СОБА ЗА ЋАСКАЊЕ

CHAT ROOM  
Вук Вукашиновић

VIII, ОШ 'Илија Вирчанин', Регионални центар за таленте Београд 1 – Земун

**РЕЗИМЕ**

Овај рад представља апликацију 'Чат соба' - комуникациону апликацију развијену у чистом C# која омогућава реалновременску размену порука између више корисника. Апликација је дизајнирана тако да омогући једноставно повезивање клијент-сервер мрежног окружења, и са својим модуларним приступом лако се надограђује. Рад објашњава функције апликације, начин рада, кориснички интерфејс, као и структуру и значајне линије кода.

**КЉУЧНЕ РЕЧИ:**  
чат соба, комуникација, C#, клијент-сервер архитектура, апликација

**ABSTRACT:**  
This paper introduces the “Chat Room” application – a communication tool developed in pure C# that enables real-time message exchange among multiple users. The application is designed to facilitate simple connection between client and server modules and can easily be upgraded with its modular approach. The paper explains the functionalities, user interface, core code structure, and key components of the application.

**KEYWORDS:**  
chat room, communication, C#, client-server architecture, application

### Увод

Апликација која је предмет овог рада омогућава комуникацију више корисника у реалном времену преко интернеtа. Апликација је развијена унутар C# окружења уз помоћ .NET библиотека. Ово омогућава брзу и стабилну конекцију. Апликација је приближена корисницима израдом корисничког интерфејса унутар Windows Forms. Овај рад је примена наученог знања са часова програмирања и самосталног истраживања.

### Настанак идеје

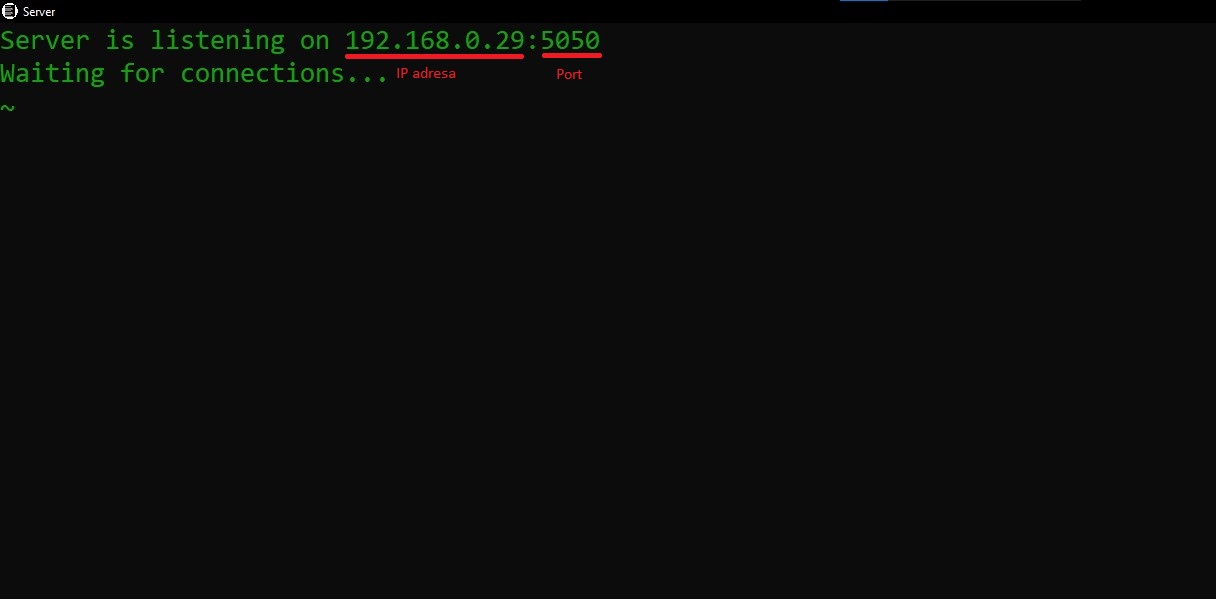
Један дан дописивао сам се са другом, запитао сам се како све то функционише. Како се врши размена порука, фајлова, видео и аудио разговора. Кренуо сам да се распитујем по интернету и да откривам како све то функционише. У мени се развила жеља да покушам да то све рекреирам.

