

AI Solution for AI needs customer



BotHub.Studio를 활용한 CHATBOT 개발

김선호 미스터마인드 CTO
| seonho.aii@gmail.com

V1.0 | Released in 2018.01

Document by Mr.MIND AI Consulting

Technology for network Appliance _ Understand User,
Create Experience **AI Consulting Inc.**

개요

BotHub.Studio 와 Python으로 챗봇 개발

BotHub.Studio는,
대화방 배포 및 관리를 위한 backend platform
개발자가 엔지니어링 문제가 아닌 자신의 논리에 집중할 수 있도록 돕는 것

1. 어떤 챗봇을 만들 것인가?
 - 영화 상영관 찾기
 - 카페에서 메뉴 주문하기
2. 어떤 메신저를 사용할 것인가?
 - Telegram
 - Facebook Messenger
3. 코드를 어디에서 실행할 것인가?
 - [BotHub.Studio](#)

환경 준비하기

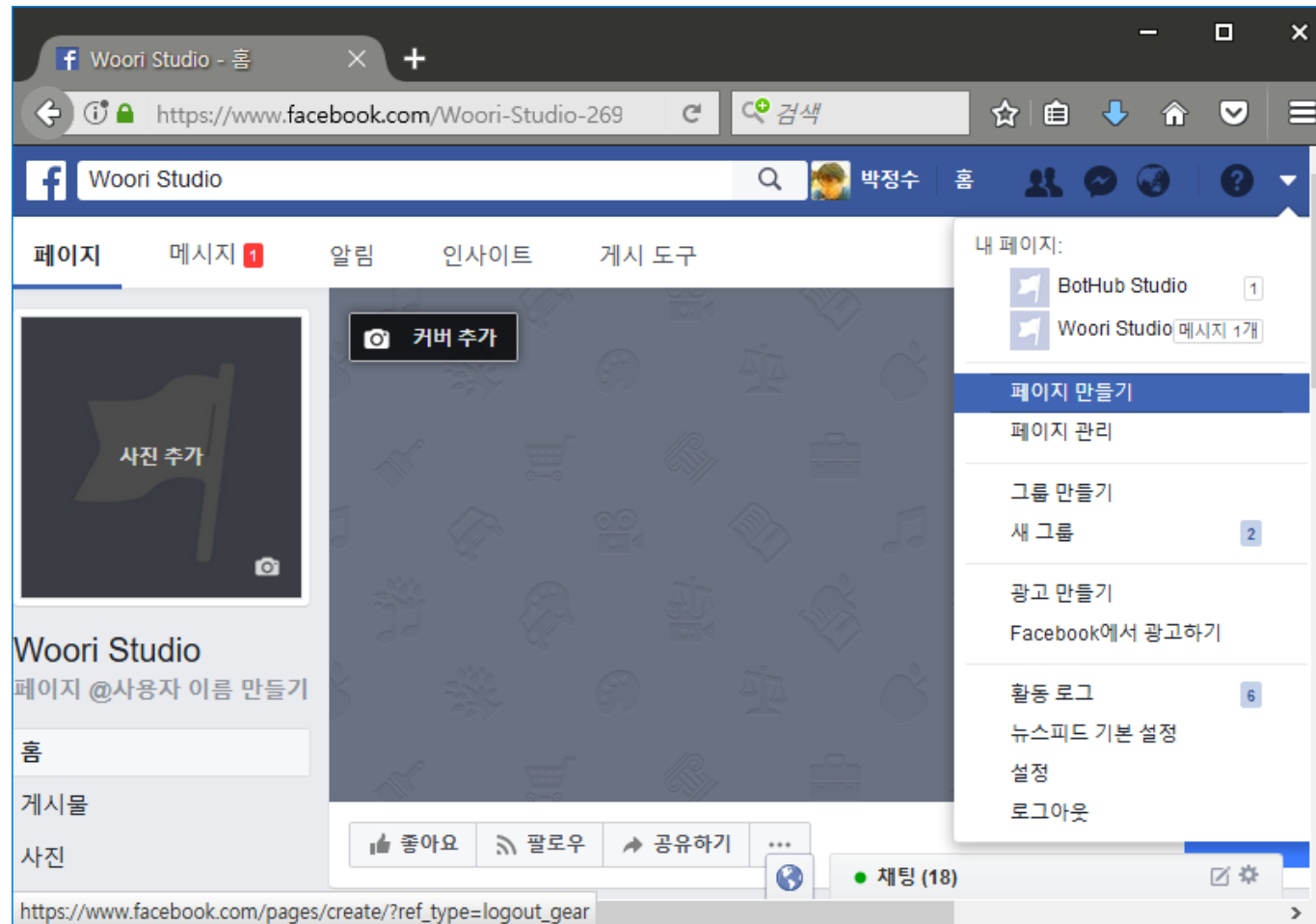
Register a ChatBot account

챗봇을 제공하려는 메시징 플랫폼에 봇을 개설

여기서는 Facebook Messenger와 Telegram의 경우를 설명

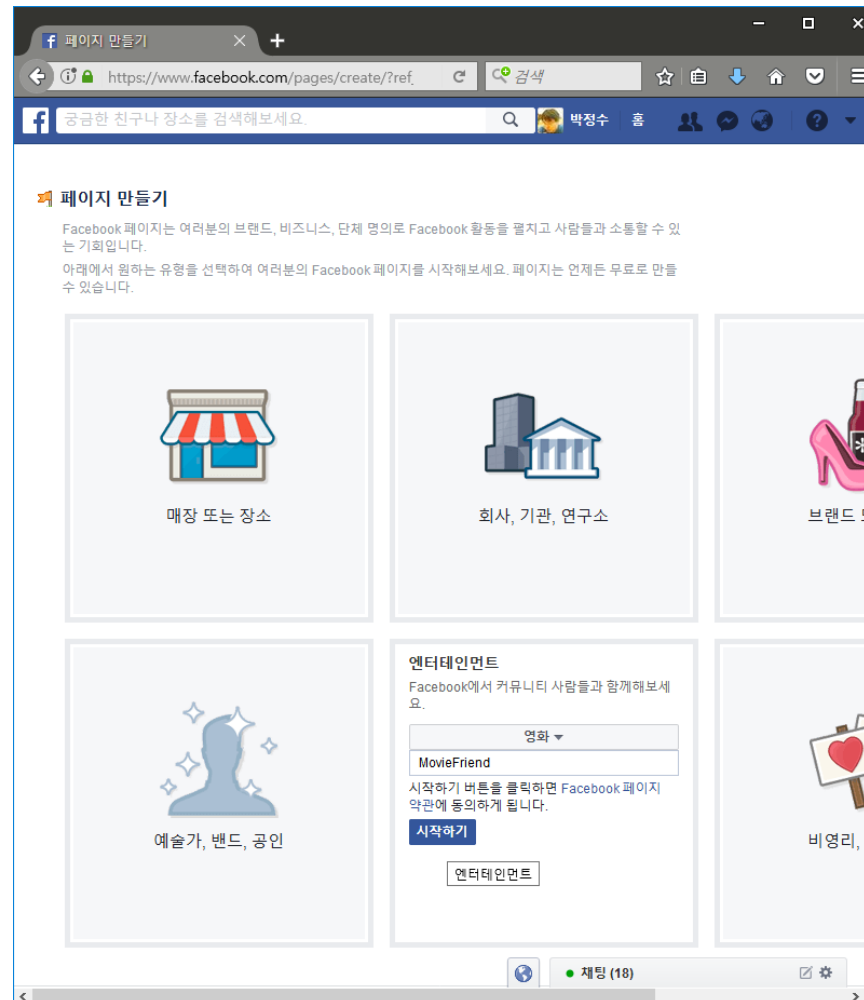
Mr.MIND Inc. © 2017 All rights reserved.

Page 6



Create a Facebook Messenger Bot account

생성할 페이지의 카테고리과 이름을 지정합니다.



Create a Facebook Messenger Bot account

새로운 앱을 생성합니다.

