

Разработка и анализ требований проектирования ПО

Средства анализа требований к ПО.
Основы UML

Лектор: Петрова О.Б.

СПбГУТ
2017

Стандарты языка UML

- Язык UML – Unified Modeling Language
- Авторы: Гради Буч, Джеймс Румбо, Айвар Якобсон
- UML 2.4.1 (2011) принят в качестве международного стандарта ISO/IEC 19505-1, 19505-2:2012.
- Спецификации UML:
 - 1.3 (1999 год)
 - 1.4 (2001 год)
 - 1.5 (2003 год)
 - 2.0 (2005 год)
 - 2.5 (март 2015 года)

Официальный сайт: www.omg.org

Назначение и элементы UML

- Используется для анализа, проектирования, документирования ПО.
- Модели UML могут быть переведены на языки программирования (C++, Java, Visual Basic, и др.).
- Элементы UML:
 - диаграммы,
 - предметы (структурные, предметы поведения),
 - отношения.

Диаграммы UML (UML 2.5)

Структурные диаграммы

Диаграмма классов.

Диаграмма компонентов.

Диаграмма объектов.

Диаграмма пакетов.

Диаграмма профилей.

Диаграмма композитной структуры.

Диаграмма развертывания.

Диаграммы поведения

Диаграмма прецедентов.

Диаграмма деятельности.

Диаграмма состояний.

Диаграммы взаимодействия:

Диаграмма последовательности.

Диаграмма коммуникаций.

Диаграмма обзора взаимодействия.

Диаграмма синхронизации.

Структурные диаграммы (Structure Diagrams)

Class diagram — диаграмма классов

Component diagram — диаграмма компонентов

Composite structure diagram — диаграмма
композиционной/составной структуры

Collaboration (UML2.0) — диаграмма кооперации

Deployment diagram — диаграмма развертывания

Object diagram — диаграмма объектов

Package diagram — диаграмма пакетов

Profile diagram (UML2.2) - диаграмма профилей

Диаграммы поведения (Behavior Diagrams)

Activity diagram — диаграмма деятельности

State Machine diagram диаграмма состояний

Use case diagram — диаграмма прецедентов

Interaction Diagrams (диаграммы взаимодействия):

Communication diagram (UML2.0) / Collaboration (UML1.x) —
диаграмма коммуникации (UML2.0) / диаграмма кооперации
(UML1.x)

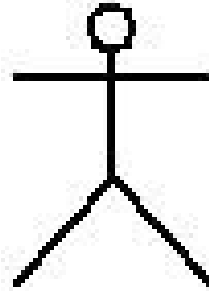
Interaction overview diagram (UML2.0) - диаграмма обзора
взаимодействия

Sequence diagram — диаграмма последовательности

Timing diagram (UML2.0) - диаграмма синхронизации (UML2.0)

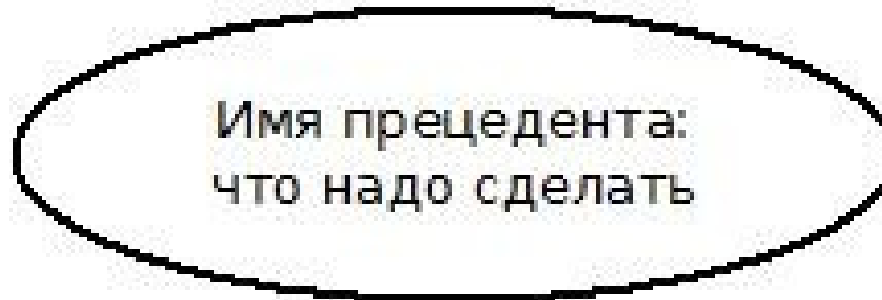
Структурные предметы UML (2)

Актер



Имя действующего лица

Прецедент
(Use Case)



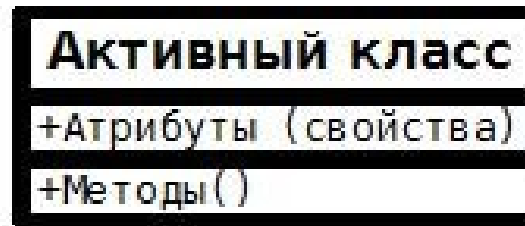
Интерфейс



Интерфейс

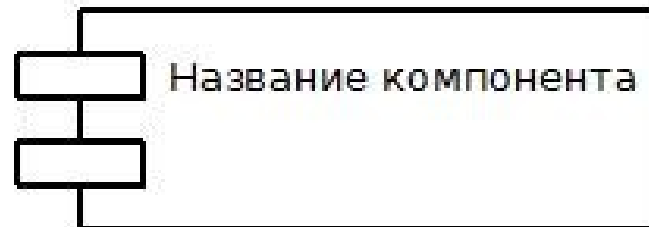
Структурные предметы UML (3)

