

Анализ и проектирование на UML

Максим Валерьевич Хлопотов

Темы лекционных занятий

- 1. Введение в UML**
2. Моделирование использования
3. Моделирование структуры
4. Моделирование поведения
5. Дисциплина моделирования
6. Примеры моделей на UML

АКТИВНОСТЬ

1. Староста группы собирает e-mail'ы, отправляет мне.
2. Я рассылаю приглашение на Piazza
3. Нужно зарегистрироваться на Piazza и создать свою тему в разделе Q&A.

Назначение UML

UML — это графический язык моделирования общего назначения, предназначенный для спецификации, визуализации, проектирования и документирования всех артефактов, создаваемых при разработке программных систем.

Модель UML

Модель UML — это конечное множество сущностей и отношений между ними.

Рассматривая модель UML с наиболее общих позиций, можно сказать, что это граф (точнее, нагруженный мульти-псевдо-гипер-орграф), в котором вершины и ребра нагружены дополнительной информацией и могут иметь сложную внутреннюю структуру. Вершины этого графа называются *сущностями*, а ребра — *отношениями*.

Диаграммы UML

Диаграммы UML – основная накладываемая на модель структура, которая облегчает создание и использование модели.

Диаграмма — это графическое представление некоторой части графа модели.

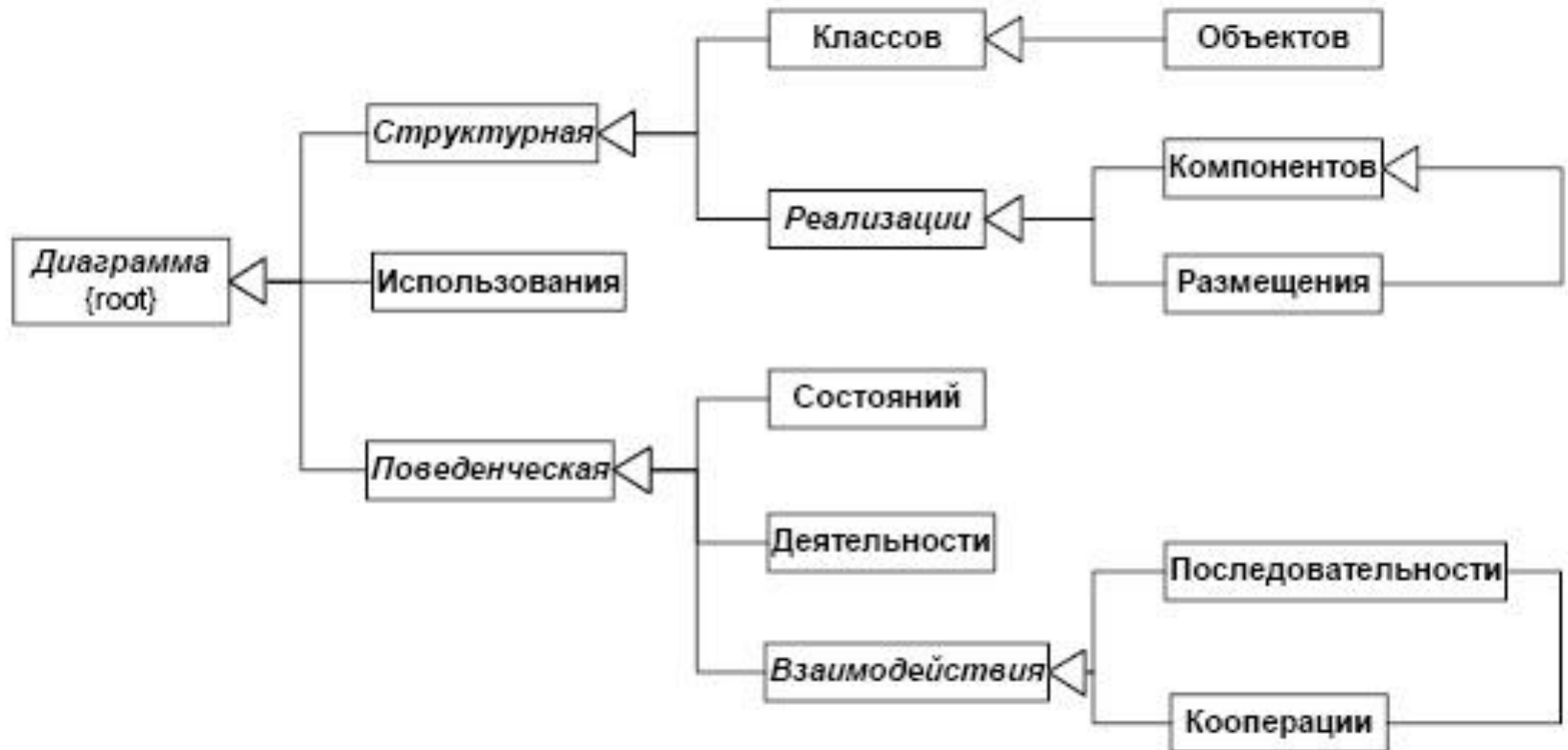
Авторы UML определили набор рекомендуемых к использованию типов диаграмм, которые получили название *канонических* типов диаграмм.

