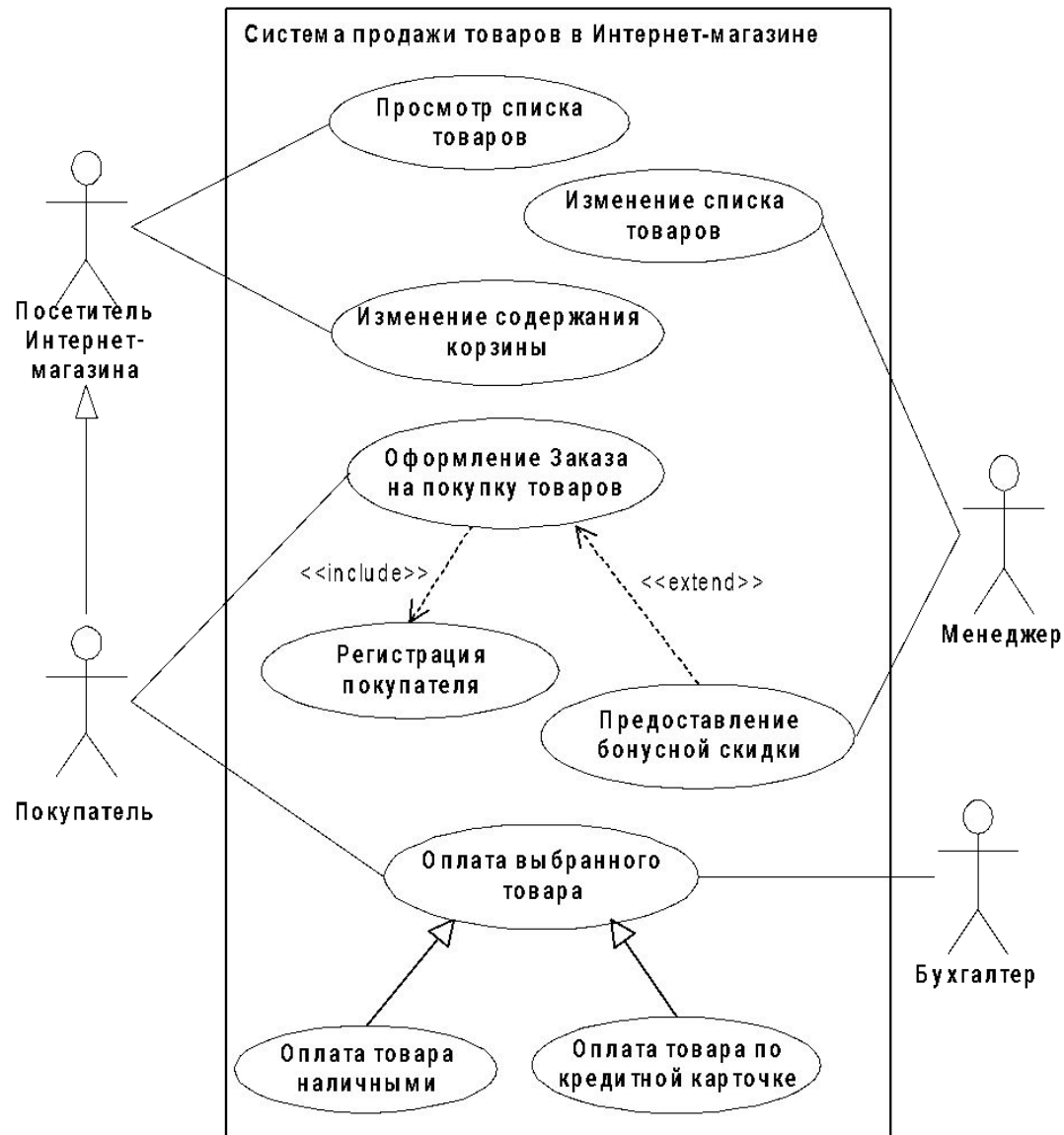
вариант использования Б



(актер А)

(актер Б)

# Пример диаграммы ВИ для системы продажи товаров в Интернет-магазине



# Формализация функциональных требований с помощью диаграммы ВИ

- *Требование (requirement)* – желательное свойство, характеристика или условие, которым должна удовлетворять система в процессе своей эксплуатации
- *Требование к ПО* – некоторое свойство ПО, которым должна обладать система или ее компонент, чтобы удовлетворять условиям контракта, положениям стандартов, формальной спецификации или технической документации
- *Управление требованиями* – это систематический подход к выявлению, организации и документированию требований к системе, а также процесс, в ходе которого вырабатывается и обеспечивается соглашение между заказчиком и разработчиком по поводу меняющихся требований к системе



# Классификация требований – модель FURPS+

- **Functionality**
  - **функциональные требования**
- Usability (требования практичности)
- Reliability (требования надежности)
- Performance (требования производительности)
- Supportability (требования обслуживания и сопровождения)

Дополнительно + IEEE 610.12.1990

- Проектные ограничения
- Требования выполнения
- Требования к GUI
- Физические требования



