**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**
**Сумський державний університет**
Факультет електроніки та інформаційних технологій
Кафедра комп'ютерних наук

«До захисту допущено»
В.о. завідувача кафедри
_____ Ігор ШЕЛЕХОВ
(підпис)
18 грудня 2023 р.

# КВАЛІФІКАЦІЙНА РОБОТА

**на здобуття освітнього ступеня магістр**

зі спеціальності 122 - Комп'ютерних наук,
освітньо-наукової програми «Інформатика»
на тему: «Інформаційно-комунікаційна технологія налаштування віртуальної захищеної мережі між віддаленими офісами»
здобувача групи ІН.м-26 Маслова Івана Олександровича

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

_____ Іван МАСЛОВ
(підпис)

Керівник,
старший викладач,
кандидат фізико-математичних наук Дмитро ВЕЛИКОДНИЙ _____
(підпис)

**Суми – 2023**

**Сумський державний університет**
Факультет електроніки та інформаційних технологій
Кафедра комп'ютерних наук

«Затверджую»
В.о. завідувача кафедри
_____ Ігор ШЕЛЕХОВ
(підпис)

# ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

**на здобуття освітнього ступеня магістр**

зі спеціальності 122 - Комп'ютерних наук, освітньо-наукової програми «Інформатика»
здобувача групи ІН.м-26 Маслова Івана Олександровича

1. Тема роботи: «Інформаційна технологія прогнозування курсу валют»
затверджую наказом по СумДУ від «06» грудня 2023 р. № 1412-VI
2. Термін здачі здобувачем кваліфікаційної роботи до 18 грудня 2023 року
3. Вхідні дані до кваліфікаційної роботи Захищені інфокомунікаційні мережі є основою функціонування сучасного цифрового світу, відповідно дослідження в цьому напрямі є актуальними та мають значну практичну цінність
4. Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити) *1) Аналіз проблеми предметної області, постановка й формування завдань дослідження. 2) Огляд технологій, що використовуються налаштування захищеної мережі. 3) Розробка інтелектуальної системи налаштування захищеної мережі між офісами 4) Аналіз результатів.*
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
6. Консультанти до проекту (роботи), із значенням розділів проекту, що стосується їх

| Розділ | Консультант | Підпис, дата | |
|---|---|---|---|
| | | Завдання видав | Завдання прийняв |
| | | | |

7. Дата видачі завдання «____» _____ 20 ___ р.

Завдання прийняв до виконання _____ Керівник _____
(підпис) (підпис)

# КАЛЕНДАРНИЙ ПЛАН

| № п/п | Назва етапів кваліфікаційної роботи | Термін виконання | Примітка |
|---|---|---|---|
| *1* | *Аналіз проблеми предметної області, постановка й формування завдань дослідження* | 08.12.2023 | |
| *2* | *Огляд технологій, що використовуються для створення захищених мереж* | 09.12.2023 | |
| *3* | *Розробка інтелектуальної системи з налаштування віртуальної захищеної мережі між віддаленими офісами* | 10.12.2023 | |
| *4* | *Аналіз отриманих результатів* | 15.12.2023 | |
| *5* | *Оформлення пояснювальної записки до кваліфікаційної роботи* | 16.12.2023 | |

Здобувач вищої освіти _____ Керівник _____
(підпис) (підпис)

# НОТАЦІЯ

**Записка:** 48 стор., 23 рис., 1 додаток, 21 джерело.

**Обґрунтування актуальності теми роботи** – Тема кваліфікаційної роботи є актуальною, оскільки присвячена розвитоку мережевих технологій та зростання потреби у захищеному та ефективному обміні даними роблять важливим вивчення та удосконалення технологій VPN та MPLS.

**Об’єкт дослідження** — передача інформації у захищених інфокомунікаційних мережах.

**Мета роботи** — розробка та оптимізація інтуїтивно зрозумілого графічного інтерфейсу для налаштування MPLS VPN, який полегшує керування мережевими налаштуваннями..

**Методи дослідження** — аналіз існуючих рішень, проектування інтерфейсу, програмування, тестування з використанням методів якісного та кількісного аналізу.

**Результати** — розроблено повнофункціональний графічний інтерфейс для VPN на базі MPLS, який демонструє поліпшення в продуктивності та зручності використання в порівнянні з існуючими системами.

ІНФОРМАЦІЙНА СИСТЕМА, НАЛАШТУВАННЯ МЕРЕЖІ, ЗАХИЩЕНИЙ ЗВ'ЯЗОК, C#, MPLS, VPN.

# ЗМІСТ

# ВСТУП

**Актуальність теми** — Дослідження роботи полягає у необхідності створення ефективних та безпечних мережевих рішень в умовах постійного збільшення обсягів даних та вимог до швидкості їх обробки. Використання технологій віртуальних приватних мереж (VPN) з інтеграцією MPLS (Multiprotocol Label Switching) стає не лише трендом, але й необхідністю для забезпечення високої продуктивності корпоративних мереж.

**Об'єкт дослідження**— процес конфігурації та управління віртуальними приватними мережами в корпоративних мережах з використанням MPLS.

**Предмет дослідження**—графічний інтерфейс користувача для налаштування VPN, який повинен бути інтуїтивно зрозумілим та забезпечувати швидке і ефективне управління мережевими налаштуваннями.

**Гіпотеза**— роботи полягає в тому, що розробка оптимізованого графічного інтерфейсу здатна значно поліпшити процес налаштування та моніторингу VPN засобами MPLS, зменшуючи час налаштування та ризик помилок.

**Наукова новизна роботи**— виявляється у розробці уніфікованого рішення, яке інтегрує новітні методики в галузі UX/UI дизайну, забезпечуючи високу ефективність управління мережевими потоками.

**Структура**— роботи включає вступ, теоретичний огляд, опис методології дослідження, практичну частину з розробкою інтерфейсу, аналіз отриманих результатів, висновки та пропозиції щодо подальших досліджень в цій області. Робота містить ілюстративні матеріали та додатки, які демонструють процес розробки та функціональні можливості створеного продукту.

# 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

## 1.1 Теоретичний огляд MPLS та VPN

Мережеві технології постійно розвиваються, щоб задовольнити зростаючі потреби в швидкості, надійності та безпеки передачі даних. Протягом останніх десятиліть, дві значущі інновації в області мережевих технологій, MPLS (Multiprotocol Label Switching) та VPN (Virtual Private Network), суттєво змінили пейзаж передачі даних, пропонуючи нові можливості та рішення для корпоративних та приватних мереж.

MPLS виник на початку 1990-х років як відповідь на потребу в більш ефективній маршрутизації трафіку у складних мережах. Ця технологія відрізняється від традиційної маршрутизації, заснованої на адресах, використовуючи короткі мітки для спрямування пакетів через мережу. Ці мітки дозволяють маршрутизаторам швидше обробляти трафік, оскільки вони можуть перемикати пакети на основі простого пошуку міток, не аналізуючи кожен раз всю інформацію заголовка пакета. Завдяки MPLS, провайдери змогли створювати віртуальні мережі зі збереженням якості сервісу (QoS) та забезпеченням більшої гнучкості і масштабованості в порівнянні з традиційними IP-мережами.

Технологія VPN, з іншого боку, почала широко використовуватися в кінці 1990-х років, коли інтернет став основним засобом комунікацій, і виникла потреба у захищеному з'єднанні між віддаленими користувачами та корпоративними мережами. VPN створює зашифрований "тунель" через публічний інтернет, який дозволяє безпечно передавати конфіденційні дані, надаючи віддаленим користувачам можливість працювати так, ніби вони знаходяться в локальній мережі організації. З розвитком криптографічних технологій та алгоритмів, VPN забезпечували все більш надійний рівень безпеки, що було особливо важливо для корпоративного сектору.

Сполучення MPLS та VPN технологій дало змогу організаціям налаштовувати свої мережеві рішення з неперевершеною гнучкістю та

контролем. MPLS пропонує поліпшену продуктивність та надійність для VPN-з'єднань, гарантуючи високу якість сервісу для чутливих до затримок додатків, таких як голосове та відео зв'язок, а також для критично важливих ділових додатків. З часом, MPLS VPN стали стандартом для багатьох міжнародних компаній, що потребують надійних і безпечних мережевих з'єднань для своїх віддалених офісів і мобільних співробітників.

Разом, MPLS і VPN відіграють критично важливу роль у сучасному мережевому світі, забезпечуючи основу для безперервної комунікації та ділових операцій у всьому світі. Вони продовжують еволюціонувати, інтегруючи новітні технології, щоб зустрічати виклики, які ставить перед ними постійно змінюваний ландшафт інформаційних технологій.

Multiprotocol Label Switching (MPLS) є високоефективною технікою мережевої інженерії, використовуваною в передачі даних у телекомунікаційних мережах. Ця технологія впроваджена для оптимізації мережевого трафіку та підвищення його швидкості, забезпечуючи більш гнучке управління даними у порівнянні з традиційними IP-мережами [1]. В MPLS, дані передаються через встановлені шляхи на основі коротких міток, що пришвидшує процес маршрутизації. Ключовою особливістю MPLS є використання міток (лейблів) для пересилання пакетів. На вході пакети в MPLS-мережу, отримує мітку, що визначає подальший маршрут через мережу. Ці мітки - це короткі, фіксованої довжини ідентифікатори, які вказують на вихідний інтерфейс і наступний стрибок (hop) у мережі. Завдяки цьому, міжмережеві комутатори (MPLS switches), замість аналізу IP-адрес, просто перевіряють мітку та швидко відправляють та пересилають пакет до наступного вузла.

Цей механізм перемикання міток допомагає мережевим операторам більш ефективно маршрутизувати дані, що призводить до зменшення затримок і більш ефективного використання мережевих ресурсів.Одним з основних застосувань MPLS є створення віртуальних приватних мереж (VPN) [2]. MPLS можна використовувати для створення приватної мережі між двома або більше офісами,

навіть якщо ці офіси географічно віддалені один від одного. Наприклад, користувачі, які володіють серверами або компаніями з офісами в різних містах або навіть у всьому світі, можуть використовувати MPLS VPN для безпечного підключення цих офісів або серверів одночасно, дозволяючи користувачам ділитися та отримувати доступ до спільних ресурсів і даних. Постачальники Інтернет-послуг (ISP) оптимізують і ідеально створюють свою мережу, а також підвищують якість клієнтів і послуг. Використовуючи MPLS для маршрутизації трафіку між усіма мережами, провайдери можуть визначати пріоритетність певних типів трафіку (наприклад, голосового або відео), забезпечуючи його доставку з мінімальною затримкою. Це дозволяє зменшити затримку та підвищити загальну якість обслуговування для кінцевих користувачів.

Таким чином, MPLS є потужною мережевою технологією, яка активно використовується для створення приватних мереж, оптимізації доставки трафіку та підвищення продуктивності та продуктивності даних, голосу та інших програм. Використовуючи перемикання міток для маршрутизації трафіку [3], MPLS забезпечує швидшу та ефективнішу передачу даних і оптимізує мережеві ресурси.

Головними перевагами MPLS :

1) Швидкість: Оскільки мітки дозволяють швидше обробляти пакети, MPLS забезпечує високу швидкість обробки трафіку порівняно з традиційною IP-маршрутизацією;

2) Гнучкість: MPLS підтримує різні типи трафіку та може бути легко інтегрований у багато існуючих мережевих технологій;

3) Якість обслуговування (QoS): MPLS дозволяє встановлювати пріоритети для різних видів трафіку, забезпечуючи краще управління пропускною спроможністю і затримкою;

4) Масштабованість: MPLS підходить для великих мереж, оскільки управління маршрутами стає простішим та ефективнішим;

5) Покращення QoS та надійності [4];

6) Підтримка створення віртуальних приватних мереж (VPN);

7) Протокол-агностичний: сумісний з різними мережевими протоколами;

8) Зменшення затримки і підвищення продуктивності;

9) Масштабованість та гнучкість.

Математично, MPLS можна розглядати через оптимізаційні моделі та алгоритми. Основна задача полягає у визначенні оптимальних шляхів для пакетів, що мінімізують загальну затримку та максимізують пропускну спроможність мережі. Це може включати використання алгоритмів таких, як Dijkstra або Bellman-Ford, для визначення найкоротших шляхів у мережі з урахуванням різних параметрів, таких як пропускна спроможність каналів та часові характеристики.

### 1.2 Quality of Service в MPLS

QoS розшифровується як Quality of Service (якість обслуговування) [5] і означає здатність мережі встановлювати пріоритети та керувати різними типами трафіку залежно від їхньої важливості або вимог. Вона використовується для забезпечення того, щоб кожен тип трафіку отримував необхідні ресурси і забезпечував постійний рівень обслуговування, навіть у періоди перевантаження.

У контексті MPLS QoS використовується для оптимізації доставки трафіку шляхом призначення відповідних рівнів пріоритету і ресурсів для різних класів обслуговування. MPLS дає змогу створювати в мережі кілька маршрутів із комутацією міток (LSP), кожен із яких має свої вимоги до QoS.