Facebook for Developers

https://developers.facebook.com

facebook for developers

새 앱 ID 만들기

Facebook을 앱이나 웹사이트로 통합합니다

표시 이름

MovieFriendBot

연락처 이메일

앱에 관한 중요한 커뮤니케이션에 사용됩니다

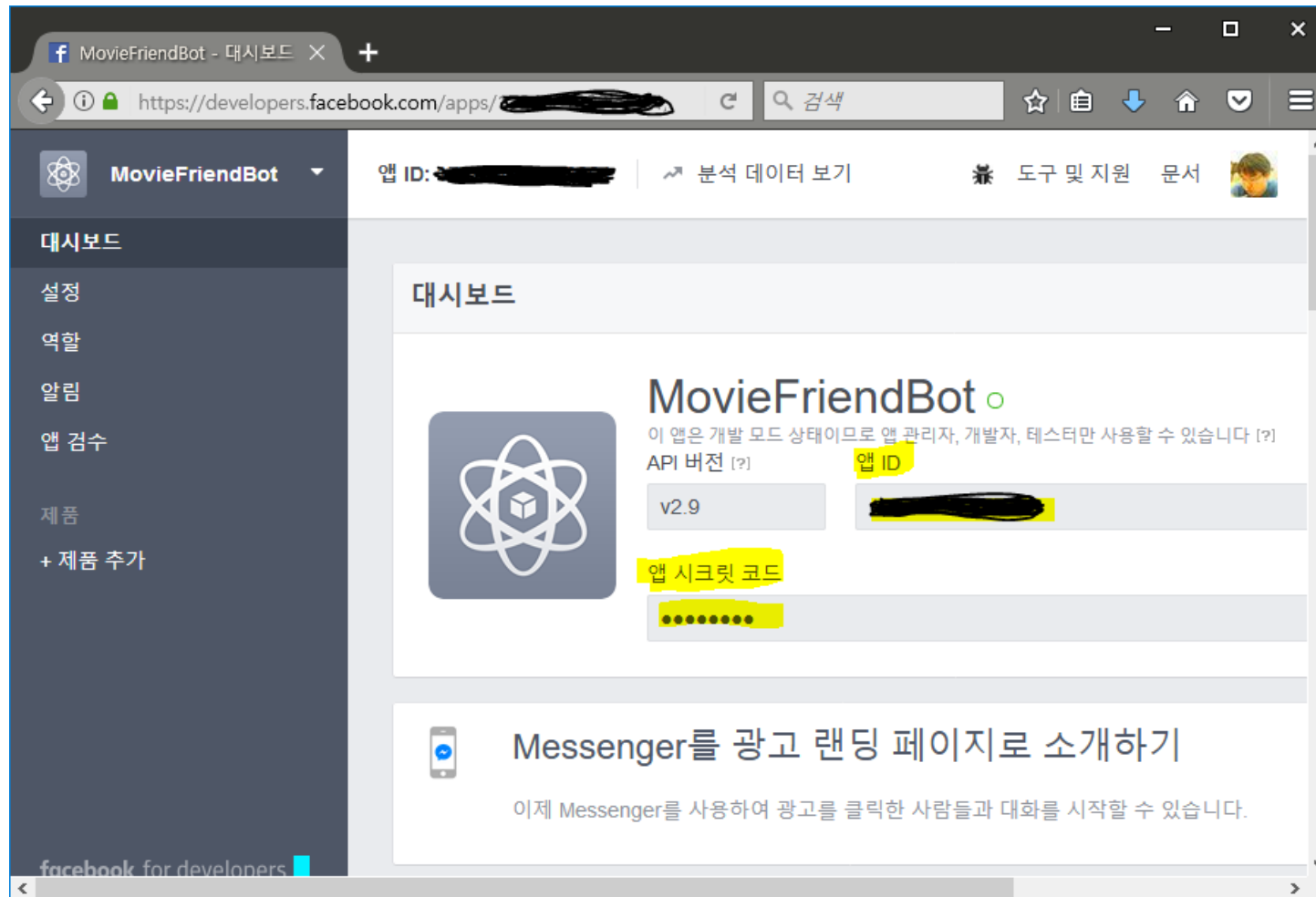
