

Servicios Multimedia: Telegram Bot

Introducción

Telegram

Telegram es una plataforma de mensajería y VOIP de origen ruso, desarrollada por los hermanos Nikolái y Pável Dúrov. La aplicación está enfocada en la mensajería instantánea, el envío de varios archivos y la comunicación en masa. Se anunció oficialmente el 14 de agosto de 2013,4 tiempo después del autoexilio de los hermanos Dúrov y antiguos compañeros de VK a otros países por posturas opuestas al gobierno de Rusia. A julio de 2021 una organización autofinanciada que lleva su nombre gestiona su desarrollo, cuyo centro de operaciones está en Dubái, Emiratos Árabes, y su matriz está registrada legalmente en las Islas Vírgenes Británicas.

Bots

Los *bot de charla* o *bot conversacional* (del inglés *chatbot*) es una aplicación software que surgen en los años 60, y que simulan mantener una conversación con una persona al proveer respuestas automáticas, las cuales son previamente establecidas por un conjunto de expertos a entradas realizadas por el usuario.

Actualmente con las aplicaciones de mensajería como *WhatsApp* o *Telegram* los *chatbots* se utilizan para muchas tareas, desde buscar documentación de un lenguaje de programación, conocer el tiempo meteorológico o estar informado de ofertas en tiendas o juegos.

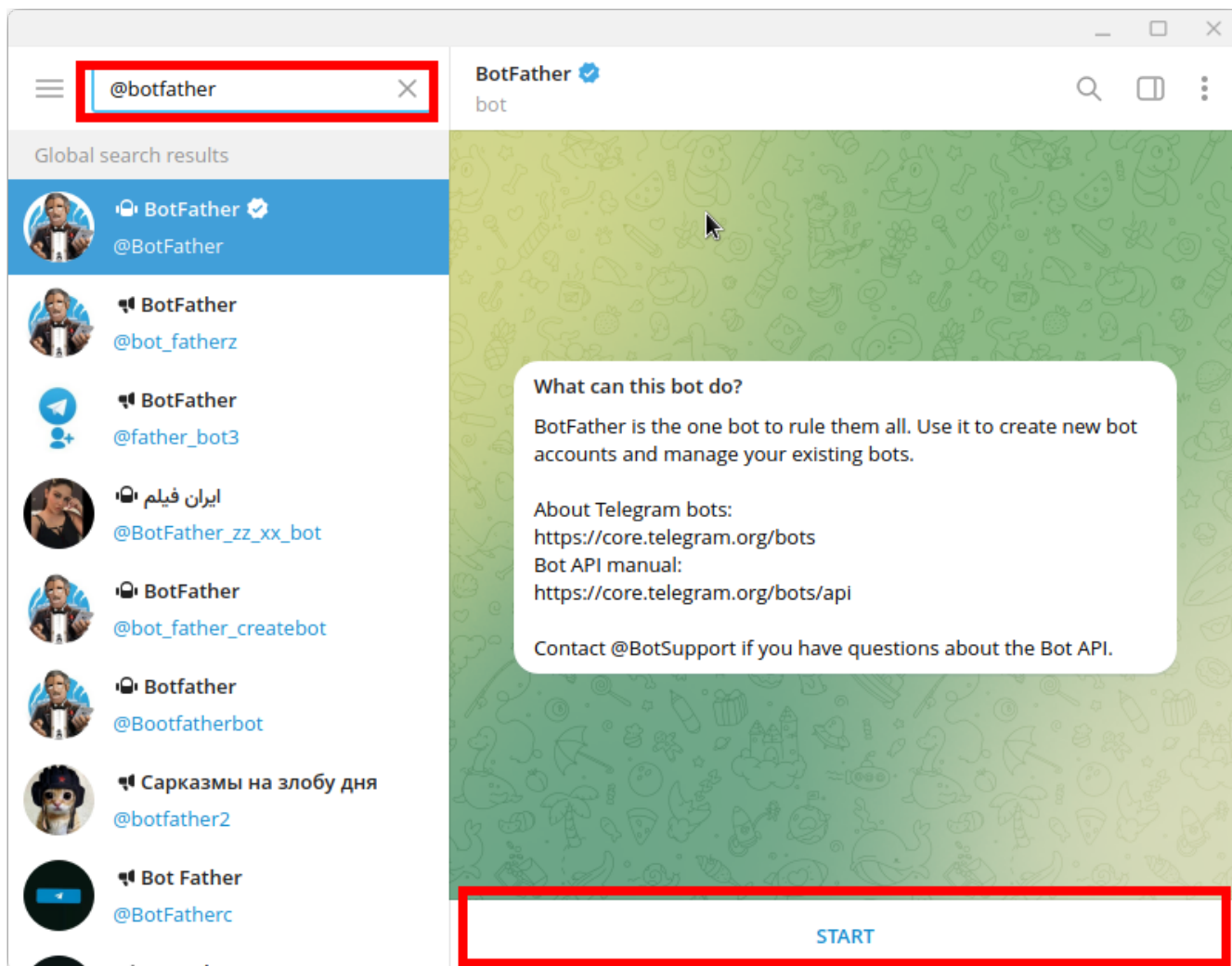
¿Por qué este nuevo *boom* con los *chatbots*? El texto usado en los chats es un interfaz universal que permite interactuar con las aplicaciones de una forma sencilla.

