

## USE CASE MODEL – kasutusjuhtude mudel

### Диаграмма прецедентов / вариантов использования

Как пользователи/деятели взаимодействуют с системой

Модель Use Case состоит из диаграмм прецедентов и их описания. Диаграммы составляются на основе функций инфосистемы.

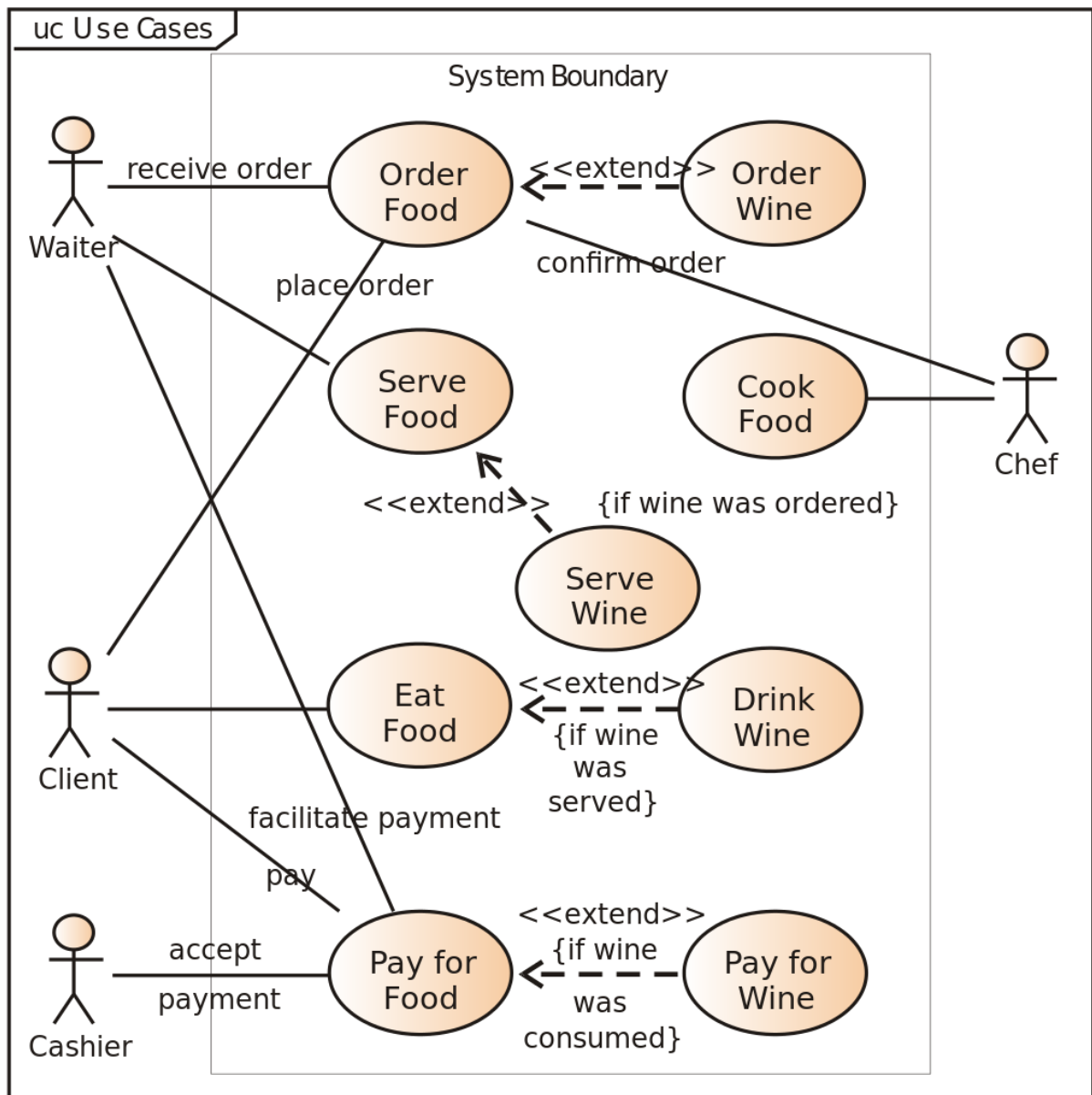
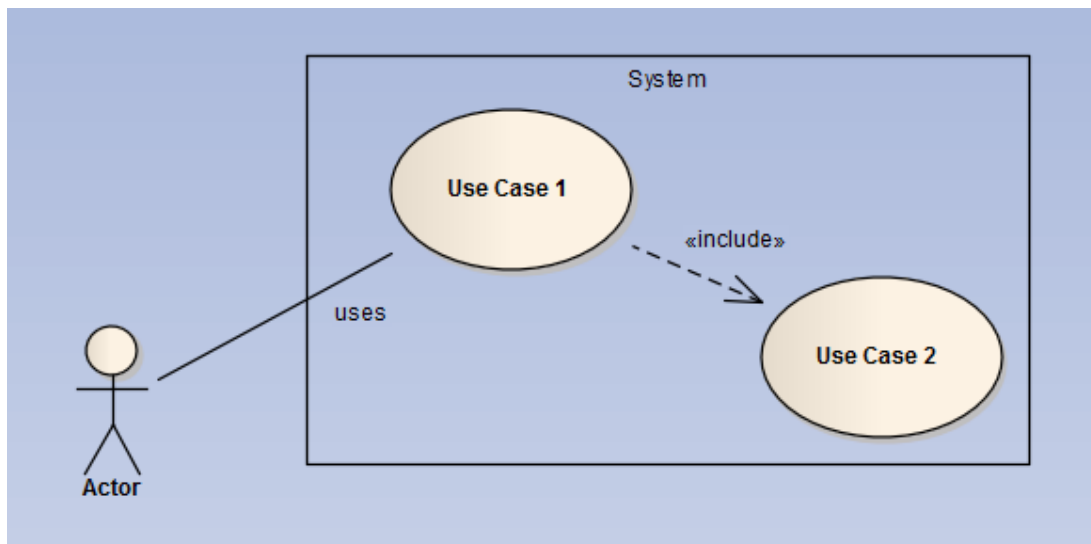