Диаграммы UML

В UML 1.x всего определено 9 канонических типов диаграмм.

- Диаграмма использования
- Диаграмма классов
- Диаграмма объектов
- Диаграмма состояний
- Диаграмма деятельности
- Диаграмма последовательности
- Диаграмма кооперации
- Диаграмма компонентов
- Диаграмма размещения

Иерархия диаграмм UML



Диаграммы UML

Диаграмма использования — это наиболее общее представление функционального назначения системы. Диаграмма использования призвана ответить на главный вопрос моделирования: что делает система во внешнем мире?

Пример

действующее лицо (эктор), ассоциация, вариант использования (прецедент), рамки системы



Диаграммы UML

Диаграмма классов — основной способ описания структуры системы. Это не удивительно, поскольку UML сильно объектно-ориентированный язык, и классы являются основным "строительным материалом" системы.

Пример

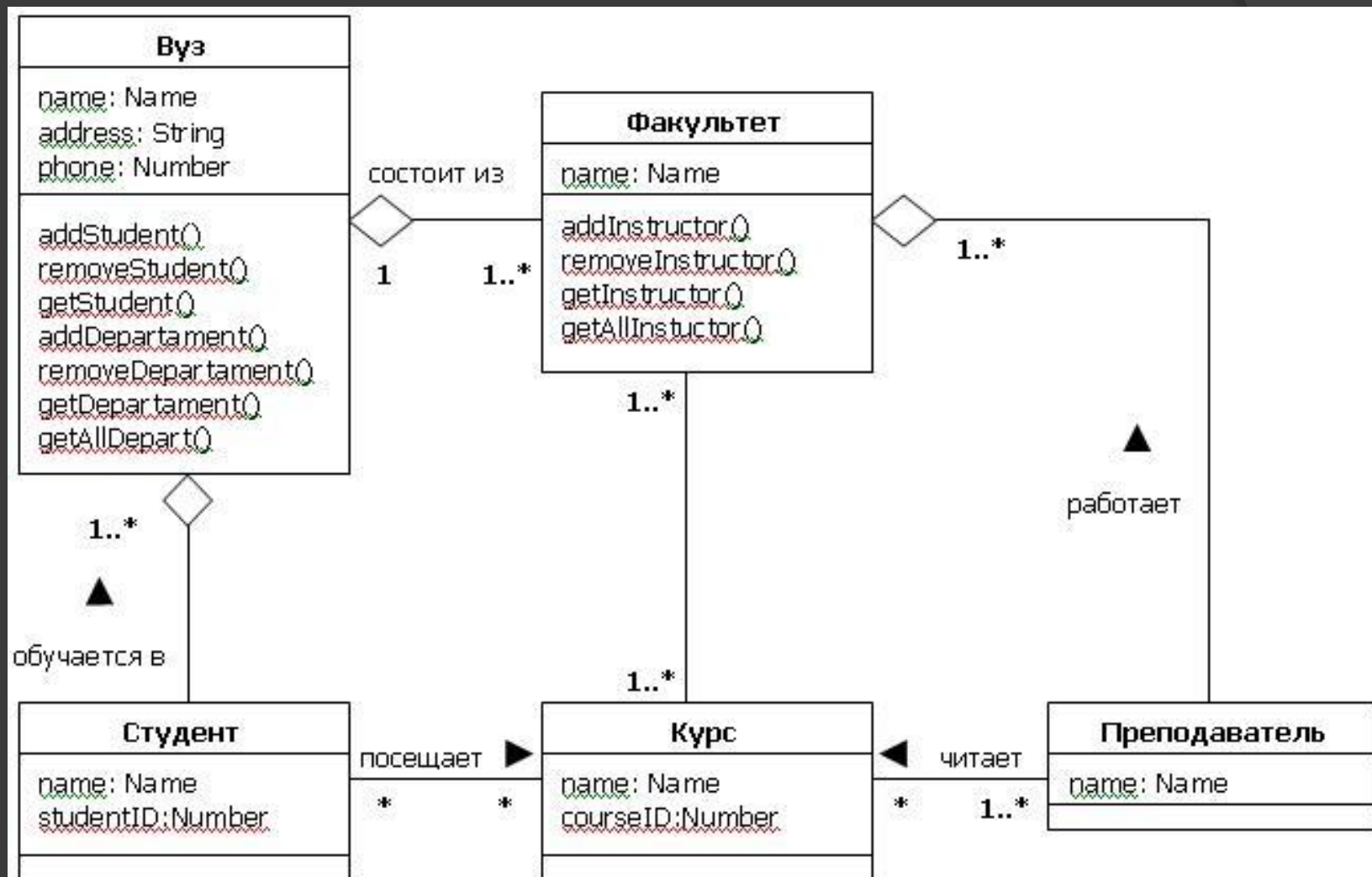


РИС 3.9

Диаграммы UML

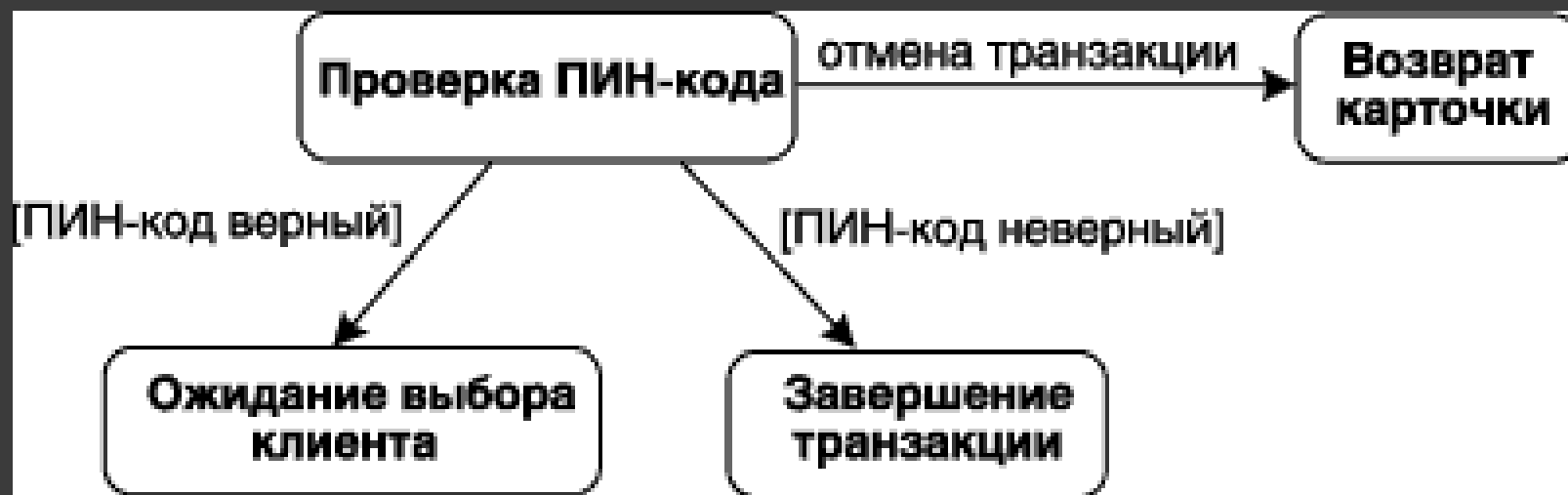
Диаграмма объектов — это частный случай диаграммы классов. Диаграммы объектов имеют вспомогательный характер — по сути это примеры, показывающие, какие имеются объекты и связи между ними в некоторый конкретный момент функционирования системы.

Диаграммы UML

Диаграмма состояний — это основной способ детального описания поведения в UML. В сущности, диаграммы состояний представляют собой граф состояний и переходов конечного автомата, нагруженный множеством дополнительных деталей и подробностей.

Пример

(изменение состояний банкомата при проверке ПИН-кода)



Диаграммы UML

Диаграмма деятельности — это, фактически, блок-схема алгоритма, в которой модернизированы обозначения, а семантика согласована с современным объектно-ориентированным подходом.

