|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  | **УНИВЕРЗИТЕТ У НОВОМ САДУ**  **ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА** |  |  |  | | --- | | **УНИВЕРЗИТЕТ У НОВОМ САДУ**  **ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА**  **НОВИ САД**  **Департман за рачунарство и аутоматику**  **Одсек за рачунарску технику и рачунарске комуникације**  **ИСПИТНИ РАД**  **Кандидат: Огњен Стојисављевић**  **Број индекса: РА155-2019**  **Предмет: Међурачунарске комуникације и рачунарске мреже 1**  **Тема рада: ПОП3 Сервер**  **Ментор рада: Немања Раковић**  **Нови Сад, Јануар 2023.** | |  | |

Садржај

[1.Задатак 3](#_Toc124865511)

[2. Концепт решења 4](#_Toc124865512)

[3.Опис решења 5](#_Toc124865513)

[3.1.SDL дијаграм 5](#_Toc124865514)

[3.1.MSC дијаграм 9](#_Toc124865515)

[4.Тестирање 11](#_Toc124865516)

[5.Закључак 14](#_Toc124865517)

[6.Литература 14](#_Toc124865518)

# 1.Задатак

Коришћењем „Microsoft Visual Studio C++“ развојног окружења реализовати пример апликације која имплементира функционалност ПОП3 мејл сервера.

Апликација очекује захтеве за успоставом везе на унапред задатом ТЦП порту. Апликација такође треба да омогући да се неким корисницима забрани приступ серверу. Корисници се идентификују мејл адресом која се шаље преко ПОП3 поруке “USER”.  
Апликација тестирати коришћењем дате ПОП3 мејл клијентске апликације(модификовати исту на начин дат испод\*). Промену стања серверске стране реализовати коришћењем језгра комуникационе програмске подршке.

(\*)При слању података ПОП3 клијент енкриптује садржај

пакета Вижнер шифром. При пријему ПОП3 сервер врши одговарајућу декрипцију и обрнуто. Кључ одредити самостално. Оригинална порука се састоји само од слова енглеског алфабета. Специјалне знаке у окивру команди ПОП3 протокола попут размака, +, - и слично је потребно енкриптовати словима енглеског алфабета.

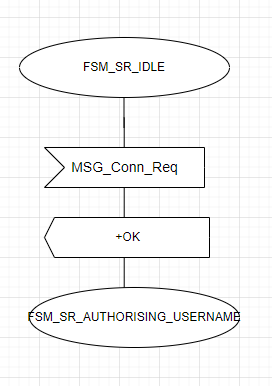
# 2. Концепт решења

* Сервер је имплементиран као један FSM аутомат
* Коришћена је WinSock библиотека за размену порука преко мреже.
* На серверској страни постоји нит која ослушкује поруке које наилазе од клијената
* Када обради клијентску поруку аутомат сервера мења своје стање у зависности од одговарајуће поруке
* Поруке које шаљу клијент и сервер енкриптоване су Вижнеровом шифром, такође и дешифроване при пријему

