**Клиент WhatsApp NET**

**Демонстрационное приложение WhatsApp Client для разработчиков .NET**

**Что такое клиент WhatsApp NET?**

[WhatsApp NET Client](http://wa-net.coding4ever.net/) - это бесплатное демо-приложение для разработчиков .NET, которое используется для разработки клиентских приложений WhatsApp на основе настольных компьютеров. Эта библиотека была разработана на технологии .NET с использованием языка программирования C#, поэтому ее также можно использовать для всех языков программирования .NET, кроме C#, таких как VB.NET, F#, C++ и других языков .NET.

В своей разработке WhatsApp NET Client использует [whatsapp-web.js](https://github.com/pedroslopez/whatsapp-web.js/) в качестве движка для связи с WhatsApp Web.

Итак, в заключение, используя этот пример, вы можете легко создавать приложения, которые требуют связи с веб-приложением WhatsApp, такие как:

* WhatsApp Blast
* [WhatsApp Бот](https://github.com/k4m4r82/OnlineTestWABot)
* И другие клиентские приложения WhatsApp

**Особенности**

* Полный *стелс-режим* (безголовый/без окна). Вы можете попрощаться с браузером chrome / firefox, который обычно запускает WhatsApp Web.
* Автоматически сохраняет сеансы входа в систему (поэтому сканируйте QR-код WA только один раз)
* Поддерживает использование [нескольких учетных записей WA,](https://github.com/WhatsAppNETClient/WhatsAppNETClientMultiAccount) примеры, которые вы можете увидеть на <https://github.com/WhatsAppNETClient/WhatsAppNETClientMultiAccount>
* Захват контактов для чтения контактов WA, чтобы результаты можно было сохранить в базе данных
* Захват групп и членов для чтения данных группы и ее членов, чтобы результаты также могли быть сохранены в базе данных.
* Отправка личных или групповых сообщений
* Отправка нескольких сообщений (широковещательная передача)
* Отправка сообщений с изображениями, аудио, видео и всеми видами документов
* Отправляйте сообщения с изображениями, аудио, видео и всеми видами документов через URL
* Отправка сообщений по типу местоположения
* ReplyMessage (цитируемое сообщение)
* Может хранить изображения, аудио, видео, все типы документов, включая vcards из входящих сообщений (вы можете указать место хранения самостоятельно)
* Может также читать сообщения с типом vcardиместоположением
* Может читать сообщения из групп и получать информацию об отправителю
* Подписка на события ChangeState для отслеживания изменений состояния подключения
* Подписаться на входящие сообщения, поэтому вам не нужно добавлять еще один объект таймера для чтения входящих сообщений
* Подпишитесь на отправленные сообщения, с помощью этой функции мы можем проверить, было ли отправленное сообщение успешным или неудачным
* Архив чата
* Удалить чат
* Выход
* Может быть легко интегрирован со всеми типами баз данных

**Требования к системе**

* Windows 8, 10 и последние версии Windows
* .NET Framework 4.5 и последняя версия .NET
* Узел.js версии 14.16.x или более поздней
* [Программный git](https://git-scm.com/downloads) (управление версиями)

**Установка**

**1. Установка Node.js**

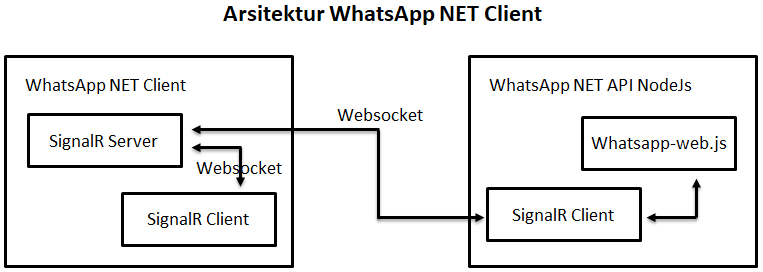
Зачем устанавливать Node.js? Потому что при разработке этого клиента WhatsApp NET задействован [whatsapp-web.js](https://github.com/pedroslopez/whatsapp-web.js/) в качестве движка для связи с WhatsApp Web, поэтому Node.js был необходим, потому что [whatsapp-web.js](https://github.com/pedroslopez/whatsapp-web.js/) сам был разработан на этой платформе.

Установить Node.js очень просто, вам просто нужно скачать установочный файл [здесь,](https://nodejs.org/en/download/) выбрав подходящую версию. После этого просто выполните установку и следуйте инструкциям.