Почео сам тиме што сам направио 'Сервер' и 'Клијент' код који су могли у минималном смислу да комуницирају један између другог. То је све било лимитирано на конзолу што као што можете да замислите није најприлагођеније корисницима. Тако да је на све то морала да се угради UI (коринички интерфејс). То је знатно олакшало интеракцију са апликацијом

### Рад са апликацијом

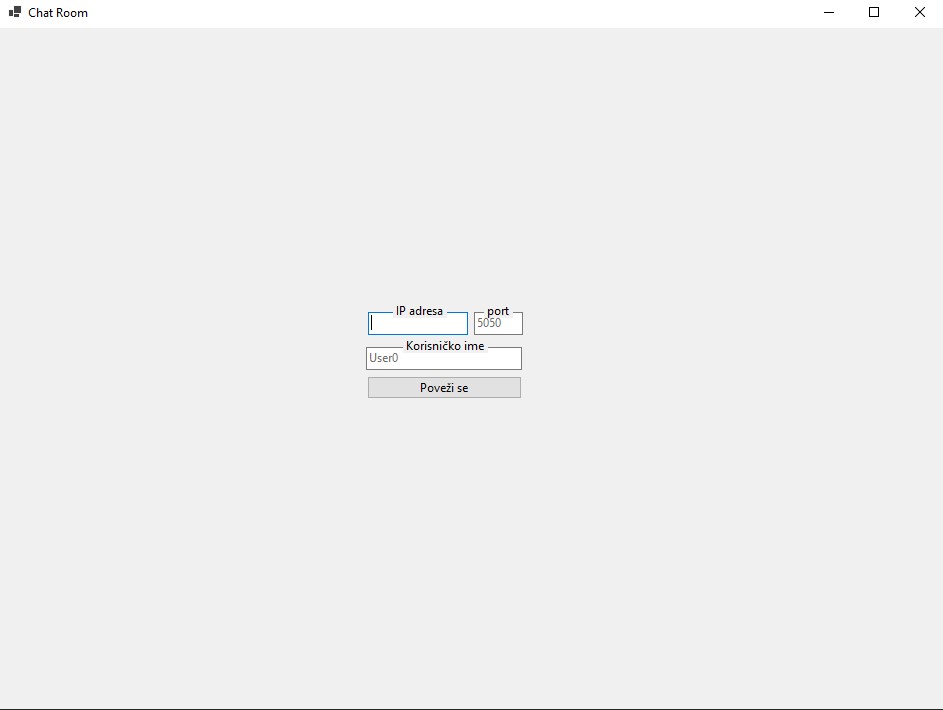
Ово поглавље представља упутство за кориснике

1. За почетак морамо да покренемо апликацију сервера преко које ће сви разговори да пролазе. Када се покрене имаћемо прозор (слика 1.1), исписаће локалну IP адресу и PORT на којој сервер ради, ово ће нам бити потребно да бисмо се повезали. Напомена уколико се у подешавањима рутера не подеси *port forwarding* конекције ће бити могуће унутар једне интернет везе, уколико је *port forwarding* подешен уместо исписане IP адресе мораће да се користи јавна адреса. Један од начина да се она сазна је да посетите [What is my ip address](https://whatismyipaddress.com/).