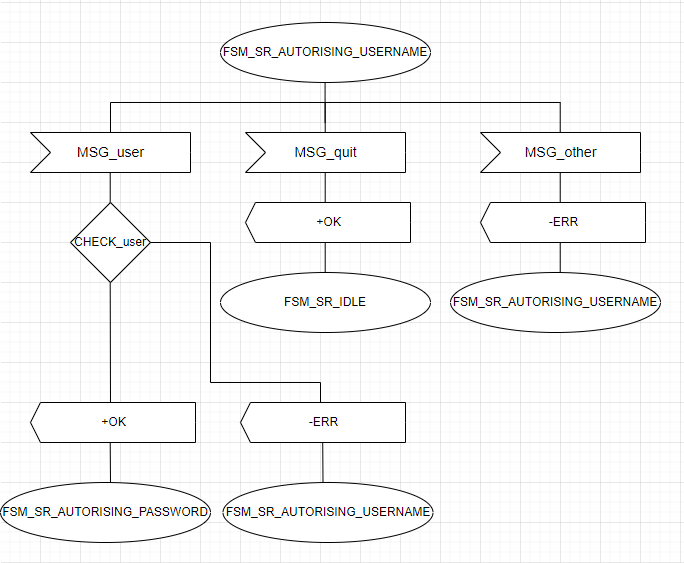
# 3.Опис решења

## 3.1.SDL дијаграм

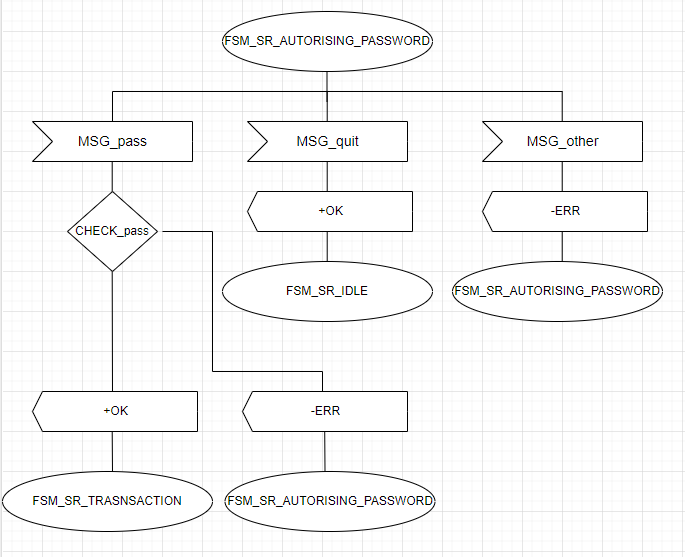
1) Почетно стање



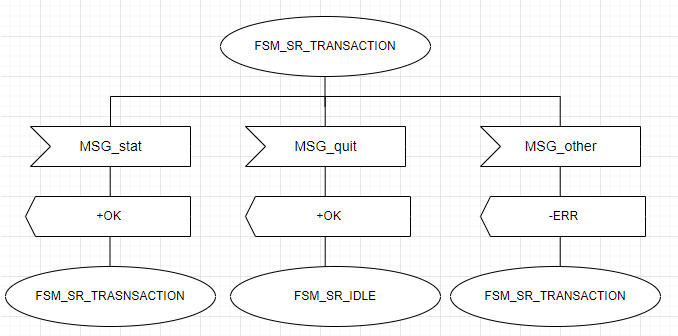
2)Уношење корисничког имена



3)Уношење шифре



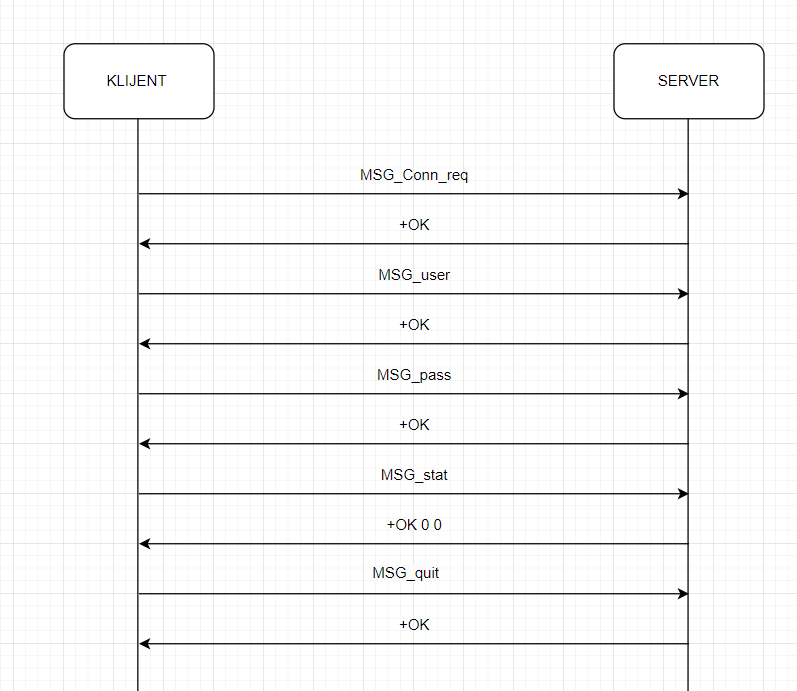
4) Команде

****

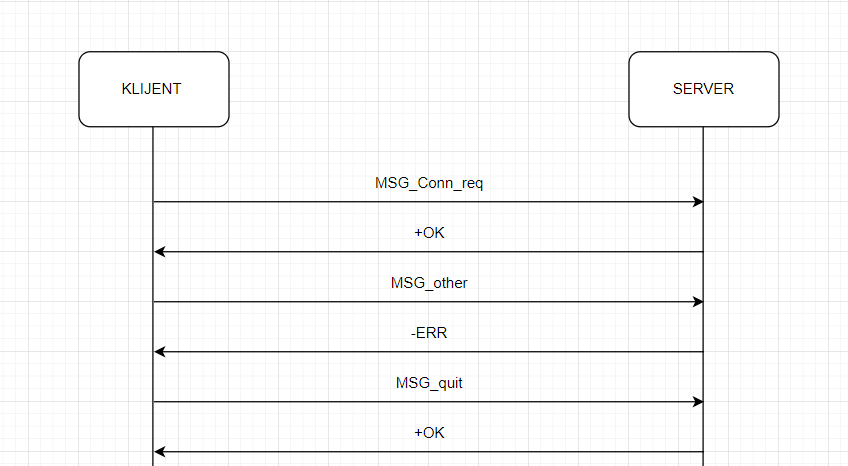
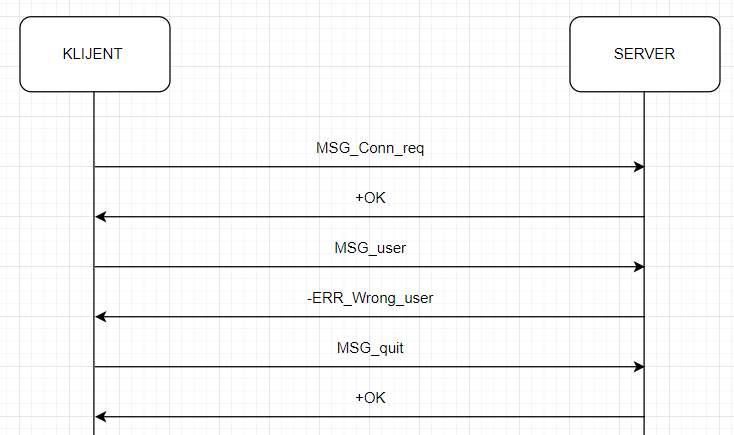