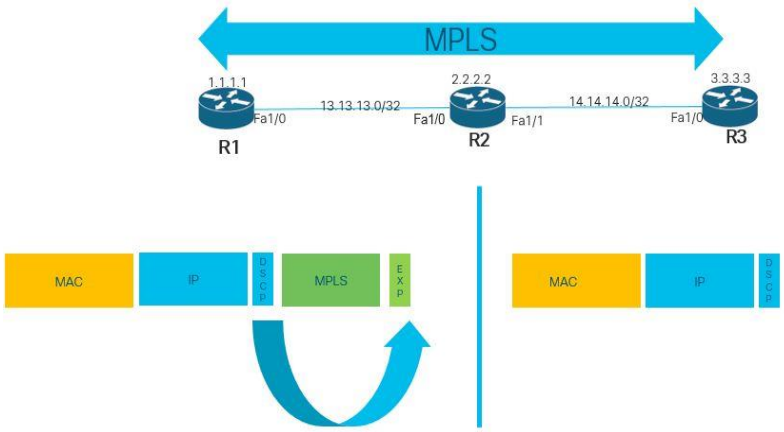


Рисунок 1.1 – Накладення міток MPLS посилання[6]

MPLS використовує різні механізми для реалізації QoS, зокрема інженерію трафіку (TE) і диференційовані послуги (DiffServ). Traffic Engineering дає змогу операторам мережі контролювати й керувати маршрутом, за яким кожен LSP проходить через мережу, що дає їм змогу розподіляти ресурси та обирати найбільш підходящий шлях на основі вимог QoS.Такі послуги, з іншого боку, дають змогу присвоювати пакетам певні мітки залежно від їхнього класу і вимог QoS. Це дає змогу мережі ідентифікувати пакети й обробляти їх відповідним чином, надаючи необхідні ресурси та встановлюючи пріоритети залежно від бажаного рівня QoS.

Завдяки QoS в MPLS оператори мереж можуть визначати пріоритети трафіку залежно від його характеристик, як-от потокове передавання в реальному часі, голосові виклики або передавання даних. Це гарантує, що критично важливий або чутливий до часу трафік, що вимагає низької затримки або високої пропускної здатності, матиме пріоритет перед некритичним або менш критичним трафіком [7].

Наприклад, у мережі, де відбуваються як голосові виклики, так і передача файлів, QoS може використовуватися для забезпечення більш високого пріоритету голосових викликів порівняно з передачею файлів. Це запобігає затримкам і обривам голосових викликів навіть у разі перевантаження мережі.

Таким чином, QoS в MPLS дає змогу визначати пріоритети й керувати

трафіком залежно від його важливості та вимог. Це гарантує, що різні типи трафіку отримають необхідні ресурси, і дає змогу мережі забезпечувати постійний рівень обслуговування навіть за різних умов роботи мережі, як на рисунку1.2. Мережа MPLS використовує мітки шляху замість мережевих адрес для спрямування трафіку. Ці мітки включають інформацію про те, який шлях комутації міток слід використовувати, щоб переконатися, що пакет потрапляє туди, куди він має потрапити.

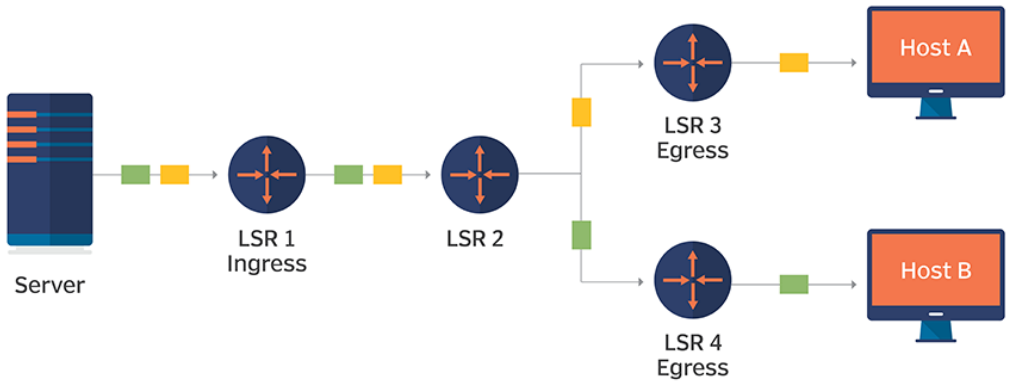


Рисунок 1.2 – Базова мережа MPLS[7].

Приклади застосування MPLS[8]:

1) Транспортування різних типів трафіку: MPLS може одночасно переносити голосовий, відео та дані, забезпечуючи високу якість кожного типу трафіку;

2) Віртуальні приватні мережі (VPN): MPLS часто використовується для створення віртуальних приватних мереж, де мітки використовуються для ізоляції трафіку різних користувачів або груп користувачів;

3) Управління трафіком: Завдяки можливості управління пріоритетами трафіку, MPLS дозволяє ефективно розподіляти навантаження у мережі, запобігаючи перевантаженням;

4) Застосування MPLS у VPN: MPLS [9] може використовуватися для створення ефективних, швидких та надійних VPN-з'єднань між віддаленими офісами;

5) Можливе використання MPLS для поліпшення якості обслуговування (QoS) в VPN, забезпечуючи високу швидкість та надійність з'єднання;

MPLS є потужним інструментом у мережевій інженерії, який забезпечує високу швидкість, гнучкість, якість обслуговування та масштабованість. Його математичне підгрунтя та оптимізаційні моделі роблять його ідеальним вибором для складних мережевих задач та інтеграції з іншими технологіями, такими як VPN

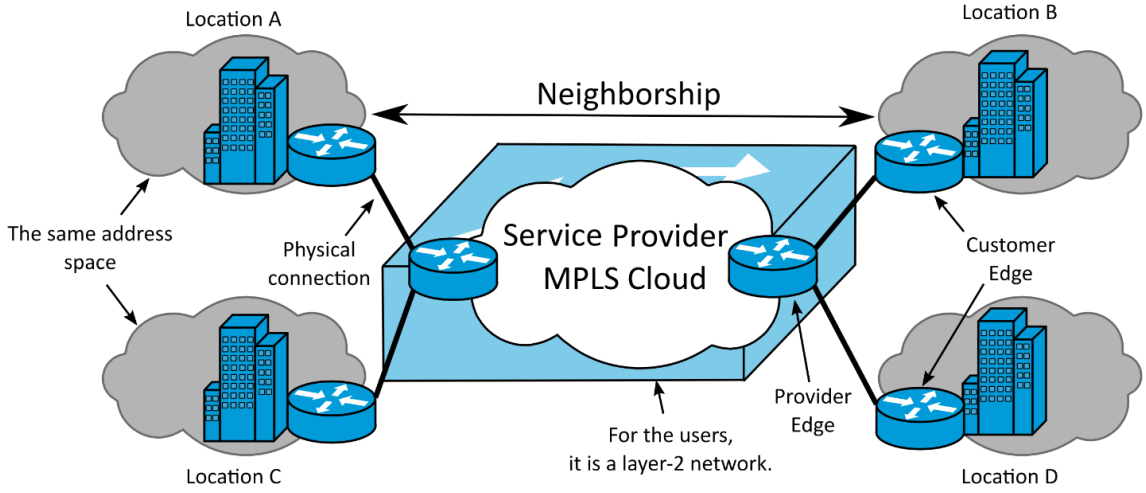


Рисунок 1.3 – Логічне подання MPLS VPN рівня 2[10]

### 1.3 Віртуальна приватна мережа VPN

Віртуальна приватна мережа (VPN)— це технологія, яка дозволяє створити безпечний і зашифрований канал зв'язку між двома або кількома точками в Інтернеті, які представляють віртуальну приватну мережу, забезпечуючи тим самим безпеку вашої діяльності в Інтернеті Конфіденційність і безпеку, і є частиною програмного забезпечення, яке створює безпечне з'єднання між пристроєм та Інтернетом. VPN можна використовувати в корпоративному середовищі та приватних користувачів для захисту даних під час передачі через загальнодоступні мережі. Коли програма VPN працює, вона створює безпечне

з’єднання (тунель) між вашим пристроєм і видаленим сервером VPN, а потім перенаправляє мережевий трафік на веб-сайт або програму, до якої ви намагаєтеся отримати доступ.

Маршрутизація трафіку через захищений сервер VPN не дозволяє стороннім особам, як-от Інтернет-провайдер, відстежувати та реєструвати ваші в Інтернеті. Це також запобігає перехопленню хакерами передачі даних в Інтернеті.

3

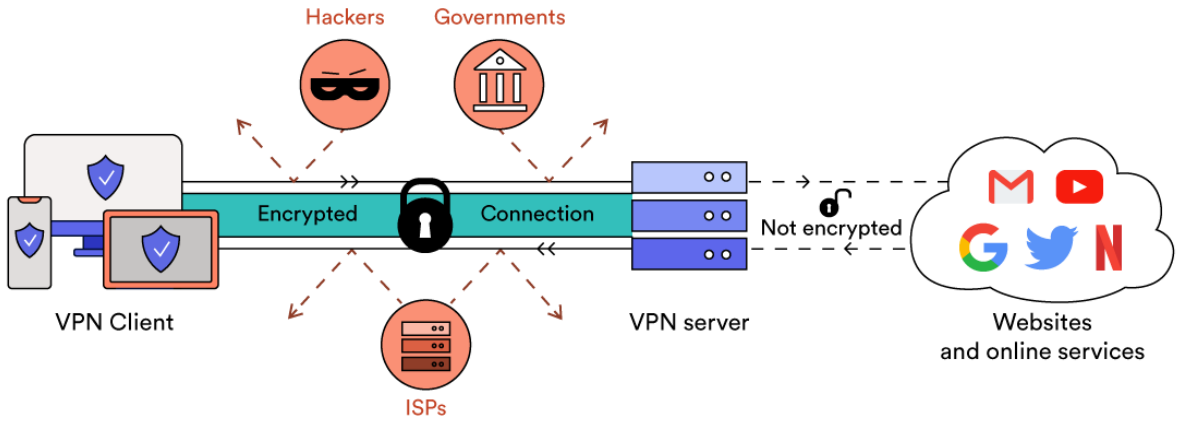


Рисунок 1.4 – Принципи Роботи VPN[11]

Створення Зашифрованого Тунелю VPN створює зашифрований тунель для передачі даних між користувачем та сервером. Цей тунель забезпечує, що всі дані, що передаються, є недоступними для сторонніх осіб, які можуть спробувати перехопити інформацію. Завдяки шифруванню, інформація стає незрозумілою для будь-кого, хто не має відповідного ключа для розшифровки.

Під час використання VPN IP-адреса користувача замінюється на IP-адресу сервера VPN. Це забезпечує справжнє відвідування користувача, забезпечуючи додаткову анонімність. Таким чином, користувачі можуть обійти геообмеження та забезпечити свою конфіденційність в Інтернеті. VPN пропонує різноманітні протоколи шифрування для забезпечення безпеки даних, наприклад IPsec,

OpenVPN, L2TP, PPTP тощо. Ці протоколи мають свої особливості щодо рівня безпеки, швидкості передачі даних і сумісності з відкритими пристроями та мережевими конфігураціями. тунель. Тунелювання — це процес упаковки даних користувача для передачі через VPN. Цей процес додатково ізолює дані користувача від інших даних, що передаються через мережу, забезпечуючи до рівня безпеки та конфіденційності.

### 1.4 Застосування VPN та технічні аспекти

Віртуальні приватні мережі (VPN) мають різноманітне застосування як в корпоративній, так і в індивідуальній сферах, а також у повсякденному житті. VPN використовує два основні типи шифрування: симетричне та асиметричне. Асиметричне шифрування включає використання пари ключів - публічного та приватного, тоді як симетричне шифрування використовує один спільний ключ для шифрування та розшифровки даних.

Протоколи Аутентифікації Аутентифікація є критично важливою для забезпечення того, що тільки авторизовані користувачі мають доступ до VPN-сервера. Використовуються різні протоколи аутентифікації, включаючи MS-CHAP, EAP та цифрові сертифікати, для забезпечення безпеки з'єднання.

Далі буде декілька детальних прикладів використання VPN як в корпоративному середовищі в безпеці так і звичайному повсякденному житті. У корпоративному середовищі VPN використовується для створення безпечного з'єднання між віддаленими співробітниками та корпоративною мережею. Це дозволяє співробітникам віддалено доступатися до внутрішніх ресурсів компанії, немов вони фізично знаходяться в офісі. Компанії часто створюють VPN для забезпечення безпечного віддаленого доступу для своїх співробітників. Це дозволяє співробітникам підключатися до внутрішньої мережі компанії з будь-якого місця, забезпечуючи конфіденційність даних і надаючи доступ до конфіденційних ресурсів.

Підключення філій: VPN використовуються для безпечного з'єднання

різних філій компанії через Інтернет. Це допомагає створити єдину мережеву інфраструктуру, що забезпечує безперебійну комунікацію, обмін даними та спільну роботу між різними офісами.

Інтеграція з бізнес-партнерами: Компанії часто співпрацюють із зовнішніми бізнес-партнерами, постачальниками або підрядниками. VPN забезпечують безпечне з'єднання між корпоративною мережею та цими зовнішніми організаціями, гарантуючи контрольований доступ до певних ресурсів або систем.

Окремі користувачі потребують захисту VPN для своєї конфіденційності в Інтернеті, особливо під час підключення до незахищених публічних мереж Wi-Fi. Крім того, VPN дозволяє обійти геообмеження та цензуру, забезпечуючи більш вільний доступ до інтернет-ресурсів. Віддалена робота: VPN все частіше використовують людей, які працюють віддалено або часто подорожчають. Підключившись до VPN, вони зможуть безпечно зберегти доступ до мережевих або хмарних програм своєї компанії, зберігаючи конфіденційність і безпеку своїх даних.

Конфіденційність в Інтернеті: мережі VPN популярні серед людей, які прагнуть підвищити свою конфіденційність в Інтернеті. Шляхом шифрування інтернет-трафіку та маскування вашої IP-адреси VPN забезпечує анонімність і запобігає виявленню або відстеженню даних. Вони особливо корисні під час доступу до публічних мереж Wi-Fi. Георозблокування: VPN можна використовувати для обходу геообмежень, розміщених веб-сайтів або потокових служб.