계속하면 Facebook 플랫폼 정책에 동의하는 것입니다

취소 앱 ID 만들기

지금 보기

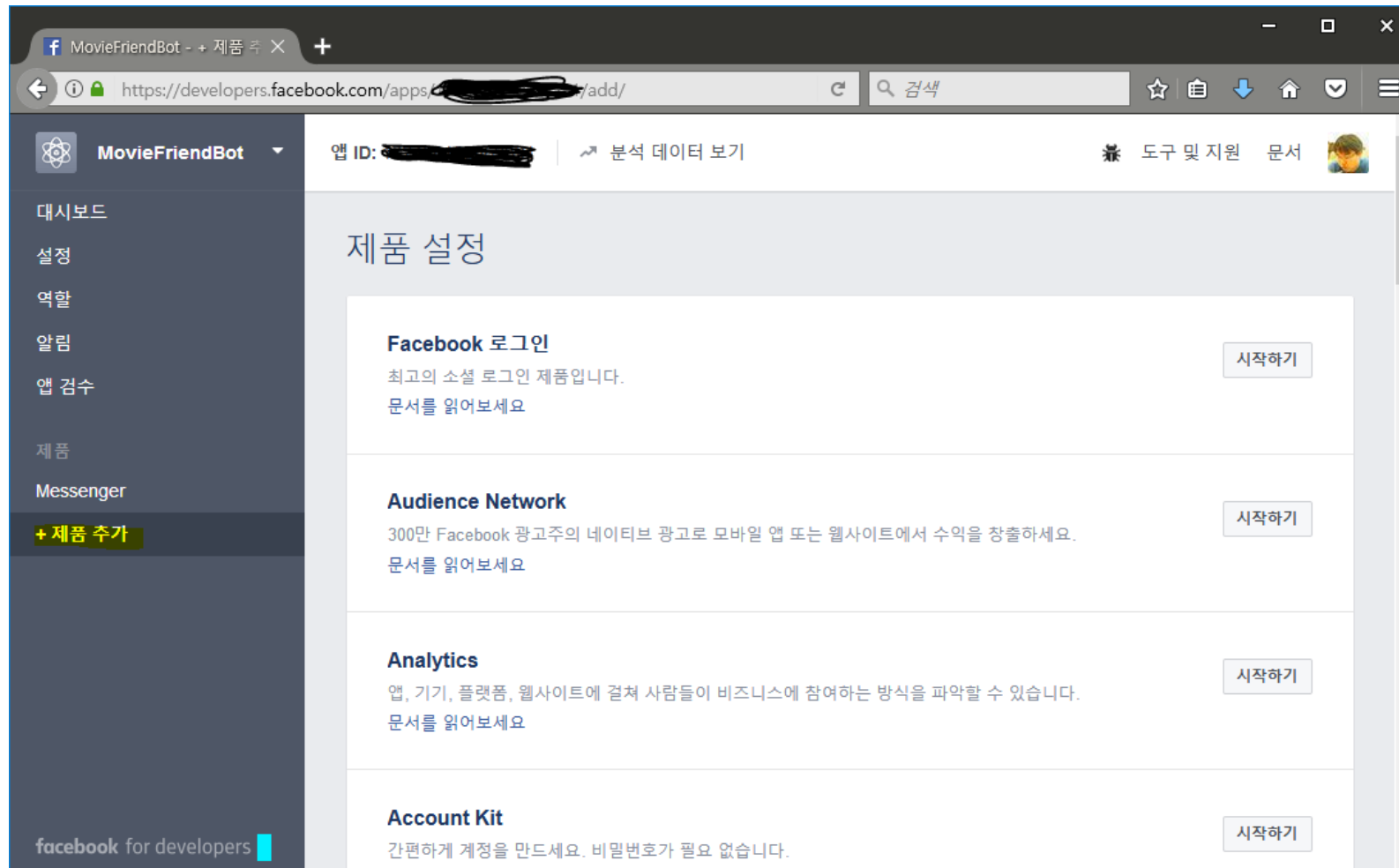
Create a Facebook Messenger Bot account

아래와 같이 대쉬보드가 나옵니다. 여기서는 앱 ID와 앱 시크릿 코드를 확인할 수 있습니다.