Obtener un token para nuestro bot

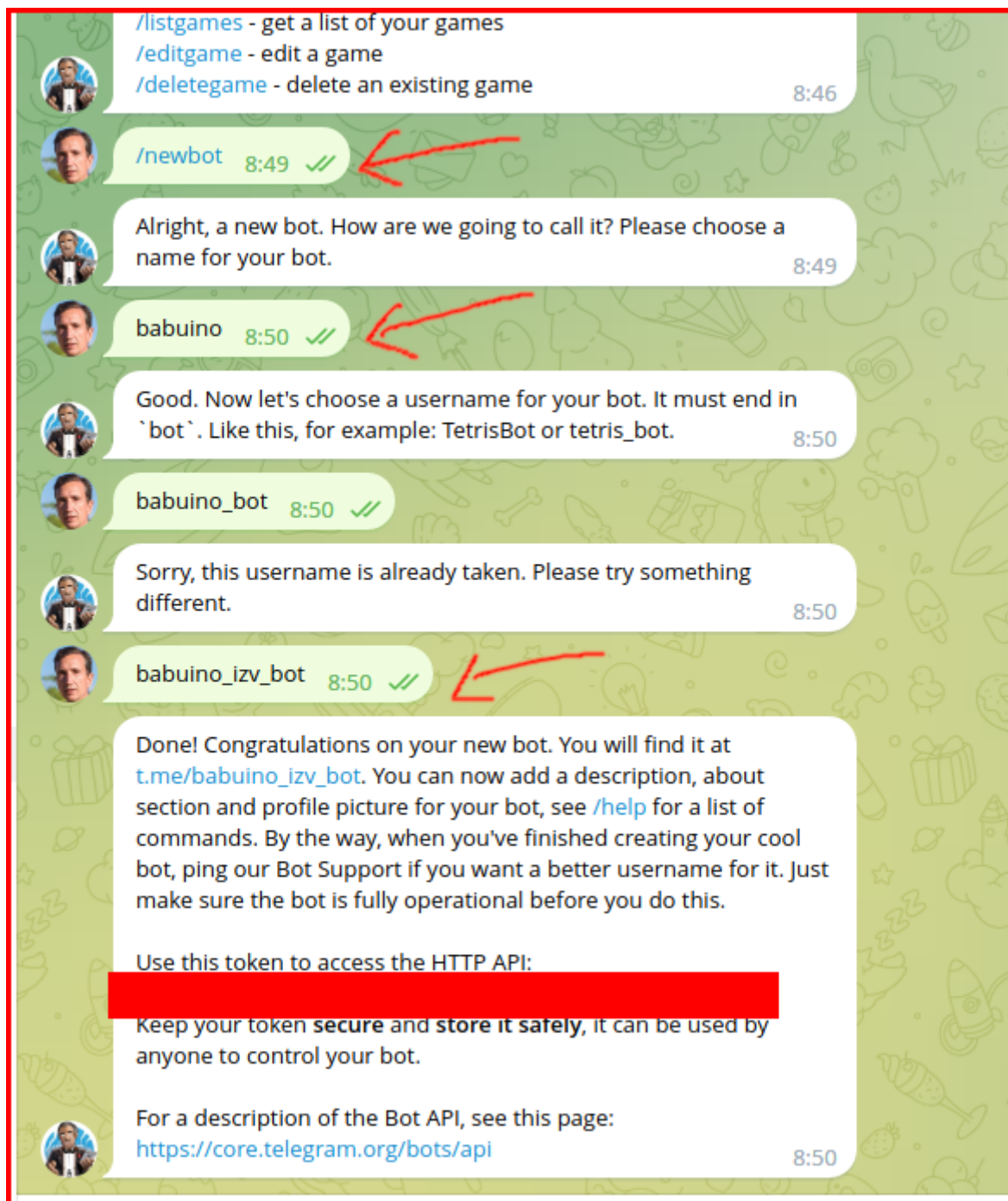
La mayoría de los sistemas modernos funcionan con autenticación tipo *OAuth*. *Open Authentication* es un estándar abierto que permite compartir información sin que el proveedor y consumidor compartan toda su identidad. El proveedor crea un *token* criptográfico y el cliente que lo posea está autorizado a realizar ciertas operaciones.