**2. Склонируйте репозиторий WhatsAppNETAPINodeJs**

[WhatsApp NET API NodeJs](https://github.com/WhatsAppNETClient/WhatsAppNETAPINodeJs) - это небольшая программа, которую я написал с использованием языка программирования java script, в котором есть модуль для доступа к [библиотекам Whatsapp-web.js](https://github.com/pedroslopez/whatsapp-web.js/) и [node-signalr.](https://www.npmjs.com/package/node-signalr)

* [Whatsapp-web.js работает](https://github.com/pedroslopez/whatsapp-web.js/) для общения с WhatsApp Web.
* [Узел-сигнальщик](https://www.npmjs.com/package/node-signalr) служит клиентом signalR, который отвечает за связь с сервером signalR, который я установил в приложении WhatsApp NET Client.

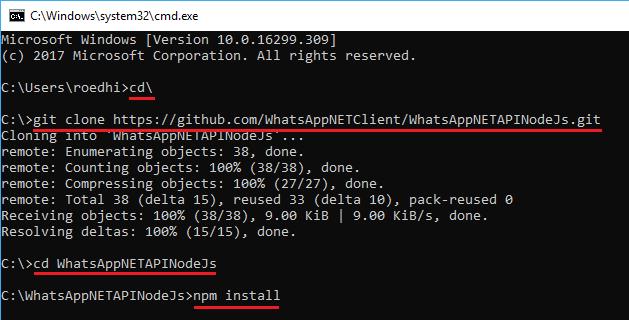


Чтобы клонировать [репозиторий WhatsApp NET API NodeJs,](https://github.com/WhatsAppNETClient/WhatsAppNETAPINodeJs) вам просто нужно загрузить программное обеспечение [git](https://git-scm.com/downloads) и установить его.

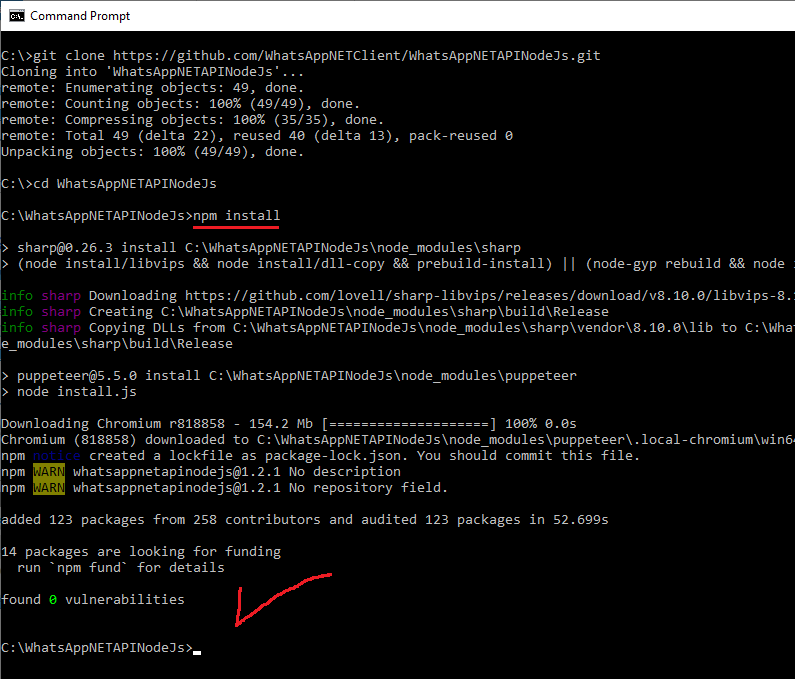
После этого запустите командную строку, затем перейдите на диск C или D, затем введите следующую команду:

https://github.com/WhatsAppNETClient/WhatsAppNETAPINodeJs.git

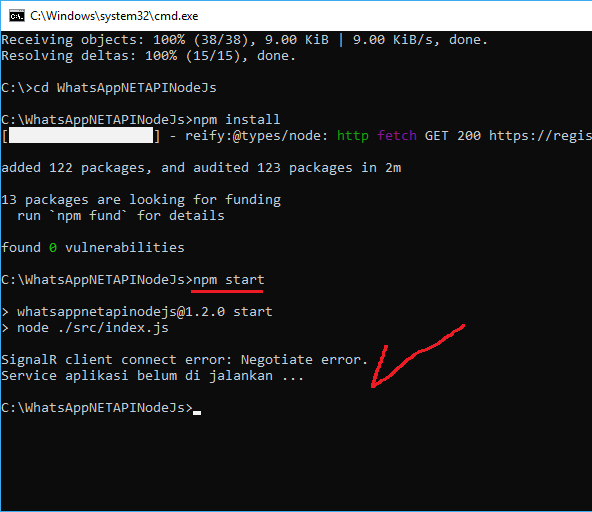
Не забудьте нажать enter, хехе.   
После этого продолжайте, набрав команду cd WhatsAppNETAPINodeJs, затем введите последнюю команду, npm install и еще раз нажмите клавишу Enter.