Пример

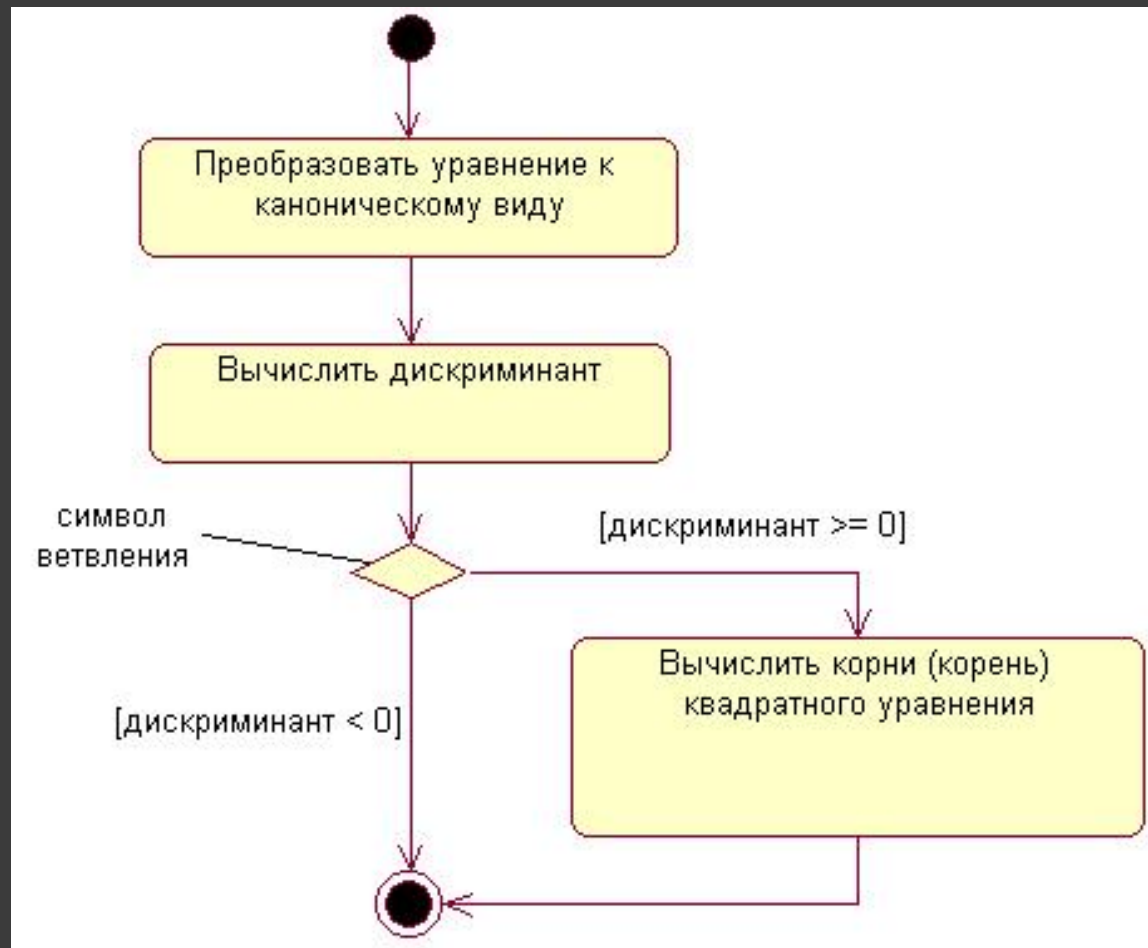
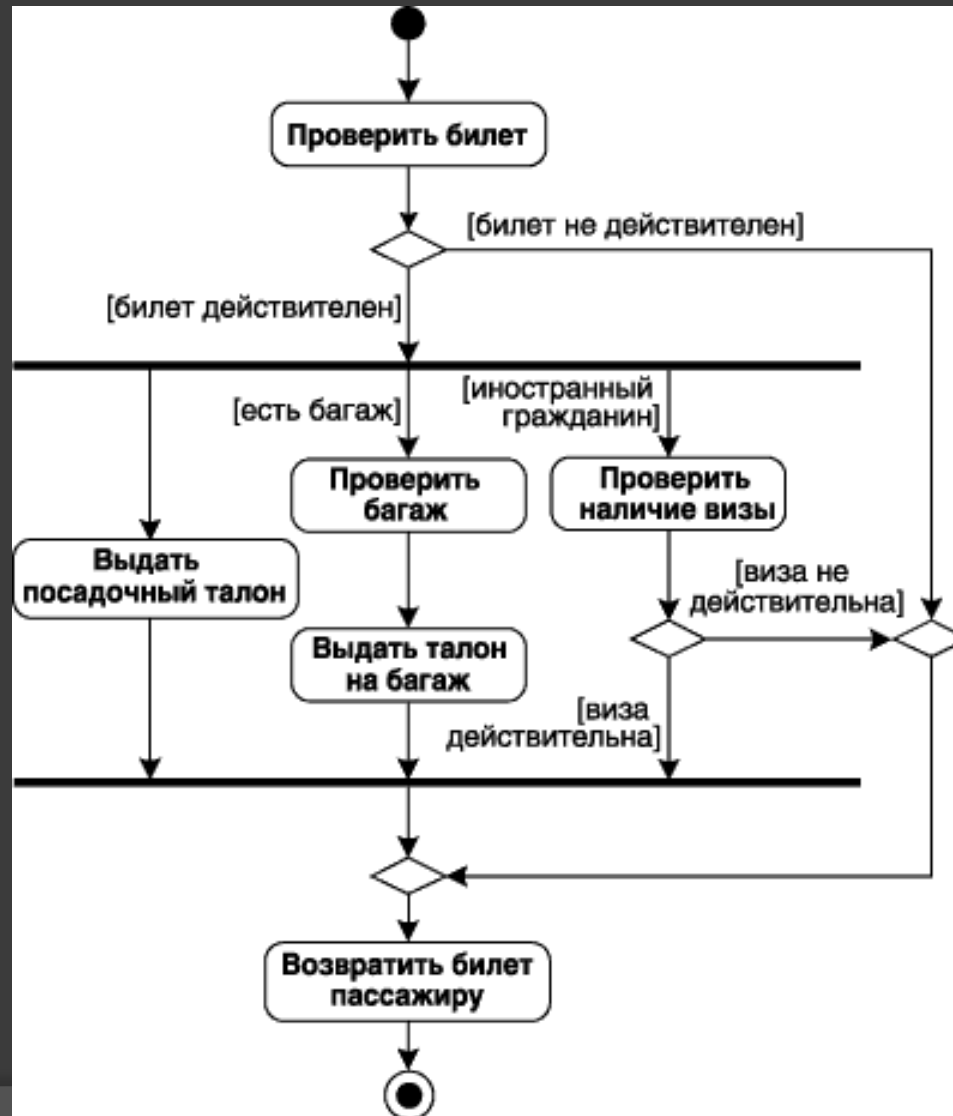


