

# Документация по проекту по дисциплине “Программирование на языке Котлин”

Кашенин А.Д, 24.Б10

Мой проект - P2P приложение для игр в локальной сети.

Технологический стек:

- 1). Jetpack Compose
- 2). Java Socket
- 3). Kotlin Serialization
- 4). Корутины

Проект устроен по модели MV:

Есть 2 отдельных проекта для бекенда и фронта, соединённые друг с другом

## Архитектура кода:

### Бекенд

1). Модуль “general”. Тут хранится общая для сервера и клиента логика, например интерфейс игры и простейшая игра - крестики-нолики. Соответственно:

В game.kt лежит интерфейс

В TicTacToeGame лежит десктопная реализация

ServerInfo - простой data класс для обмена JSon о информации сервера при автопоиске

Network Config - я хотел сделать автопоиск по портам, но передумал. Для дебага оставил этот файл

### 2). Модуль server

Я плохо декомпозирую код, поэтому всего один файл MainServer, который осуществляет всю работу сервера. Сервер запускается на IP адресе, полученном из getLocalIpAdress(), проверяет что потенциальный порт свободен, поддерживает автонахождение сервера. Так же можно выбрать имя серверу, чтобы при автопоиске оно прикольно отображалось

### 3). Модуль client

Не особо отличается от сервера. Он асинхронно в методе selectGoodServers сканирует все найденные IP сервера для нахождения запущенного сервера. Но это не всегда работает, поэтому мной была добавлена функция подключения к конкретному известному IP. В моей реализации это значит, что при запуске сервера мы видим IP, после чего подключаемся по нему из клиента

### Фронтенд

Тут один модуль

#### 1). ClientComposable

Логика отрисовки клиента отличается от логики отрисовки сервера, т.к. клиенту нужно поддерживать автовыбор запущенных серверов. Этот файл просто переопределяет некоторые методы клиента чтобы автовыбор корректно отрисовывался

#### 2). MainActivity

Это файл с основной отрисовкой поля выборов, в нём происходит запуск приложения и т.п. В общем - просто центральный файл

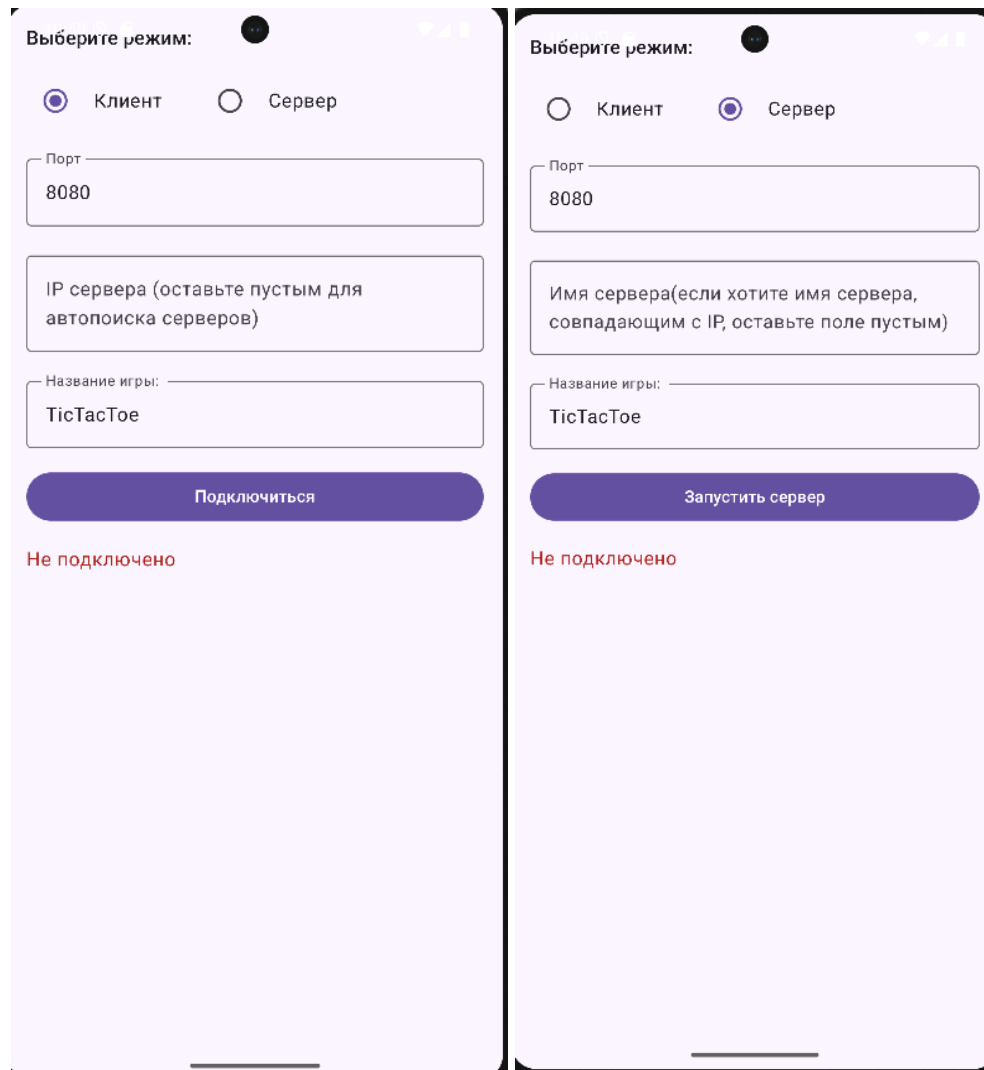
### 3). TicTacToeComposable

реализация TicTacToe с переопределёнными методами для отрисовки поля и считывания данных с помощью UI

Гайд как пользоваться приложением.