Підключившись до сервера VPN в іншому місці, люди можуть отримати доступ до вмісту, який може бути обмеженим або недоступним у їхньому регіоні.

Безпечний доступ до Інтернету: звичайні користувачі можуть використовувати VPN для захисту свого Інтернет-з’єднання та захисту

конфіденційної інформації від хакерів або кіберзагроз. Це особливо важливо під час використання публічних мереж Wi-Fi, які часто є незахищеними.

Потокове передавання: VPN можна використовувати для доступу до потокових платформ і бібліотек вмісту, доступних в інших країнах. Це дає користувачам доступ до більш широкого спектру фільмів, телешоу та інших медіафайлів.

Підсумовуючи, сфера застосування VPN дуже широка, охоплюючи бізнес, особисте життя та повсякденне життя. Вони забезпечують безпечний віддалений доступ, безпечну співпрацю, захищають конфіденційність, обходять обмеження, забезпечують безпеку в Інтернеті та покращують взаємодію з користувачем у сценаріях.

### 1.5 Потреби Користувачів та Вимоги до Графічного Інтерфейсу

Зручність користування є одним з ключових факторів у дизайні графічного інтерфейсу, особливо для систем налаштування VPN та управління мережевими налаштуваннями. Інтуїтивно зрозумілий інтерфейс дозволяє користувачам легко налаштовувати та виконувати необхідні задачі без глибоких технічних знань [12]. Наприклад, інтерфейси програмного забезпечення Cisco VPN демонструють як використання яскравих, чітких іконок та вкладок може спростити процес налаштування з'єднань та моніторингу стану мережі. Важливо, щоб інтерфейс надавав легкий доступ до ключових функцій, таких як підключення/відключення VPN, зміна налаштувань мережі, та перегляд статусу з'єднання.

Ефективна візуалізація даних є важливою для користувачів, які потребують швидкого розуміння стану мережі. Графічний інтерфейс повинен включати засоби для відображення ключових метрик, таких як швидкість з'єднання, використання пропускної спроможності, і стан з'єднань VPN. Діаграми, графіки та кольорові індикатори можуть бути використані для надання цієї інформації у зрозумілій та легко читаємій формі. Наприклад, інтерфейси,

схожі на ті, що використовуються в системах моніторингу мережі, таких як SolarWinds або PRTG Network Monitor, можуть надавати динамічні графіки, що відображають реальний час використання мережі [13].

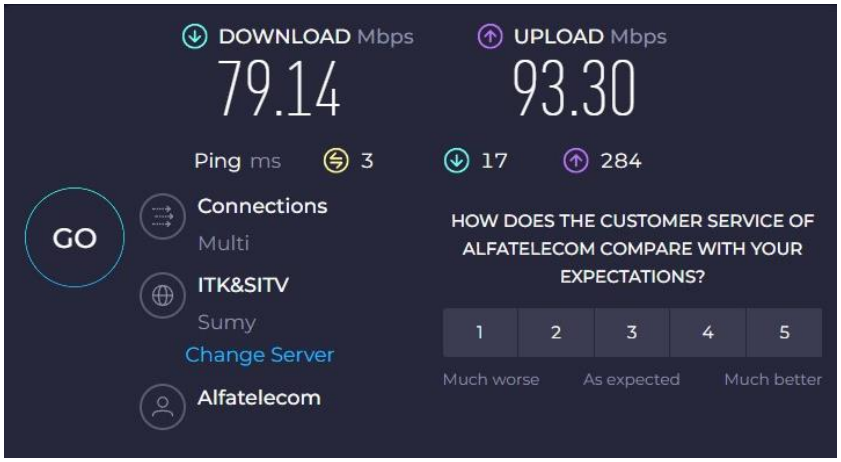


Рисунок 1.5 – Приклад заміру швидкості з ресурсу speedtest.net

Гнучкість у налаштуваннях та можливість кастомізації інтерфейсу є важливими для задоволення різноманітних потреб користувачів. Інтерфейс повинен дозволяти легке досягнення різних рівнів налаштування - від базових опцій для менш досвідчених користувачів до більш складних налаштувань для професіоналів. Можливість налаштування інтерфейсу, така як зміна теми або розташування елементів, дозволяє користувачам створити робоче середовище, яке найкраще відповідає їхнім особистим перевагам та потребам. Прикладом може служити інтерфейс NordVPN,(рис 1.5) який пропонує прості, але потужні налаштування, дозволяючи користувачам легко керувати своїми VPN-підключеннями.


Рисунок 1.6 – Інтерфейс програми NordVPN

Наявність вбудованих ресурсів підтримки та навчальних матеріалів є важливою складовою графічного інтерфейсу. Це включає в себе доступ до довідкових матеріалів, пояснення опцій налаштувань та швидкі посібники для вирішення поширених проблем. Інтерфейси, які надають інтуїтивно зрозумілі підказки та інтерактивну допомогу, можуть значно підвищити задоволеність користувачів та знизити час, необхідний для вирішення технічних питань. Прикладом може служити інтерфейс Microsoft Windows VPN, який містить посібники та підказки для спрощення процесу налаштування VPN-з'єднань.

Загалом, графічний інтерфейс для систем налаштування VPN повинен бути збалансованим, включаючи зручність, інтуїтивність, гнучкість, а також ефективні засоби візуалізації та підтримки. Це дозволить користувачам ефективно взаємодіяти з системою, незалежно від їх технічних знань та досвіду

### 1.6 Постановка задачі

Основною метою кваліфікаційної магістерської роботи є розробка графічного інтерфейсу для налаштування VPN-з'єднань на основі MPLS, який включає кілька ключових завдань. По-перше, необхідно проаналізувати існуючі рішення в цій галузі, визначити їх сильні та слабкі сторони, виявити потреби користувачів, які не задовольняються в існуючих системах. На основі цього аналізу слід розробити детальні вимоги до графічного інтерфейсу, включаючи функціональність, ергономіку, безпеку та інші важливі аспекти. Цей продукт має задовольняти наступним ключовим вимогам:

- Висока зручність і простота використання інтерфейсу, адаптованого для користувачів без глибоких технічних знань у галузі мережевих технологій;
- Гнучкість налаштувань, що дозволяє керувати різними аспектами VPN-з'єднань, включаючи безпеку, маршрутизацію та пропускну спроможність;
- Високий рівень безпеки передачі даних, використовуючи сучасні шифрувальні протоколи;

- Сумісність з різними операційними системами та мережевими обладнаннями

Для досягнення поставленої мети необхідно вирішити наступні кроки:

1) Глибокий аналіз існуючих рішень та збір вимог користувачів для забезпечення релевантності та користувацької цінності розроблюваного продукту;

2) Проектування графічного інтерфейсу, зорієнтованого на зручність та інтуїтивність використання;

3) Розробка та інтеграція мережевих алгоритмів для ефективної роботи з MPLS та VPN;

4) Тестування продукту для забезпечення його стабільності, безпеки та продуктивності;

5) Отримання зворотного зв'язку від кінцевих користувачів та вдосконалення продукту на основі цих даних.

Предмет дослідження охоплює розробку та оптимізацію графічного інтерфейсу для налаштування MPLS VPN, а також інтеграцію мережевих алгоритмів для забезпечення ефективної та безпечної роботи системи.

Наукова новизна полягає у створенні уніфікованого графічного рішення, яке інтегрує передові практики UX/UI дизайну з складними мережевими алгоритмами, створюючи новаторський продукт у сфері VPN-технологій.

Практична значимість дослідження виражається у забезпеченні більш ефективного та безпечного управління корпоративними мережами, що може значно підвищити продуктивність роботи та знизити витрати на IT-інфраструктуру для бізнесу. Розроблений продукт може бути використаний у різних галузях, де потрібно безпечне та ефективне управління мережевими з'єднаннями.

.

# 2 ОПИС ІНФОРМАЦІЙНОЇ СИСТЕМИ

## 2.1 Аналіз обмежень з точки зору продуктивності та безпеки

Пропускна Спроможність та Швидкість одним з основних обмежень систем VPN є зниження пропускної спроможності та швидкості інтернет-з'єднань. Шифрування та дешифрування даних вимагає додаткових обчислювальних ресурсів, що може сповільнити передачу даних, особливо при використанні високої рівня шифрування або при з'єднанні з віддаленими або перевантаженими VPN-серверами.В VPN [14] затримка може бути вищою порівняно з прямими інтернет-з'єднаннями через обхід додаткових мережевих вузлів. Це може бути критичним для додатків, чутливих до затримок, таких як відеоконференції або онлайн-ігри.

Деякі протоколи VPN, особливо старіші, такі як PPTP, мають відомі вразливості. Хакери можуть експлуатувати ці слабкості для перехоплення або навіть зламу шифрування, ставлячи під загрозу конфіденційність даних користувачів. Не всі VPN-провайдери дотримуються строгих політик щодо незберігання логів користувачів. Це створює потенційні ризики для приватності користувачів, оскільки дані про їхню онлайн-активність можуть бути збережені та використані.

Продуктивність та безпека MPLS є ключовими значеннями для мереж та систем які створені на цій базі. Розгортання та обслуговування MPLS-мереж є складними та дорогими, вимагаючи спеціалізованого обладнання та кваліфікованого персоналу. Це робить MPLS недоступним для малих та середніх підприємств. Розширення MPLS-мережі часто вимагає значних капіталовкладень, оскільки це може включати фізичне додавання обладнання та збільшення мережевої інфраструктури [15].

Клієнти MPLS залежать від провайдера мережевих послуг, що може становити ризик для безпеки даних. Компрометація інфраструктури провайдера може призвести до витоку даних. MPLS не надає вродженого шифрування на

рівні кінцевого користувача, що може створювати вразливості у випадку внутрішніх загроз. Інтеграція додаткових механізмів шифрування та аутентифікації може бути необхідною для забезпечення повної безпеки. Аналіз продуктивності та безпеки існуючих систем VPN та MPLS показує, що обидві технології мають свої переваги та обмеження. Хоча VPN пропонує вищий рівень конфіденційності та анонімності, він також стикається з проблемами, пов’язаними зі швидкістю та надійністю. MPLS, з іншого боку, пропонує більш високу швидкість і надійність, але вимагає більше інвестицій і складності впровадження. Обидві системи потребують ретельного керування та підтримки для забезпечення максимальної продуктивності та безпеки.

Налаштування VPN між віддаленими офісами за допомогою технології MPLS включає кілька етапів, які вимагають планування, налаштування та тестування. По-перше, необхідно проаналізувати вимоги до мережі, включаючи пропускну здатність, якість обслуговування (QoS) [16] і вимоги безпеки, щоб забезпечити адекватні послуги для всіх віддалених офісів. Після цього вам слід вибрати постачальника телекомунікаційних послуг, який підтримує MPLS, і розробити топологію VPN, враховуючи розташування головного та віддаленого офісу.

Одним із ключових аспектів налаштування є конфігурація PE (Provider Edge) роутерів у віддалених офісах для з'єднання з MPLS мережею провайдера. Для кожного віддаленого офіса потрібно налаштувати VRF (Virtual Routing and Forwarding), щоб ізолювати трафік в мережі. Це також включає налаштування MPLS L3 VPN, встановлення маршрутизації між VRFs та конфігурацію BGP(Border Gateway Protocol) або інших протоколів маршрутизації між роутерами.

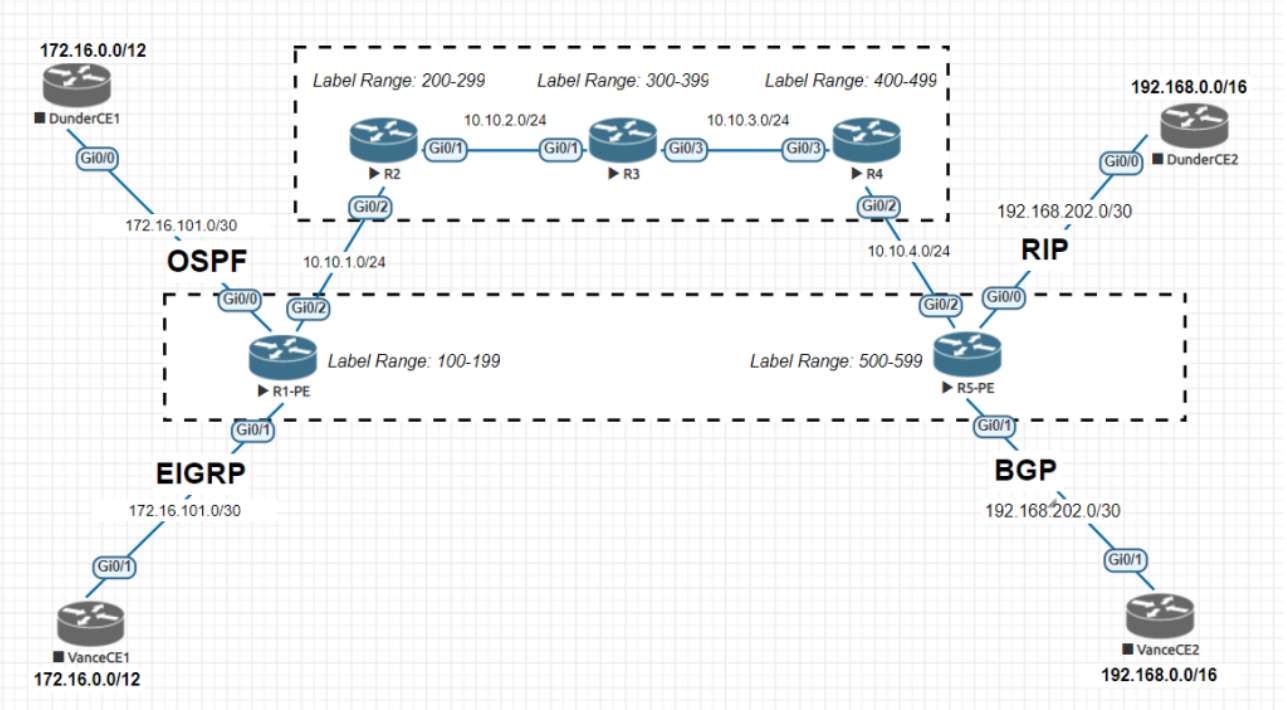


Рисунок 2.1 – Принцип роботи MPLS VPN рівня 3[17]

Для забезпечення необхідної якості обслуговування для різних типів трафіку важливо визначити та налаштувати політики QoS. Ці політики QoS потрібно застосувати на роутерах, щоб керувати пріоритетами та гарантувати пропускну здатність для важливих даних. Крім того, важливо забезпечити додаткові шари безпеки, наприклад, використовуючи IPsec [18] для шифрування трафіку між сайтами, а також встановити брандмауери та фільтрацію трафіку для управління доступом до мережі.