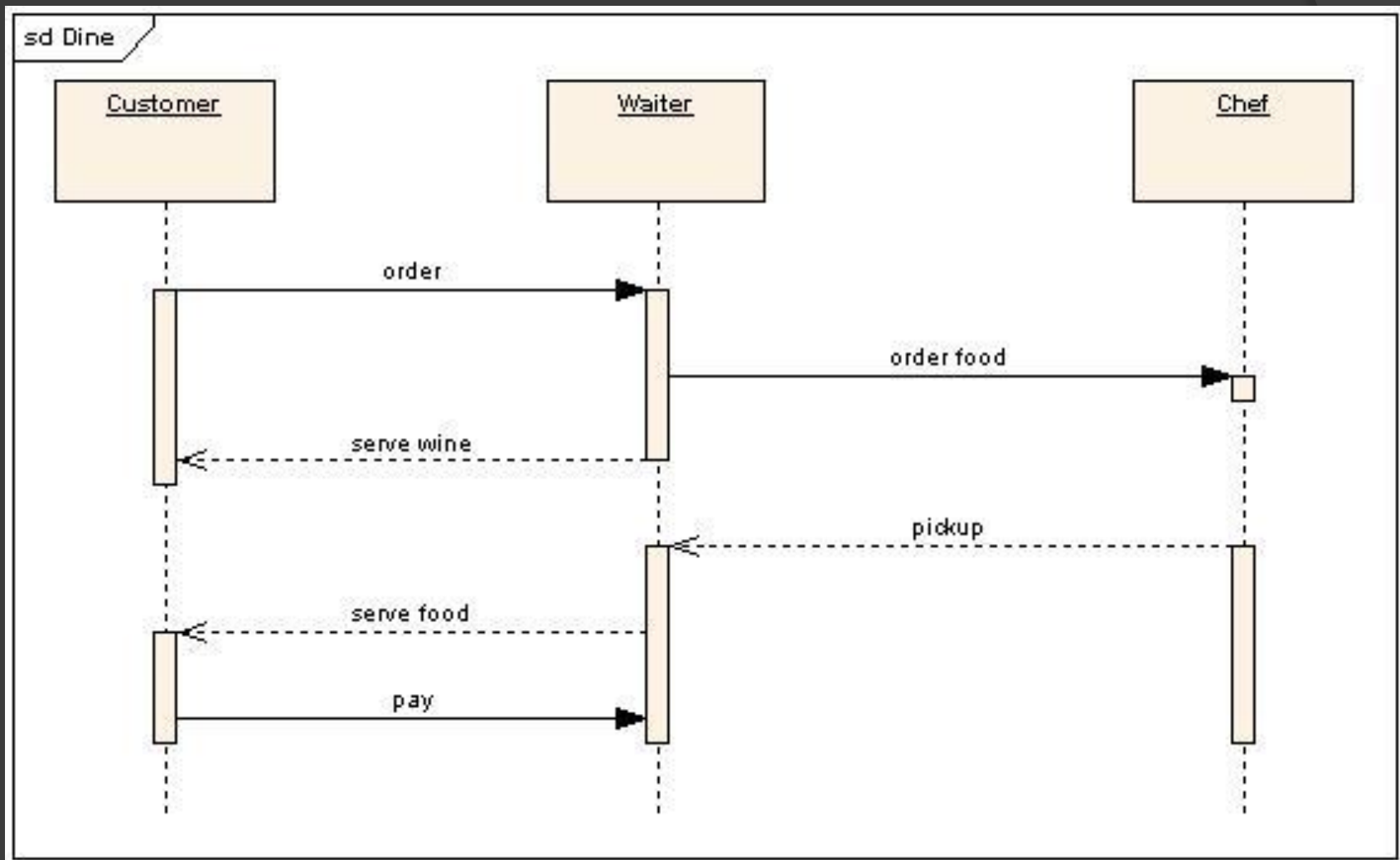
Диаграмма деятельности (пример)