Затем дождитесь завершения установки необходимых модулей Node.js. Что касается видов экрана, процесс более или менее похож на приведенный на скриншоте. Но, возможно, на вашем ПК / ноутбуке вы получите несколько другой вид... ;-)



Тогда что делать, если вы хотите протестировать установку успешно или нет? Это просто, вы просто вводите команду npm start, если результат более или менее похож на рисунок ниже, я предполагаю, что установка прошла успешно :-).



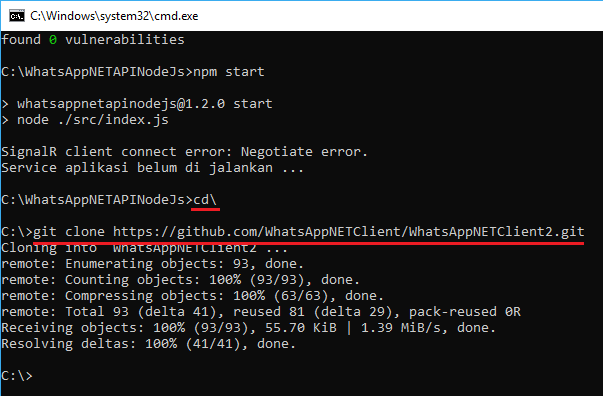
Давайте просто перейдем к следующему шагу :-)

**3. Клонирование репозитория с примером использования клиента WhatsApp NET**

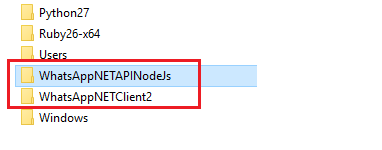
Эти шаги на самом деле такие же, как и раньше просто вам нужно изменить каталог, но по сути это то же самое. Но для шагов здесь я все еще использую диск C для клонирования репозитория для [примера WhatsApp NET Client.](https://github.com/WhatsAppNETClient/WhatsAppNETClient2)

Поскольку наша позиция теперь находится в каталоге C:\WhatsAppNETAPINodeJs,мы должны сначала вернуться к корню диска C, введя команду cd\,а затем ввести следующую команду:

https://github.com/WhatsAppNETClient/WhatsAppNETClient2.git

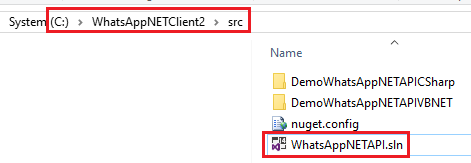


Хорошо, здесь означает, что шаги установки были завершены, если вы откроете проводник Windows, а затем переместитесь на диск C, он получит два новых каталога с именем WhatsAppNETAPINodeJsи WhatsAppNETClient2,чтоявляется результатом предыдущих шагов.



**Открытие проекта примера использования клиента WhatsApp NET**

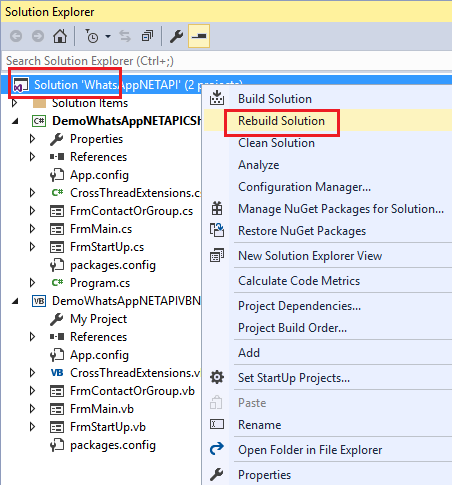
Это просто, просто откройте файл WhatsAppNETAPI.sln в C:\WhatsAppNETClient2\src.



