

# Проектирование архитектуры программных систем

Работа с требованиями

Семинар первый

27-31 января 2025

# Цель семинара

- Что такое архитектура ПО?
- Связь архитектуры ПО и бизнес-требований
- Использование бизнес-требований в качестве исходных данных
- Проектирование архитектуры ПО
- Моделирование предметной области
- Моделирование прецедентов
- Проверка согласованности моделей

Что такое архитектура ПО?

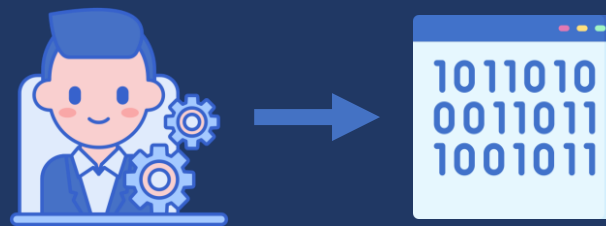
gpt: **Архитектура программного обеспечения (АПО)** — это структурный план или концепция, описывающая, как программная система устроена, как ее компоненты взаимодействуют друг с другом и как достигаются функциональные и нефункциональные требования

От него зависят качество, удобство сопровождения и результат.

Хорошо спроектированная архитектура упрощает разработку и повышает качество системы.

Важность бизнес-требований для  