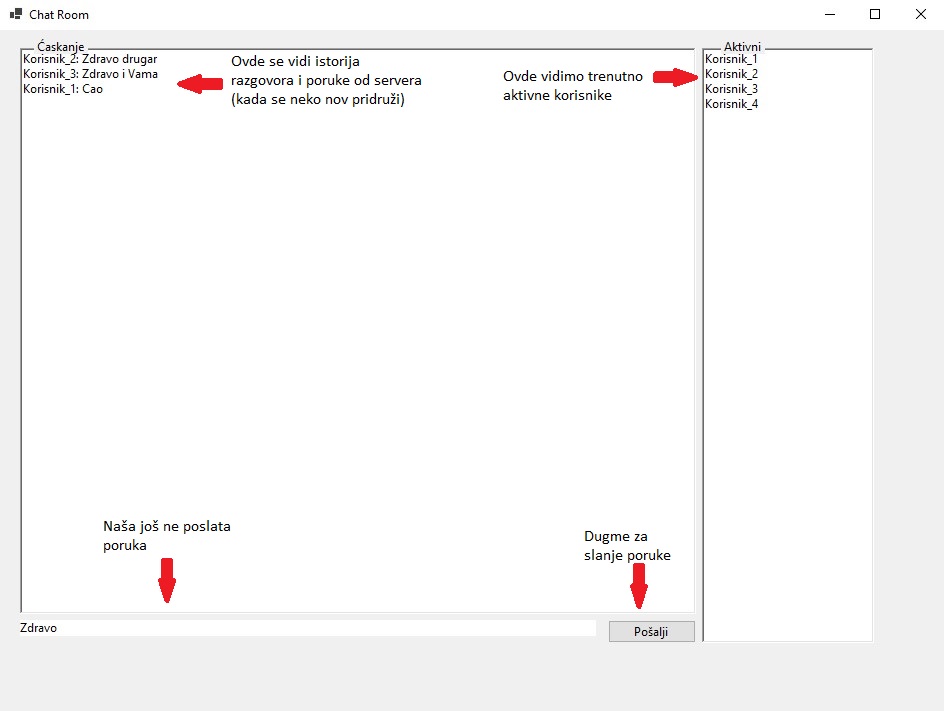
Слика 1.1 Конзола сервера

1. Након тога покрећемо Chat Room апликацију и у њој попуњавамо поља са слике (слика 2.1), IP adresa I PORT су адреса и порт нашег сервера, а у претходном кораку смо видели како да их сазнамо. Корисничко име само говори, уносите име под којим ће вас други људи у ћаскању видети, и кликнемо дугме “Poveži se”.

Слика 2.1 Изглед почетног менија

1. Након повезивања имамо следећи мени (слика 2.2) и уједно и главни мени у коме се врши дописивање. У њему можемо видети разговор, тренутно активне кориснике, и шаљемо поруке

