# Опис використаних технологій і тестування

## Використані технології

1. C++   
 - Основна мова програмування для реалізації серверної частини та обробки клієнтських запитів. Використовується для роботи з низькорівневими API та забезпечення високої продуктивності.

2. Sockets (Winsock2)   
 - Бібліотека для роботи з мережевими підключеннями, що дозволяє реалізувати TCP/IP протоколи. Використовується для встановлення з'єднань між сервером і клієнтами, передачі даних у режимі реального часу.  
  
3. WinHTTP   
 - Використовується для HTTP-запитів, якщо серверна або клієнтська логіка потребує обміну даними з веб-ресурсами.  
  
4. Мультипоточність (Threads)   
 - Реалізація паралельної обробки клієнтських запитів через бібліотеку <thread>. Забезпечує високу продуктивність і асинхронність при роботі з великою кількістю клієнтів.  
  
5. Синхронізація потоків (std::mutex)   
 - Механізм забезпечення безпечного доступу до спільних ресурсів, таких як список підписаних клієнтів, що запобігає конфліктам при обробці одночасних запитів.  
  
6. Windows API   
 - Нативні функції Windows для керування потоками, роботи з мережею та іншими операціями.  
  
7. Microsoft Visual Studio   
 - Інтегроване середовище розробки (IDE), яке використовується для написання, компіляції, налагодження та тестування проєкту. Застосовується для зручності роботи з C++.  
  
8. GUI (Windows Forms)   
 - Використовується для створення клієнтської програми з графічним інтерфейсом. Реалізує інтуїтивно зрозумілий інтерфейс для користувачів із функціями підписки на різні сервіси.  
  
9. Структура проєкту   
 - Логічний поділ коду на модулі:  
 - Server: Реалізує серверну частину (SubscriptionServer.cpp), яка обробляє клієнтські підключення та запити.  
 - Client: Реалізує клієнтську частину (ReaderThread.cpp, MyForm.cpp), яка підключається до сервера для отримання даних.

## Використане тестування

1. Тестування серверної частини  
 - Консольний клієнт:  
 Для перевірки роботи сервера на етапі розробки був створений консольний клієнт, який дозволяв відправляти запити до сервера та отримувати відповіді.  
 Тестування включало:  
 - Встановлення з'єднання між клієнтом і сервером.  
 - Перевірку коректної обробки підключення кількох клієнтів одночасно.  
 - Надсилання запитів на різні сервіси підписок (погода, курси акцій, курси валют).  
  
2. Тестування клієнтської частини  
 - Графічний інтерфейс (Windows Forms):  
 Після успішного тестування сервера був протестований основний інтерфейсний додаток.  
 Перевірялися такі аспекти:  
 - Підключення до сервера з GUI.  
 - Вибір і підписка на різні сервіси через графічний інтерфейс.  
 - Відображення отриманих даних (погода, курси акцій, курси валют) у відповідних елементах GUI.  
 - Перевірка коректності введення даних у полях інтерфейсу.  
 - Тестування стабільності клієнта при частих або некоректних запитах.  
 - Функціональне тестування:  
 Перевірка повного циклу взаємодії клієнта і сервера, включаючи ініціалізацію підключення, підписку, отримання даних та відключення.