Після налаштувань слід провести всебічне тестування, щоб переконатися, що всі офіси мають стабільне з'єднання через MPLS VPN. Це включає перевірку працездатності з'єднань, якості обслуговування, а також безпеки передачі даних. Використання інструментів моніторингу дозволяє відстежувати продуктивність мережі та швидко реагувати на будь-які проблеми. Також важливо забезпечити доступ до технічної підтримки від провайдера MPLS та внутрішніх IT-спеціалістів для регулярних оновлень конфігурації та обладнання, з метою відповідати змінам у вимогах до мережі та безпеці.

## 2.2 Принцип передачі даних

Принцип передачі даних між мережами є фундаментальним аспектом сучасних комунікацій. В основі цього процесу лежить мережева модель, що дозволяє різним комп'ютерним системам обмінюватися інформацією через набір стандартизованих протоколів і інтерфейсів. Ключовим елементом у цій моделі є мережеві маршрутизатори та комутатори, які спрямовують пакети даних від вихідного пункту до призначення, використовуючи найбільш ефективні шляхи. Коли дані відправляються з одного пристрою, вони спочатку пакуються у вигляді пакетів, які містять самі дані та заголовок із метаданими. Ці метадані включають в себе адресу відправника та отримувача, тип даних, контрольну суму для перевірки помилок та іншу службову інформацію, необхідну для коректної маршрутизації та доставки пакетів. Після створення пакета, мережева карта відправника починає процес передачі, відправляючи пакети через фізичний кабель або бездротове з'єднання до найближчого мережевого пристрою.

На кожному етапі маршрутизації мережеві пристрої, такі як комутатори та маршрутизатори, використовують таблиці маршрутизації, щоб визначити, куди направити пакет далі. Ці таблиці складаються з набору правил, які базуються на IP-адресах, масках підмереж та інших параметрах, що визначають оптимальні шляхи для кожного типу трафіку. У складних мережах, де маршрути можуть змінюватися через збої або перевантаження, маршрутизатори постійно оновлюють свої таблиці, використовуючи протоколи динамічної маршрутизації. При досягненні кінцевої точки, пакети даних проходять через ряд перевірок. Система призначення аналізує контрольну суму, щоб переконатися, що дані не були пошкоджені під час передачі. Після цього вона збирає пакети у вихідні дані, перевіряє, що всі пакети отримано, і відправляє підтвердження назад до відправника. Якщо деякі пакети втрачені або пошкоджені, система може запитати повторну відправку цих пакетів.

В процесі передачі даних також важливу роль відіграють протоколи вищого рівня, такі як TCP, які забезпечують надійність з'єднань, контролюючи потік та послідовність пакетів, а також вирішують проблеми, пов'язані з

перевантаженням мережі. Це дозволяє створювати стабільні та надійні канали зв'язку між віддаленими системами, незалежно від їх фізичного розташування.
На сучасному етапі розвитку мережевих технологій важливу роль відіграє швидкість та безпека передачі даних. Шифрування та VPN-технології дозволяють забезпечити конфіденційність і цілісність інформації, що переміщається через публічні мережі. Сучасні алгоритми шифрування, такі як AES та RSA, забезпечують високий рівень захисту, роблячи практично неможливим несанкціонований доступ або модифікацію даних.

Таким чином, передача даних від однієї мережі до іншої є складним процесом, який включає багатоетапну перевірку, маршрутизацію, шифрування і контроль. Цей процес постійно оптимізується, щоб відповідати зростаючим вимогам до швидкості, надійності та безпеки в епоху глобальної інформатизації.

При передачі великих обсягів даних через мережу важливо враховувати цілий ряд аспектів, що забезпечують ефективність та надійність цього процесу. Однією з ключових умов є наявність стабільної та достатньо широкої пропускної здатності мережі, яка може витримати пікові навантаження без значних затримок або втрат даних. Це особливо важливо в сучасному цифровому світі, де організації залежать від швидкої та безперебійної передачі інформації.

Врахування якості сервісу (Quality of Service, QoS) є невід'ємним аспектом планування мережі, особливо коли мова йде про передачу даних, чутливих до затримок. Конфігурація QoS дозволяє визначити пріоритети для різних типів трафіку, забезпечуючи, що критично важливі дані обробляються з першочерговістю. Це забезпечує, що важливі додатки, такі як системи відеоконференцій та голосового зв'язку, функціонують безперебійно навіть під час інтенсивних періодів передачі даних.

Іншим важливим фактором є безпека передаваних даних. Зі зростанням обсягів передаваних даних зростає й ризик несанкціонованого доступу або витоку інформації. Використання шифрування, такого як SSL/TLS та VPN-технологій, є необхідним для забезпечення конфіденційності та цілісності даних.

Криптографічне шифрування даних перед їхньою передачею через публічні або незахищені мережі запобігає можливості перехоплення та читання даних третіми сторонами.

Крім того, при передачі великих обсягів даних необхідно враховувати можливість відновлення після збоїв. Системи управління передачею даних повинні включати механізми для автоматичного повторення невдалих передач, здатність до швидкого відновлення після збоїв у мережі, а також алгоритми для перевірки цілісності отриманих даних.

Останнім, але не менш важливим аспектом є масштабованість мережевих рішень. Інфраструктура мережі має бути спроектована таким чином, щоб можна було легко розширювати пропускну спроможність та мережеві ресурси з мінімальними зусиллями. Це гарантує, що зі збільшенням обсягу даних, які потрібно передати, мережа зможе витримати додаткове навантаження без деградації продуктивності або якості сервісу.

Розуміння та впровадження цих принципів є критично важливими для забезпечення успішної та ефективної передачі великих обсягів даних у корпоративних мережах. Сучасні технології та методології надають організаціям інструменти, необхідні для вирішення цих викликів, забезпечуючи продуктивність, безпеку та надійність в процесі передачі даних.

### 2.3 Детальний опис архітектури системи та Інтерфейсу

Розробка системи VPN для Windows, з особливим акцентом на використання C# для створення користувацького інтерфейсу, вимагає ретельного планування та розуміння архітектури. Система повинна бути стабільною, безпечною та зручною для користувача, забезпечуючи легке управління VPN-з'єднаннями та високий рівень захисту даних. Ця архітектура повинна інтегрувати кілька ключових компонентів: VPN-клієнт, сервер VPN, системи шифрування, мережевий стек Windows і графічний користувацький інтерфейс (GUI). Такий підхід забезпечує безпеку, надійність та зручність використання.

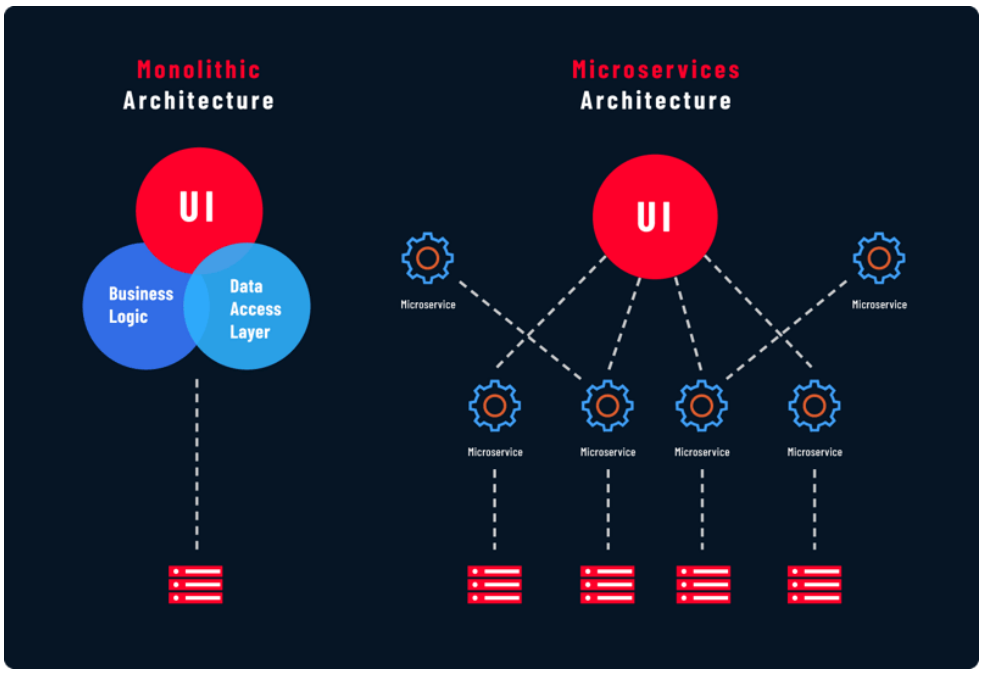


Рисунок 2.2 – Клієнт-Серверна Модель[19]

Клієнт-Серверна Модель основою системи є клієнт-серверна архітектура. Серверна частина відповідає за управління VPN-з'єднаннями, [20] аутентифікацію користувачів, маршрутизацію трафіку та обробку шифрування. Клієнтська частина, реалізована у C#, забезпечує користувацький інтерфейс для налаштування з'єднань, моніторингу стану та управління налаштуваннями безпеки.Користувацький Інтерфейс користувача реалізується за допомогою мови програмування C# із використанням WPF (Windows Presentation Foundation) для створення графічного інтерфейсу. WPF дозволяє створювати сучасні, багатофункціональні інтерфейси з розширеними можливостями візуалізації. Це може включати інтерактивні елементи управління, такі як кнопки, випадаючі меню, слайдери для налаштування з'єднань, а також інструменти для візуалізації стану мережі та статистики трафіку.

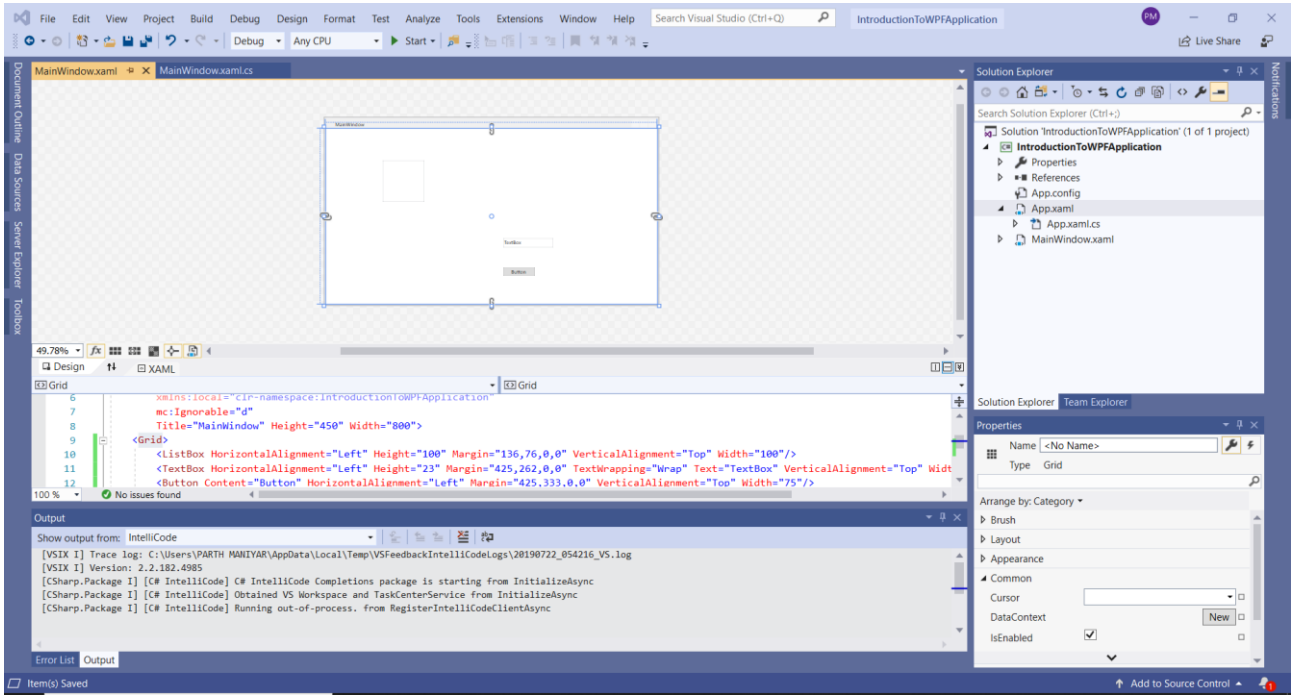


Рисунок 2.3 – WPF (Windows Presentation Foundation) AES

Серверні системи повинні підтримувати стандартні протоколи VPN, такі як IPsec і OpenVPN, щоб забезпечити сумісність і безпеку. У додатку до основних функцій VPN, сервер також відповідає застосуванню політики безпеки, таких як брандмауер, фільтрація трафіку та блокування небажаних з’єднань. Основне увага приділяється безпеці, зокрема шифрування даних. Використовуйте алгоритми шифрування, такі як AES або RSA, щоб забезпечити захист даних користувача під час передачі через загальнодоступну мережу. Важливо переконатися, що ключі шифрування зберігаються в безпечному середовищі та не мають доступу до них [21].