Диаграммы UML

Диаграмма последовательности — это способ описать поведение системы "на примерах". Фактически, диаграмма последовательности — это запись протокола конкретного сеанса работы системы (или фрагмента такого протокола). В объектно-ориентированном программировании самым существенным во время выполнения является посылка сообщений взаимодействующими объектами.

Пример



Диаграммы UML

Диаграмма кооперации (в UML 2 – диаграмма коммуникации) семантически эквивалентна диаграмме последовательности.

Фактически, это такое же описание последовательности обмена сообщениями взаимодействующих объектов, только выраженное другими графическими средствами.

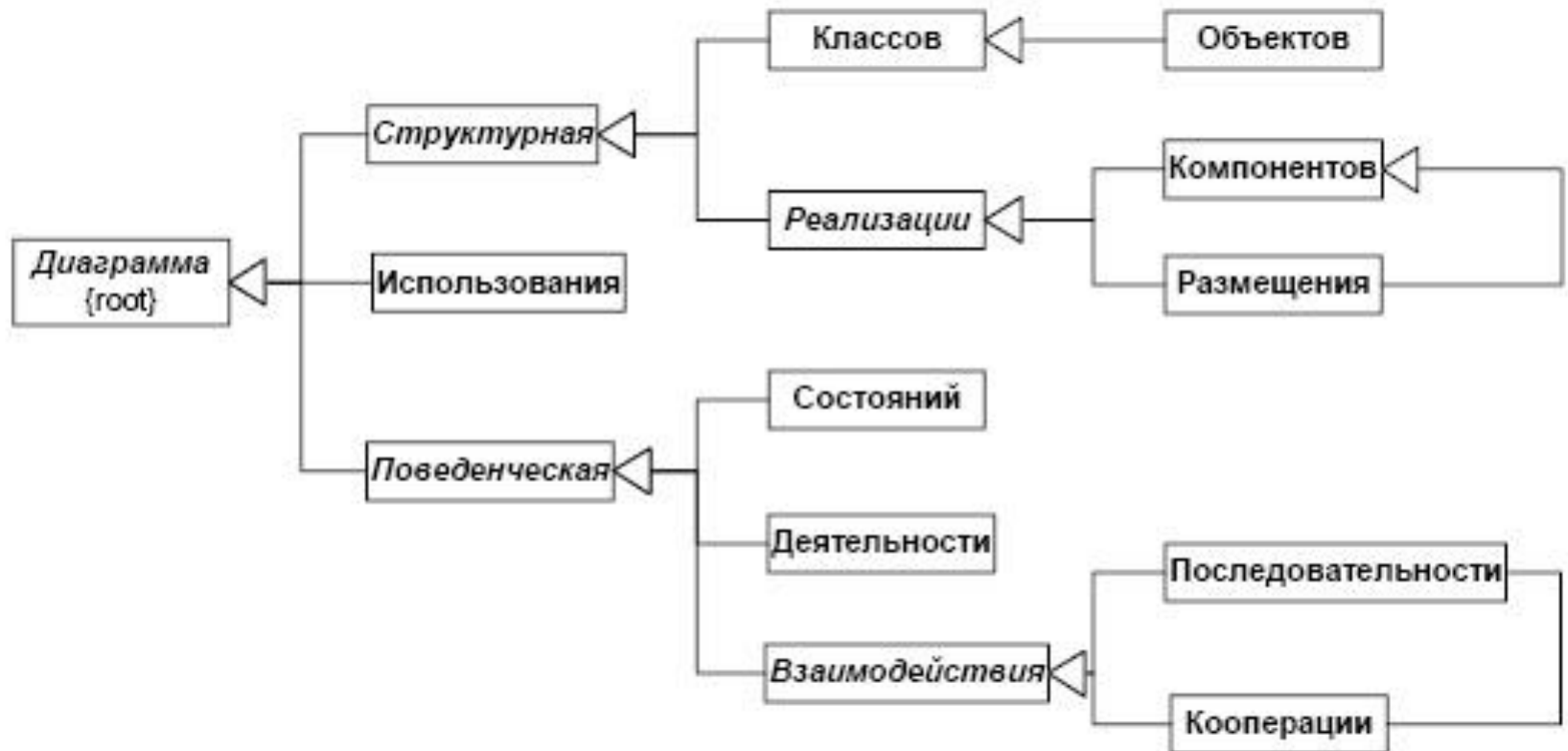
Диаграммы UML

Диаграмма компонентов — это, фактически, список артефактов, из которых состоит моделируемая система, с указанием некоторых отношений между артефактами. Наиболее существенным типом артефактов программных систем являются программы. Таким образом, на диаграмме компонентов основной тип сущностей — это компоненты (как исполнимые модули, так и другие артефакты), а также интерфейсы (чтобы указывать взаимосвязь между компонентами) и объекты (входящие в состав компонентов).