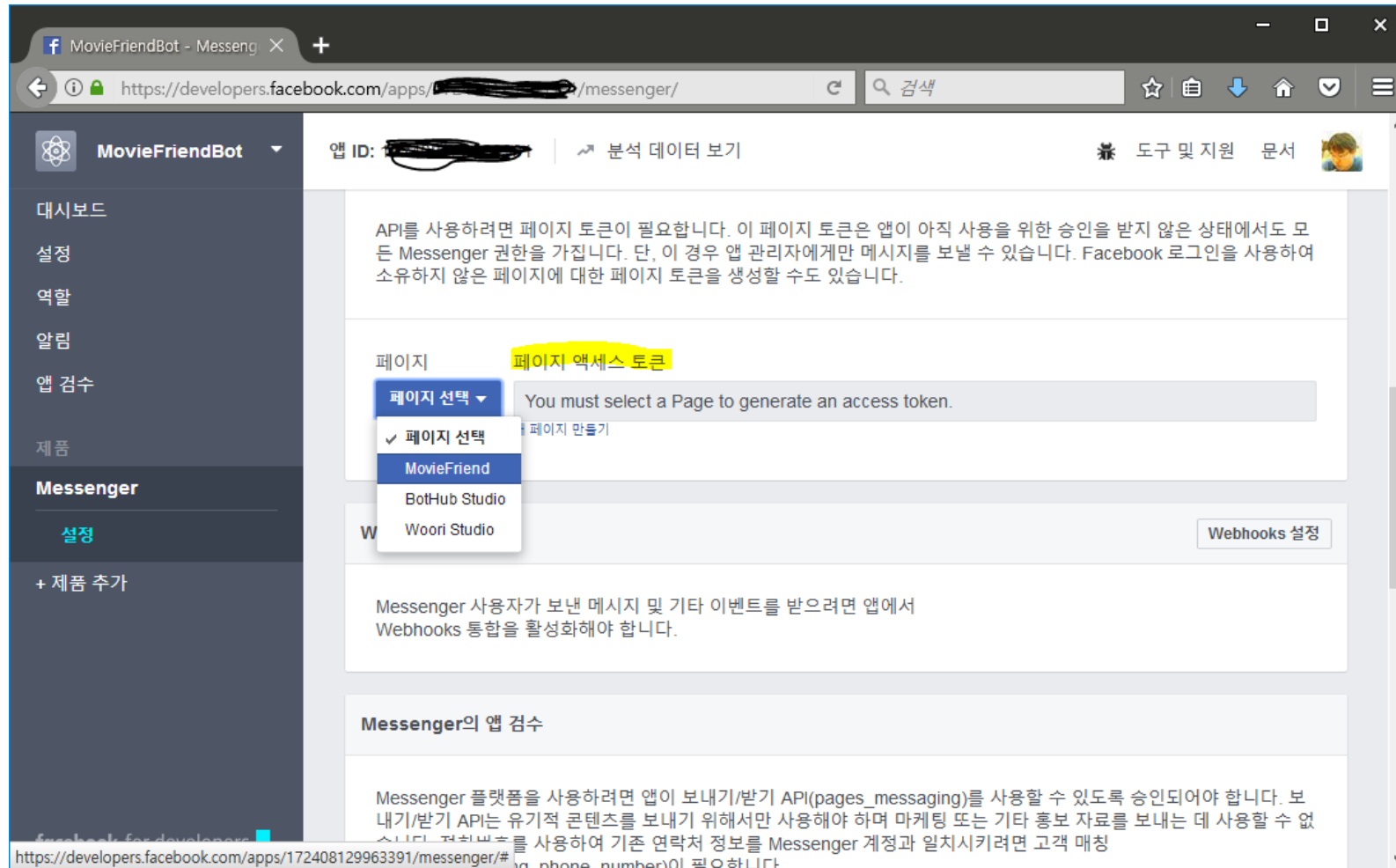
Create a Facebook Messenger Bot account

앱에 제품을 추가합니다. Messenger라는 제품 항목에서 시작하기 버튼을 클릭합니다.