проектирования архитектуры ПО

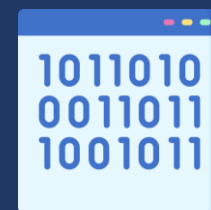


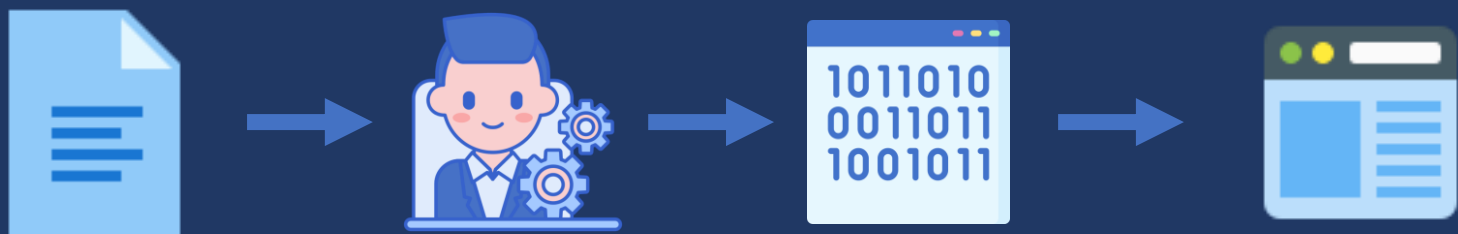


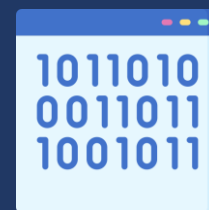


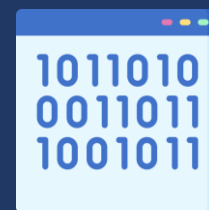


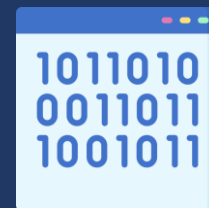


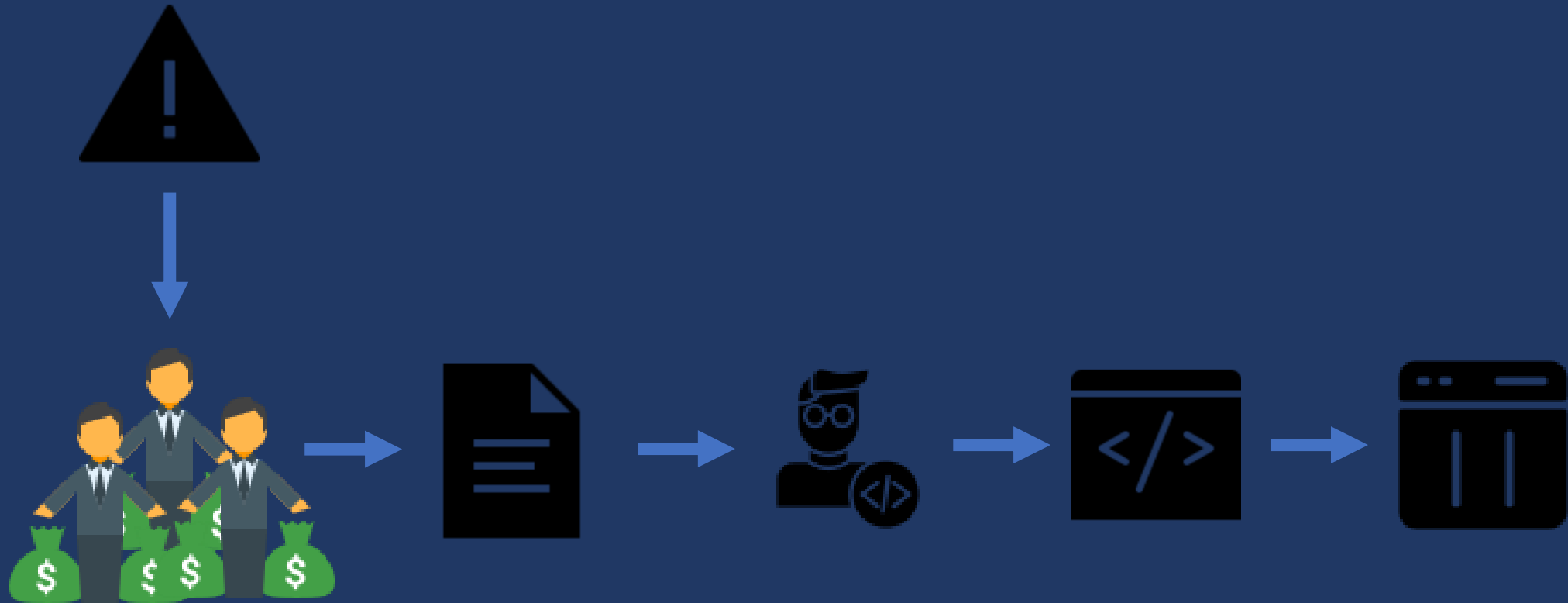




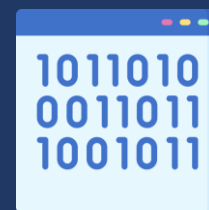


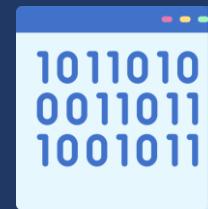




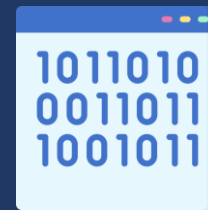


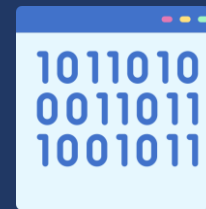


