



Co-funded by the
Erasmus+ Programme
of the European Union



Диаграмма вариантов использования (use case diagram)

Описание функций проектируемой игры

Локальные тренинги 24-25 мая 2017 г.
Лектор: Тихонова Оксана Анатольевна



Co-funded by the
Erasmus+ Programme
of the European Union



UML - общецелевой язык визуального моделирования, который разработан для спецификации, визуализации, проектирования и документирования компонентов программного обеспечения, бизнес-процессов и других систем.

Основная цель UML - графически изобразить архитектуру проекта



Co-funded by the
Erasmus+ Programme
of the European Union



University Enterprises Cooperation
In Game Industry In Ukraine

UML и gamedev

В последнее время игроделы стали чаще использовать UML, т.к. данная технология полезна при разработки игр.

Примеры:

1. Если проект большой – тяжело разбираться с кучей скриптов.
2. Если над проектом работают более 1 человека – и например, нужно согласовать некоторые детали проекта или быстро ввести в курс дела нового члена команды.
3. Документация игры получается намного информативней, если в ней есть диаграммы.
4. ГЛАВНОЕ - UML можно внедрить на любом этапе создания проекта.



Co-funded by the
Erasmus+ Programme
of the European Union



University Enterprises Cooperation
In Game Industry In Ukraine

Основной этап разработки игры – формулировка набора требований к продукту (*техническое задание*)



В языке UML для формализации функциональных требований применяются диаграммы использования.



Co-funded by the
Erasmus+ Programme
of the European Union

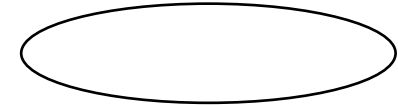


На диаграмме использования изображаются:

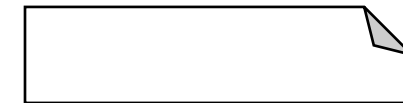
1. актёры — группы лиц или систем, взаимодействующих с нашей системой; графическое обозначение



2. варианты использования (прецеденты) — сервисы, которые наша система предоставляет актёрам; графическое обозначение



3. комментарии; графическое обозначение



4. отношения между элементами диаграммы



Co-funded by the
Erasmus+ Programme
of the European Union



University Enterprises Cooperation
In Game Industry In Ukraine

Виды отношений:

- Отношение ассоциации (association).
- Отношение расширения (extend relationship).
- Отношение включения (include relationship).
- Отношение обобщения (generalization relationship). Означает, что два и более актера могут взаимодействовать с одним и тем же множеством прецедентов одинаковым образом.