Create a Facebook Messenger Bot account

챗봇을 운영할 페이지와 연동합니다. 여기서는 페이지 액세스 토큰을 확인할 수 있습니다.



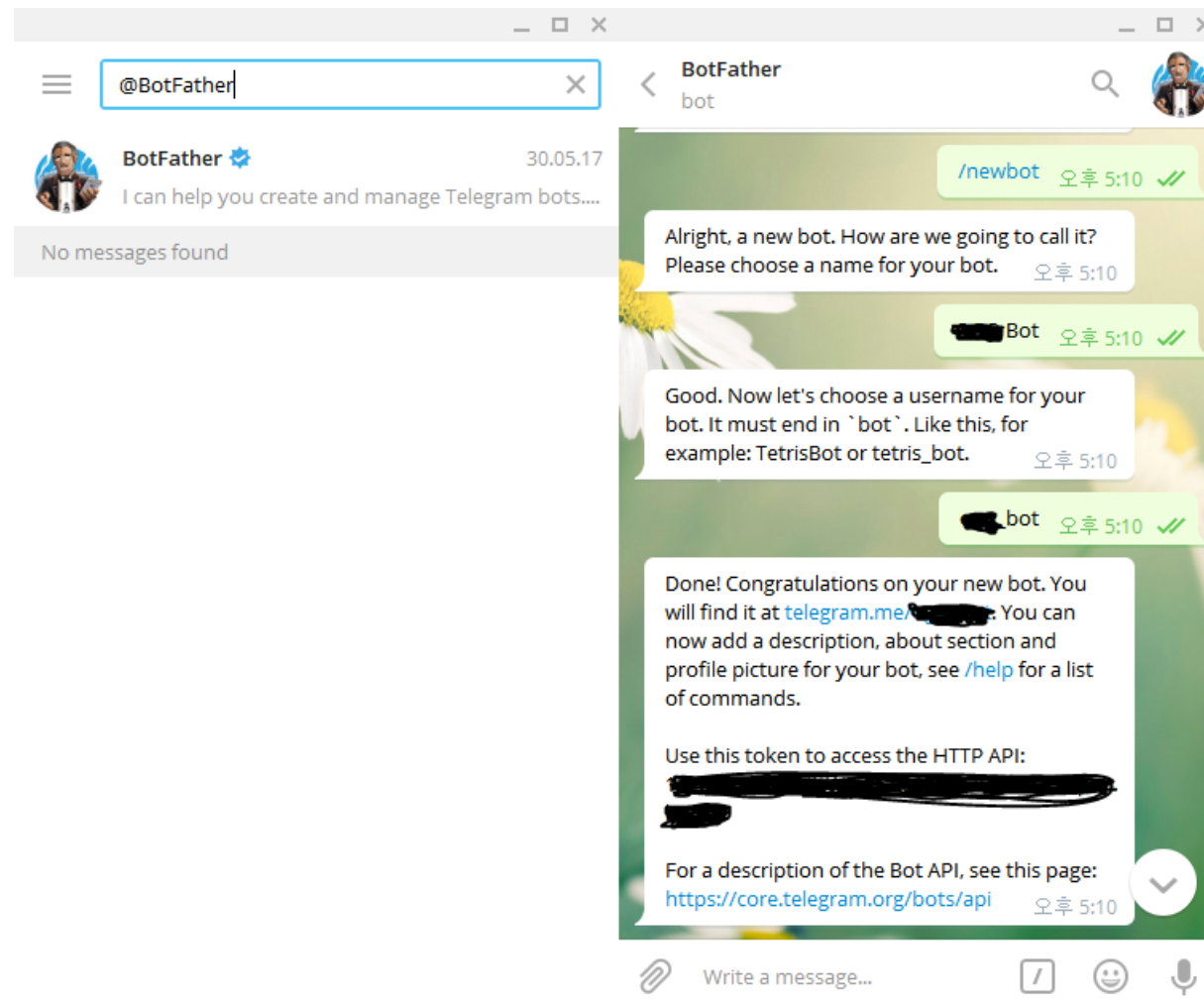
Create a Facebook Messenger Bot account

페이스북 챗봇을 연동하기 위해서는 아래 세 가지 정보를 기록해둡니다.