Активный класс

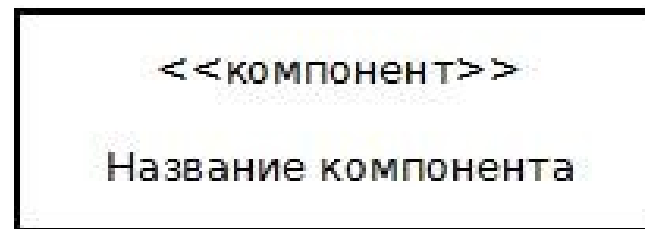


Компонент

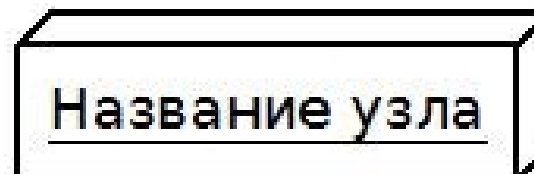
UML 1.2



UML 2.0

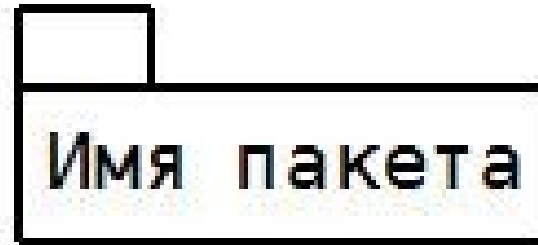


Узел



Структурные предметы UML (4)

Пакеты



Комментарии
(аннотации)