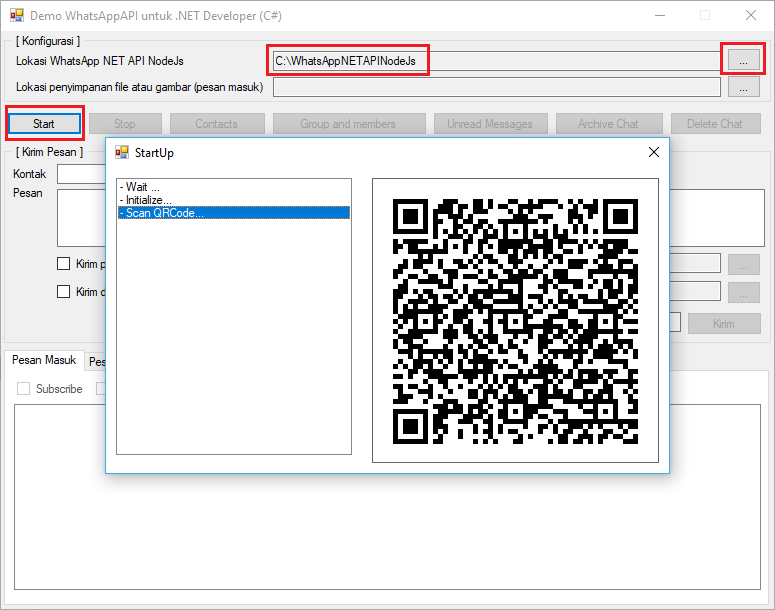
Пример использования клиентской библиотеки WhatsApp NET, которую я создал на двух языках, а именно C# и VB.NET.

Насколько я помню, создавал этот проект я с помощью Visual Studio .NET 2015 Community, поэтому, по крайней мере, вы должны были установить Visual Studio .NET 2015 Community. А вообще будет лучше, если установите более новую версию типа 2022, так как заканчивается 2021 год... :-)

Хорошо, после открытия WhatsAppNETAPI.sln, щелкните правой кнопкой мыши решение WhatsAppNETAPI-> Rebuild Solution. Этот шаг направлен на восстановление всех [пакетов nuget, используемых](https://www.nuget.org/packages/WhatsAppNETAPI)  в этом проекте.



После этого вы можете сразу запустить проект, нажав клавишу F5, затем установить расположение каталога WhatsAppNETAPINodeJs, которое является результатом предыдущих шагов, затем нажать кнопку *Пуск.*



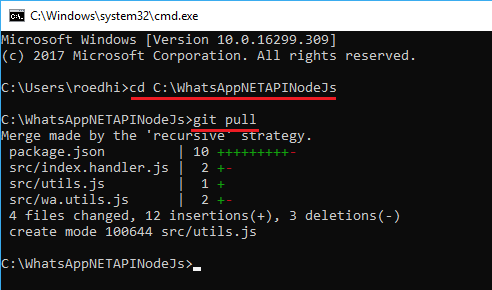
**Обновление проекта**

**1. Обновление репозитория WhatsAppNETAPINodeJs**

Чтобы обновить [репозиторий WhatsApp NET API NodeJs](https://github.com/WhatsAppNETClient/WhatsAppNETAPINodeJs) очень просто, вы запускаете его сначала командную строку, а затем входите в каталог WhatsAppNETAPINodeJs с помощью командного компакт-диска,после чего введите следующую команду:

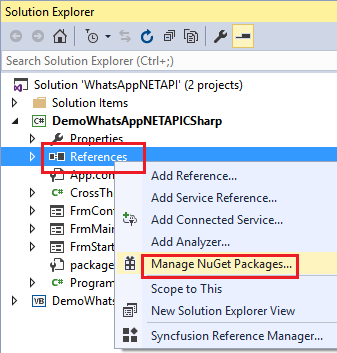
git pull

Затем дождитесь завершения процесса обновления. В случае успеха процесс выглядит примерно так:

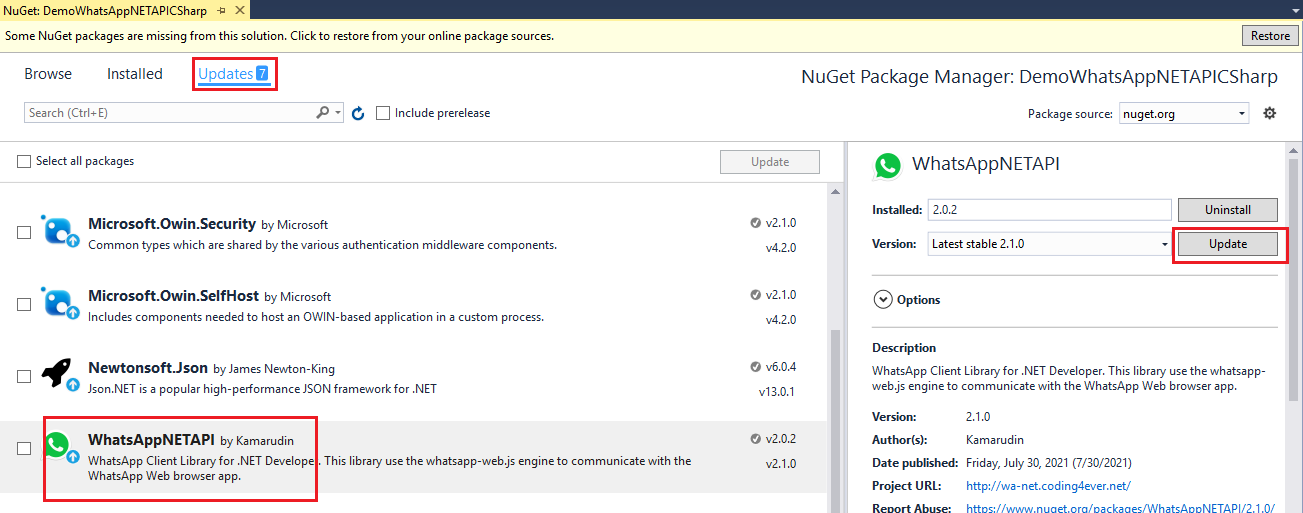


**2. Обновление project .NET**

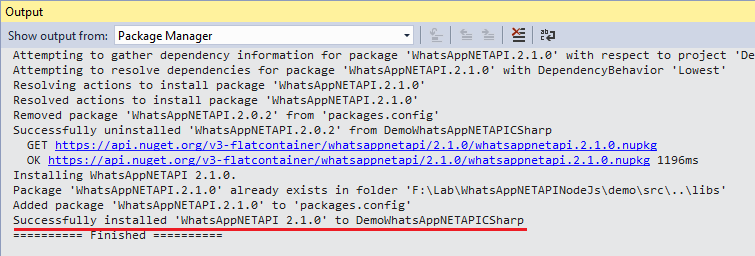
Следующим шагом является обновление содержимого пакета [WhatsAppNETAPI](https://www.nuget.org/packages/WhatsAppNETAPI/)  в вашем проекте .NET. Хитрость заключается в том, чтобы сначала открыть проект, а затем щелкнуть правой кнопкой мыши Ссылки->Управление пакетом Nuget...



Затем в окне Package Nugget вы выбираете панель Обновленияи находите пакет [WhatsAppNETAPI,](https://www.nuget.org/packages/WhatsAppNETAPI/)  затем нажимаете кнопку Обновить.



Дождитесь завершения процесса обновления. Выглядит это примерно так:

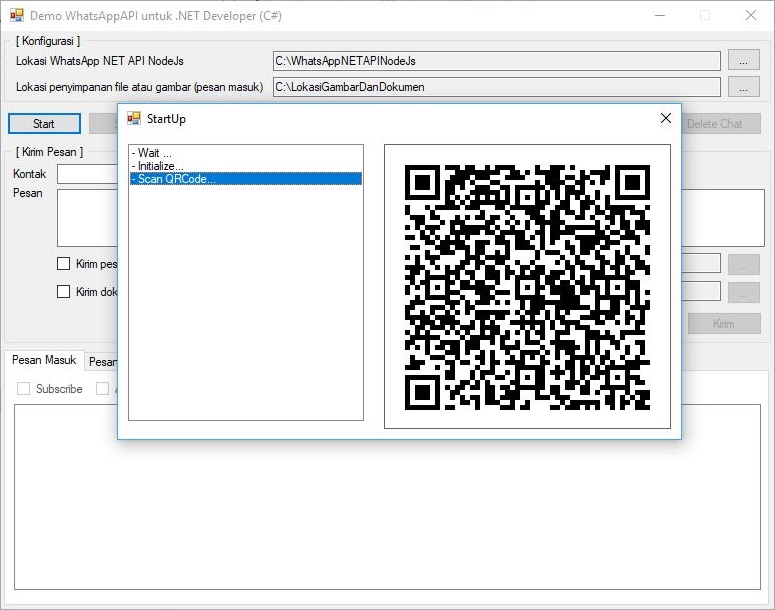


**Сообщение об ошибке или ошибке**

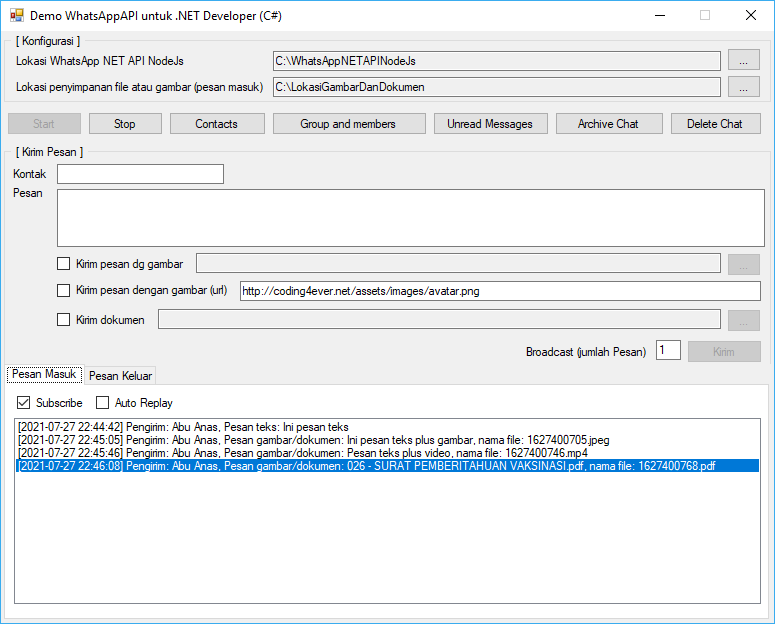
Технически при разработке приложения явно невозможно быть на 100% свободным от багов. Поэтому, если вы обнаружите ошибку или ошибку при использовании клиентской библиотеки WhatsApp NET, пожалуйста, сообщите об этом на странице [https://github.com/WhatsAppNETClient/WhatsAppNETClient2/issues.](https://github.com/WhatsAppNETClient/WhatsAppNETClient2/issues)

**Пример использования клиентской библиотеки WhatsApp NET**

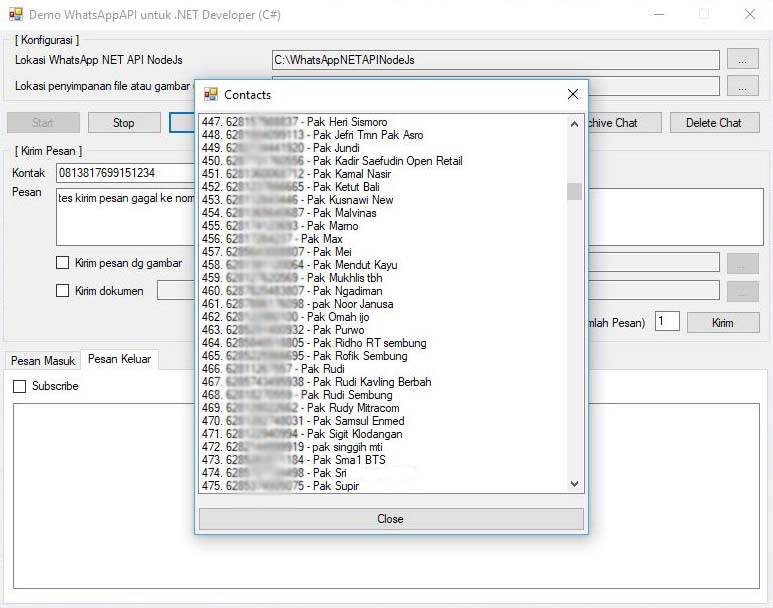
Начального сканирования QRCode достаточно, для следующего подключения он больше не нужен, так как эта библиотека автоматически сохраняет сеанс аутентификации.



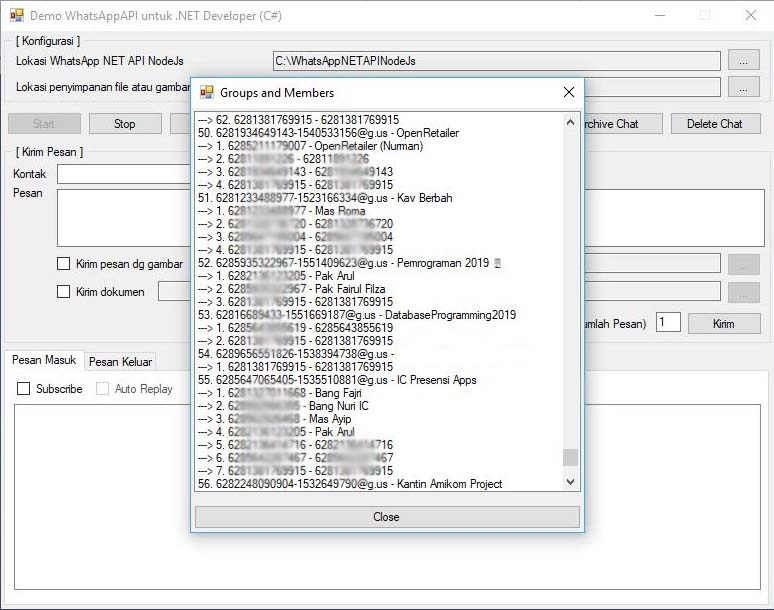
Например, подписываясь на входящее сообщение, вы можете получить некоторую информацию, такую как дата/ время, номер отправителя / имя, сообщение и имя файла, если сообщение содержит изображения, аудио, видео и другие файлы документа.