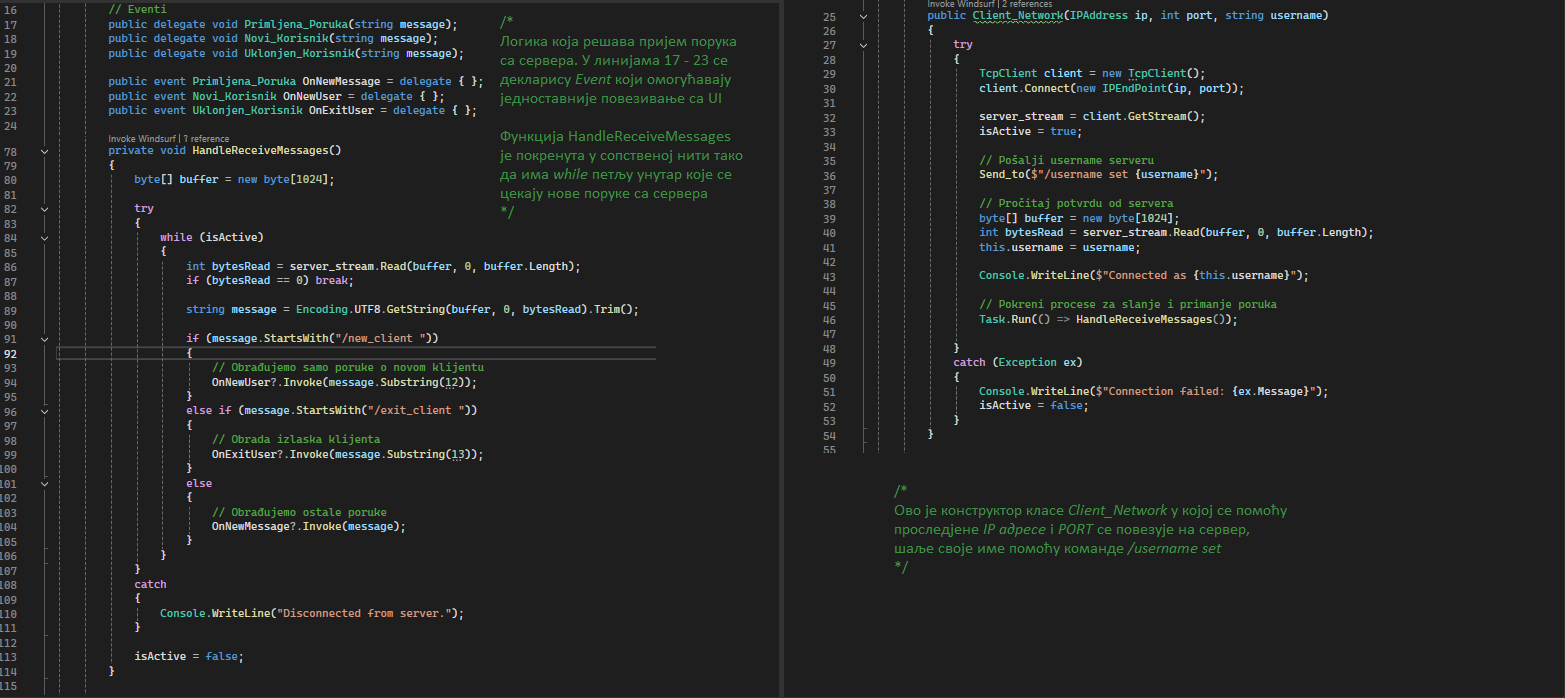
Слика 2.2 Изглед главног менија



### Опис Апликације

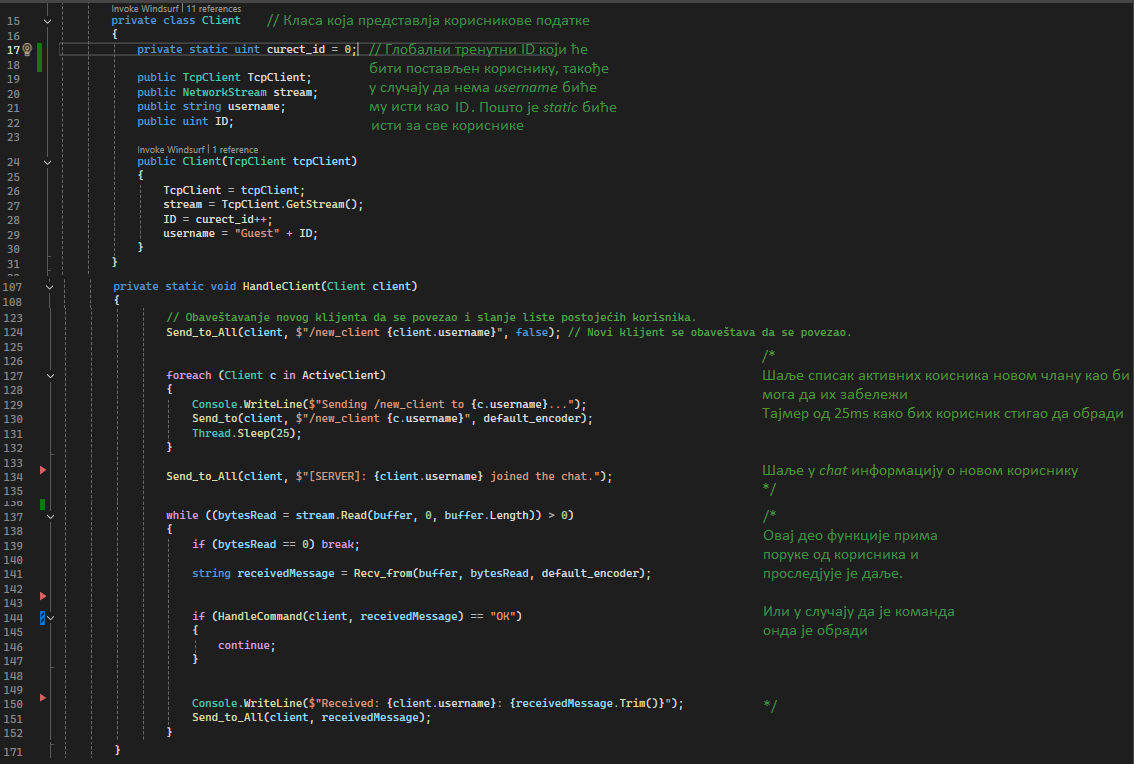
Апликација је развијена у чистом C# уз коришћењ .NET оквира. Примарни фокус је био на дизајнирању сервер-клијент конекције који је накнадно надограђен да користи UI. Апликацију би поделио у два главна дела, сервер и клијент класе. Конекција је базирана на socket и TCP протоколом. Постоје разне библиотеке које су ово имплементирале тако да је лакше да се користи и на бољи начин али ја сам изабрао да радим овако јер се на овај начин много више може научити рад са socket-има.