Предметы поведения UML

Взаимодействие



Конечный автомат



Вид деятельности



Отношения UML

Зависимость



Ассоциация



Агрегация



Композиция



Обобщение



Реализация



Диаграмма Use Case

Между актером и прецедентом – ассоциативная связь

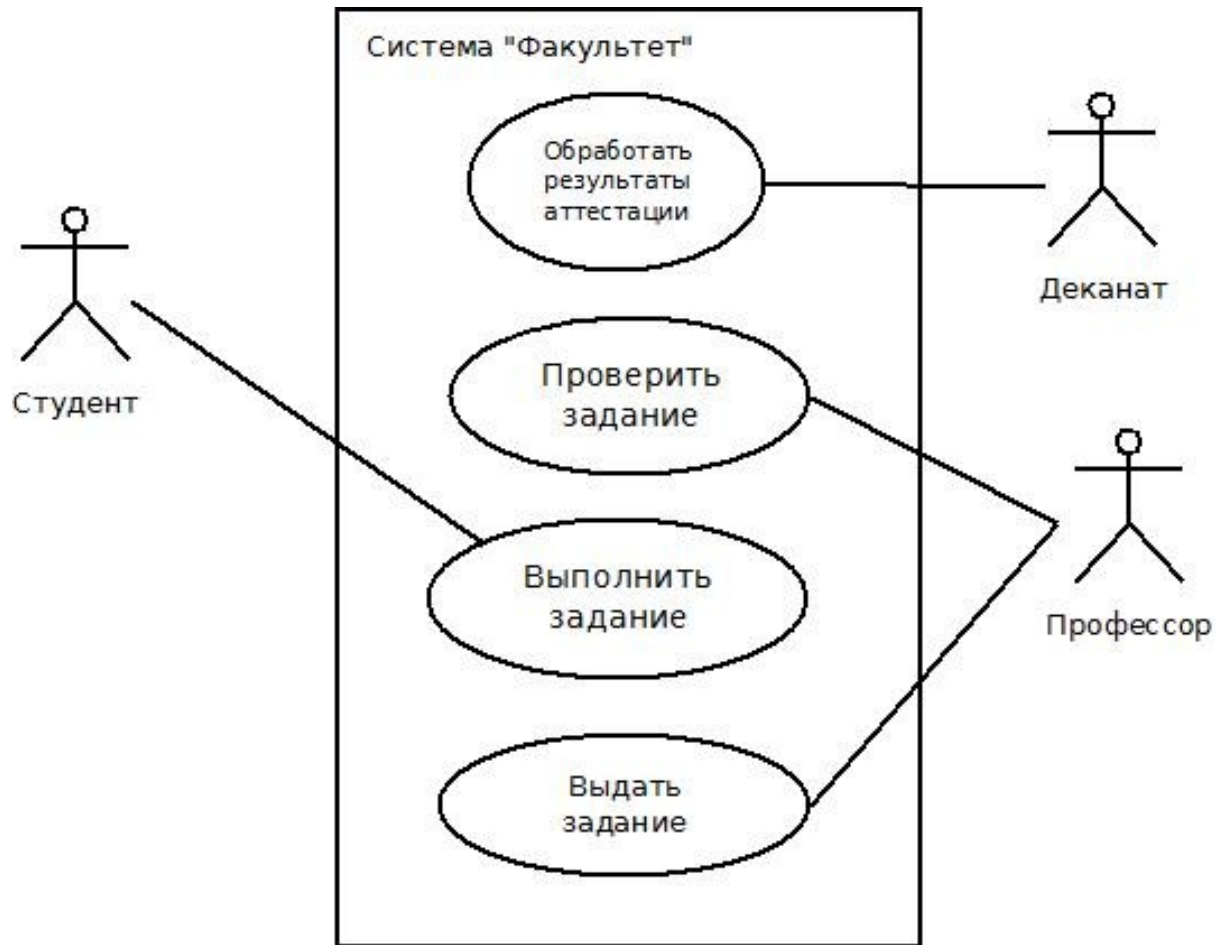


Диаграмма Use Case (2)