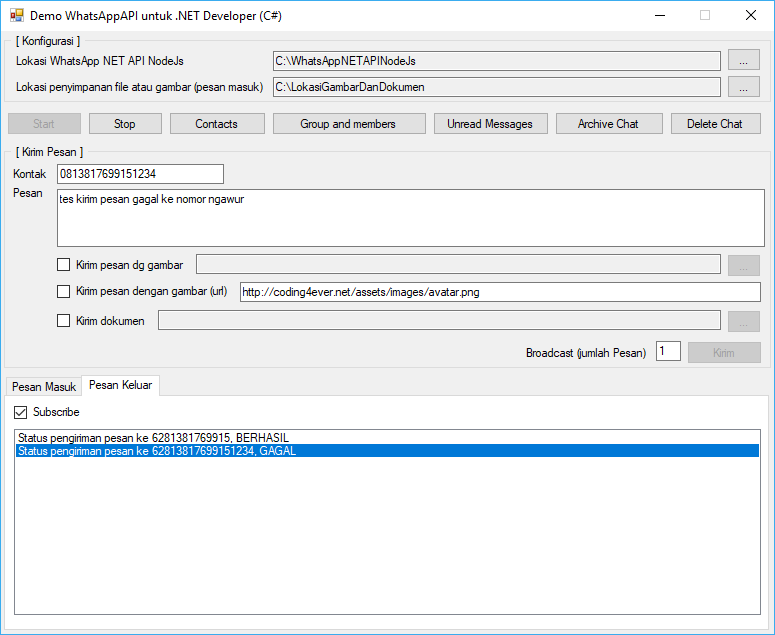
Пример результатов функции захвата контактных данных, в этой функции вы можете получить контактный номер / имя, краткое имя и проверенное имя.



Пример результатов функции захвата группы и ее членов, в этой функции вы можете получить информацию о названии группы и ее членах (контактный номер /имя, краткое имя и проверенное имя).



Пример подписки на исходящее сообщение, в этой функции вы можете получить статус отправки сообщения, было ли оно успешным или неудачным (контактный номер не был зарегистрирован в WhatsApp).



**Информация о выпуске**

**v2.3.0 (2021-08-19)**

* Новые возможности:
  + Добавлены [функции WA с несколькими учетными записями](https://github.com/WhatsAppNETClient/WhatsAppNETClientMultiAccount)

В версии [2.3.0](https://www.nuget.org/packages/WhatsAppNETAPI/2.3.0) во всех существующих обработчиках событий есть дополнительный параметр sessionId,поэтому его приходится настраивать вручную.

Примеры некоторых настраиваемых обработчиков событий:

**private void OnStartupHandler(строковое** сообщение)

**приватный void OnScanMeHandler**(**строка** qrcodePath)

**приватный void OnReceiveMessageHandler**(whatsAppNETAPI.Message message)

Все, что вам нужно сделать, это добавить параметр строкитипа sessionIdследующим образом:

**private void OnStartupHandler**(строковое сообщение, **строка** sessionId)

**private void OnScanMeHandler**(**строка** qrcodePath, **строка** sessionId)

**private void OnReceiveMessageHandler**(сообщение WhatsAppNETAPI.Message, **строковый** сеансId)

**v2.2.0 (2021-08-12)**

* Новые возможности:
  + Добавлена возможность получения информации о группе без участника
  + Добавлена возможность отправки местоположения
  + Добавлена функция выхода из системы
* Исправлены ошибки:
  + Подписаться на статус исходящих сообщений, отправляемых в группу
* Улучшать:
  + Прочитав сообщения из группы, уже можно получить информацию об отправителю сообщения
  + Добавлена информация о версии WhatsApp Web и Chrome при инициализации