Nuestro primer objetivo es por tanto obtener un token para que nuestro bot pueda comunicarse. El programa encargado de darnos permiso es el *@botfather* (el padrino de los bots).

1. Buscaremos en Telegram *@botfather*



1. Comenzaremos una conversación con `@botfather` pulsando el botón *START*
2. Escribiremos `/newbot`.
3. Nos preguntará el nombre del bot. Por ejemplo el mío se llamará *babuino*
4. Luego nos pregunta el nombre de usuario del bot, que debe terminar en *bot*. Por ejemplo lo llamaré *babuino_bot*. Este nombre estaba ocupado así que elijo *babuino_izv_bot*.
5. Nuestro bot estará accesible en `t.me/<nombre-usuario-bot>`. En mi caso `t.me/babuino_izv_bot`
6. Nos concederá un token para autenticarnos y acceder a través del API web.



Asegurát de guardar el token en lugar seguro o cualquiera podría manipular a nuestro bot.

Crear el primer bot

Crearemos nuestro primero bot con ayuda de Docker. Crearemos una imagen para que la instalación de las herramientas sea lo más sencilla posible.

Token del API

Guardaremos en un fichero de texto llamado por ejemplo `token.txt` el token que nos proporcionó *BotFather*

token.txt

```
BOT_TOKEN=pon-tu-token
```

Librería de python para Telegram API

Crearemos un fichero de texto llamado `requirements.txt` que contendrá una línea por cada librería que necesitemos en Python.

Example 1. `requirements.txt`

```
pyTelegramBotAPI
```

Dockerfile

Ahora crearemos un fichero Dockerfile para crear nuestra imagen.

Example 2. `Dockerfile`

```
FROM python:3 ①  
WORKDIR /usr/src/app ②  
COPY requirements.txt requirements.txt ③  
RUN pip install --no-cache-dir -r requirements.txt ④  
COPY ./bot.py . ⑤  
CMD ["python", "./bot.py"] ⑥
```

- ① Partimos de la imagen oficial de Python (en docker hub)
- ② El directorio de trabajo de nuestra aplicación
- ③ Copiamos las librerías requeridas al contenedor
- ④ Instalamos las librerías que necesitamos
- ⑤ Copio el bot (ahora lo haremos)
- ⑥ El punto de entrada del contenedor es ejecutar nuestra aplicación

Programando bots

Ahora tenemos que programar nuestro primer bot en el fichero `bot.py`.