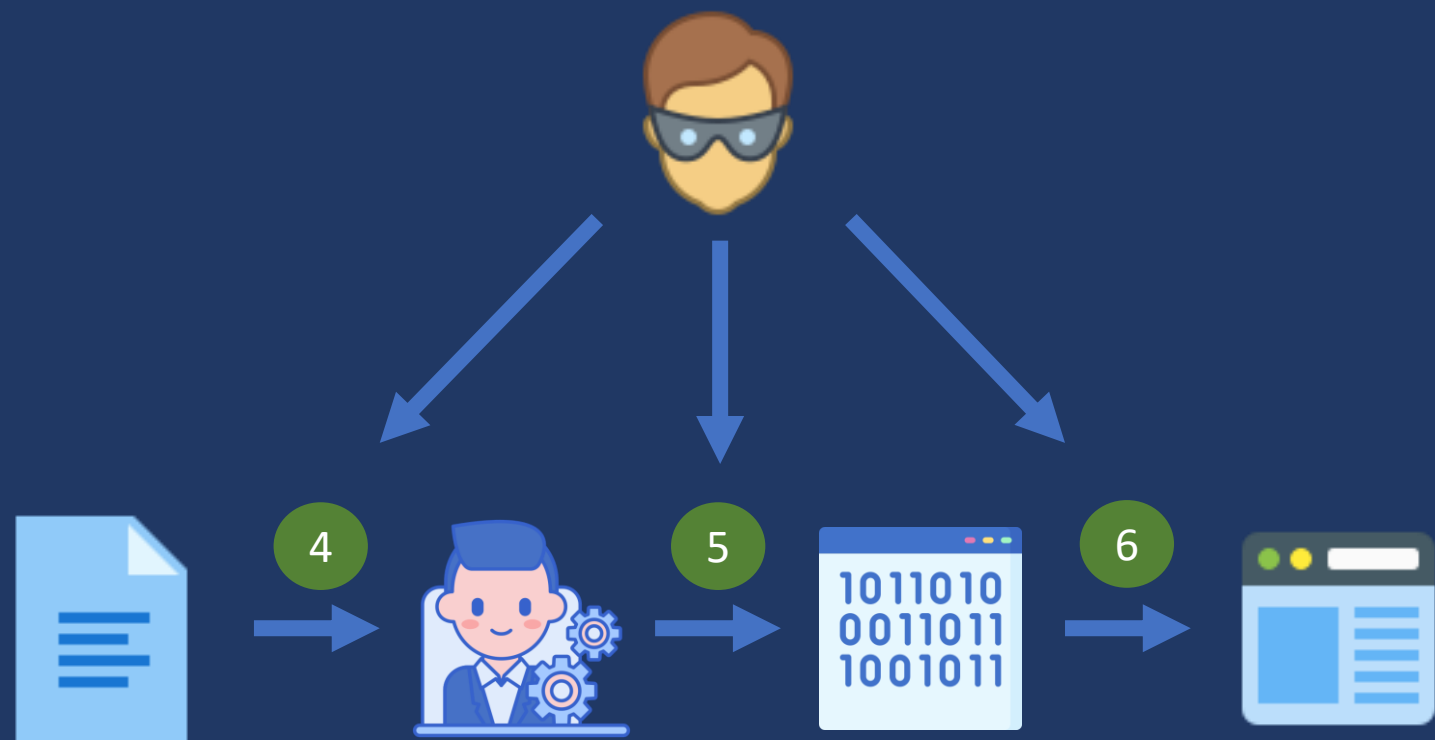




# Связь архитектуры ПО и бизнес-требований



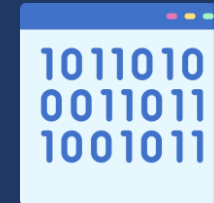




# Использование бизнес-требований в качестве исходных данных



ADR



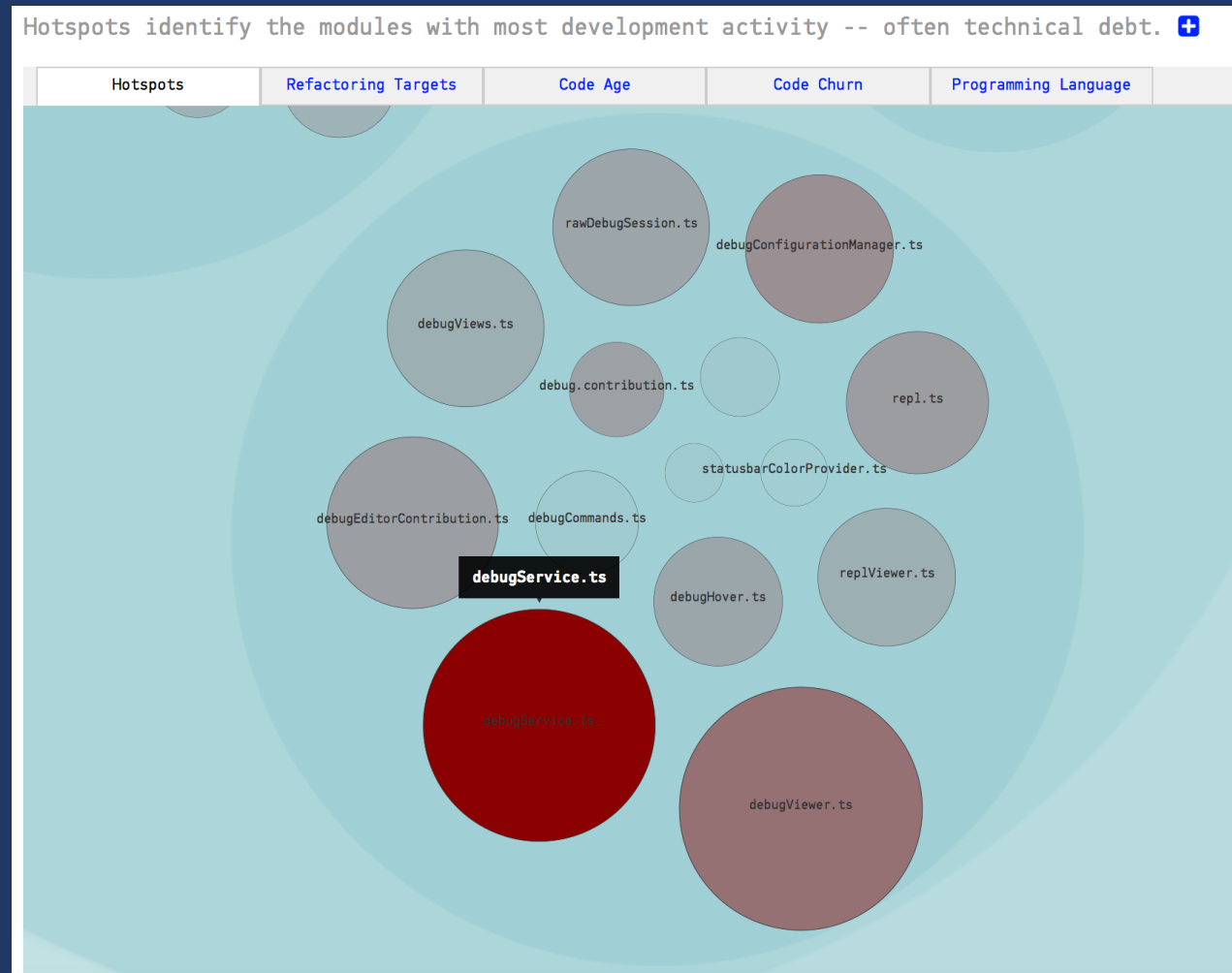


# Проектирование архитектуры ПО

# Точки зрения на архитектуру ПО

- Волатильность (изменения, нагрузка)
- Характер данных
- Технологии
- Организационный подход
- Предметное ориентирование