Диаграмма прецедентов показывает актеров, прецеденты и отношения между ними:

- Какие актеры выполняют тот или иной прецедент
- Какие прецеденты включают другие прецеденты



**Actor** – актер / пользователь / деятель

**Use Case** – прецедент

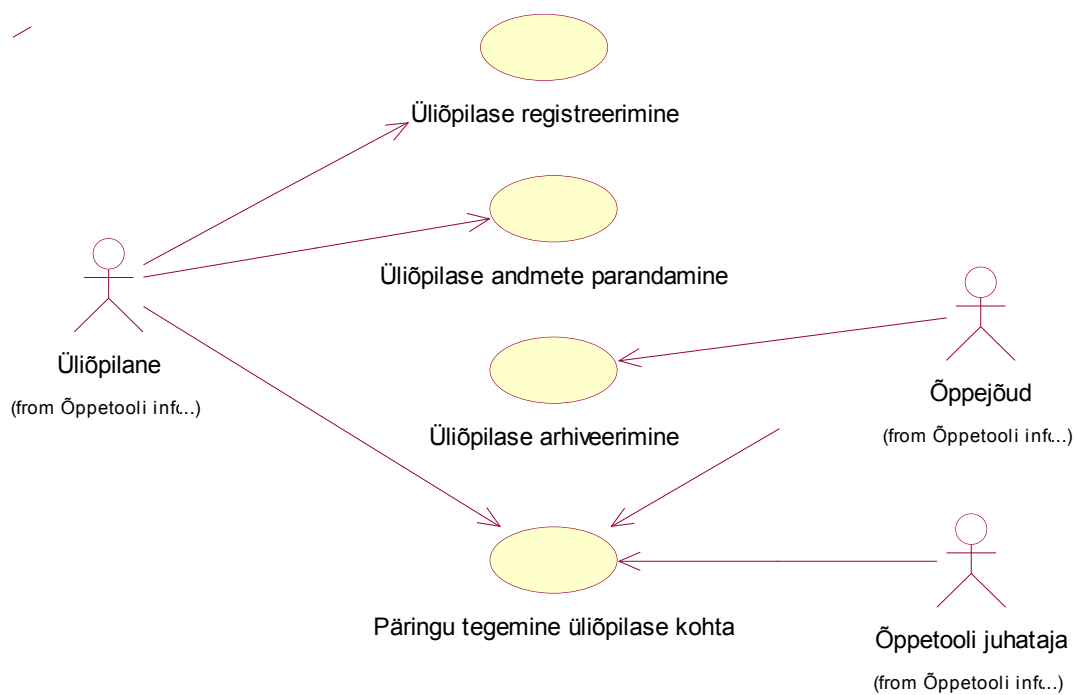
**System** - границы системы

**Типы связей:** между деятелем и прецедентом, между прецедентами

Можно добавлять пояснения и комментарии

### Пример: Учет студентов

Use Case диаграмма учета студентов:



### Краткое описание прецедента

**Название:** UC1.3 Запрос информации о студенте

**Деятели:** директор кафедры, преподаватель, студент

**Цель:** получить обзор об учебе студента

**Описание:** У пользователей системы возникает потребность получения информации о студенте и его успехах в учебе. Система проверяет права пользователя на совершение запроса и получения информации. Пользователь составляет запрос. Студент может получить информацию только о себе. Директор кафедры и преподаватели могут делать запросы о всех студентах.

### Подробное описание прецедента

**Название:** UC2.8 Регистрация результатов проверки знаний

**Деятели:** преподаватель

**Цель:** Регистрация проверки знаний студентов и результатов

**Подцель:** Ускорить фиксирование точного состояния оценок студента

**Описание:** Студент прошел проверку знаний. Преподаватель оценивает знания студента и регистрирует результат.