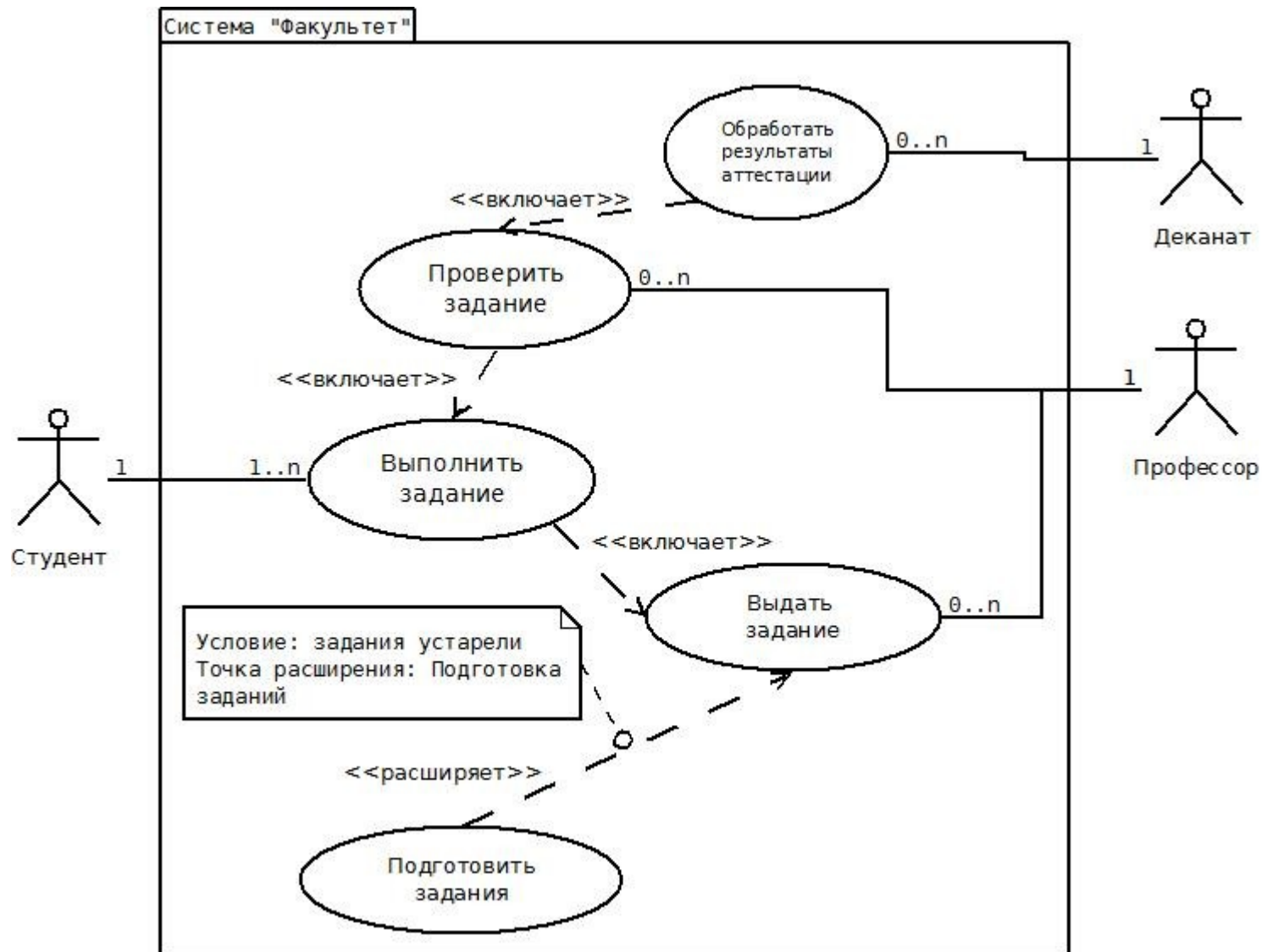


Диаграмма (видов) деятельности



Диаграмма классов

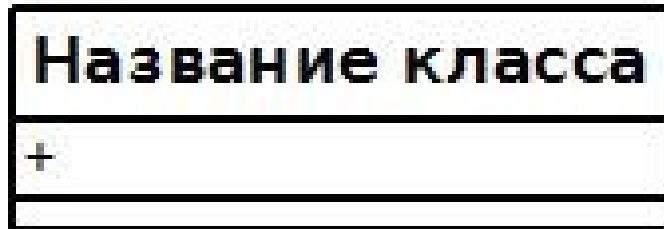


