Чат – Клиент-Сървър конзолно приложение

Проект по Разпределени ИТ архитектури

**Изготвил: Асен Пенчев, факултетен №71810**

**Проверил:…………………..................**

**/ас. Христо Христов/**

Съдържание

Условие на задачата.................................................................................................................................. 3

Преглед на алгоритмите за реализиране на задачата............................................................................ 3

Архитектура на приложението и реализация на проекта...................................................................... 4

Проведени тестове и измервания............................................................................................................ 7

Условие на задачата:

Разглеждаме софтуерната архитектура за разпределени системи клиент-сървър. Най-общо можем да характеризираме тази архитектура като услуги изисквани от клиентите и предоставяни от сървърите. Сървърите доставящи услугите се държат и проектират различно от клиентите изискващи услугите.

Някой от по важните характеристики на двете страни в тази архитектура са както следва:

Характеристики на Клиента

- Подава заявки

- Изчаква отговор

- Свързва се до малък брой сървъри едновременно

- Взаимодейства си с крайните потребители чрез графичен интерфейс

Характеристика на Сървъра

- Пасивност

- Чака за заявки от клиенти

- При получаване на заявки, ги обработва и след това отговаря

- Получава заявки от голямо количество клиенти

- Не контактува директно с крайният потребител

Задачата ни се състои да моделираме определен тип клиент-сървър приложение, а именно такова за обмяна на малки текстови съобщения. Приложението се състои от две части (програми) работещи напълно независимо една от друга - chat сървър и chat клиент. По предварително дефиниран протокол за работа на клиента със сървъра трябва да се реализират двете части на приложението.

Преглед на алгоритмите за реализиране на задачата

Архитектура на приложението и реализация на проекта

Проведени тестове и измервания

**SimpleChatApp**

This project was made in the process of learning about sockets and threads. Idea was to make a simple Server - Clinet chat program (one to many) using broadcasting technique. In this process I made 4 sub projects:

* [SimpleChatApp](https://github.com/MilosVulikic/SimpleChatApp/tree/master/SimpleChatApp) - Console Server
* [SimpleChatAppClient](https://github.com/MilosVulikic/SimpleChatApp/tree/master/SimpleChatAppClient) - Console Client
* [SimpleChatFormClient](https://github.com/MilosVulikic/SimpleChatApp/tree/master/SimpleChatFormClient) - Form Client
* [SimpleChatFormServer](https://github.com/MilosVulikic/SimpleChatApp/tree/master/SimpleChatFormServer) - Form Server

**Getting Started**

Easiest way to test this application is to download [program exe files](https://github.com/MilosVulikic/SimpleChatApp/blob/master/SimpleChatApp%20.exe%20files.rar) and run any server- client combination you want.

**How to test it**

Since this application has 2 types of server apps it is advised to use one. If using both server application simultaneously, use different sockets. Take into account server is only receiving and broadcasting messages. Use more clients on one server to simulate regular chat. You have many possible options so it is up to you how to test it. Here are some ideas how to test:

Test console server - console client (take into account, both server and client ip address and port can only be changed in the code editor)

start - SimpleChatApp.exe (s)

start - SimpleChatAppClient.exe (1)

start - SimpleChatAppClient.exe (2)

Enter inquired names for (1) and (2) and start chatting

Test console server - form client (take into account, server ip address and port can only be changed in the code editor)

start - SimpleChatApp.exe (s)

start - SimpleChatFormClient (1)

start - SimpleChatFormClient (2)

Enter inquired names for (1) and (2) or use preset value and start chatting

Test form server - console client (take into account, client ip address and port can only be changed in the code editor)

start - SimpleChatFormServer.exe (s)

start - SimpleChatAppClient.exe (1)

start - SimpleChatAppClient.exe (2)

Enter inquired names for (1) and (2) and start chatting

Test console server - form client

start - SimpleChatFormServer.exe (s)

start - SimpleChatFormClient (1)

start - SimpleChatFormClient (2)

Enter inquired names for (1) and (2) or use preset value and start chatting

Use (s) with preset or manually change value of port and IP

NOTICE:

* You can use more clients than 2 simultaneously
* You can use console and form clients connected to the same server

Another way to test it is to clone files and opent the project on your by opening **SimpleChatApp.sln**

**Installing**

No installation is needed

**Running the tests**

For now tests are to be run manually by using the program. You can try to use clients on different network, accompanied by changes of ip&port values.

**Versioning**

[GitHub Visual studio extension](https://visualstudio.github.com/) used for versioning. For the versions available, see the [tags on this repository](https://github.com/MilosVulikic/SimpleChatApp/commits/master). Note, due to the simplicity of the project, I mostly used master branch.

**Authors**

* **Milos vulikic**

**License**

This is freeware for only learning purpose. If you manage to earn money by using this project, which is impossible , you will gain my sincere admiration.

**Acknowledgments**

* Idea of sharing this code is to share my insight and knowledge gained in the process of learning about sockets and threads
* Some of the concepts I will use on the next project