**Типичный ход событий:**

Действие деятеля	Ответ системы
1. Преподаватель входит в систему.	2. Система проверяет права пользователя.
3. Преподаватель желает оценить знания студента..	4. Система выводит на экран список предметов.
5. Преподаватель выбирает предмет, по которому была проверка знаний.	6. Система выводит на экран список студентов, изучающих этот предмет.
7. Преподаватель выбирает студента, которого он хочет оценить.	8. Система выводит данные студента.
9. Преподаватель регистрирует результат проверки знаний студента.	10. Система сохраняет проверку знаний и результат.

**Альтернативы:** Шаг 3 - если система не может распознать пользователя, то он не может зафиксировать оценку студента. Система оповещает об этом пользователя.

**Предпосылки:** предварительно создана проверка знаний, куда студент зарегистрировался.

**Ожидаемый результат:** Результаты проверки знаний зарегистрированы.

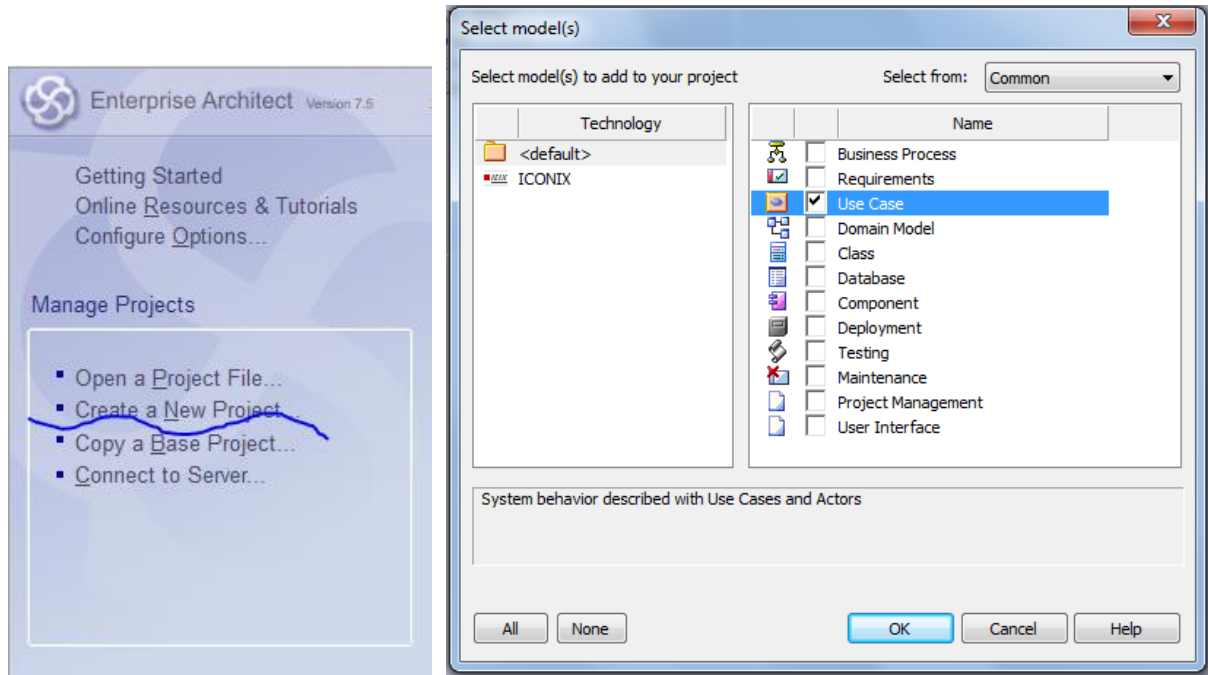
**UML программы:** Visio, Rational Rose, Enterprise Architect, ArgoUML (<http://argouml.tigris.org/>)