# Functionality – функциональные требования

- **Функциональные требования** определяют действия, которые должна быть способна выполнить система, без рассмотрения физических особенностей их реализации
- Тем самым функциональные требования определяют внешнее поведение системы
- Лучше всего они описываются в форме *модели вариантов использования*
- Каждому функциональному требованию в этом случае будет соответствовать отдельный *вариант использования*

# Спецификация ВИ с помощью текстовых сценариев

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

Актер	Цель № 1	Успех	Исключение № 1	Примечания
			Исключение № 2	
			Исключение № 3	
Цель № 2	Успех	Исключение № 1		
		Исключение № 2		
		Исключение № 3		

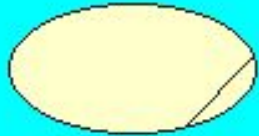
# Показатели качества для модели вариантов использования

- Все ли функциональные требования описываются вариантами использования?
- Не содержит ли модель вариантов использования ненужное поведение, которое отсутствует в требованиях?
- Действительно ли в модели необходимы все выявленные связи включения, расширения и обобщения?
- Правильно ли произведено деление модели на пакеты вариантов использования?
- Стала ли модель в результате деление на пакеты проще и удобнее для восприятия и сопровождения?
- Можно ли на основе модели вариантов использования составить четкое представление о функционировании системы в контексте ее пользователей?

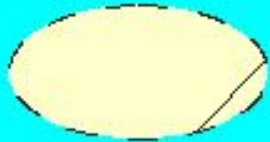
# Последовательность разработки вариантов использования

- Определить главных (первичных) актеров и определить их цели по отношению к системе
- Специфицировать все базовые (основные) варианты использования
- Выделить цели базовых ВИ, интересы актеров в контексте этих ВИ, предусловия и постусловия ВИ
- Написать успешный сценарий выполнения базовых ВИ
- Определить исключения (неуспех) в сценариях ВИ и написать сценарии для всех исключений
- Выделить ВИ исключений и изобразить их со стереотипом «extend»
- Выделить общие фрагменты функциональности ВИ и изобразить их отдельными ВИ со стереотипом «include»

# UML Profile for Business Modeling



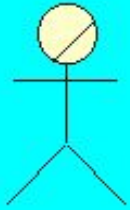
Business use case



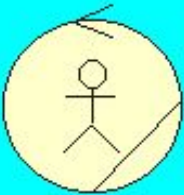
Business use-case realization

- Бизнес вариант использования – элемент модели, предназначенный для представления отдельного бизнес процесса
- Реализация бизнес варианта использования – описывает реализацию отдельного бизнес варианта использования в терминах кооперации объектов, экземпляров сотрудников и бизнес сущностей

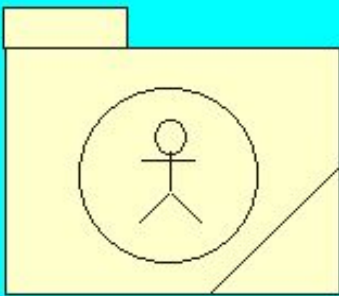
# UML Profile for Business Modeling



Business actor



Business worker



Organization unit

- Бизнес актер – индивидуум, группа, организация, компания или система, которые взаимодействуют с моделируемой системой (компанией), но не входят в нее. Примеры – клиенты, поставщики, партнеры.
- Сотрудник – индивидуум, который действует внутри моделируемой системы (компании), взаимодействует с другими сотрудниками и манипулирует бизнес сущностями.
- Организационная единица – пакет, в состав которого могут входить сотрудники, бизнес сущности, реализации бизнес вариантов использования, диаграммы языка UML и другие организационные единицы

# Типичные ошибки при разработке диаграмм вариантов использования

- Превращение диаграммы вариантов использования в диаграмму деятельности за счет желания отразить все функциональные действия
- Инициатором действий является разрабатываемая система
- Спецификация атрибутов и операций классов до того, как сформулированы все варианты использования
- Задание слишком кратких имен вариантам использования
- Описание вариантов использования в терминологии, непонятной пользователям системы или заказчику
- Отсутствие описаний альтернативных последовательностей действий
- Тратится слишком много времени на решение вопросов о том, какие стереотипы и ассоциации использовать на диаграмме



# Самостоятельное задание №1

- Выполнить текущее тестирование: вопросы 7-11
- На основе заданных сценария №1 и сценария №2 разработать диаграмму вариантов использования для АТМ

## **Сценарий №1 выполнения варианта использования "Снятие наличных по кредитной карточке"**

### Главный раздел

- *Вариант использования:* Снятие наличных по кредитной карточке
- *Актеры:* Клиент Банкомата, Банк
- *Цель:* Получение требуемой суммы наличными
- *Краткое описание:* Клиент использует свою карточку для снятия наличных. 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 ПИН-кода кредитной карточки