- App ID
- App Secret Code
- Page Access Token

Create a Telegram Bot account

텔레그램의 검색창에서 @BotFather 를 찾습니다. /newbot 명령을 사용하여 새로운 봇을 생성합니다.



Create a Facebook Messenger Bot account

텔레그램 챗봇을 연동하기 위해서는 아래 정보를 기록해둡니다.

- Access Token

Register an account for Bot Hosting

챗봇을 구동하려면 서버가 필요합니다.

여기서는 무료로 챗봇 호스팅을 제공하는 [BotHub.Studio](https://bothub.io) 를 사용합니다. 우선 홈페이지로 가서 회원 가입을 마칩니다.



Register an account for Bot Hosting

로컬 컴퓨터에서 셸 창을 열어 아래와 같이 CLI 도구를 설치합니다.

```
pip install bothub-cli
```

그리고 아래와 같이 계정을 연결합니다.

```
bothub configure
```

새로운 프로젝트를 생성합니다.

```
mkdir [your bot_name]  
cd [your bot_name]  
bothub init
```


Register an account for Bot Hosting

그러면 아래와 같이 기본 코드가 생성됩니다.

```
.
|-- bothub
|   |-- bot.py
|   `-- __init__.py
|-- bothub.yml
|-- requirements.txt
`-- tests
```

Register an account for Bot Hosting

메신저를 연결해봅시다.

```
bothub channel add telegram --api-key=<api-key>  
bothub channel add facebook --app-id=<app-id> --app-secret=<app-secret> --page-access-token=<page-access-token>
```

프로젝트를 새로 생성하면 기본으로 EchoBot 코드가 들어있습니다. Bot을 서버에 구동해봅시다.

```
bothub deploy
```

이제 메시징 플랫폼에 등록된 챗봇을 찾아서, 챗봇이 잘 동작하는지 확인해봅시다.
bothub-cli 및 bothub 패키지의 자세한 사용법은 Pypi([bothub-cli](#), [bothub](#))에서 확인할 수 있습니다.

기본 코드

- ▼ BasicBot
 - ▶ .bothub-meta
 - ▼ bothub
 - __init__.py
 - bot.py
 - ▶ dist
 - ▶ tests
 - bothub.yml
 - requirements.txt

```

1  # -*- coding: utf-8 -*-
2
3  from __future__ import (absolute_import, division, print_function, unicode_literals)
4
5  from bothub_client.bot import BaseBot
6
7
8  class Bot(BaseBot):
9      """Represent a Bot logic which interacts with a user.
10
11      BaseBot superclass have methods belows:
12
13      * Send message
14      * self.send_message(message, chat_id=None, channel=None)
15      * Data Storage
16      * self.set_project_data(data)
17      * self.get_project_data()
18      * self.set_user_data(data, user_id=None, channel=None)
19      * self.get_user_data(user_id=None, channel=None)
20
21      When you omit user_id and channel argument, it regarded as a user
22      who triggered a bot.
23      """
24
25      def handle_message(self, event, context):
26          """Handle a message received
27
28          event is a dict and contains trigger info.
29
30          {
31              "trigger": "webhook",
32              "channel": "<name>",
33              "sender": {
34                  "id": "<chat_id>",
35                  "name": "<nickname>"
36              },
37              "content": "<message content>",
38              "raw_data": <unmodified data itself webhook received>
39          }
40          """
41          self.send_message('Echo: {}'.format(event['content']))
42

```

LET'S MAKE CHATBOT !!



감사합니다