**v2.1.0 (2021-07-30)**

* Новые возможности:
  + Добавлена функция ReplyMessage (цитируемое сообщение)
  + Добавлена возможность отправлять графические, аудио, видео и документальные сообщения по URL
  + Добавлена возможность сохранения сообщений vcard в виде файла
  + Добавлено событие ChangeState для отслеживания изменений состояния подключения
* Улучшать:
  + Чтение входящих сообщений с типом locationи vcard(данные уже в виде объекта)

**v2.0.2 (2021-07-25)**

* Заменил библиотеку [Selenium Webdriver](https://www.selenium.dev/documentation/en/webdriver/) на библиотеку [whatsapp-web.js](https://github.com/pedroslopez/whatsapp-web.js/) в качестве основного движка для взаимодействия с WhatsApp Web.
* *Полнорежимный силуман* (безголовый/без окна)
* Несовместим с примерами проектов, в которые по-прежнему используется библиотека версии 1.3.x
* Захват групп и членов для чтения данных группы и ее членов, чтобы сохранить результаты в базе данных
* Улучшена функция отправки сообщений с изображениями, аудио, видео и всеми типами документов
* Может сохранять изображения, аудио, видео и все типы документов из входящих сообщений (вы можете указать место хранения самостоятельно)
* Может также читать сообщения с типом контактаиместоположением
* Архив чата
* Удалить чат

**v1.3.3 (2021-02-02)**

* Незначительное обновление

**v1.3.2 (2020-11-22)**

* Исправлены ошибки:
  + Отправляйте сообщения, содержащие *эмодзи,* в *скрытом режиме* (безголовый/без окна).
* Улучшать:
  + Метод прослушивания/подписки для входящих сообщений.
  + Способ отправки сообщения

**v1.3.1 (2020-07-20)**

* Исправлены ошибки:
  + Отправив фотографию контакта, который не зарегистрирован в WA contacts, отправленное сообщение становится дубликатом. Пример: Тест сообщения, отправленный как  **тесттест.**
* Улучшать:
  + Режим *силуман* (безголовый/без окна).

**v1.3.0 (2020-07-17)**

* Новые возможности:
  + Функция подписки на сообщение отправляется, с помощью этой функции мы можем проверить, было ли отправлено сообщение или нет.
* Исправлены ошибки:
  + Отправив сообщение контакту, который не зарегистрирован в WA contacts, отправленное сообщение становится дубликатом. Пример: Тест сообщения, отправленный как  **тесттест.**

**v1.2.0 (2020-07-13)**

* Новые возможности:
  + Добавлен *скрытый* режим (безголовый/без окна)
  + Добавлена функция захвата контактов
* Улучшать:
  + Method SendMessageAsync, BroadcastMessage dan ListenIncomingMessageAsync

**v1.1.0 (2020-07-05)**

* Новые возможности:
  + Добавлена возможность отправлять сообщения с картинками
  + Добавлена возможность отправки файлов документов
* Улучшать:
  + Добавлена поддержка .NET Framework 4.0
  + Функция подписки на входящие сообщения
* Исправлены ошибки:
  + Отправка контакта с недопустимым номером
  + Отправка многостроочных сообщений во время трансляции
* Устаревшие:
  + Свойство больше не используется, поэтому просто отключите его в старом коде
  + Метод больше не используется, пожалуйста, используйте метод *SendMessage* вместо этого

**v1.0.5 (2020-06-24)**

* Исправлены ошибки:
  + Отправка многострочных сообщений
* Улучшать:
  + Функция подписки на входящие сообщения

**v1.0.0 (2020-06-22)**

* Первый релиз
* Отправка сообщения
* Отправка нескольких сообщений (широковещательная передача)
* Прослушивание/подписка на входящие сообщения, поэтому нет необходимости снова добавлять объект таймера для чтения входящих сообщений
* Автоматическое воспроизведение входящих сообщений
* Автоматический вход (поэтому нет необходимости сканировать WA QRCode туда и обратно, только один раз)
* Все существующие методы, кроме метода подключения к браузеру WA/Chrome, сделаны асинхронными (асинхронно), поэтому они не блокируют пользовательский интерфейс при выполнении.
* Поддержка всех типов баз данных

© 2020 - 2021 Камарудин. На базе темы [Jekyll](https://jekyllrb.com/) & [The SlateКодирование блога4ever](https://github.com/pages-themes/slate/)   
 | [rudi.krsoftware@gmail.com](mailto:rudi.krsoftware@gmail.com)

Опубликовано с ❤️ GitHub Pages