# Раздел Типичный ход событий

- 1. Клиент вставляет кредитную карточку в устройство чтения банкомата
- 2. Банкомат передает информацию о кредитной карточке в Банк
- 3. Банк проверяет информацию о кредитной карточке
- *Исключение №1:* Кредитная карточка недействительна (утрачена)
- *Исключение №2:* Кредитная карточка просрочена
- 4. Банкомат предлагает ввести ПИН-код
- 5. Клиент вводит PIN-код
- 6. Банкомат проверяет ПИН-код
- *Исключение №3:* Введенный ПИН-код неверный
- *Исключение №4:* Клиент ввел неверный ПИН-код 3 раза
- 7. Банкомат отображает опции меню
- 8. Клиент выбирает снятие наличных со своего счета
- 9. Банкомат предлагает ввести требуемую сумму

# Раздел Типичный ход событий

- 10. Клиент вводит требуемую сумму
- 11. Банкомат делает соответствующий запрос в Банк
- 12. Банк проверяет введенную сумму
- Исключение №5: Требуемая сумма превышает сумму на счете клиента
- 13. Банк изменяет состояние счета клиента
- 15. Клиент получает свою кредитную карточку
- Исключение №6: Клиент выбрал печать чека
- 14. Банкомат предлагает клиенту забрать его кредитную карточку
- 16. Банкомат выдает наличные и предлагает забрать их клиенту
- 17. Клиент получает наличные
- 18. Банкомат отображает сообщение о готовности к дальнейшей работе

# Раздел Исключений

- Исключение №1. Кредитная карточка недействительна (утрачена)
- 4. Банкомат блокирует кредитную карточку
- 18. Банкомат отображает сообщение о готовности к дальнейшей работе
  
- Исключение №2: Кредитная карточка просрочена
- 4. Банкомат предлагает клиенту забрать его кредитную карточку
- 15. Клиент получает свою кредитную карточку
- 18. Банкомат отображает сообщение о готовности к дальнейшей работе
  
- Исключение №3. Введенный ПИН-код неверный
- 4. Банкомат предлагает ввести ПИН-код
- 5. Клиент вводит ПИН-код

# Сценарий №2 "Получение справки о состоянии счета"

## Главный раздел

- *Вариант использования:* Получение справки о состоянии счета
- *Актеры:* Клиент Банкомата, Банк
- *Цель:* Получение информации о балансе
- *Краткое описание:* Клиент использует свою карточку для получения справки о состоянии счета. Банкомат обеспечивает доступ к счету клиента. Банкомат выдает клиенту справку в форме чека.
- *Тип:* Базовый
- *Ссылки на другие варианты использования:* Включает в себя ВИ:
  - Проверка ПИН-кода кредитной карточки

# Типичный ход событий

- 1. Клиент вставляет кредитную карточку в устройство чтения банкомата
- 2. Банкомат передает информацию о кредитной карточке в Банк
- 3. Банк проверяет информацию о кредитной карточке
- Исключение №1: Кредитная карточка недействительна (утрачена)
- Исключение №2: Кредитная карточка просрочена
- 4. Банкомат предлагает ввести ПИН-код
- 5. Клиент вводит PIN-код
- 6. Банкомат проверяет ПИН-код
- Исключение №3: Введенный ПИН-код неверный
- Исключение №4: Клиент ввел неверный ПИН-код 3 раза

# Типичный ход событий

- 7. Банкомат отображает опции меню
- 8. Клиент выбирает получение справки о состоянии счета
- 9. Банкомат делает соответствующий запрос в Банк
- 10. Банкомат предлагает клиенту забрать его кредитную карточку
- 11. Клиент получает свою кредитную карточку
- 12. Банкомат выдает справку о состоянии счета и предлагает забрать ее клиенту
- 13. Клиент получает справку о состоянии своего счета
- 14. Банкомат отображает сообщение о готовности к дальнейшей работе

# Раздел Исключений

- Исключение №1. Кредитная карточка недействительна (утрачена)
- 4. Банкомат блокирует кредитную карточку
- 14. Банкомат отображает сообщение о готовности к дальнейшей работе
- Исключение №2: Кредитная карточка просрочена
- 4. Банкомат предлагает клиенту забрать его кредитную карточку
- 11. Клиент получает свою кредитную карточку
- 14. Банкомат отображает сообщение о готовности к дальнейшей работе
- Исключение №3. Введенный ПИН-код неверный
- 4. Банкомат предлагает ввести ПИН-код
- 5. Клиент вводит ПИН-код
- Исключение №4: Клиент вводит неверный ПИН-код 3 раза
- 4. Банкомат блокирует кредитную карточку
- 18. Банкомат отображает сообщение о готовности к дальнейшей работе



# Раздел Исключений

- Исключение №4: Клиент вводит неверный ПИН-код 3 раза
- 4. Банкомат блокирует кредитную карточку
- 18. Банкомат отображает сообщение о готовности к дальнейшей работе
- Исключение №5. Требуемая сумма превышает сумму на счете клиента
- 9. Банкомат предлагает ввести новую сумму
- 10. Клиент вводит новую требуемую сумму
- Исключение №6: Клиент выбрал печать чека
- 16.1. Банкомат предлагает клиенту забрать чек
- Примечание. Клиент может отказаться от выполнения транзакции "Снятие наличных по кредитной карточке" при введении ПИН-кода, при выборе типа транзакции и при вводе суммы.