Example 3. `bot.py`

```
import os ①  
import telebot ②
```

```

BOT_TOKEN = os.environ.get('BOT_TOKEN') ③
bot = telebot.TeleBot(BOT_TOKEN) ④

# Registrar manejadores de mensajes

@bot.message_handler(commands=['start', 'hola']) ⑤
def enviar_bienvenida(mensaje): ⑥
    bot.reply_to(mensaje, "Hola, ¿como estás?")

@bot.message_handler(func=lambda msg: True) ⑦
def enviar_echo(mensaje):
    bot.reply_to(mensaje, mensaje.text)

# Escuchar peticiones

bot.infinity_polling()

```

- ① Importamos `os` para leer la variable del entorno
- ② Importamos las librería de Telegram
- ③ Tomamos la variable de entorno que contiene el token
- ④ Creamos el `bot`
- ⑤ Registramos un manejador para los mensajes `/start` y `/hola`
- ⑥ Cualquier nombre es válido para un manejador
- ⑦ Repetiremos lo que el usuario escriba, para ello siempre devolvemos `True` desde la función anónima (`lambda`)

Iniciar el servicio

Crearemos nuestra imagen con Docker, ejecutando el siguiente comando en el directorio donde se sitúa nuestro `Dockerfile`

```
docker build -t telegram-bot-app .
```

Esto construirá nuestra imagen, instalará los paquetes necesarios y copiará la aplicación. Podremos comprobar con `docker image ls` que tenemos creada una imagen llamada `telegram-bot-app:latest`.

Ejecutaremos el servidor

```
docker run -dti --env-file token.txt --name babuino-bot telegram-bot-app
```

Hemos pasado a las variables de entorno el contenido del fichero `token.txt` con el argumento `--env-file`. El nombre de nuestro contenedor será `babuino-bot` y la imagen que usamos para crearlo será

telegram-bot-app que creamos anteriormente.