Графическое обозначение





Co-funded by the
Erasmus+ Programme
of the European Union

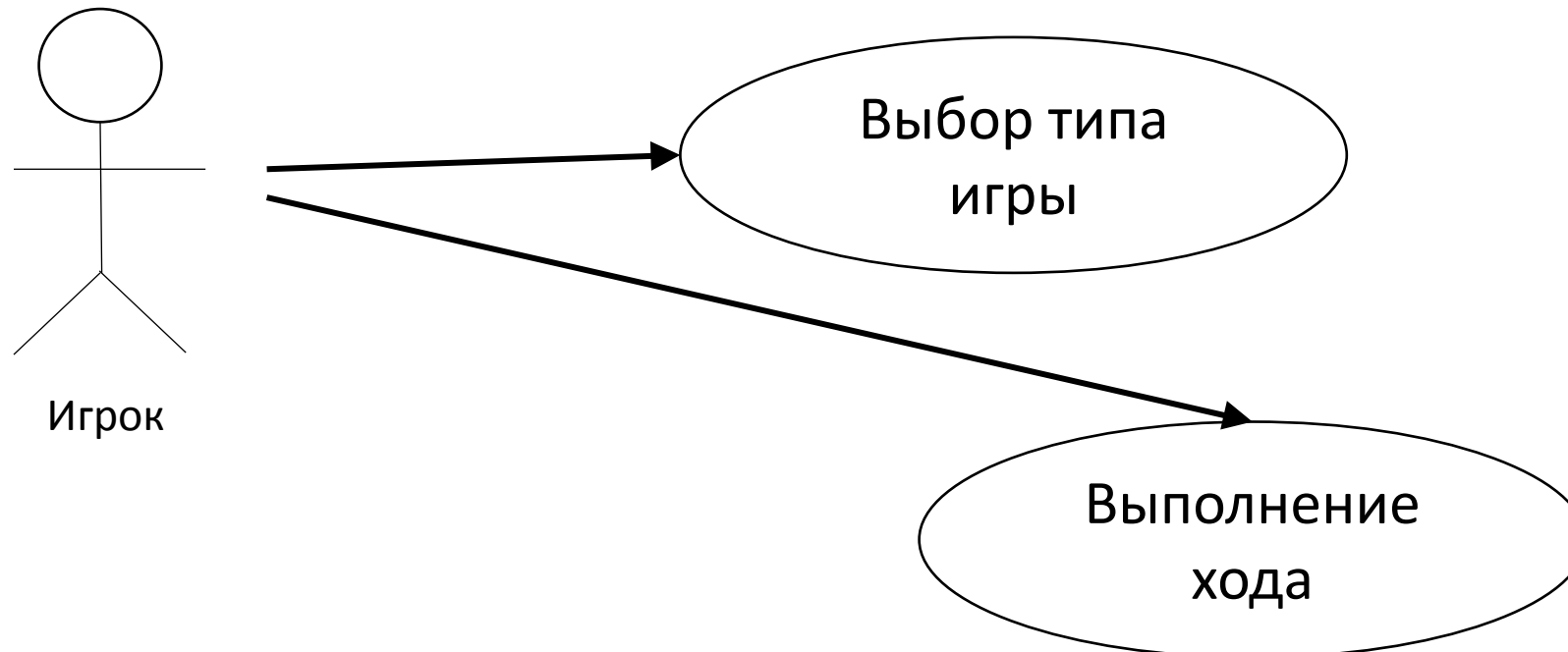


University Enterprises Cooperation
In Game Industry In Ukraine

Отношение ассоциации (association).

Отражает возможность использования актёром прецедента.