* Клијент: Ова класа има за задатак да успостави и одржава конекцију са сервером

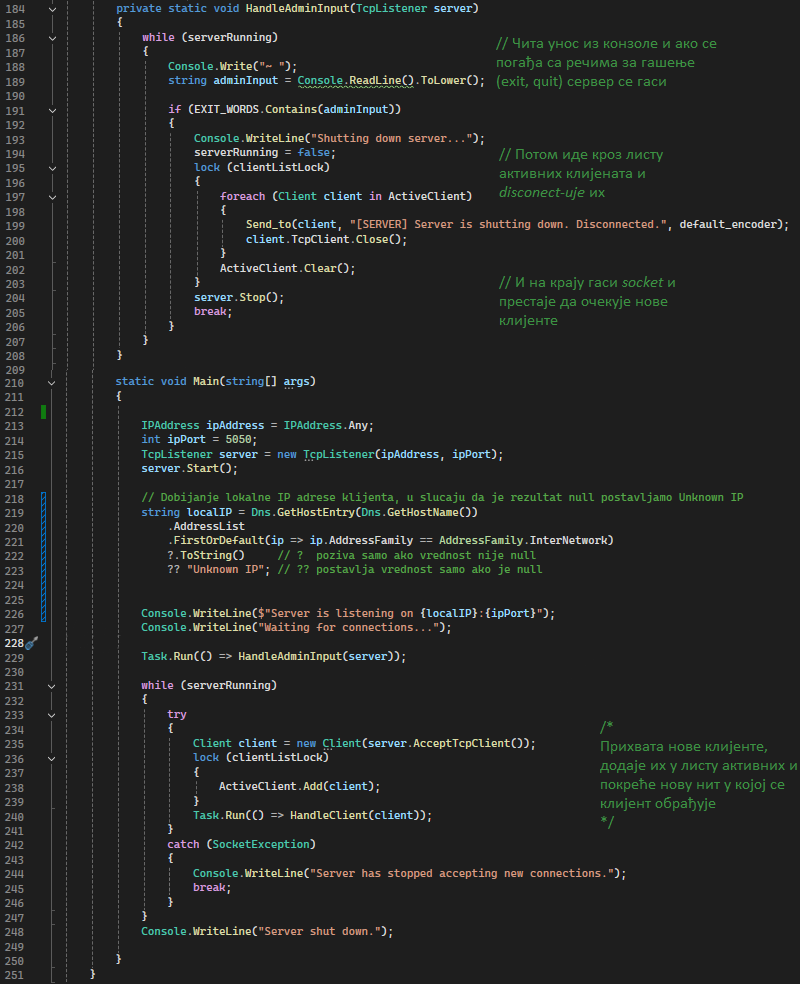


Слика 3.1 Структура Клијента

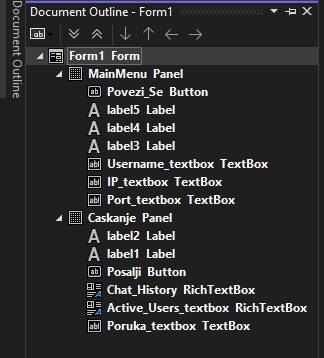
* Сервер: Задужен за обраду клијената прослеђивање порука.  
  на слици 4.1 можемо видети класу која је представља клијенте на серверу (15-31),

и функцију која обрађује везу са корисницима (107-171). На слици 4.2 можемо видети функцију која обрађује кориснички унос у конзоли, и видимо *Main* функцију која има задатак да покрене сервер.