Диаграмма классов

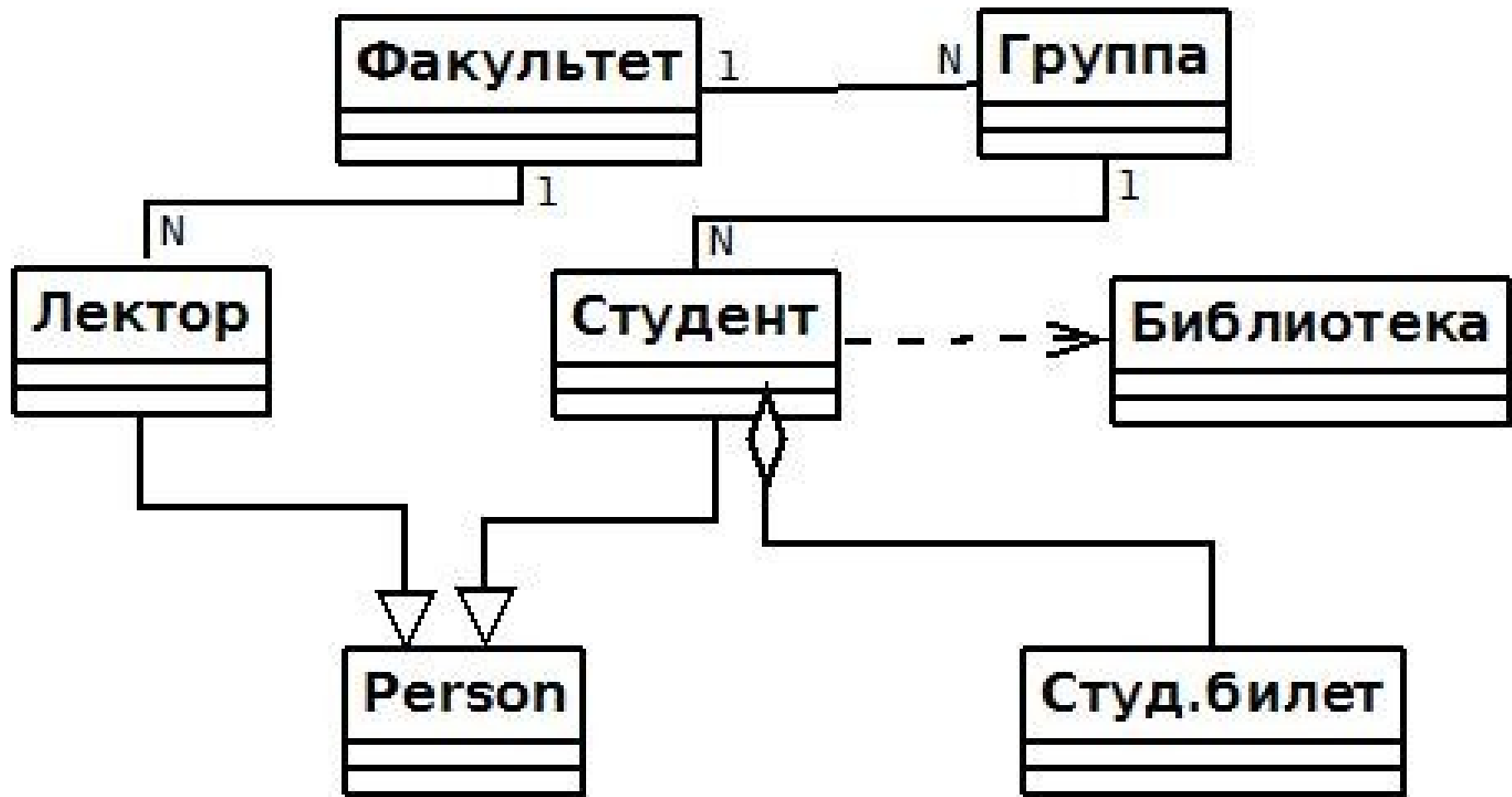


Диаграмма объектов

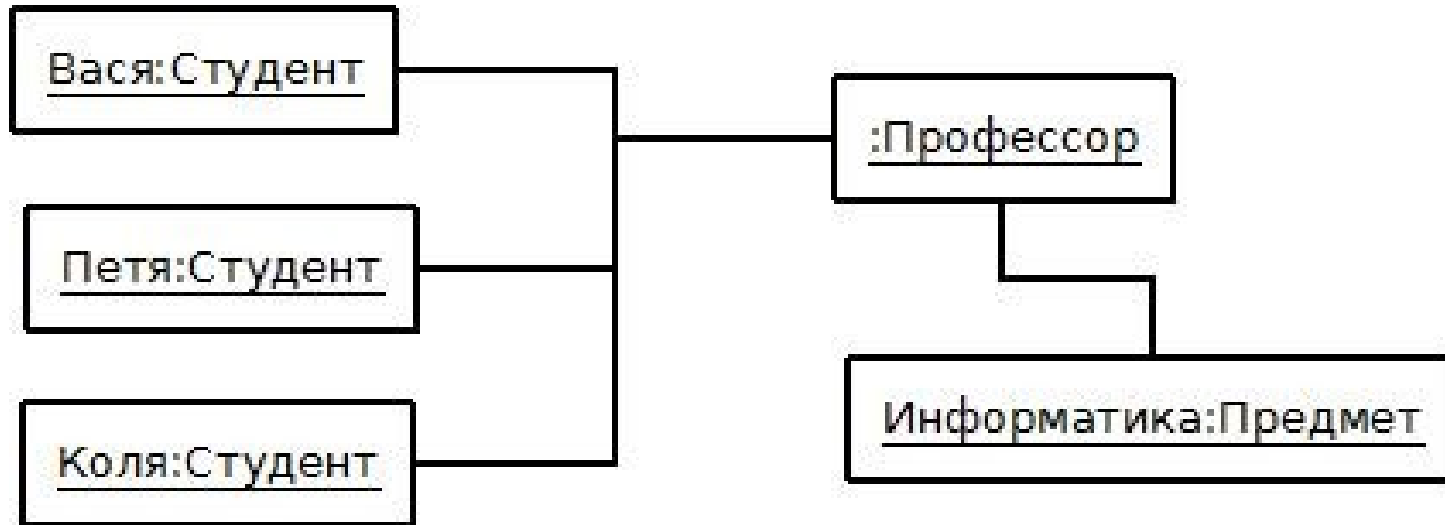