Puedes ver el contenedor ejecutándose con `docker ps`

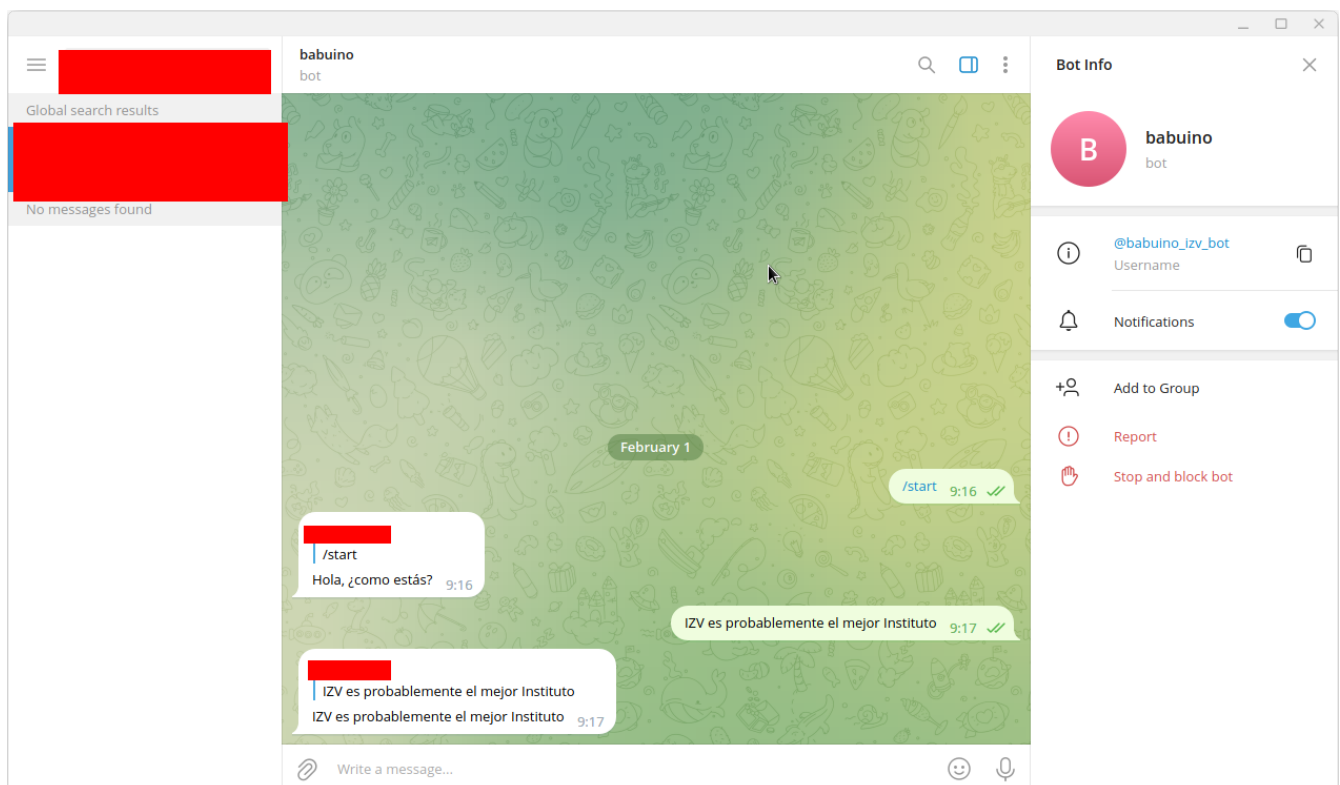
```
fraya@pc129-15:~/telegram-bot/docker$ docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
c2c52e5d0c4f	telegram-bot-app	"python ./bot.py"	2 seconds ago	Up 1 second
babuino-bot				

Para parar el contenedor usa `docker stop babuino-bot` y para borrar el contenedor `docker rm babuino-bot`. Si quieres volver a crear la imagen, borra la antigua con `docker rmi telegram-bot-app` y repite los pasos anteriores.

Probando la aplicación

El último paso es probar la aplicación.



En cuanto iniciamos la conversación con el bot se lanza el mensaje `/start` por lo que nuestro bot responde con el saludo. Después, todo lo que digamos es repetido por el bot. Si escribimos `/hola`, de la misma forma recibimos el saludo.

Ahora debes ser tú el que programe a *Skynet*, Happy hacking!!

Derechos de autor

Este documento tiene derechos de autor (c) 2023 por el equipo de profesores del IES Zaidín

Vergeles. Los colaboradores se listan más abajo. Se puede distribuir y modificar bajo los términos de la GNU General Public License versión 3 o posterior o la Creative Commons Attribution License, versión 4.0 o posterior. Todas las marcas registradas mencionadas en esta guía pertenecen a sus propietarios legítimos.

Referencias

1. Colaboradores de Wikipedia. Bot conversacional [en línea]. Wikipedia, La enciclopedia libre, 2022 [fecha de consulta: 25 de enero del 2023]. Disponible en https://es.wikipedia.org/w/index.php?title=Bot_conversacional&oldid=146991806.
2. <https://www.xataka.com/basics/bots-telegram-que-como-funcionan-recomendados-para-empezar>
3. <https://www.freecodecamp.org/news/how-to-create-a-telegram-bot-using-python/>

Colaboradores

- De esta edición: Fernando Raya

Comentarios y sugerencias

Puede dirigir cualquier clase de comentario o sugerencia acerca de este documento a la lista de correo del equipo de documentación: fraya@ieszaidinvergeles.org