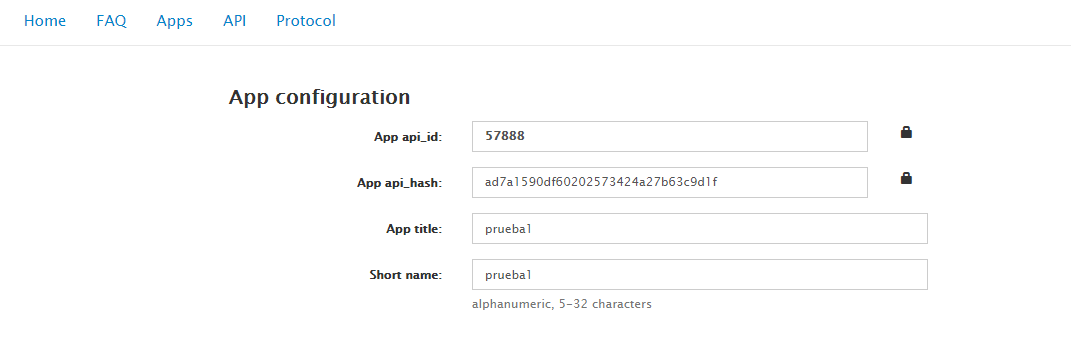
# **Pruebas de interacción con Telegram**

## **Herramienta utilizada**

* **TLSharp** – Librería implementada en C# que proporciona las herramientas básicas para interactuar con Telegram y enviar información a otros usuarios

## **Requisitos para el uso de la herramienta**

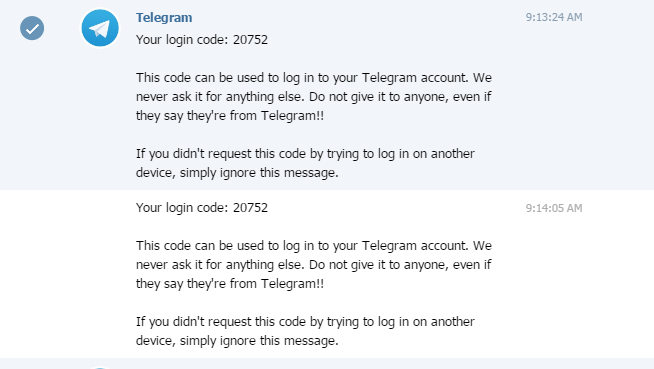
* Tener una cuenta y número telefónico registrados en Telegram
* Crear una cuenta de desarrollador (Ver Enlaces) para obtener los valores de API\_ID y API\_HASH



## **Herramientas disponibles en TLSharp**

Las siguientes herramientas requieren los valores API\_ID y API\_HASH obtenidos anteriormente, además del número telefónico asociado a los mismos

* Solicitar código de autenticación de usuario
  + Solicita un código de autenticación el mismo que será enviado a la cuenta Telegram asociada a un número telefónico.



* Autenticación de usuario
  + Permite asociar la aplicación a la cuenta registrada con el número telefónico
  + Requiere el código de autenticación ya solicitado
* Obtener la lista de contactos del usuario
  + Para realizar envíos de información de cualquier tipo (mensajes, imágenes, archivos, etc.) es necesario que el usuario destino se encuentre en esta lista de contactos
* Enviar mensajes
  + El número telefónico/usuario destino debe estar en la lista de contactos
* Enviar mensajes a canales
  + Se pueden enviar hasta un máximo de 30 mensajes por segundo hacia Telegram
* Enviar fotos/imágenes
  + Se pueden enviar imágenes de cualquier tamaño pero el método para enviarlas varía si el archivo pasa los 10mb
* Enviar archivos
  + Mismo procedimiento que con las imágenes
* Recuperar el historial de charlas de usuario telegram

## **Instalación**

* Instalable desde el administrador de NuGets (nombre TLSharp)
* Métodos disponibles:
  + IsPhoneRegisteredAsync
  + SendCodeRequestAsync
  + MakeAuthAsync
  + SignUpAsync
  + GetContactsAsync
  + SendMessageAsync
  + SendTypingAsync
  + GetUserDialogsAsync
  + SendUploadedPhoto
  + SendUploadedDocument
  + GetFile
  + UploadFile
  + SendPingAsync
  + GetHistoryAsync

## Prototipo realizado

# **Enlaces**

* Registrarse como desarrollador - <https://my.telegram.org/apps>
* Herramienta TLSharp - <https://github.com/sochix/TLSharp>