Основною метою даної роботи є розробка графічного інтерфейсу для налаштування VPN-з'єднань на основі MPLS, який включає кілька ключових завдань. По-перше, необхідно проаналізувати існуючі рішення в цій галузі, визначити їх сильні та слабкі сторони, виявити потреби користувачів, які не задовольняються в існуючих системах. На основі цього аналізу слід розробити детальні вимоги до графічного інтерфейсу, включаючи функціональність, ергономіку, безпеку та інші важливі аспекти.

Особливу увагу слід приділити розробці інтерфейсу користувача, який

повинен бути не тільки функціональним, але й інтуїтивно зрозумілим, щоб забезпечити зручність використання навіть для менш досвідчених користувачів.

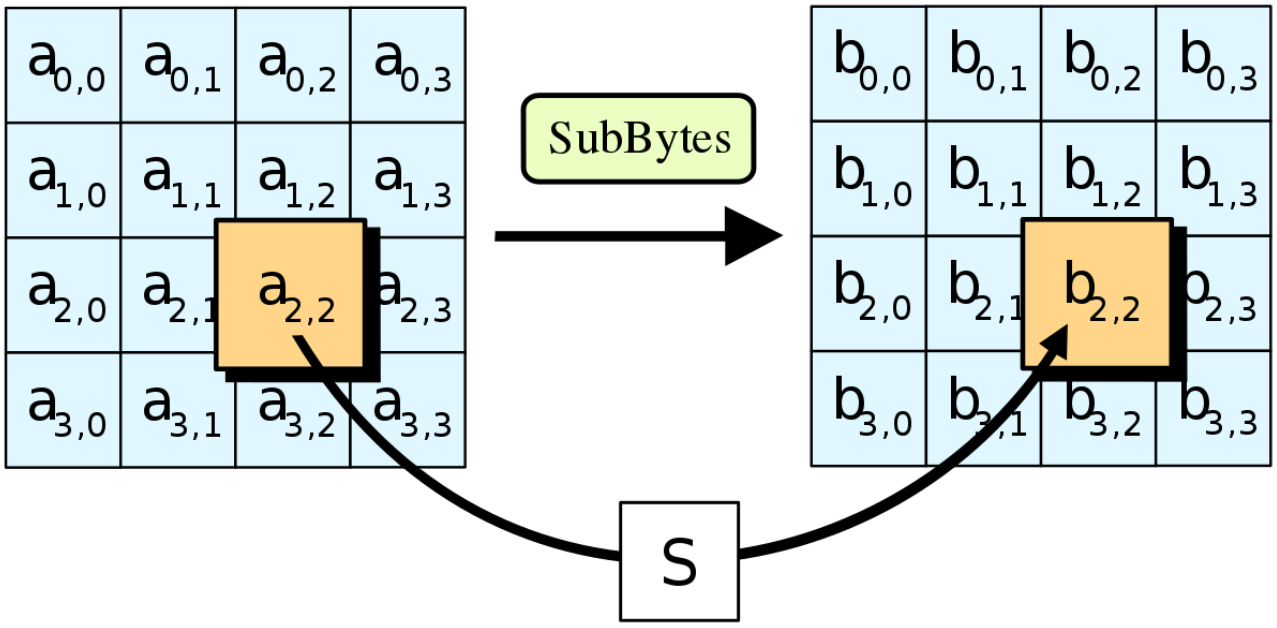


Рисунок 2.4 – AES

Система має надавати можливості для управління VPN-з'єднаннями, включаючи налаштування, створення профілів користувачів, та моніторинг стану з'єднань. Інструменти моніторингу повинні забезпечувати детальну статистику використання мережі, виявлення проблем та автоматизоване управління пропускною спроможністю та інтеграцію з операційною системою Windows вимагає врахування особливостей Windows API для забезпечення стабільної та ефективної роботи системи. Це включає управління мережевими адаптерами, налаштування файрволів, та взаємодію з іншими системними компонентами.

Основна архітектура користувацький Інтерфейс на C# та WPF:

1) Використання C# у поєднанні з Windows Presentation Foundation (WPF) для створення настільного додатку, що надасть інтуїтивно зрозумілий графічний інтерфейс для налаштування VPN та MPLS;

2) Інтерфейс включатиме необхідні елементи керування для введення налаштувань мережі, управління профілями VPN та перегляду статусу з'єднання;

3) Інтеграція командної строки у програму для введення допоміжних команд та скриптів. Це дозволить користувачам швидко виконувати розширені або спеціалізовані налаштування;

4) Вбудована функція для тестування швидкості інтернет-з'єднання, яка дозволить користувачам оцінити ефективність VPN-з'єднання в реальному часі;

5) Інтеграція функціоналу для налаштування та управління MPLS-параметрами, включаючи налаштування маршрутів, політик QoS (Quality of Service) та інших важливих параметрів мережі/

Реалізація Функціоналу

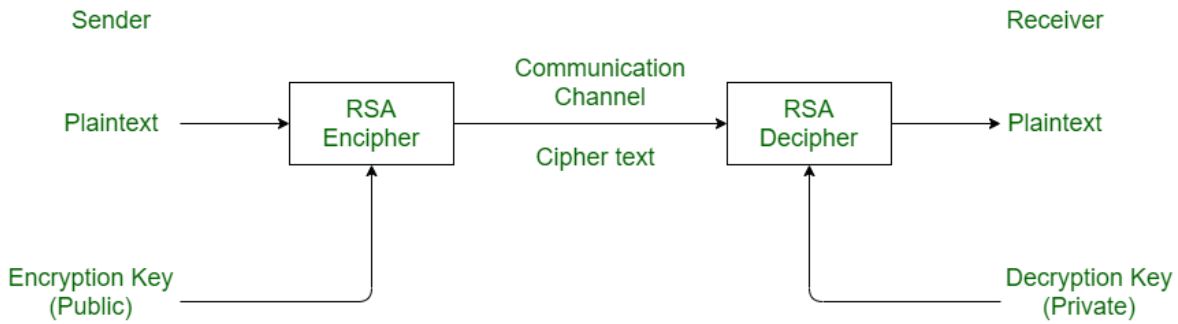


Рисунок 2.5 – RSA

Інтеграція з Серверною Частиною:

1) Використання API або інших механізмів комунікації для зв'язку між настільним додатком та сервером, який керує VPN/MPLS. Це може включати в себе відправку налаштувань, команд та запитів статусу;

2) Впровадження функціональної командної строки в межах програми, що дозволить користувачам вводити команди для швидкого доступу до розширених функцій або виконання специфічних скриптів;

3) Розробка вбудованого модуля для вимірювання швидкості інтернет-з'єднання, який може включати в себе тестування пінгу, завантаження та вивантаження даних;

4) Реалізація інтерфейсу для керування MPLS, включаючи налаштування маршрутизації, управління пропускною спроможністю та іншими аспектами MPLS-конфігурації;

5) Реалізація високого рівня безпеки для зберігання користувацьких даних та передачі конфіденційної інформації, особливо при взаємодії з серверною частиною.

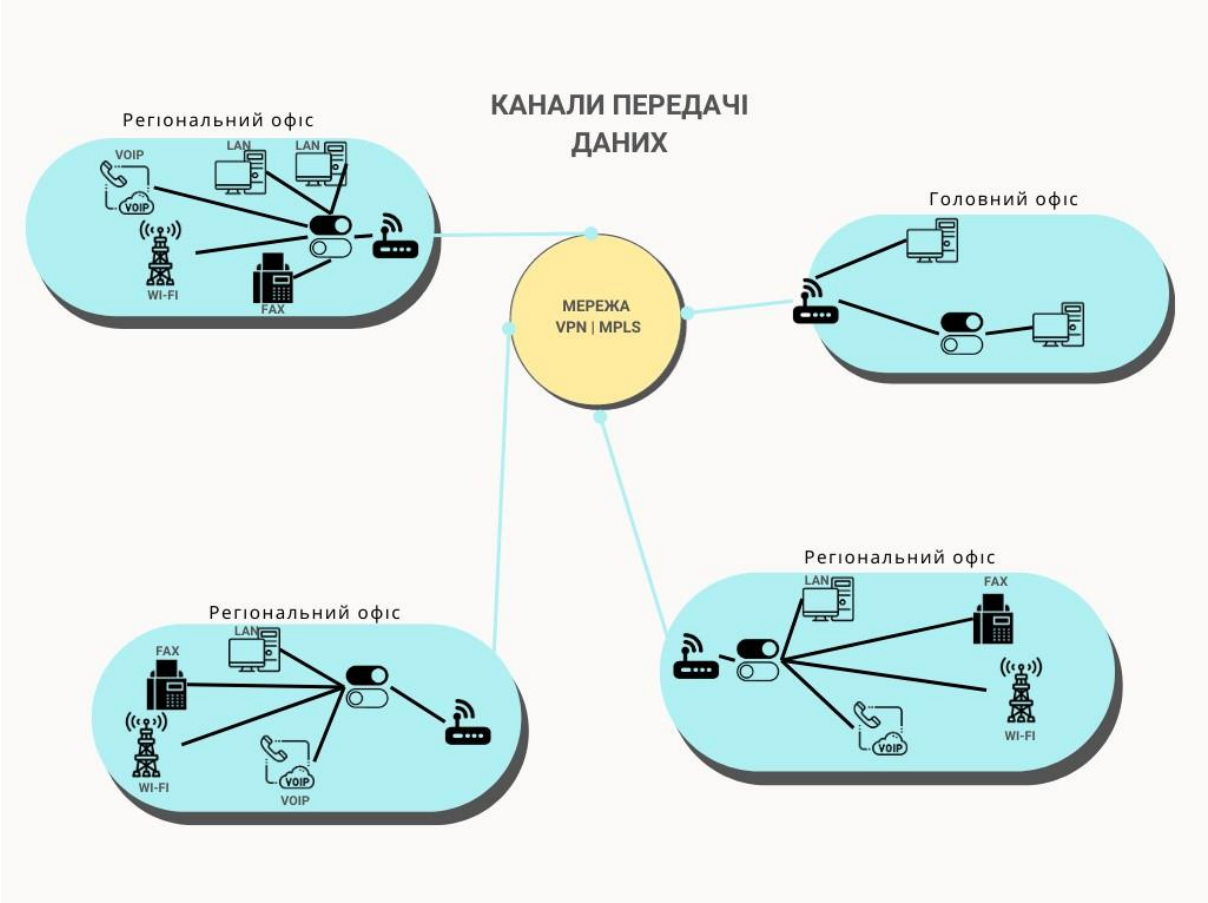


Рисунок 2.6 – Схема взаємодії між офісами в мережі

### 2.4 Опис основних компонентів клієнт-сервер системи.

В архітектурі програмного забезпечення, що використовує клієнт-серверну модель, взаємодія між програмами заснована на чіткому розподілі відповідальності та функцій. Ось як це виглядає у випадку системи VPN на основі MPLS.

Модульність: C# дозволяє створювати модульний код, тим самим спрощуючи керування складними програмами. Розробники можуть використовувати такі шаблони проектування, як Model-View-ViewModel (MVVM), щоб чітко відокремити бізнес-логіку від інтерфейсу користувача. Зв’язування даних WPF підтримує двостороннє зв’язування даних, дозволяючи елементам інтерфейсу автоматично синхронізуватися з моделлю даних. Це зменшує потребу в ручному оновленні інтерфейсу користувача та спрощує реакцію інтерфейсу на зміни даних. WPF надає гнучку систему стилів і тем, яка дозволяє створювати узгоджений візуальний дизайн у всій програмі. Розробники можуть універсально визначати стилі для кнопок, текстових полів та інших елементів і застосовувати їх у всій програмі.

Через цей інтерфейс користувачі можуть вводити параметри конфігурації, такі як ідентифікаційні дані для підключення до VPN, налаштування мережі, та вибір типу шифрування. Вікно командної строки інтегроване в програму як додатковий інструмент для досвідчених користувачів, які бажають вводити команди безпосередньо або виконувати скрипти для автоматизації задач. Додаткові модулі, такі як інструменти для тестування швидкості інтернету та моніторингу MPLS, дають користувачам можливість оцінювати та оптимізувати мережеві з'єднання.

Модульність: C# дозволяє створювати модульний код, тим самим спрощуючи керування складними програмами. Розробники можуть використовувати такі шаблони проектування, як Model-View-ViewModel (MVVM), щоб чітко відокремити бізнес-логіку від інтерфейсу користувача. Зв’язування даних WPF підтримує двостороннє зв’язування даних, дозволяючи елементам інтерфейсу автоматично синхронізуватися з моделлю даних. Це

зменшує потребу в ручному оновленні інтерфейсу користувача та спрощує реакцію інтерфейсу на зміни даних. WPF надає гнучку систему стилів і тем, яка дозволяє створювати узгоджений візуальний дизайн у всій програмі. Розробники можуть універсально визначати стилі для кнопок, текстових полів та інших елементів і застосовувати їх у всій програмі.

**Серверна Частина**

Управління VPN-з'єднаннями в корпоративному середовищі є критично важливим завданням, яке покладається на серверну частину системи. Серверна частина функціонує як центральний вузол, що керує всіма з'єднаннями, аутентифікацією користувачів, шифруванням трафіку, а також маршрутизацією даних між користувачами та корпоративними ресурсами.