# Волатильность



# Характер данных



# Технологии

Фронтенд  
команда



Веб-интерфейс  
<Представление>

Бэкенд  
команда



Бэкенд  
<Бизнес-логика>

АБД  
команда



База данных Postgre  
<Данные>

База данных Mongo DB  
<Данные>

# Организационный подход

Антарктическая  
команда



Веб-интерфейс  
<Представление>

Африканская  
команда



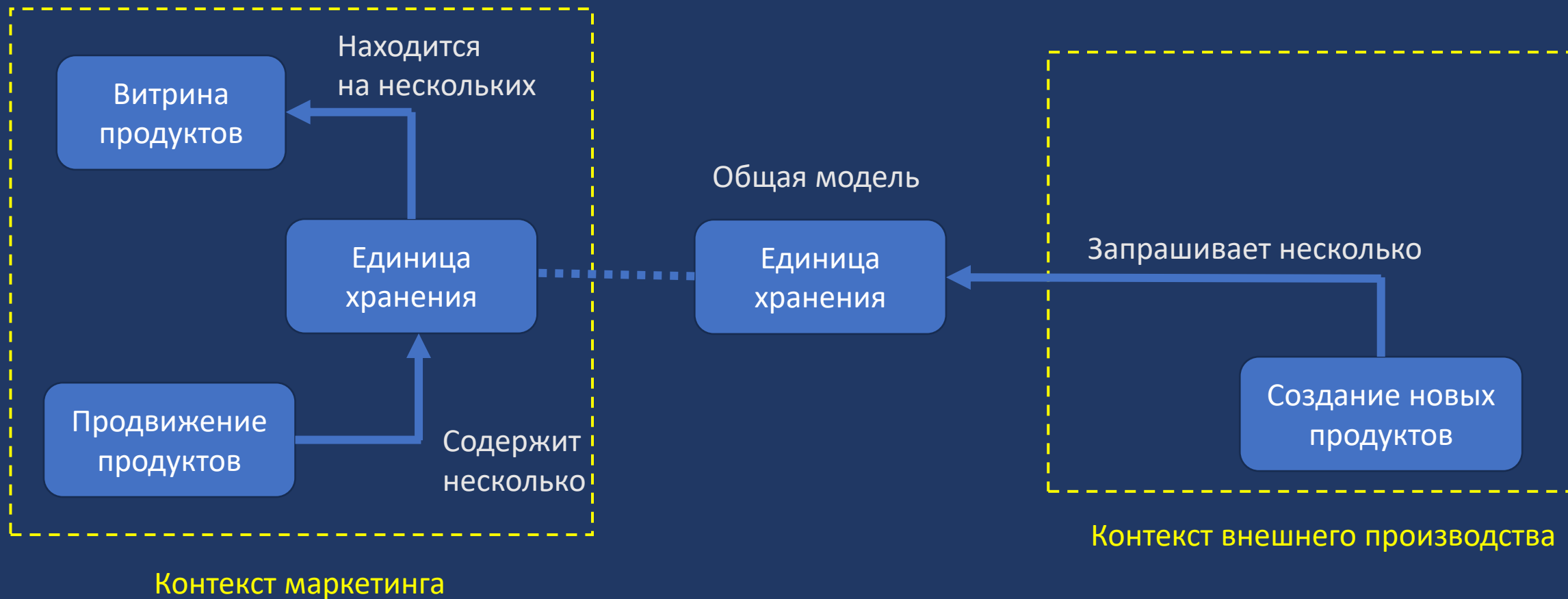
Бэкенд  
<Бизнес-логика>

Арктическая  
Команда



База данных  
<Данные>

# Предметное ориентирование



Архитектура ПО

**должна быть максимально устойчивой**  
с течением времени



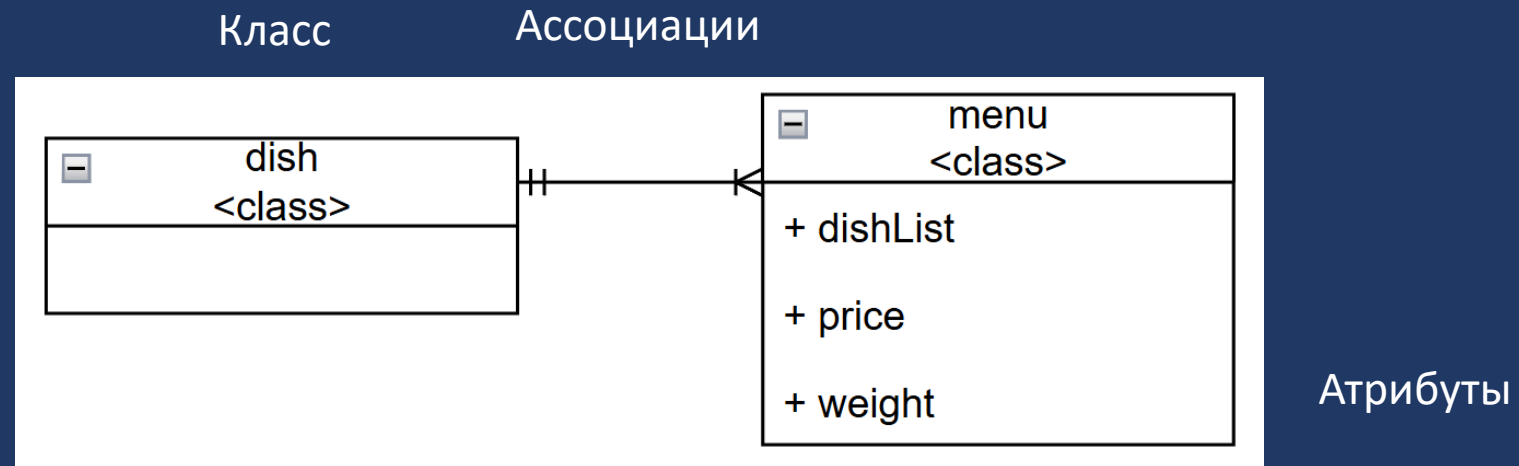
# Моделирование предметной области

**Модель предметной области** (domain model / conceptual object model) – визуальное представление концептуальных классов или объектов реального мира в терминах предметной области.

- Объекты предметной области или концептуальные классы;
- Ассоциации между концептуальными классами;
- Атрибуты концептуальных классов.

# Моделирование предметной области

- **Класс (class)** — описывает группу объектов с одинаковыми свойствами (атрибутами), поведением (операциями), типами отношений и семантикой.
- **Ассоциация (association)** — это структурное отношение между двумя и более классами. Ассоциации показывают отношения между классами на том же уровне абстракции, на котором находятся сами классы.
- **Атрибут (attribute)** — это свойства объектов. Атрибуты, в отличие от ассоциаций, скрывают зависимости и их одностороннюю природу.



# Моделирование предметной области

Этапы проектирования модели предметной области:

1. Выделение классов
2. Подготовка словаря данных
3. Выделение ассоциаций
4. Выделение атрибутов объектов и связей
5. Объединение и организация классов с помощью наследования
6. Переход к следующей итерации: уточнение модели
7. Группировка классов в пакеты

# Задание

Необходимо разработать информационную систему для автоматизации сети небольших кафе и ресторанов.