\* **ArgoUML:** Create -> New Use Case Diagram

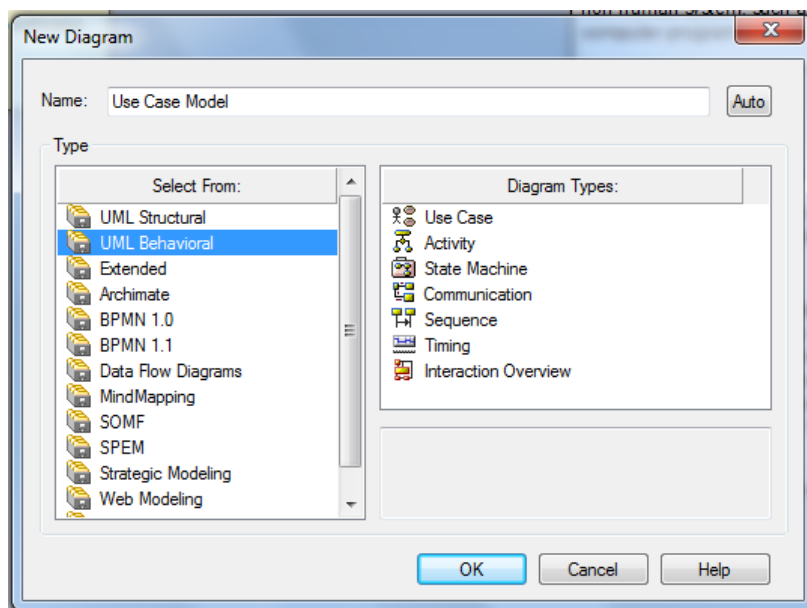
Экспорт диаграмм: File -> Export Graphics

\* **Visio:** File -> New-> Software -> UML Model Diagram, раздел UML Use Case

\* **Enterprise Architect:** File->Create a New Project -> сохраните .eap fail.



Project -> New Diagram -> UML Behavioral -> Use Case



Экспорт диаграмм: Diagram -> Save Image или Copy Image.

## **Задание: USE CASE MODEL - Модель вариантов использования / прецедентов**

Каждый студент должен выбрать, по крайней мере, одну основную функцию инфосистемы, для которой будет описывать диаграмму вариантов использования (в результате будет, по крайней мере, одна Use Case диаграмма, если делаете проект вдвоем, то соответственно - две диаграммы).

Каждый прецедент должен быть описан, по крайней мере, кратко (название, деятели, описание, при необходимости цель). Самый сложный прецедент должен быть описан более детально (название, деятели, описание, цель, типичный ход событий, альтернативы, предварительные условия/предпосылки, ожидаемый результат/последствия). Если делаете вдвоем, то - два подробно описанных прецедента.

**NB! 10.10.2014 срок сдачи следующей части проекта!**