**-**Напомена: Ово је само пример серверске апликације, где је могуће додати још команди ПОП3 протокола у стање трансакције.

## 3.1.MSC дијаграм

1)Пример успешне комуникације



2)Примери неуспешне комуникације

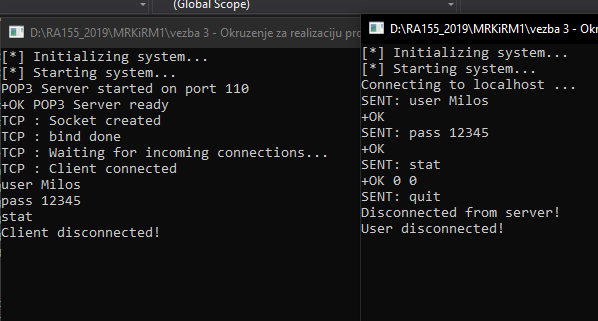


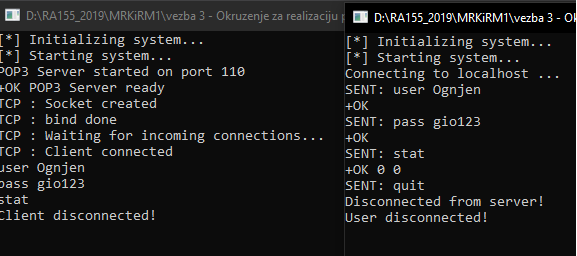
# 4.Тестирање

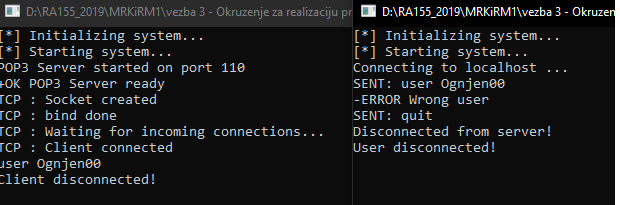
- У следећим сликама приложено је различите врсте стања аутомата у којим може да се нађе, такође и пријављивање више корисника.