Сервер VPN, який зазвичай розміщується у вигляді одного або декількох віртуальних або фізичних серверів у дата-центрі або в хмарному середовищі, має високий рівень доступності та надійності. Ці сервери мають здатність обробляти велику кількість одночасних VPN-сесій та гарантувати стабільність і безперебійність роботи мережі.

Сервер виконує аутентифікацію користувачів, перевіряючи їхні облікові дані перед наданням доступу до VPN. Аутентифікація може здійснюватися через власні бази даних користувачів, LDAP, Active Directory або за допомогою сторонніх ідентифікаційних провайдерів. Авторизація включає в себе перевірку прав користувачів на доступ до певних ресурсів.

Після успішної аутентифікації, сервер налаштовує зашифрований тунель для захищеної передачі даних. Шифрування використовує стандартизовані алгоритми, такі як AES або SSL/TLS, щоб забезпечити, що дані не можуть бути прочитані або змінені під час передачі через публічні мережі.

Сервер також керує маршрутизацією даних між віддаленими користувачами та корпоративною мережею. Це включає в себе налаштування мережевих маршрутів, які визначають, як і куди будуть перенаправлятися пакети

даних, забезпечуючи оптимальну продуктивність та мінімізуючи затримку.

Сервери обладнані інструментами для моніторингу стану з'єднань, використання пропускної спроможності та виявлення нештатних ситуацій. Адміністратори мережі мають доступ до докладних звітів та логів, що дозволяє їм оперативно реагувати на інциденти та здійснювати необхідні коректівні дії.
Сервери VPN часто налаштовані на високу доступність, використовуючи технології кластеризації та балансування навантаження, щоб забезпечити стійкість до збоїв та здатність до швидкого відновлення після непередбачених відключень.

Серверна частина системи VPN є фундаментом для забезпечення безпеки та продуктивності віддаленого доступу до корпоративних ресурсів. Вона має бути ретельно спроектована та правильно налаштована для відповідності строгим вимогам корпоративної безпеки та надійності, забезпечуючи при цьому гнучкість та масштабованість для адаптації до змінних бізнес-потреб.
напиши більш делальніше про Взаємодія між Клієнтом та Сервером

**Взаємодія між Клієнтом та Сервером у VPN-Системі**

У системах VPN, які базуються на клієнт-серверній архітектурі, взаємодія між клієнтом (користувачем) та сервером (центральним вузлом) є ключовою для забезпечення безпечного і ефективного підключення до мережі. Цей процес інтеграції включає декілька етапів та компонентів, які працюють разом для створення зашифрованого тунелю через який можуть передаватися конфіденційні дані.

Коли користувач намагається підключитися до VPN, процес розпочинається з встановлення зв'язку з VPN-сервером. Клієнт, посилає запит на підключення до сервера, який, у свою чергу, вимагає аутентифікації користувача. Аутентифікація може відбуватися через різні методи, включаючи паролі, токени або сертифікати.

Після успішної аутентифікації сервер ініціює створення зашифрованого

тунелю. Сервер налаштовує параметри шифрування та відправляє їх клієнту, який використовує ці параметри для шифрування даних перед їх відправкою. Такий підхід гарантує, що всі дані, які передаються через публічну мережу, захищені від перехоплення або витоку.

VPN-сервер керує маршрутизацією трафіку в межах зашифрованого тунелю. Він визначає, яким чином пакети даних будуть направлятися до віддалених офісів або вхідних точок корпоративної мережі. Сервер може використовувати технологію MPLS для оптимізації шляхів трафіку і забезпечення кращої продуктивності мережі.

Взаємодія також включає постійний моніторинг стану з'єднань та трафіку. Клієнтська програма може надсилати запити на сервер для отримання інформації про стан з'єднання або швидкість інтернету. Це дозволяє користувачам виявляти та діагностувати проблеми з мережею.

Серверна частина інтегрується з існуючою мережевою інфраструктурою, управляючи обладнанням MPLS, маршрутизаторами, комутаторами та іншими пристроями. Це забезпечує гнучке управління мережею та можливість швидкого реагування на зміни у бізнес-процесах або технологічних потребах.

Ефективна взаємодія між клієнтом та сервером є основою для успішної реалізації VPN-системи. Вона дозволяє не лише забез печити безпеку та конфіденційність даних, але й гарантує високу продуктивність і доступність корпоративних ресурсів для віддалених користувачів, використовуючи сучасні протоколи та технології

# 3 ІНФОРМАЦІЙНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ

## 3.1 Огляд програмного рішення

Програма для управління VPN, розроблена на C#, , функціонує як централізована платформа для налаштування та управління VPN-з'єднаннями між декількома віддаленими офісами за допомогою технології MPLS. Інтерфейс програми розроблений таким чином, щоб бути зручним для користувача, забезпечуючи просту навігацію і зрозумілі опції для налаштування і моніторингу VPN-з'єднань. Розробка, відбувалася в середовищі розробки, такому як Visual Studio, яке надає широку підтримку для розробки на C# та WPF. Логіка програми інкапсульована в класах і методах, які обробляють різні аспекти.

Інтерфейс використовує Windows Presentation Foundation (WPF) для відображення графічного інтерфейсу користувача (GUI), який відомий своєю багатою презентацією і можливістю створювати відокремлені, підтримувані додатки.

Користувачі можуть додавати або змінювати налаштування VPN-сервера, включаючи імена, адреси, протоколи та облікові дані. Це здійснюється за допомогою взаємодії з мережевими компонентами операційної системи управління VPN.

Окремі розділи для налаштувань клієнта і сервера дозволяють детально контролювати кожен кінець VPN-з'єднання. Це може включати налаштування шифрування, мережевих масок і таблиць маршрутизації, гарантуючи, що обидва кінці VPN-тунелю правильно налаштовані.

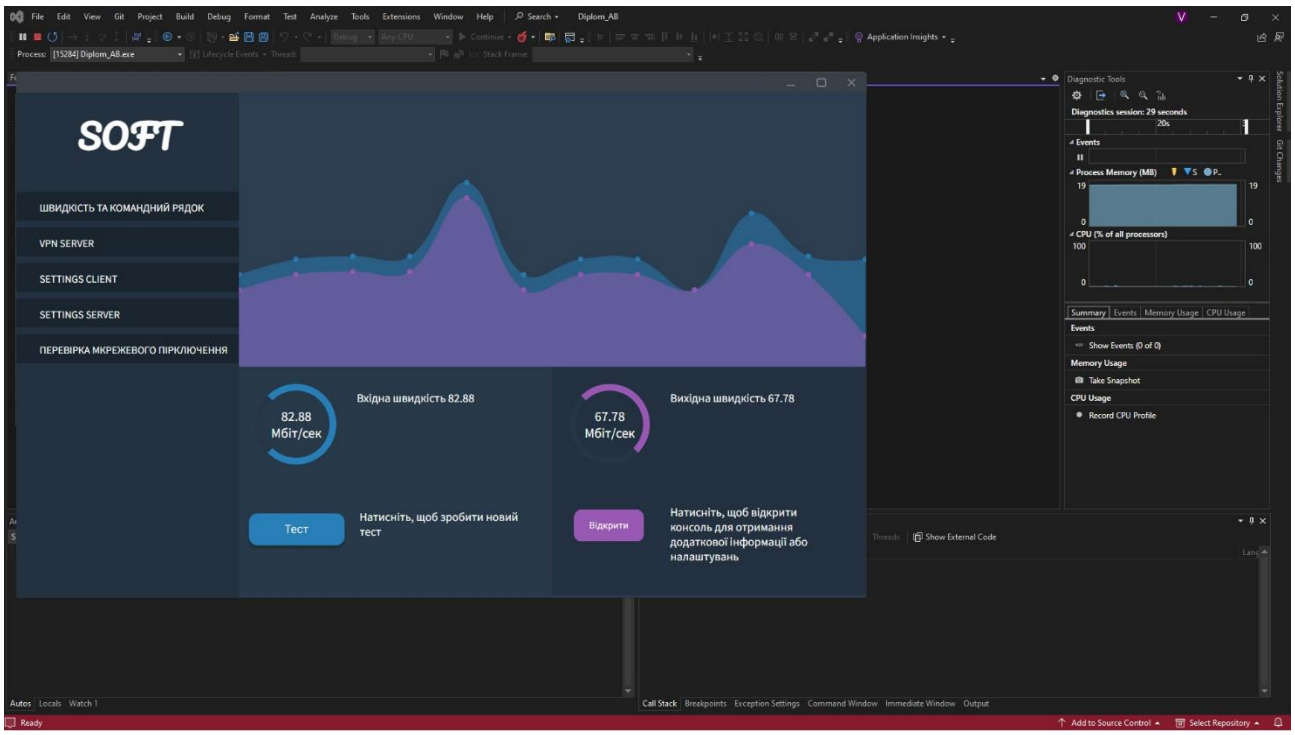


Рисунок 3.1 – Запуск програми в WPF через Visual Studio

Програма містить функції для перевірки цілісності та швидкості мережевих з'єднань, як правило, шляхом надсилання тестових пакетів через VPN-тунель для вимірювання затримок і пропускної здатності. Яка в реальному часі показує швидкість та входу та виходу даних, та виводе їх в динамічний графік котрий показує пікові значення мережі.

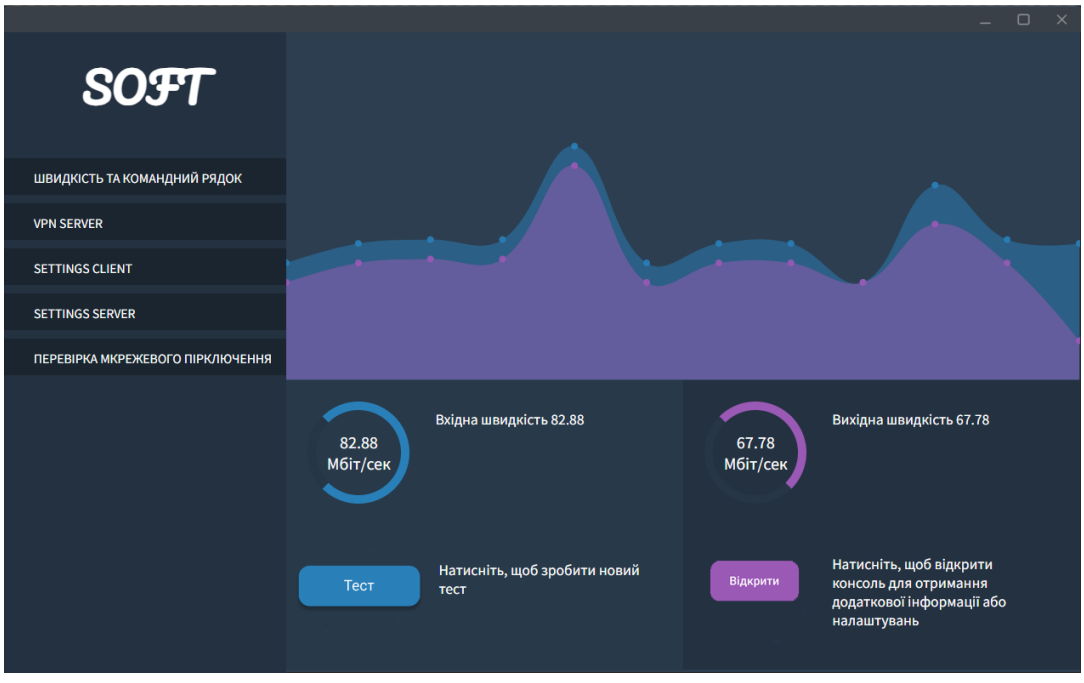


Рисунок 3.2 – Показу швидкості у додатку

Для досвідчених користувачів, та більш глибинного налаштування даних та перегляду інформації створена інтеграція з командним рядком або PowerShell, що дозволяє їм виконувати сценарії або команди для більш складних конфігурацій або завдань автоматизації, та бути більш зручною для виявлення проблем.

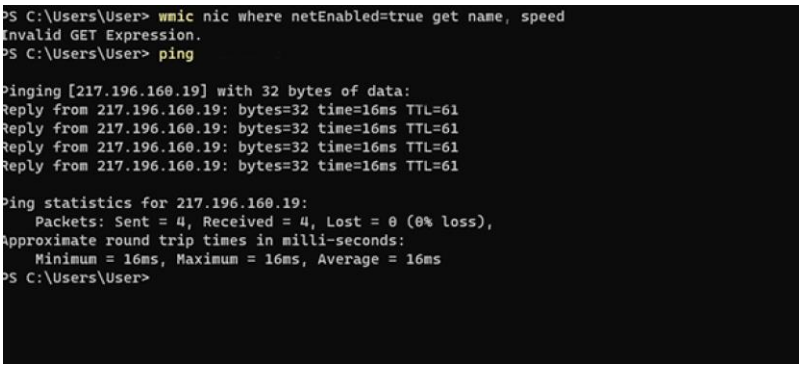


Рисунок 3.3 – Командний рядок у додатку

Таким чином візуальний аспект програми займає одну з ключових ролей, але й має більш глибокий функціонал для моніторингу та налаштувань за допомогою командного рядка.

### 3.2 Приклад роботи застосунку

Програма інтегрує налаштування MPLS, дозволяючи користувачам визначати мітки і шляхи для ефективної маршрутизації пакетів даних, що має вирішальне значення для підтримки високої продуктивності в корпоративних мережах. Система надає оновлення в режимі реального часу про стан VPN-з'єднань і мережевих умов. Безпека є наріжним каменем програми, з надійними механізмами автентифікації для запобігання несанкціонованому доступу. Для шифрування, , використовуються стандартні протоколи, такі як SSL/TLS.