Диаграммы UML

Диаграмма размещения (диаграмма развёртывания) немногим отличается от диаграммы компонентов. Фактически, наряду с отображением состава и связей компонентов здесь показывается, как физически размещены компоненты на вычислительных ресурсах во время выполнения.

Иерархия диаграмм UML



Представления

Все аспекты моделируемой системы не удастся описать с единой точки зрения.

Моделировать сложную систему следует с нескольких различных точек зрения, каждый раз принимая во внимание один аспект моделируемой системы и абстрагируясь от остальных.

Этот тезис является одним из основополагающих принципов UML.

Представления

Выделим три представления:

- представление использования (что делает система полезного?);
- представление структуры (из чего состоит система?);
- представление поведения (как работает система?).

Представления

Выделим три представления:

- представление использования;
- представление структуры;
- представление поведения.

Представления

Представление использования призвано отвечать на вопрос, что делает система полезного.

Определяющим признаком для отнесения элементов модели к представлению использования является, по нашему мнению, явное сосредоточение внимание на факте наличия у системы внешних границ, то есть выделение внешних действующих лиц, взаимодействующих с системой, и внутренних вариантов использования, описывающих различные сценарии такого взаимодействия.

Описывается диаграммой использования.

Представления

Представление структуры призвано отвечать на вопрос: из чего состоит система.

Определяющим признаком для отнесения элементов модели к представлению структуры является явное выделение структурных элементов — составных частей системы — и описания взаимосвязей между ними. Принципиальным является чисто статический характер описания, то есть отсутствие понятия времени в любой форме, в частности, в форме последовательности событий и/или действий.

Описывается диаграммами классов, а также, если нужно, диаграммами компонентов и размещения и, в редких случаях, диаграммами объектов.

Представления

Представление поведения призвано отвечать на вопрос: как работает система.

Определяющим признаком для отнесения элементов модели к представлению поведения является явное использования понятия времени, в частности, в форме описания последовательности событий/действий, то есть в форме алгоритма.

Описывается диаграммами состояний и деятельности, а также диаграммами взаимодействия в форме диаграмм кооперации и/или последовательности.

Примерные темы

Список тем проектов для выбора:

1. Программное обеспечение банкомата.

Обзор: банкомат по карте позволяет снимать наличные со счета по и/или печатать справку об остатке на счете.

2. Информационная система библиотеки.

Обзор: информационная система библиотеки позволяет искать книги в своем каталоге, учитывать выдачу книг на руки и возврат книг, а также позволяет добавлять книги в фонд и списывать их.

3. Информационная система поликлиники.

Обзор: информационная система поликлиники позволяет ставить и снимать больных с учета, записывать больных на прием к врачам, учитывать факт приема, а также позволяет вести историю болезни (медицинскую карту) больного.

Примерные темы

4. Информационная система деканата.

Обзор: информационная система деканата позволяет принимать и отчислять студентов, вести учет успеваемости по итогам сессии, переводить студентов из группы в группу и с курса на курс.

5. Информационная система склада.

Обзор: информационная система склада позволяет учитывать поступление и уход товаров со склада, а также определять место хранения товаров на складе.

6. Система учета рабочего времени.

Обзор: Система учета рабочего времени позволяет руководителям выдавать задания и отслеживать ход их выполнения, а исполнителям – вести учет рабочего времени, затраченного на выполнение каждого задания.

Примерные темы

7. Информационная система жилищного агентства.

Обзор: информационная система жилищного агентства позволяет квартиросъемщикам подобрать и снять жилье, а владельцам жилья – предложить и сдать.

8. Информационная система технической экспертизы.

Обзор: информационная система технической экспертизы позволяет соискателям грантов подавать заявки, независимым экспертам оценивать заявки, а держателям фонда принимать решение о выдаче грантов по результатам экспертизы заявок.

9. Система продажи билетов на футбол.

Обзор: система продажи билетов позволяет покупать и сдавать билеты и абонементы на матчи, проходящие на одном стадионе с нумерованными местами через несколько одновременно работающих касс.