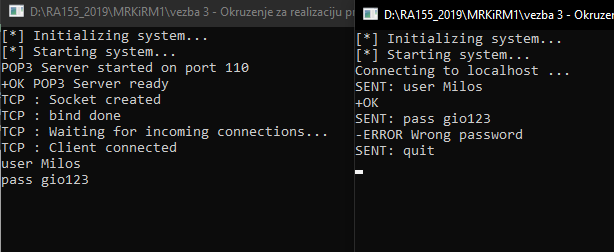
- На конзолама су исписивани декриптовани пакети, а криптоване поруке односно пакете посматрали смо помоћу WireShark.

1) Успешна комуникација

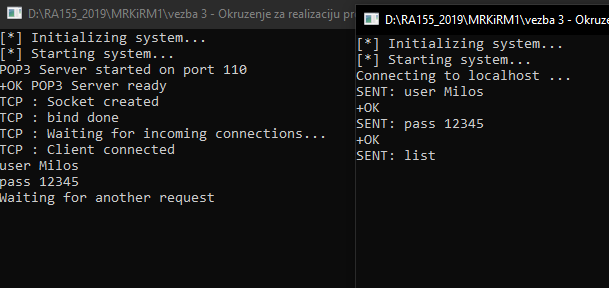




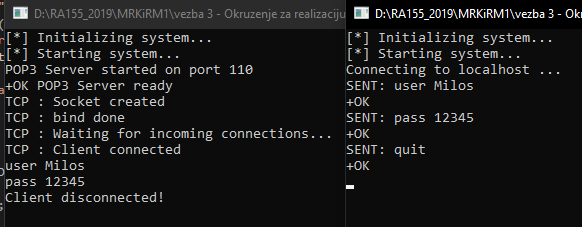
2)Погрешно корисничко име или шифра

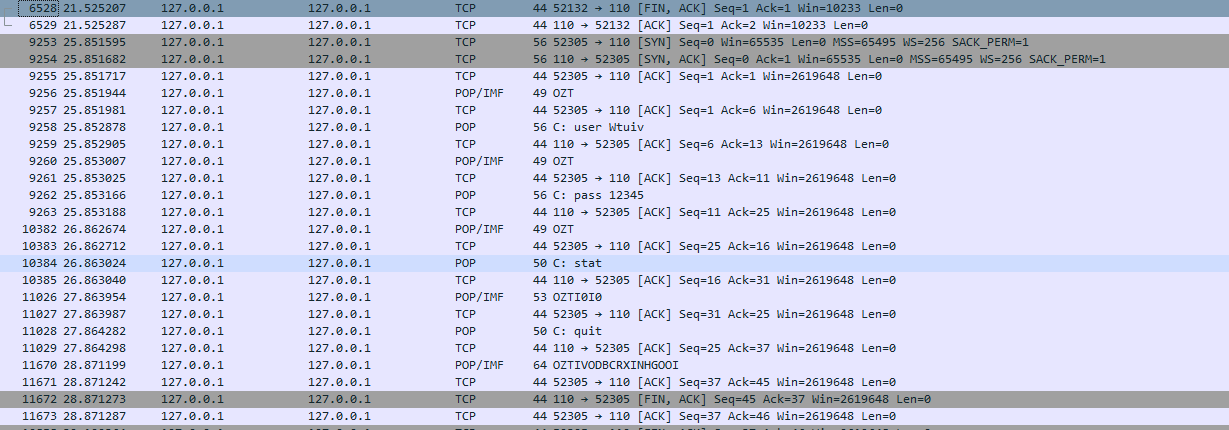


3) Неподржана команда



4)Излазак из апликације



5) Енкриптован садржај 

# 5.Закључак

- Направљена серверска апликација је доста ограниченог типа и у питању је само пример који био могао да се доради.

- Додатне измене које би могле да се додају су више команди, SQLite база података за мејлове, остатак ПОП3 сервер команди, база за регистровање корисника...

# 6.Литература

- ПОП3 документација

- Материјали са вежби из предмета МРКиРМ1