Графическое обозначение





Co-funded by the
Erasmus+ Programme
of the European Union

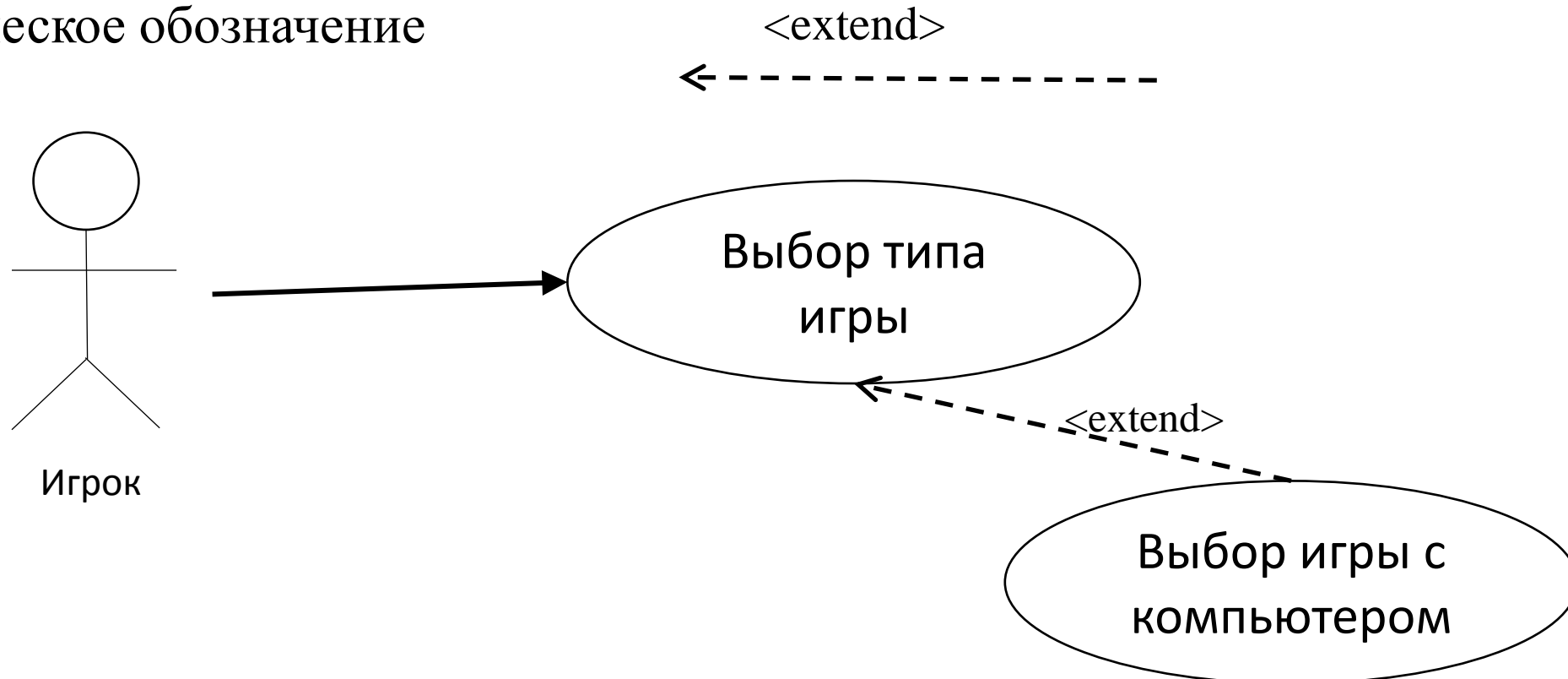


University Enterprises Cooperation
In Game Industry In Ukraine

Отношение расширения (extend relationship).

Отражает **возможное** присоединение одного варианта использования к другому в некоторой точке (*точке расширения*).