Этапы выполнения практических работ

Подготовительный этап. Выбор инструментов

1 этап. Анализ предметной области

2 этап. Моделирование использования (лабораторная работа 1)

3 этап. Моделирование структуры (лабораторная работа 2)

4 этап. Моделирование поведения (лабораторная работа 3)

5 этап. Реализация прототипа

6 этап. Приёмо-сдаточные испытания (лабораторная работа 4)

Этапы выполнения практических работ

Подготовительный этап. Выбор инструментов

1. Выбрать инструмент моделирования (инструмент должен быть доступен).
2. Выбрать инструмент разработки (инструмент должен быть доступен и знаком).
3. Выбрать инструмент подготовки презентаций и документации (инструмент должен быть доступен)
4. Проверить совместимость инструментов (необходимо проверить возможность экспорта диаграмм из инструмента моделирования в инструмент подготовки презентаций, совместимость инструментов моделирования и разработки).

Этапы выполнения практических работ

1 этап. Моделирование использования

5. Провести обзор и анализ выбранной предметной области.

6. Дать характеристику объекту автоматизации, подготовить обзор аналогов (текстовый документ)

7. Определить основные действующие лица, определить основные функциональные требования, составить диаграмму вариантов использования.

Этапы выполнения практических работ

1 этап. Моделирование использования

8. Определить, какие из вариантов использования (не менее трех) будут уточняться при последующем моделировании

9. Определить нефункциональные и специальные требования, если они необходимы, и объединить все требования в единый документ (текстовый документ с диаграммами использования, защищаемый артефакт).

10. Для выбранных вариантов использования составить диаграммы деятельности (защищаемый артефакт).

Этапы выполнения практических работ

2 этап. Моделирование структуры

11. Составить словарь предметной области.
12. Составить «одностраничное» описание проекта.
13. Провести идентификацию классов.
14. Определить связи между классами (обобщение, ассоциации).

Этапы выполнения практических работ

2 этап. Моделирование структуры

15. Доработать диаграмму классов. Идентифицировать классы на основе технического задания, словаря предметной области и реализованных вариантов использования. Спроектировать схему хранимых данных диаграммы классов.

16. Составить сводную диаграмму (или несколько диаграмм) классов, на которой должны быть отражены все классы и интерфейсы, задействованные на других диаграммах.

17. Выделить компоненты системы и определить их интерфейсы.

18. Составить диаграмму компонентов и диаграмму размещения (по выбору), описывающую структуру системы в целом.

Этапы выполнения практических работ

3 этап. Моделирование поведения

19. Выделить класс или классы, поведение экземпляров которых зависит от истории.

20. Составить диаграмму (или диаграммы) состояний, описывающую поведение выбранных классов.

21. Реализовать выбранные варианты использования диаграммами последовательности

22. Реализовать выбранные варианты использования диаграммами кооперации (коммуникации).

Этапы выполнения практических работ

4 этап. Реализация прототипа

23. Разработать документацию программной системы.

24. Разработать и отладить код программы на выбранном инструменте разработки.

25. Разработать план тестирования программы с определением значений параметров (качественных характеристик системы).

26. Разработать графический интерфейс пользователя в виде экранных форм.

Этапы выполнения практических работ

5 этап. Приемо-сдаточные испытания

27. Проверить согласованность и корректность всех диаграмм.

28. Составить детальный проект архитектуры, согласованный с техническим заданием, диаграммы использования, диаграммы деятельности, диаграммы последовательности, диаграммы коммуникации, диаграммы состояний, диаграммы компонентов или размещения (защищаемый артефакт).

29. Определить план презентации для представления результатов разработки.

30. Провести презентацию продолжительностью 10 минут, представив основные результаты выполненной разработки (защищаемый артефакт).

Этапы выполнения практических работ

Основываясь на рекомендуемых учебных материалах и путем анализа выбранной предметной области, поэтапно разрабатываются модель приложения, прототип программного обеспечения и программная документация для выбранной предметной области. Результаты работы по каждому этапу оформляются в виде указанных выше артефактов и защищаются группой на очной встрече с преподавателями.

Выводы

- Знание UML является необходимым, но не является достаточным условием построения разумных моделей программных систем.
- UML имеет синтаксис, семантику и прагматику, которые нужно знать и использовать с учетом особенностей реальной задачи и инструмента.
- Модель UML состоит из описания сущностей и отношений между ними.
- Элементы модели группируются в диаграммы и представления для наилучшего описания моделируемой системы с различных точек зрения.

Выводы

- Модель UML состоит из описания сущностей и отношений между ними.
- Диаграмма — это графическое представление некоторой части графа модели.
- Для удобства обзора сущности в UML можно подразделить на четыре группы: *структурные; поведенческие; группирующие; аннотационные.*
- В UML используются четыре основных типа отношений: *зависимость; ассоциация; обобщение; реализация.*
- Элементы модели группируются в диаграммы и представления для наилучшего описания моделируемой системы с различных точек зрения.