Програма надає можливості налаштування для різних сценаріїв VPN-з'єднань, що дозволяє мережевим адміністраторам пристосовувати налаштування до своїх конкретних потреб.

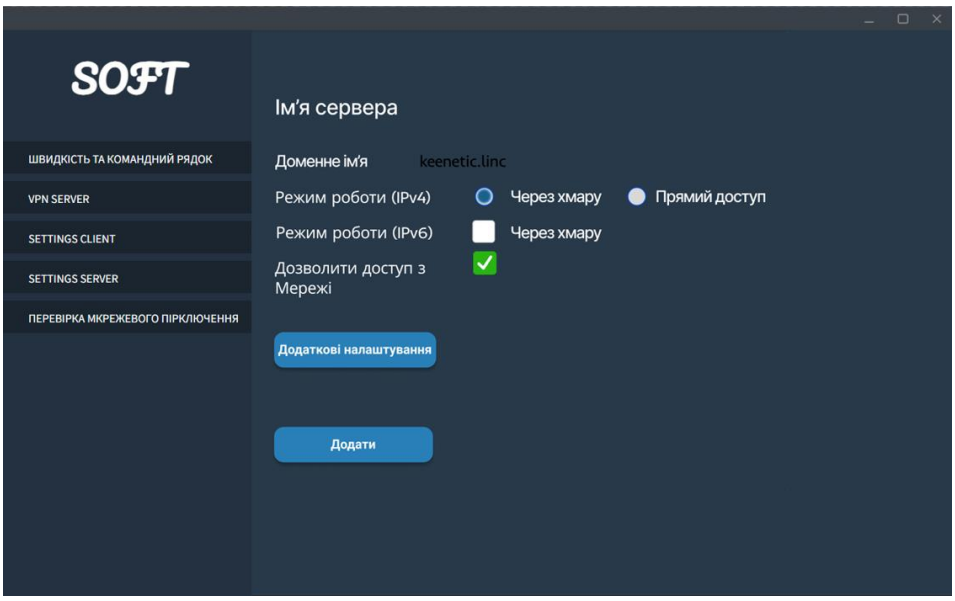


Рисунок 3.4 – Початкове налаштування для сервера

Щоб встановити нове VPN-з'єднання, потрібно вести необхідну інформацію у відповідні поля в графічному інтерфейсі. Після введення даних програма перевіряє їх і застосовує конфігурацію до мережевого стеку системи або зв'язується з VPN-сервером для встановлення з'єднання. Користувач може відстежувати стан з'єднання і виконувати тести в додатку, щоб переконатися, що все працює належним чином. Для технічного обслуговування або усунення несправностей програма надає можливість перегляду деталів журналів та діагностики за допомогою командного рядку.

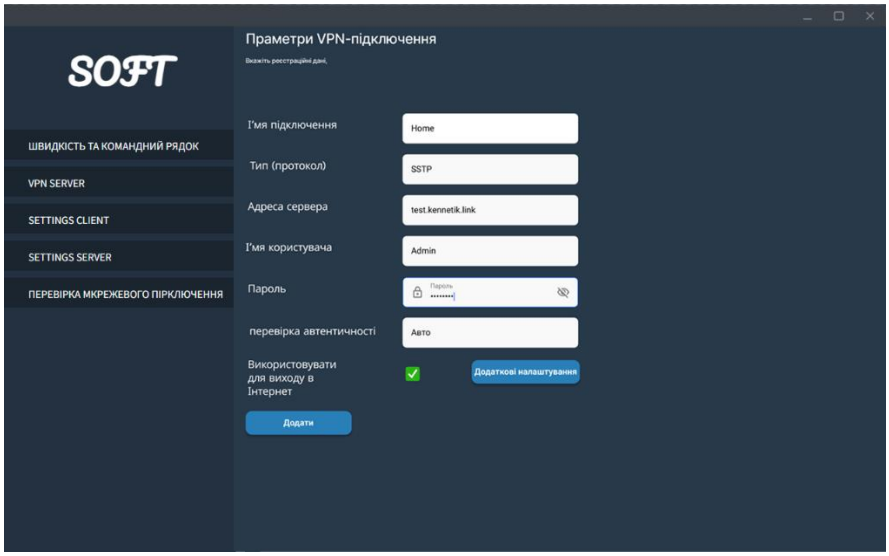


Рисунок 3.5 – Налаштування до параметрів VPN-підключення

Параметри з’єднання VPN включають ім’я з’єднання, яке ідентифікує профіль VPN у налаштуваннях мережі користувача. Тип протоколу (наприклад, SSTP) визначає метод для шифрування та передачі інформації. Адреса сервера показує на віддалену кінцеву точку з’єднання VPN, а облікові дані ім’я користувача та пароль забезпечують автентифікацію до мережі та доступ до неї. Інші параметри (наприклад, «Використовувати для входу в Інтернет») дозволяють налаштувати, чи весь трафік має проходити через VPN.

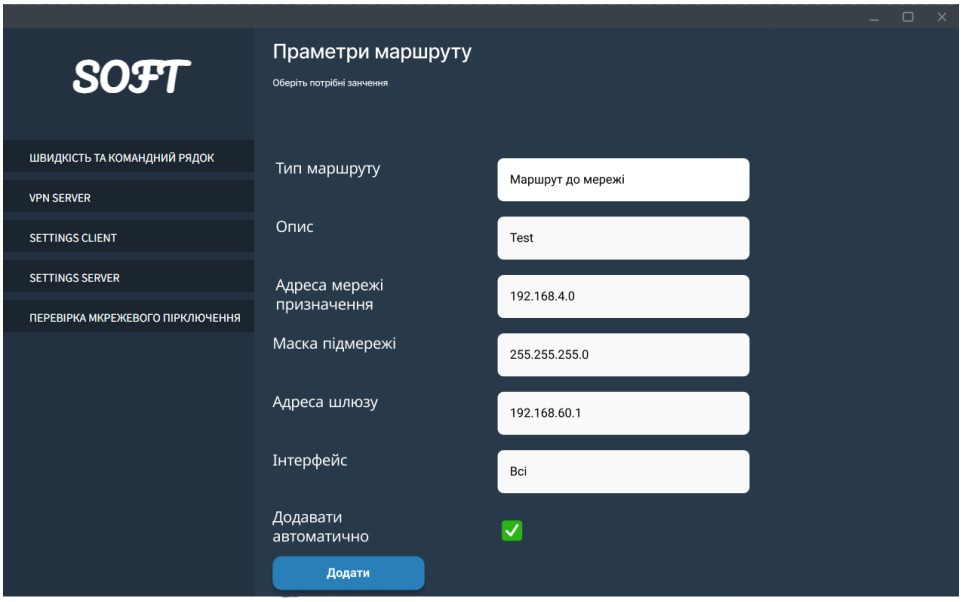


Рисунок 3.6 – Налаштування до параметру маршруту

Параметри маршруту визначають налаштування мережевої маршрутизації для VPN. Тип маршруту вказує, чи це стандартний або специфічний маршрут до певної підмережі. Адреса мережі призначення та маска підмережі визначають цільовий діапазон IP-адрес, до яких буде застосовуватися маршрут. Адреса шлюзу вказує на інтерфейс або пристрій, через який пакети повинні бути направлені для досягнення вказаної підмережі.

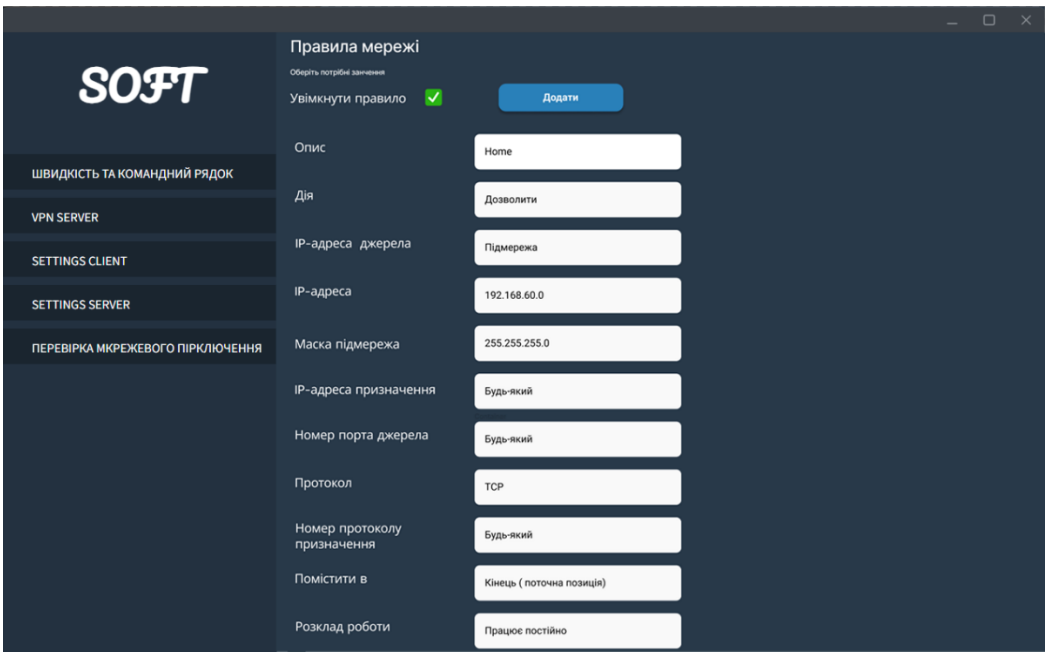


Рисунок 3.7 – Налаштування до правил мережі

Налаштування до правил мережі дозволяє адміністраторам визначати та управляти доступом до мережевих ресурсів через VPN. Включає в себе можливість дозволити або заборонити трафік до певних IP-адрес або підмереж, встановлюючи конкретні маски підмережі та правила маршрутизації. Діапазон IP-адрес джерела та призначення, а також номери портів і протоколи, що використовуються для фільтрації трафіку, задають гранулярний контроль над тим, як дані переміщаються через VPN.

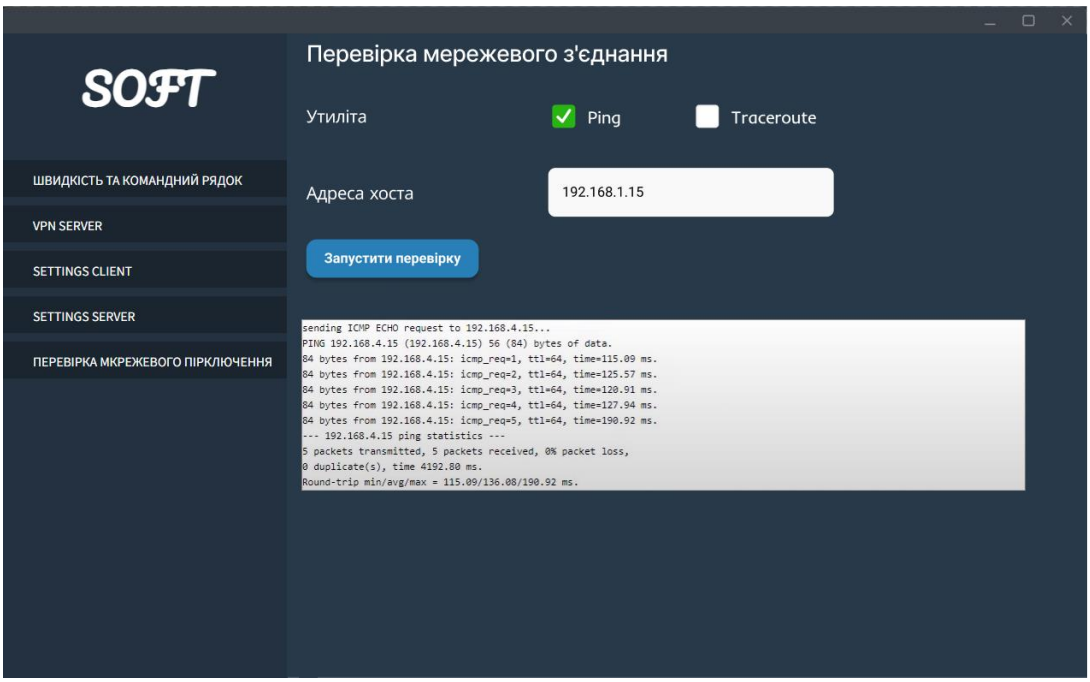

Рисунок 3.8 – Перевірка мережевого з’єднання VPN-тунелю

Використання WPF дозволяє відокремити графічний інтерфейс від бізнес-логіки, що полегшує підтримку та розширення програми.

Дотримуючись цих принципів, додаток для управління VPN слугує потужним інструментом для мережевих адміністраторів, забезпечуючи ефективне та безпечне управління віддаленими офісними з'єднаннями.

Після детального та правильного та налаштування користувач додатку має можливість зайти в мережу та передавати дані по тунелю з різних девайсів та надавати і зчитувати дані. Для переходу до мережі котра була підключина потрібно вети дані для входу в пункт швидкий доступу Рис.3.9.