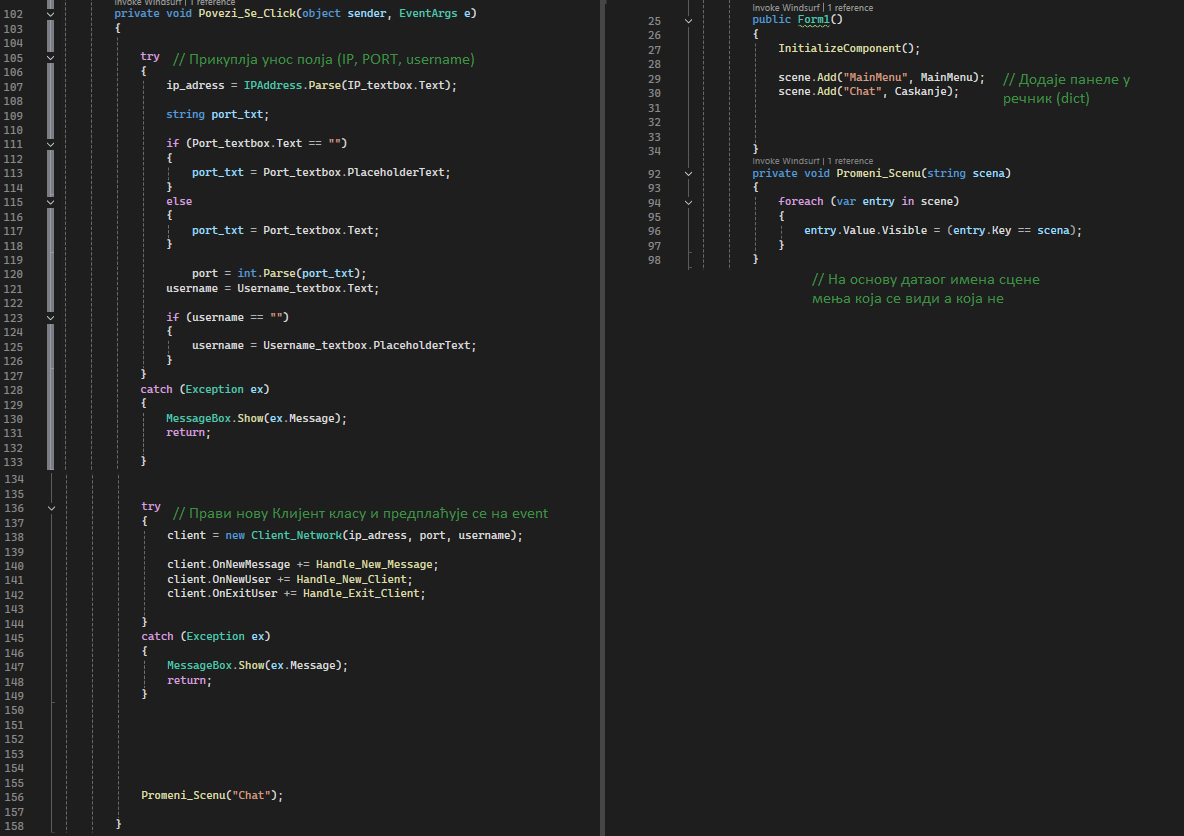
Слика 4.1 Структура класе која чува податке клијената и обрада клијената



Слика 4.2 Обрада корисничког уноса и функција за покретање сервера

* UI: То је само спајање Windows Forms и класе Client\_Network (слике 5.2). Како бих лако могао да мењам 'сцене' додао сам два панела, један је, видљив други није када се мења 'сцена' само се ротирају који је панел видљив, а који није (слика 5.1). **

Слика 5.1 Приказ WF дизајна



Слика 5.2 Функција за повезивање на сервер и Промена 'сцене'

**ЗАКЉУЧАК**

Апликација 'Чат соба' је развијена са циљем истраживања методологије за комуникацију у реалном времену преко мреже. Пројекат је пружио практично искуство у раду са мрежним протоколима, сокет програмирању, и изради графичко корисничког интерфејса.

Апликација је врло забавна и функционална, нуди разна поља за усавршавање, попут оптимизација и безбедности.

Уколико сте заинтересовани за цео код, и да пробате апликацију можете то урадити на GitHub страници: [Chat Room](https://github.com/Vukile2801w/Chat_Room)

### Захвалница

Захваљујем се мојим наставницима Рељи Ћурчин и Дијани Стефановић који су ми објаснили било које недоумице.

**Литература**

1. Званична документација за .NET: [docs.microsoft.com](https://docs.microsoft.com/)