1). Нужно скачать приложение и включить его на 2 устройствах в одной локальной сети(подойдёт даже если одно будет раздавать интернет, а другое подключаться к нему)

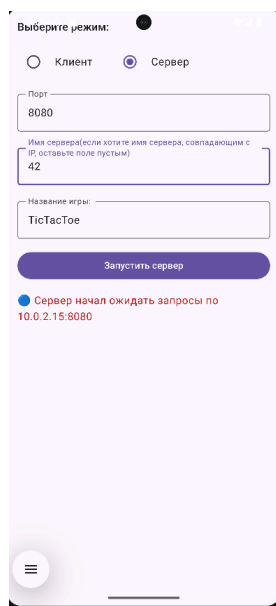
2). Вам откроется такая картина:



Соответственно моё приложение P2P, и сначала нужно запустить сервер на одном из устройств, потом подключиться

Про название игры - пока реализована только одна игра и это поле буквально ничего не делает, но можно сделать больше игр и тогда приложение будет поддерживать больше игр.

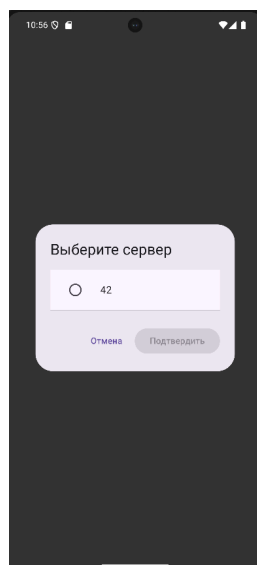
Запустим сервер и посмотрим варианты развития событий:



Имя сервера - для автоподключения.

1 сценарий - автоподключение(может не всегда работать из-за проблем с определением IP)

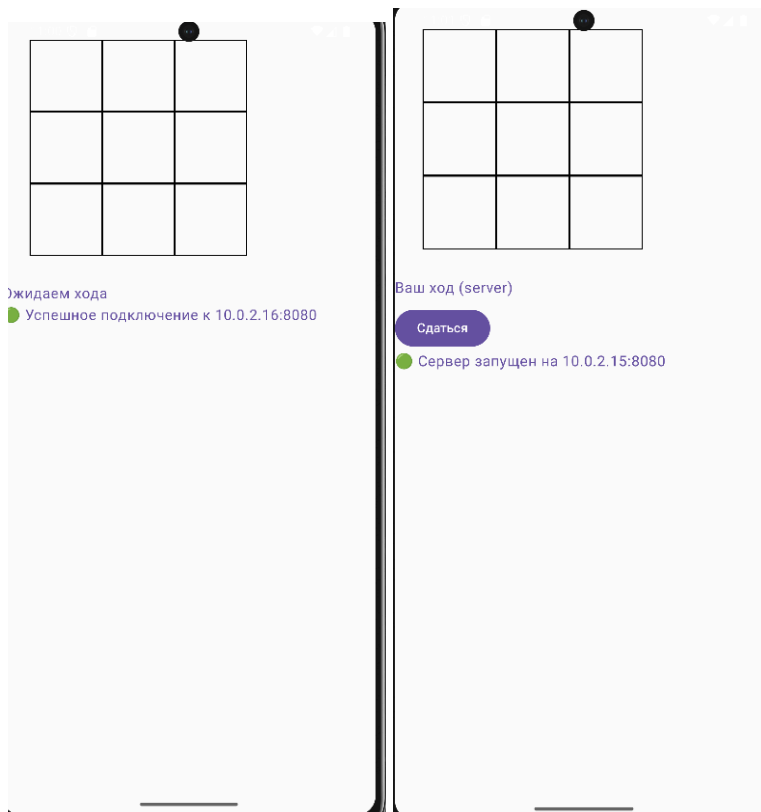
Если мы оставим поле с IP пустым, то если в той же локальной сети запущен сервак, то после нажатия на кнопку “подключиться” появится что-то такое:



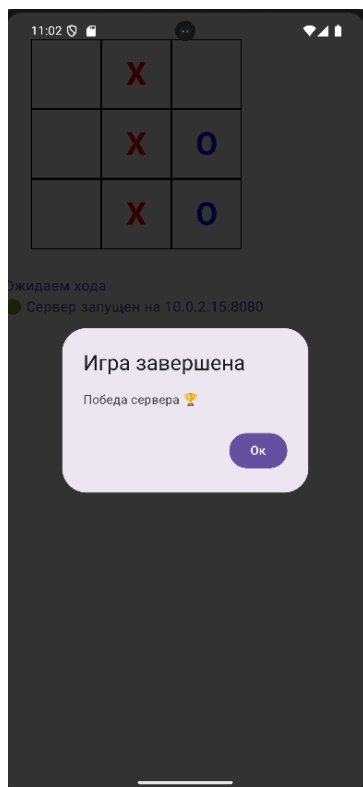
Выбираем его, жмём “подтвердить”, и начинается игра

2 сценарий - прямой ввод IP - шага с выбором сервера по имени нет

После подключения появляется UI с игрой(начинает сервер)



Сами квадратики кликабельны, просто жмёте на них и играете. Если хотите сдаться - жмёте на кнопку.



После окончания игры появляется такая плашка, после чего можно закрывать приложение и начинать заново

