



FACULTAD DE CIENCIAS INTELIGENCIA ARTIFICIAL

Chatbot con ChatterBot

Equipo: Skynet Scribes

Número de practica: 01

Carlos Daniel Cortés Jiménez
420004846

Sarah Sophía Olivares García
318360638

Marco Silva Huerta
316205326

Juan Daniel Barrera Holan
417079372

Laura Itzel Tinoco Miguel
316020189

Profesora: Cecilia Reyes Peña

Ayudante teoría: Karem Ramos Calpulalpan

Ayudante laboratorio: Tania Michelle Rubí Rojas

Semestre 2024-2

Fecha de entrega:
14 de Febrero del 2024

1. Documentación

1.1. Organización del proyecto

Dentro del archivo .zip, el proyecto se compone de la siguiente forma

■ Carpeta Principal: PRACTICA01	Carpeta Principal: PRACTICA01
• Carpeta: IMA	Guarda las imágenes del proyecto
• Carpeta: src	Archivos ejecutables para el bot
○ Carpeta: venv	Contiene la configuración del entorno
○ Carpeta: templates	Contiene el front del chatbot
◇ HTML: index.html	HTML: index.html
○ PYTHON: bot.py	Lógica del programa
○ JSON: conver.json	Es información del entrenamiento
○ BD : db.sqlite3	Guarda el <i>aprendizaje</i> del bot
• PDF : practica01.pdf	PDF : practica01.pdf

1.2. Como ejecutar Chat Dog

Antes de iniciar a conversar con Chat Dog necesitamos esto en nuestra computadora:

1. Iniciamos nuestra terminal

2. Revisamos la versión de python para evitar conflictos con el entorno.

```
1 Skynet : $ python --version
2 Skynet : $ Python 3.7.9
```

3. Navegar hasta la carpeta donde esta el archivo bot.py

```
1 Skynet : $ cd Practica01/src
2 Skynet : /src $ ls
3 bot.py conver.json templates nenv
```

4. Vamos a instalar flask para que nuestro bot se vea correctamente en la web

```
1 Skynet : /src $ pip install flask
```

■ (Se descargarán los archivos necesarios)

5. Lo que sigue es activar nuestro entorno, primero crearemos la carpeta

■ Para Windows

```
1 Skynet : /src $ python -m venv venv
```

■ Para Linux

```
1 Skynet : /src $ python -m venv venv
```

6. Ahora vamos a activar el entorno

- Para Windows

```
1 Skynet : /src $ venv\Scripts\activate
2 (venv) Skynet : /src $
```

- Para Linux

```
1 Skynet : /src $ source_venv/bin/activate
2 (venv) Skynet : /src $
```

- Sabremos que esta activo el entorno al ver del lado izquierdo: (venv)

7. Ahora instalaremos CharrterBot

- Para Windows

```
1 (venv) Skynet : /src $ python -m pip install chatterbot==1.0.4 pytz
2 (venv) Skynet : /src $
```

- Para Linux

```
1 (venv) Skynet : /src $ python -m pip install chatterbot==1.0.4 pytz
2 (venv) Skynet : /src $
```

8. El siguiente paso es ejecutar el bot con el comando: `python bot.py`

```
1 (venv) Skynet : /src $ ls
2 bot.py  conver.json  templates  nenv
3 (venv) Skynet : /src $ python bot.py
```

- A continuación veremos varias lineas como estas

```
1 (venv) Skynet : /src $ python bot.py
2 [nltk_data] Downloading package averaged_perceptron_tagger to
3 [nltk_data] C:\Users\youruser\AppData\Roaming\nltk_data...
4 List Trainer: [#####] 100%
5 * Serving Flask app 'bot'
6 * Debug mode: on
```

- La siguiente linea es importante porque es el puerto donde verás a Chat-Dog

```
1 * Running on http://127.0.0.1:5000
```

- Tal vez en tu computadora sea otro puerto, pero así debe vers. Copia y pega en tu navegador de preferencia (`http://127.0.0.1:5000`)

```
1 * Running on http://127.0.0.1:5000
2 [nltk_data] Package stopwords is already up-to-date!
3 List Trainer: [#####] 100%
4 * Debugger is active!
```

9. Esto te deberá permitir ver la interfaz de Chat-Dog

10. Y listo podrás empezar a conversar con él

11. Para terminal el proceso basta con que en la terminal ejecutes: `Ctrl+C`

```
1 * Running on http://127.0.0.1:5000
2 [nltk_data] Package stopwords is already up-to-date!
3 List Trainer: [#####] 100%
4 * Debugger is active!
5
6 Ctrl+C
7 (venv) Skynet : /src $
```

1.3. Lógica del Código

Necesitamos hacer las siguientes importaciones en el código para tener el funcionamiento de de python con las aplicaciones web (para eso es flask), se necesita por supuesto la importación de chatterbot para construir el bot y trainers es utilizado para hacer el entrenamiento y hable de manera más fluida con el usuario. Finalmente json para trabajar con estos archivos pues son los que contienen toda la información de entrenamiento.

```
1 from flask import Flask, render_template, request, jsonify
2 from chatterbot import ChatBot
3 from chatterbot.trainers import ListTrainer
4 import json
```

Listing 1: Fragmento de código Python

Entonces trabajaremos con un chatbot, debemos crearlo y eso se hace así:

```
1 chatbot = ChatBot("ChatDog")
```

Listing 2: Fragmento de código Python

ChatDog es el nombre con el cual distinguimos a nuestro bot y es la referencia para hacer acciones específicas.

2. Investigación

3. Propósito

El propósito principal del chatbot es brindar un servicio educativo e informativo sobre animales domésticos y sus cuidados veterinarios. Busca ser una herramienta que ayude a los usuarios a comprender mejor cómo cuidar adecuadamente a sus mascotas.

Funcionalidades básicas:

- **Mensaje de saludo:** El chatbot debe ser capaz de saludar a los usuarios de manera amigable, utilizando una variedad de saludos para crear una experiencia acogedora desde el inicio de la interacción.
- **Descripción de servicios:** Proporciona una descripción clara y concisa de los servicios que ofrece
- **Datos interesantes:** Proporciona curiosidades y datos interesantes sobre animales domésticos, como sus habilidades naturales, comportamientos únicos, etc.
- **Despedida cordial:** Al finalizar la interacción manda un mensaje de despedida cordial, agradeciéndoles por utilizar el servicio.

Referencias

- [Cox21] Gunther Cox. *ChatterBot*. <https://chatterbot.readthedocs.io/en/stable/>. 2021.
- [Bre22] Martin Breuss. *ChatterBot: Build a Chatbot With Python*. <https://realpython.com/build-a-chatbot-python-chatterbot/>. 2022.
- [Dyo22] Abdelhadi Dyouri. *How To Make a Web Application Using Flask in Python 3*. <https://www.digitalocean.com/community/tutorials/how-to-make-a-web-application-using-flask-in-python-3>. 2022.