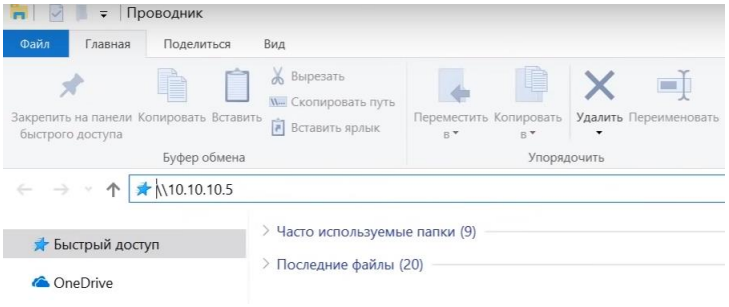


Рисунок 3.9 – Перехід до мережі

Після входу в мережу Users стає доступним весь її контент та матеріали, які вільно можна передавати та перегляди, як ще зображено на Рис.3.10-3.11

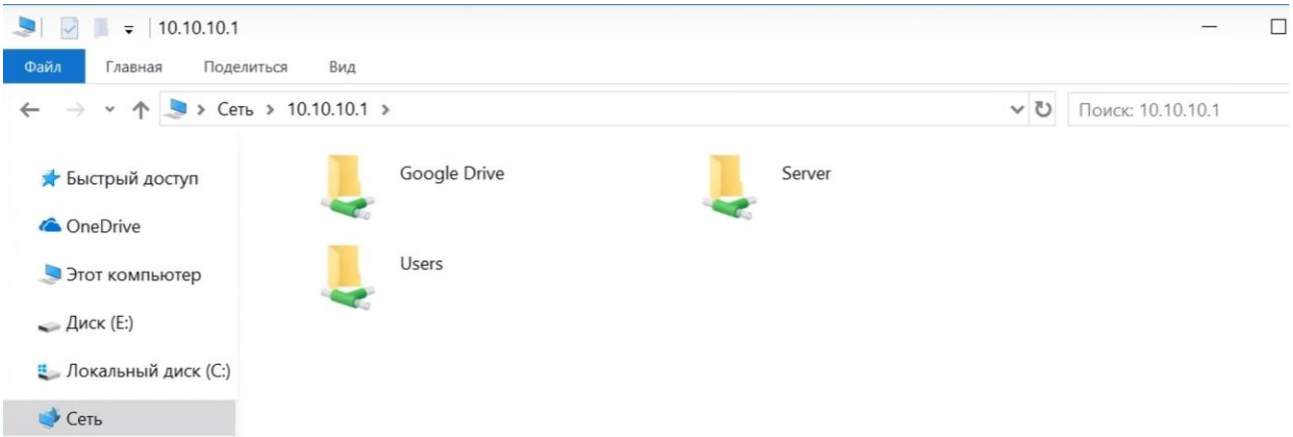


Рисунок 3.10 – Підключені мережі

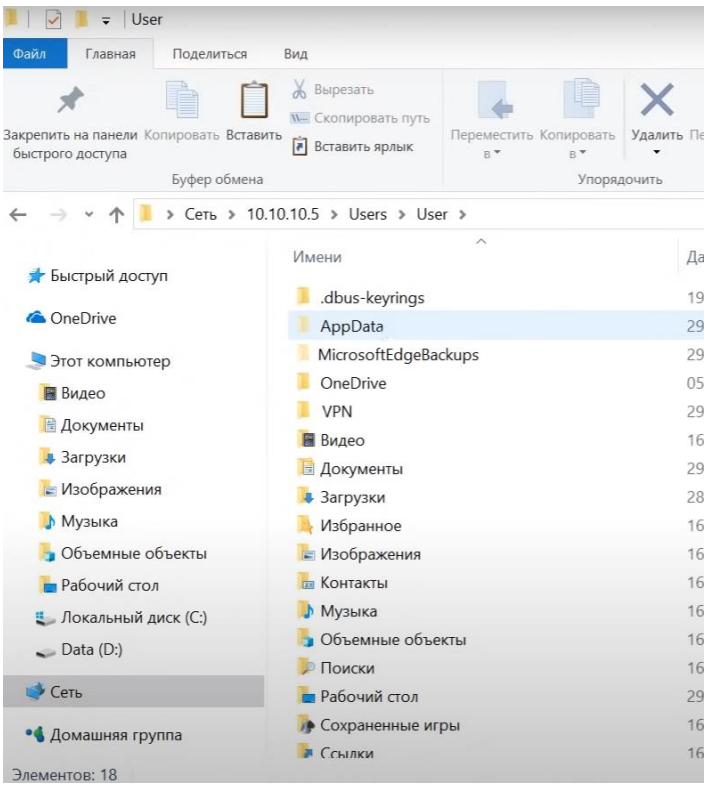


Рисунок 3.11 – Вхід в мережу Users

### 3.3Аналіз системи

Додаток має інтерфейс, який був розроблений з акцентом на інтуїтивність і легкість використання, що виокремлює його серед аналогічних рішень на ринку. Він інтегрує візуальні підказки та контекстно-залежні меню, сприяючи швидкій адаптації користувачів і зниженню кривої навчання. Що стосується функціональності, додаток включає спеціалізовані засоби для налаштування MPLS, які дозволяють оптимізувати пропускну здатність мережі і автоматизувати маршрутизацію, що є рідкісними серед стандартних VPN-рішень.

У сфері продуктивності, додаток демонструє ефективність, при роботі з даних або в мережах з високим навантаженням. Це досягається завдяки вдосконаленому алгоритму обробки даних та оптимізації мережевих запитів. З точки зору масштабованості, додаток показує високу гнучкість, дозволяючи з легкістю додавати нові віддалені офіси та користувачів без необхідності в істотних змінах інфраструктури або перерв у роботі сервісу.

Безпека в додатку стандартам, що включає протоколи шифрування та аутентифікації, які забезпечують захист даних на рівні.

Додаток має великий базу для росту та додавання нових можливостей та нового функціоналу.

# ВИСНОВКИ

У результаті проведеного дослідження було розроблено та реалізовано графічний інтерфейс для налаштування VPN на основі MPLS, який демонструє значне поліпшення в ефективності та зручності управління мережевими налаштуваннями у корпоративних середовищах. Дослідження підкреслило важливість інтуїтивного дизайну та легкості використання в сучасних мережевих системах, де час і точність налаштувань відіграють критичну роль.

Розроблений інтерфейс забезпечує користувачам гнучкі та ефективні засоби для керування VPN-з'єднаннями, зменшуючи потребу в глибоких технічних знаннях і сприяючи швидшому впровадженню мережевих змін. Також, реалізація сучасних принципів безпеки та шифрування у додатку гарантує надійність та безпеку передаваних даних.

Додатково, праця внесла свій вклад у розвиток наукових знань в області мережевих технологій, зокрема у створенні користувацьких інтерфейсів для складних технічних систем. Результати дослідження можуть бути використані для подальшого розвитку та удосконалення мережевих рішень, а також слугувати базою для майбутніх наукових праць у цій галузі.

Таким чином, виконана робота демонструє значний потенціал для підвищення продуктивності та ефективності управління мережами в корпоративних середовищах і вносить значний вклад у розвиток технологій VPN та MPLS.

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ


1. MPLS QOS LLM [Електронний ресурс] – Режим доступу https://protocoholic.com/2019/04/18/mpls-qos/
2. Microservices: what are they and why use them? [Електронний ресурс] – Режим доступу https://blog.sparkfabrik.com/en/guides/microservices-what-are-they-and-why-use-them
3. Layer 3 MPLS VPN – Lab 1: Underlay, MPLS and LDP [Електронний ресурс] – Режим доступу https://ccieme.wordpress.com/2021/05/13/layer-3-mpls-vpn-lab-1-underlay-mpls-and-ldp/
4. Recent trends in MPLS networks: technologies, applications and challenges LDP [Електронний ресурс] – Режим доступу https://ietresearch.onlinelibrary.wiley.com/doi/full/10.1049/iet-com.2019.6129
5. Segment Routing with the MPLS Data Plane [Електронний ресурс] – Режим доступу https://rap.mirror.cyberbits.eu/rfcs/rfc8660.pdf
6. An In-depth Guidance to Multiprotocol Label Switching (MPLS)[Електронний ресурс] – Режим доступу https://medium.com/@allan0123120/an-in-depth-guidance-to-multiprotocol-label-switching-mpls-1dd2b5cb8f6d
7. Smart hybrid SDN approach for MPLS VPN management on digital environment [Електронний ресурс] – Режим доступу https://link.springer.com/article/10.1007/s11235-019-00603-6
8. Exploring Complex MPLS VPN Applications: Models and Implementations for Modern Communication Demands [Електронний ресурс] – Режим доступу https://www.researchsquare.com/article/rs-3347362/v1
9. Scalability, Consistency, Reliability and Security in SDN Controllers: A Survey of Diverse SDN Controllers [Електронний ресурс] – Режим доступу https://link.springer.com/article/10.1007/s10922-020-09575-4
10. Evaluation of 6PE and 6VPE techniques in MPLS-VPN networks for video streaming [Електронний ресурс] – Режим доступу https://ieeexplore.ieee.org/abstract/document/9690563
11. A Comparative Analysis of Unicast Routing Protocols for MPLS-VPN [Електронний ресурс] – Режим доступу https://www.researchgate.net/profile/Maaz-Ahmad-7/publication/358892644_A_Comparative_Analysis_of_Unicast_Routing_Protocols_for_MPLS-VPN/links/6384f9bb554def61937e9836/A-Comparative-Analysis-of-Unicast-Routing-Protocols-for-MPLS-VPN.pdf
12. Evaluating Performances of VPN Tunneling Protocols Based on Application Service Requirements [Електронний ресурс] – Режим доступу https://link.springer.com/chapter/10.1007/978-981-16-7597-3_36
13. Application of MPLS Tunnel Service L2TP-VPN Optimization Concept with Traffic Engineering Method for Looping-Protection Service Analysis [Електронний ресурс] – Режим доступу https://journals.pan.pl/Content/126547/PDF/16.pdf

14. Traffic Engineering and QoS in a Proposed MPLS-VPN https://ieeexplore.ieee.org/document/9301135
15. Distributed ADS-B system based on MPLS VPN [Електронний ресурс] – Режим доступу https://ieeexplore.ieee.org/document/9339279
16. Network Automation for CE Router with Route Leaking in MPLS-VPN Network [Електронний ресурс] – Режим доступу https://ieeexplore.ieee.org/document/9944460
17. Experimental Design and Teaching Research of a MPLS VPN Network Based on BGP [Електронний ресурс] – Режим доступу https://ieeexplore.ieee.org/document/10107884
18. Research on the application of cross-domain VPN technology based on MPLS BGP Network [Електронний ресурс] – Режим доступу https://ieeexplore.ieee.org/document/10148288
19. Configuring MPLS Cloud Providers with Virtual Private Network [Електронний ресурс] – Режим доступу https://link.springer.com/chapter/10.1007/978-981-15-2256-7_76
20. Toward a new SDN based approach for smart management and routing of VPN-MPLS [Електронний ресурс] – Режим доступу networkshttps://www.researchgate.net/profile/Ayoub-Bahnasse/publication/344468197_Toward_a_new_SDN_based_approach_for_smart_management_and_routing_of_VPN-MPLS_networks/links/63218b320a70852150f235b6/Toward-a-new-SDN-based-approach-for-smart-management-and-routing-of-VPN-MPLS-networks.pdf
21. Toward a new SDN based approach for smart management and routing of VPN-MPLS networks [Електронний ресурс] – Режим доступу https://www.researchgate.net/publication/344468197_Toward_a_new_SDN_based_approach_for_smart_management_and_routing_of_VPN-MPLS_networks

# ДОДАТОК

```
using System;
using System.Collections.Generic;

public class Graph
{
    private readonly int _vertices;
    private readonly List<KeyValuePair<int, int>>[] _edges;

    public Graph(int vertices)
    {
        _vertices = vertices;
        _edges = new List<KeyValuePair<int, int>>[vertices];

        for (int i = 0; i < vertices; i++)
        {
            _edges[i] = new List<KeyValuePair<int, int>>();
        }
    }

    public void AddEdge(int startVertex, int endVertex, int weight)
    {
        _edges[startVertex].Add(new KeyValuePair<int, int>(endVertex, weight));
    }

    public int[] ShortestPath(int source)
    {
        int[] distances = new int[_vertices];
        bool[] shortestPathTreeSet = new bool[_vertices];

        for (int i = 0; i < _vertices; i++)
        {
            distances[i] = int.MaxValue;
            shortestPathTreeSet[i] = false;
        }

        distances[source] = 0;

        for (int count = 0; count < _vertices - 1; count++)
        {
            int u = MinDistance(distances, shortestPathTreeSet);
            shortestPathTreeSet[u] = true;

            foreach (var edge in _edges[u])
            {
                int v = edge.Key;
                int weight = edge.Value;

                if (!shortestPathTreeSet[v] && distances[u] != int.MaxValue && distances[u] + weight <
distances[v])
                {
                    distances[v] = distances[u] + weight;
                }
            }
        }
```

```
        return distances;
    }

    private int MinDistance(int[] distances, bool[] sptSet)
    {
        int min = int.MaxValue;
        int minIndex = -1;

        for (int v = 0; v < _vertices; v++)
        {
            if (!sptSet[v] && distances[v] <= min)
            {
                min = distances[v];
                minIndex = v;
            }
        }

        return minIndex;
    }
}

public class Program
{
    public static void Main()
    {
        int vertices = 6;  // Example number of vertices
        Graph g = new Graph(vertices);

        // Adding edges with weights to the graph
        g.AddEdge(0, 1, 2);
        g.AddEdge(0, 2, 4);
        g.AddEdge(1, 2, 1);
        g.AddEdge(1, 3, 7);
        g.AddEdge(2, 4, 3);
        g.AddEdge(3, 5, 2);
        g.AddEdge(4, 5, 5);

        int sourceVertex = 0;  // Starting vertex for finding shortest paths
        int[] distances = g.ShortestPath(sourceVertex);

        Console.WriteLine($"Shortest distances from source vertex {sourceVertex}:");
        for (int i = 0; i < vertices; i++)
        {
            Console.WriteLine($"Distance to vertex {i} is {distances[i]}");
        }
    }
}
```