Графическое обозначение





Co-funded by the
Erasmus+ Programme
of the European Union

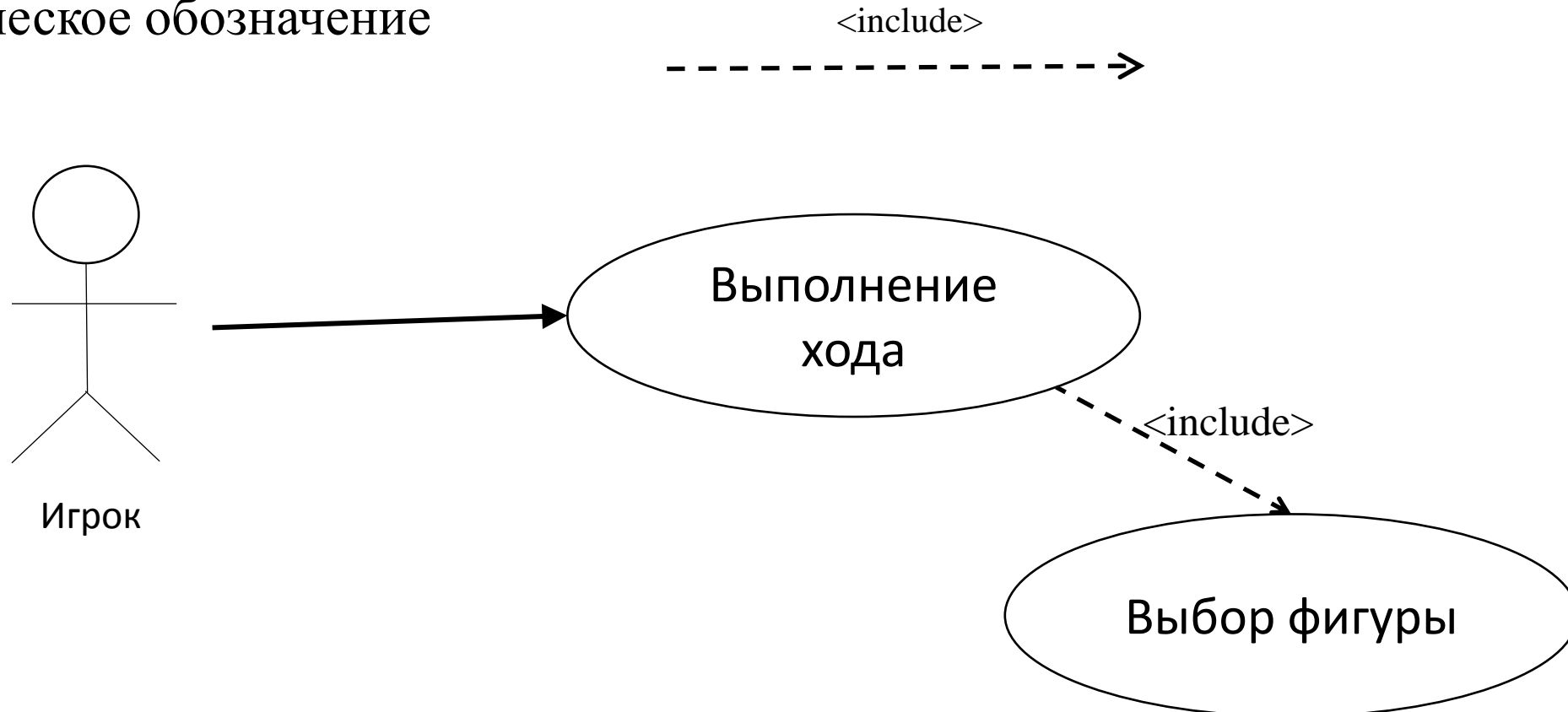


University Enterprises Cooperation
In Game Industry In Ukraine

Отношение включения (include relationship).

Указывает на то, что поведение одного прецедента включается в некоторой точке в другой прецедент в качестве составного компонента.

Графическое обозначение





Co-funded by the
Erasmus+ Programme
of the European Union

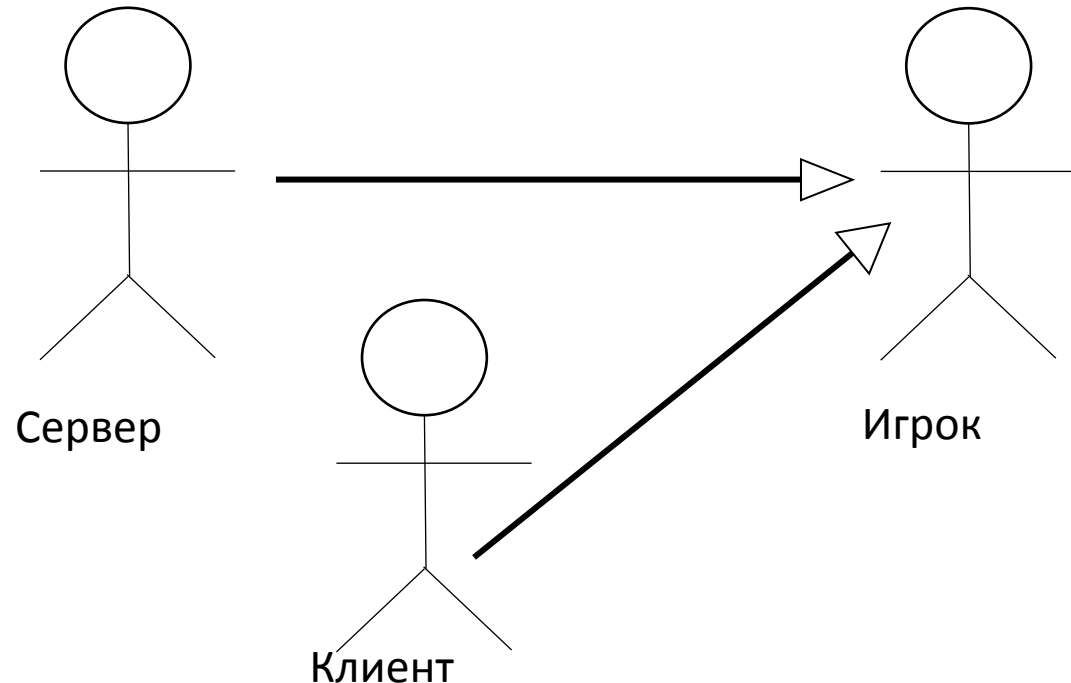
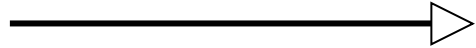


University Enterprises Cooperation
In Game Industry In Ukraine

Отношение обобщения (generalization relationship).

Означает, что два и более актера могут взаимодействовать с одним и тем же множеством прецедентов (или других актеров) одинаковым образом.