Диаграмма состояний

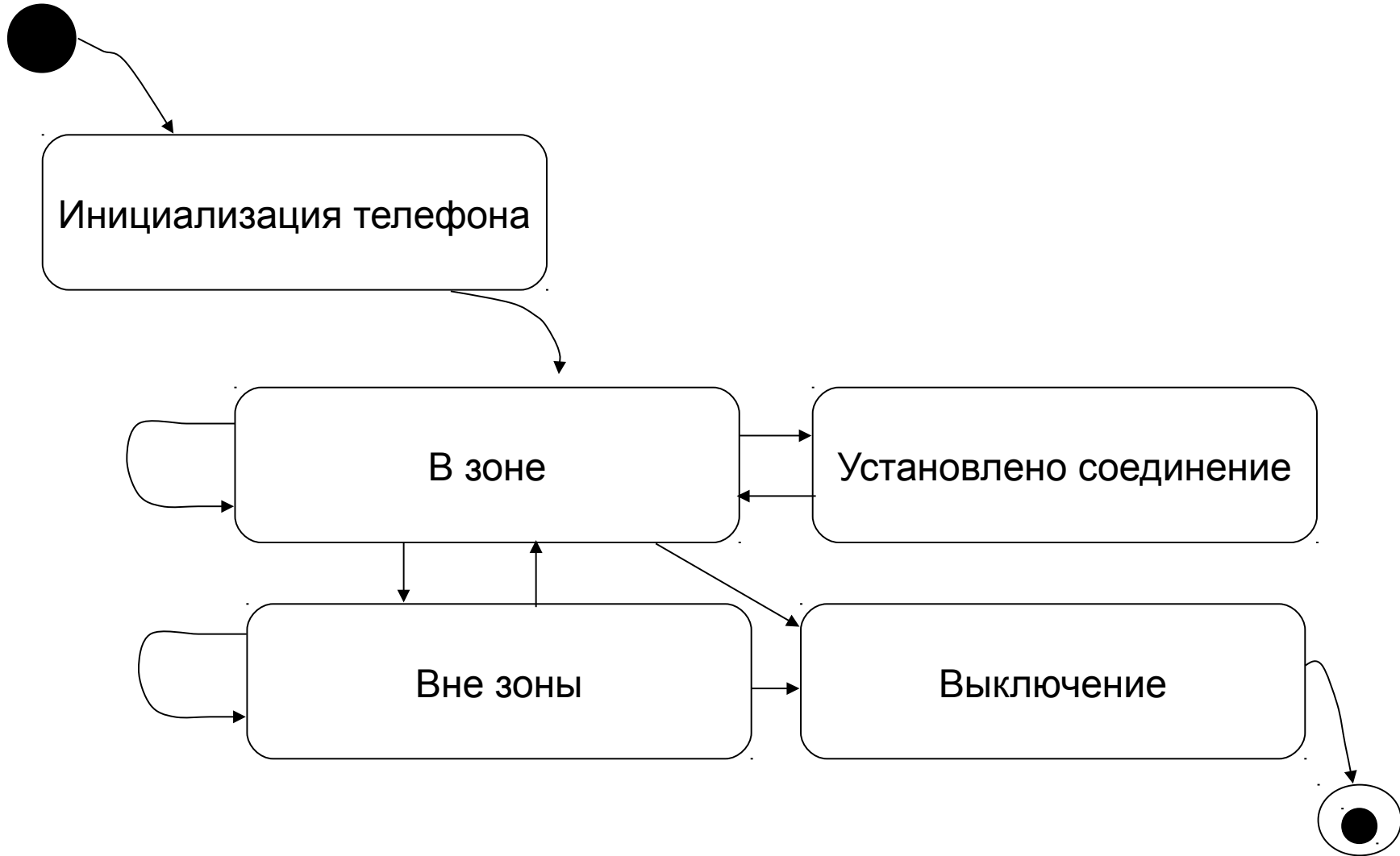


Диаграмма последовательности

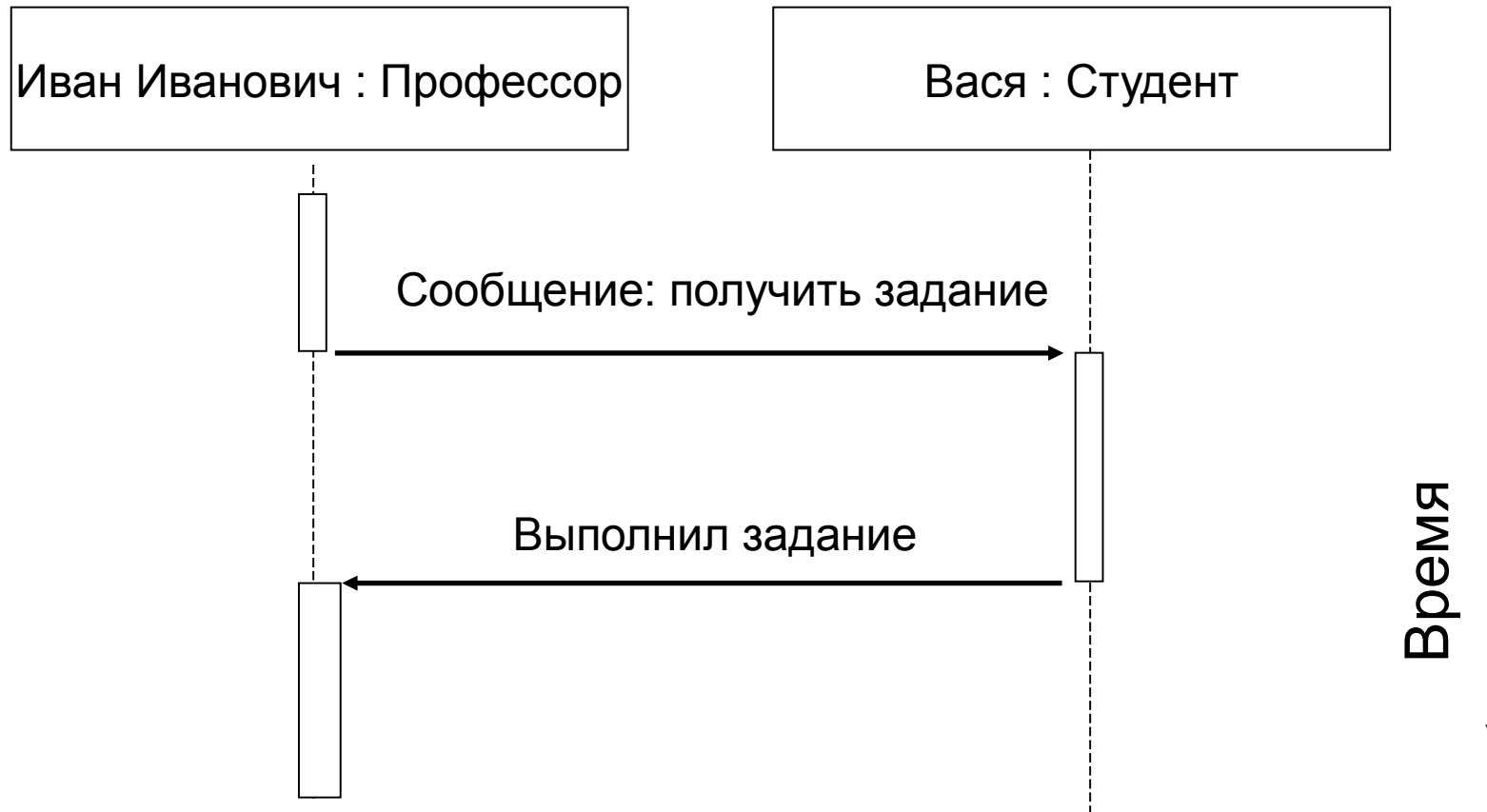


Диаграмма компонентов

Артефакт:

- таблицы,
- файлы данных,
- исполняемые файлы,
- документы,
- динамически загружаемые библиотеки.



Компонент:

- программная реализация класса или нескольких классов

Интерфейс

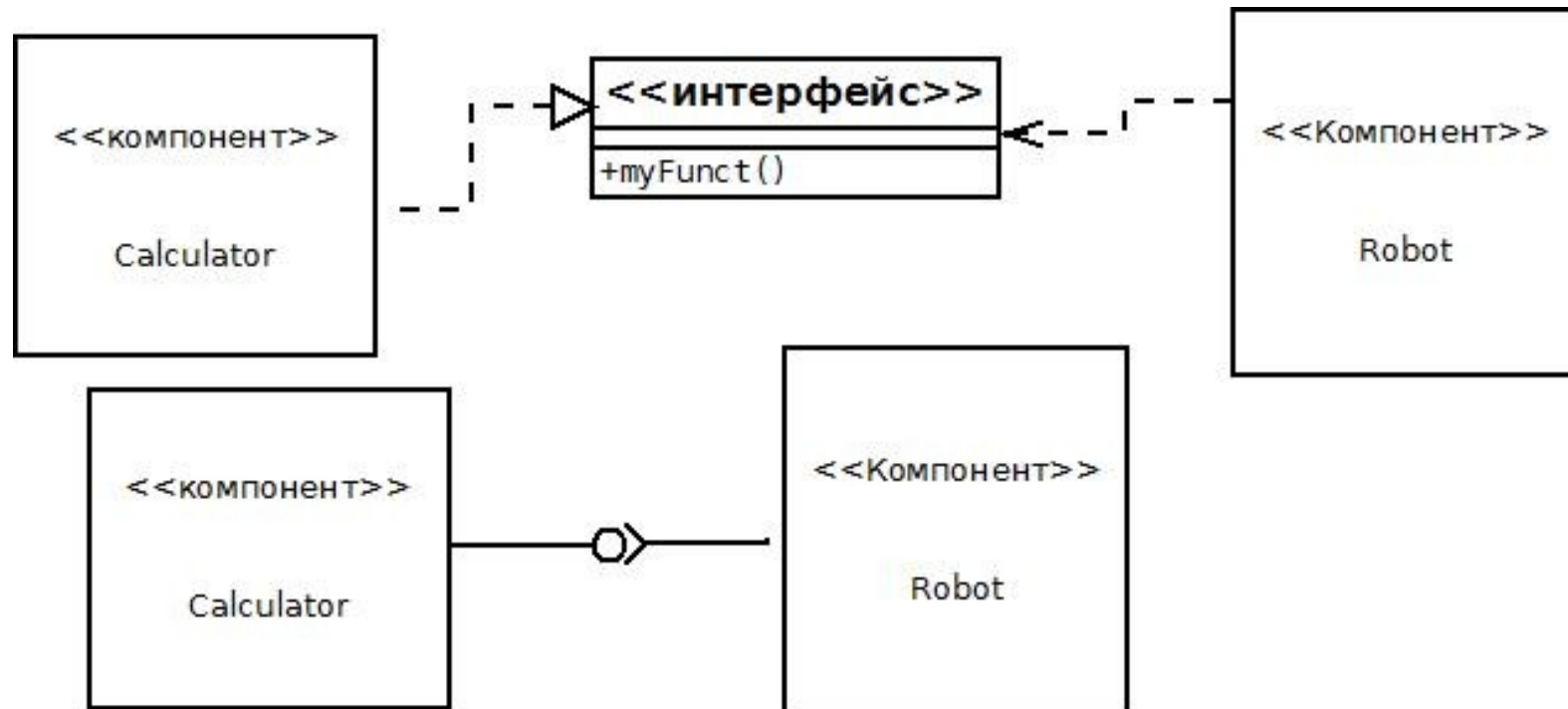
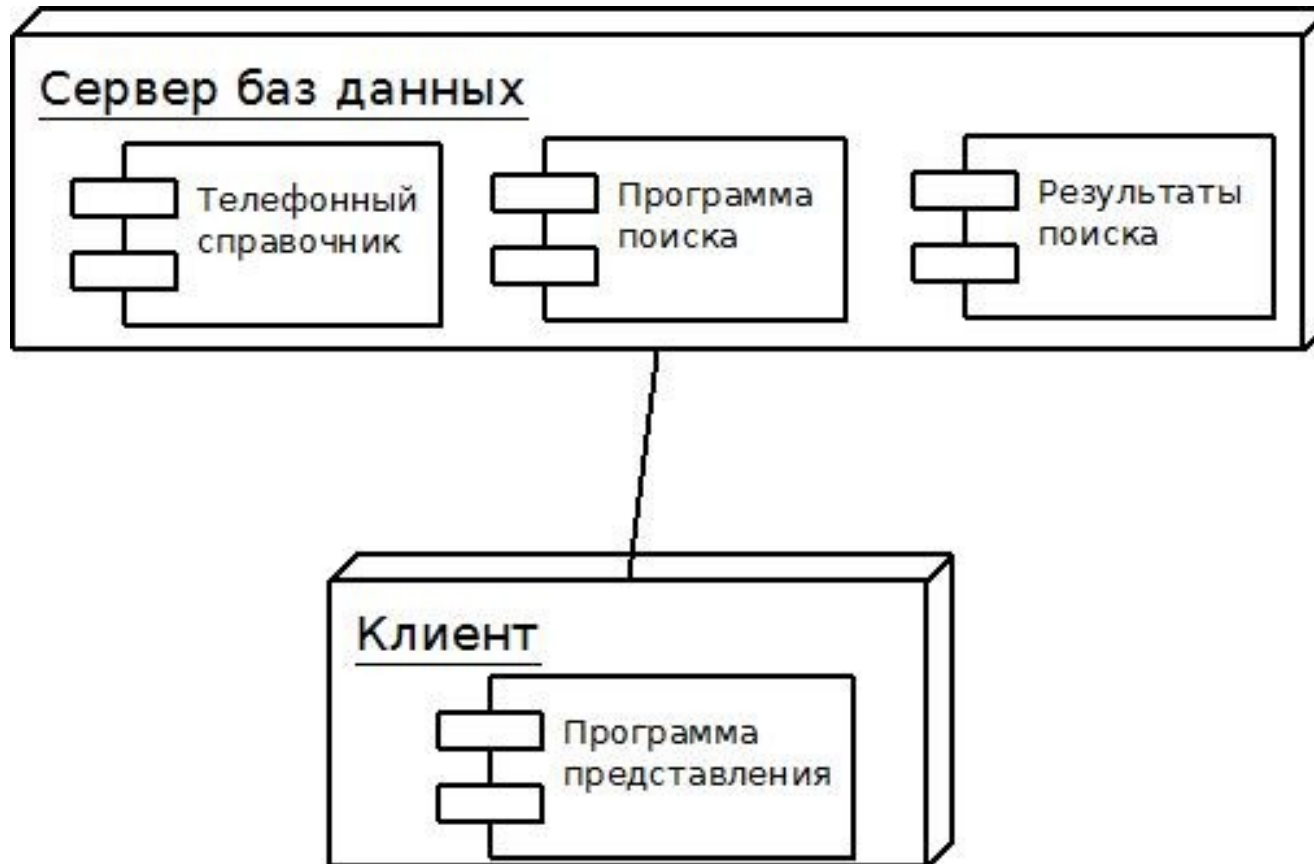


Диаграмма развертывания



Вопросы

- Где можно применять диаграммы UML?
- Какие виды диаграмм наилучшим образом представляют структуру ПО?
- Какие виды диаграмм представляют описание поведения ПО во времени?
- Какие виды диаграмм наилучшим образом представляют требования к ПО?

заголовок