# Описание архитектуры программы

# Архитектурное представление

Данный документ состоит из 3 разделов.

В разделе «Логическое представление» представлена диаграмма классов с кратким описанием назначения классов, так же представлена диаграмма активности.

В разделе «Представление прецедентов» представлена диаграмма прецедентов с кратким описанием основных прецедентов.

В разделе «Представление развертывания» представлена диаграмма развертывания, отражающая распределение компонентов по узлам сети.

# Логическое представление



На данной диаграмме отображены все классы системы и связи между ними.

### Назначение классов:

**Player**: дочерний класс для классов MainPlayer и SlavePlayer. Предоставляет общие функции игроков.

**MainPlayer:** класс для игрока- организатора. Предоставляет функции игрока-организатора. Именно игрок-организатор создает игру, выбирает параметры, подключает игроков, сохраняет/восстанавливает игру.

**SlavePlayer**: класс для ведомого игрока. Он не выполняет никаких функций, только кидает фишки.

**ListOfPlayers**: список игроков. Позволяет добавлять игроков в список, удалять игроков из списка, получать активного игрока и др.

**Controller**: класс, управляющий работой программы.

**MainDlg:** класс, отвечающий за реализацию пользовательского интерфейса.

**iNetworkStruct:** предоставляет интерфейс внутренней структуры сети.

**iSerializable:** Интрфейс, описывающий возможности сериализации данных.

**iDrawable:** Интерфейс, описывающий возможности отображения данных.

**Data:** реализует интерфейсы iNetworkStruct, iDrawable, iSerializable.

**NetworkPetry:** класс, реализующий сеть Петри.

**position\_iterator**: итератор для обхода позиций.

**transition\_input\_it:** Итератор для обхода входных позиций перехода.

**transition\_iterator:** Итератор для обхода переходов.

**transition\_output\_it:** Итератор для обхода выходных позиций перехода.

**MaxFiredException:** Исключение, генерируемое при зацикливании сети.



## Представление прецедентов



### Основные прецеденты:

1. **«Создание игры».** Данный прецедент предназначен для создания игры, задания ее параметров, идентификации пользователей. Прецедент включает в себя прецедент **«Задание параметров»**, который описывает процесс задания параметров сети, в случае локальной игры – идентификации пользователей, в случае сетевой игры – действия ведомых игроков для подключения к игре.
2. **«Играть».** Описывает, непосредственно, процесс игры. Определяет действия игроков во время процесса игры. Включает в себя прецедент **«Отображение текущего статуса»**, который описывает действия по отображению состояния игры после каждого хода игрока, вывод имени победителя.
3. **«Сохранение/восстановление игры».** Данные прецеденты описывают последовательность действий по сохранению/восстановлению текущей конфигурации игры.

## Представление развертывания



В разработанной системе возможны 2 варианта игры:

* Локальная;
* Сетевая.

При локальном варианте игры на локальной машине запускается экземпляр программы и происходит, непосредственно, процесс игры.

При выборе сетевого варианта игры выбирается Сервер, на котором игрок-организатор запускает экземпляр программы. Так же все ведомые игроки на удаленных компьютерах запускают экземпляры программы и пытаются подключиться к Серверу. После успешного подключения всех ведомых игроков начинается процесс игры.