Основной услугой кафе сети «Пельмешки» является обслуживание клиентов. Клиент формирует, используя бумажное меню, состав своего обеда или ужина как набора блюд и напитков. Официант проводит обслуживание клиента, выписывает чек. Клиент оплачивает чек (наличными или банковской картой) и по своему усмотрению может оставить чаевые официанту.

Каждое кафе самостоятельно проводит закупку продуктов для своих нужд. Согласно рецепту на приготовление каждого блюда расходуется определенный состав продуктов и ингредиентов. За состав блюд и рецептуру в каждом кафе отвечает шеф-повар данного кафе, но он руководствуется общим базовым меню всей сети. Цены блюд во всей сети регулируются централизованно. Но отдельные кафе могут незначительно менять цены в том случае, если рецептура блюда в конкретном кафе отличается. Кафе могут иметь уникальные блюда, отсутствующие во всей сети.

Кафе не обязаны предлагать все блюда, имеющиеся в базовом меню сети.

Внедряемая информационная система должна иметь терминалы в каждом ресторане сети «Пельмешки». Она будет использоваться для учета продуктов в каждом кафе, подсчета потребляемых клиентами блюд, числа обслуженных клиентов, ведения кадровых вопросов сети и т. п. Для руководства сети ключевое значение имеет централизованный учет движения денежных средств, включая возможность расчета средней стоимости чека как характеристики конкретного кафе.

Система должна иметь возможность формировать различные отчеты, включая расчёт среднего чека по отдельному кафе и сети в целом за заданный период времени, отчёт с оценкой занятости персонала в кафе, а также отчёты по доходам и расходам как по отдельным кафе, так и по сети в целом по заданным периодам времени.

# Модель прецедентов

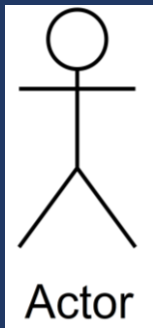
**Модель прецедентов** (use-case diagram/model) – это тип поведенческой диаграммы UML, который позволяют визуализировать различные типы ролей в системе и то, как эти роли с ней взаимодействуют.