Графическое обозначение



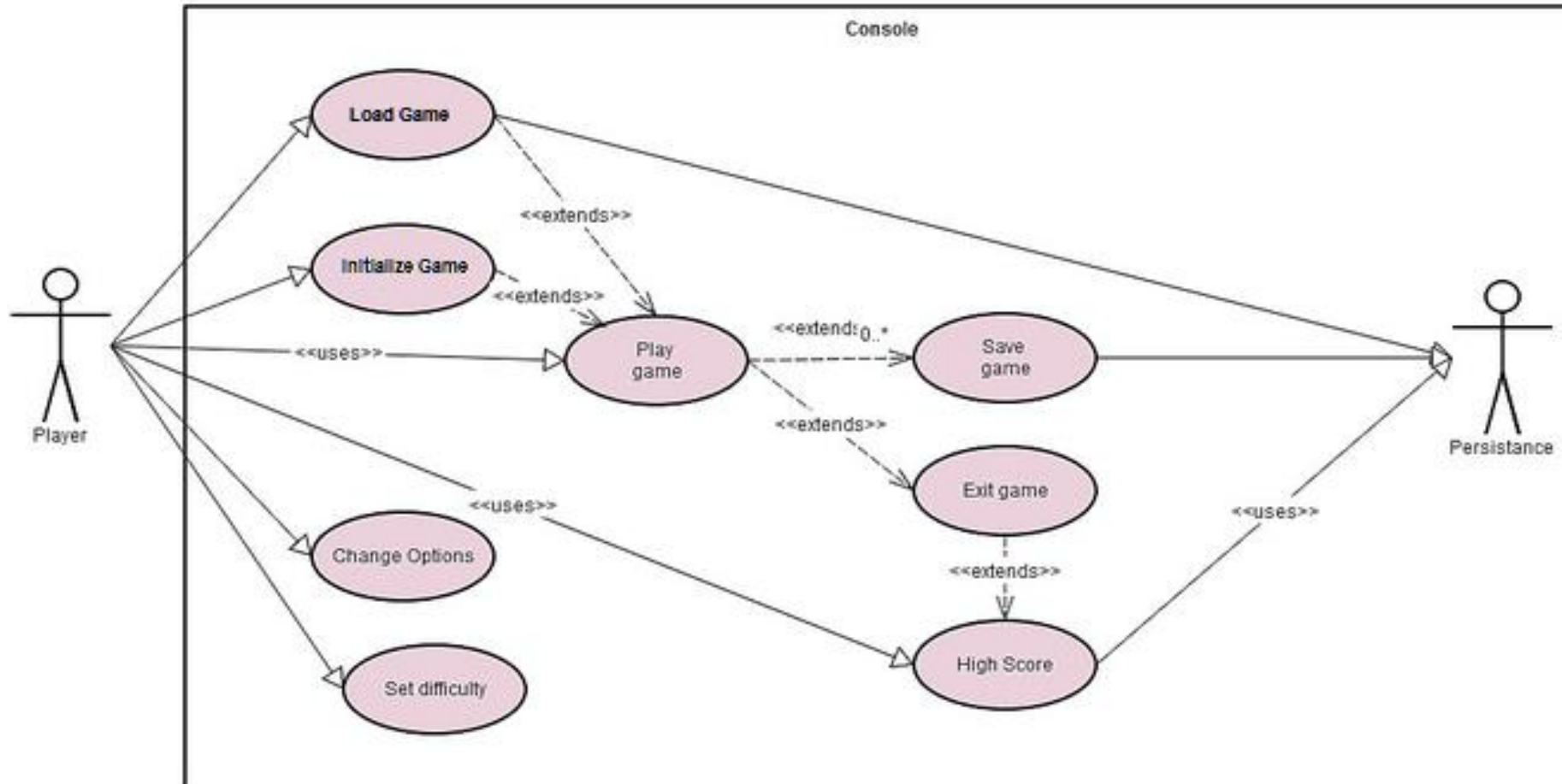


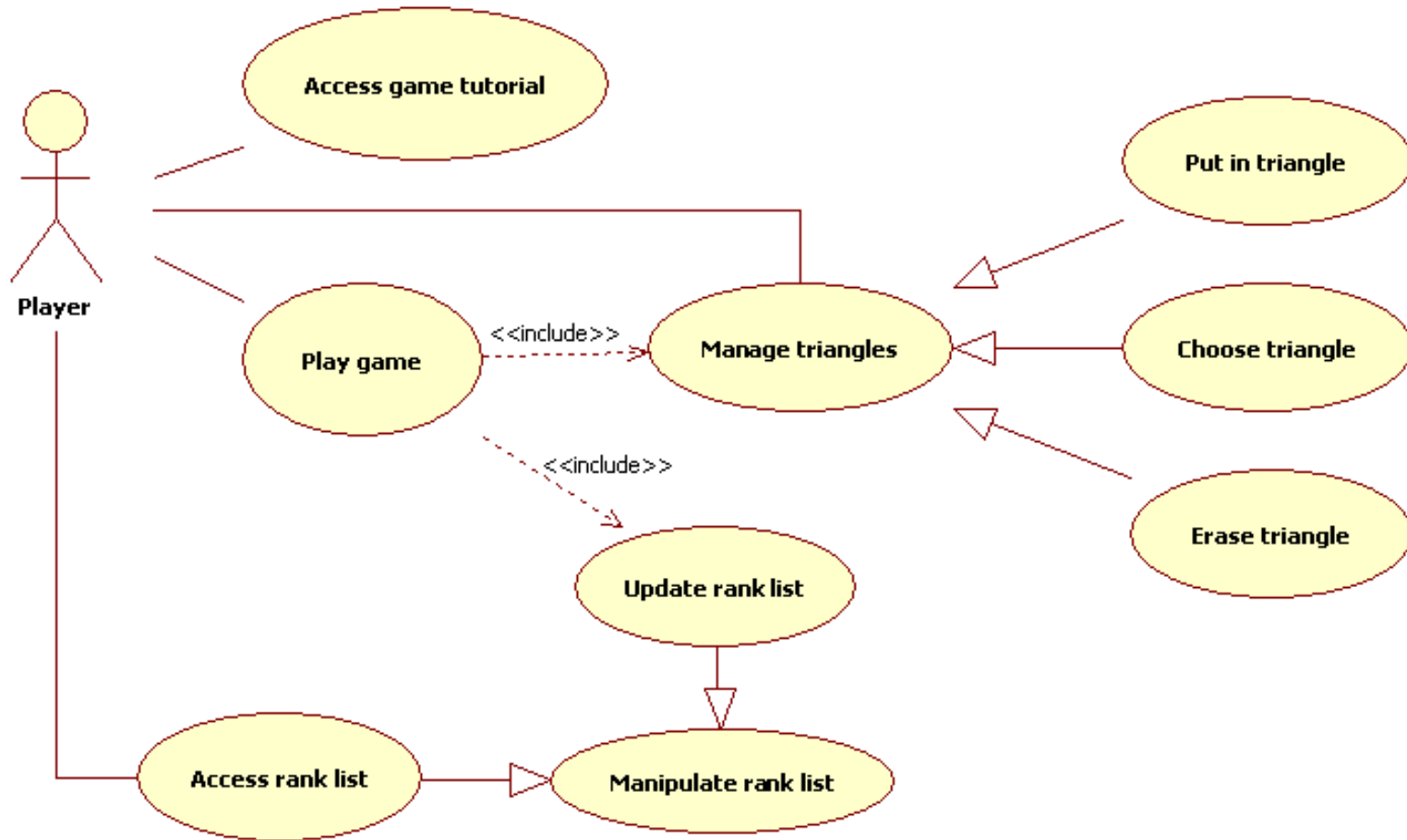
Co-funded by the
Erasmus+ Programme
of the European Union



University Enterprises Cooperation
In Game Industry In **Ukraine**

Примеры диаграммы использования





Игрок может выбрать игру, посмотреть учебник по игре или получить доступ к ранжированию в меню игры, которое отображается при запуске программы. Когда начался игровой этап, программа будет считывать соответствующий шаблон этапа и назначать треугольники заданной позиции в качестве инициализации.

Во время игры игрок может управлять треугольниками с заданным цветом, например, выбирать ориентацию треугольника, перемещать треугольник в нужное ему положение и вставлять треугольник. Тем временем игровая программа будет продолжать проверять, стереть ли треугольники, которые три или более одинаковых цветных треугольника склеились.

Когда игра окончена, игрок может ввести свое имя и обновить ранжированный список