Этапы проектирования модели:

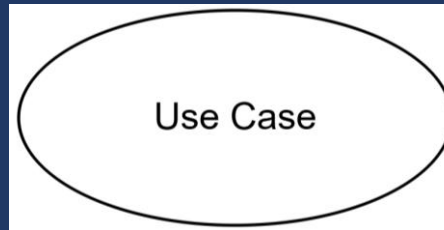
- Определить границы проектируемого ПО
- Выделить роли пользователей проектируемого ПО
- Сопоставить роли с ключевыми функциями, описанных в требованиях к ПО
- Составление модели прецедентов

# Модель прецедентов

- **Пользователь/действующее лицо** (actor) – это объект/субъект или их множество, непосредственно взаимодействующих с проектируемым ПО, но не являющиеся его частью
- **Прецедент/Вариант использования** (use-case) – это связный элемент функциональности, предоставляемый проектируемым ПО при взаимодействии с действующими лицами



Пользователь  
проектируемого ПО



Вариант  
использования

# Модель прецедентов

- **Отношение ассоциации** (association relationship) – предназначен только для соединения действующих лиц и прецедентов с целью отображения информации о том, какие прецеденты могут быть использованы каждым действующим лицом
- **Отношение обобщения** (generalization relationship) – обобщение одного действующего лица / прецедента к другому (от частного случая к общему).
- **Отношение включения** (include relationship) – указывает на то, что некоторое заданное поведение для одного прецедента включает в качестве составного компонента поведение другого прецедента
- **Отношение расширения** (extend relationship) - определяет взаимосвязь экземпляров отдельного прецедента с более общим прецедентом, свойства которого определяются на основе способа совместного объединения данных экземпляров.

association relationship

Отношение  
ассоциации

"include" →

Отношение  
включения

generalization relationship →

Отношение  
обобщения

"extend" →

Отношение  
расширения

# Задание

Необходимо разработать информационную систему для автоматизации сети небольших кафе и ресторанов.

Основной услугой кафе сети «Пельмешки» является обслуживание клиентов. Клиент формирует, используя бумажное меню, состав своего обеда или ужина как набора блюд и напитков. Официант проводит обслуживание клиента, выписывает чек. Клиент оплачивает чек (наличными или банковской картой) и по своему усмотрению может оставить чаевые официанту.

Каждое кафе самостоятельно проводит закупку продуктов для своих нужд. Согласно рецепту на приготовление каждого блюда расходуется определенный состав продуктов и ингредиентов. За состав блюд и рецептуру в каждом кафе отвечает шеф-повар данного кафе, но он руководствуется общим базовым меню всей сети. Цены блюд во всей сети регулируются централизованно. Но отдельные кафе могут незначительно менять цены в том случае, если рецептура блюда в конкретном кафе отличается. Кафе могут иметь уникальные блюда, отсутствующие во всей сети.

Кафе не обязаны предлагать все блюда, имеющиеся в базовом меню сети.

Внедряемая информационная система должна иметь терминалы в каждом ресторане сети «Пельмешки». Она будет использоваться для учета продуктов в каждом кафе, подсчета потребляемых клиентами блюд, числа обслуженных клиентов, ведения кадровых вопросов сети и т. п. Для руководства сети ключевое значение имеет централизованный учет движения денежных средств, включая возможность расчета средней стоимости чека как характеристики конкретного кафе.

Система должна иметь возможность формировать различные отчеты, включая расчёт среднего чека по отдельному кафе и сети в целом за заданный период времени, отчёт с оценкой занятости персонала в кафе, а также отчёты по доходам и расходам как по отдельным кафе, так и по сети в целом по заданным периодам времени.



# Задание

Согласовать модель предметной области и модель прецедентов

- Избыточное/недостаточное число акторов
- Избыточное/недостаточное количество функций
- Согласованность функций и предметных областей
- Наличие фрагмента функций в смежной предметной области

# Итоги семинара

- Узнали, что такое архитектура ПО
- Определили целесообразность связи архитектуры ПО и бизнес-требований
- На практике использовали бизнес-требования в качестве исходных данных
- Приступили к проектированию архитектуры ПО
- Сформировали предметную область
- Смоделировали прецеденты
- Выполнили проверку на согласованность моделей

# Далее в программе

- Спецификация прецедента: что это и с чем его едят
- 4С-моделирование: матрица глазами архитектора ПО

# Обратная связь

