

Desarrollo de software ChatMedic para la interacción y accesibilidad de medicamentos

Eddy Mercedes Ariza Osma

Sebastian Rodriguez Charcas

Resumen- Desarrollo de una aplicación Python, bajo la versión 3.8.10, que permite realizar una interacción entre el usuario y un chatbot predefinido por algunas opciones relevantes, para la consulta y solicitud de medicamentos, según parametrización de una base de datos no relacional con MongoDB, adicional se cuenta con una aplicación Python que contiene los diferentes servicios REST para consulta, creación y actualización de medicamentos o flujo del proceso.

Abstract – Development of a Python a application under versión 3.8.10, which allows an interaction between the user and a chatbot predefined by some relevant options, for the consultation and request of medicines, according to the parametrization of a non relational database with MongoDB, additionally, there is a Python application that contains the different REST services for consultation, creation, and updating of medications or process flow

INTRODUCCIÓN

Un chatbot es un software de inteligencia artificial diseñado para permitir la interacción con los usuarios para satisfacer cualquier solicitud o demanda de información o servicios. Este tipo de robot debe poder mantener o simular conversaciones para gestionar solicitudes. La interfaz de usuario conversacional debe estar enfocada y perfectamente ajustada para brindar una experiencia multicanal más personalizada, con menos barreras de entrada, las 24 horas del día, los 7 días de la semana y el anonimato total.

ChatMedic proporcionará información sencilla sobre medicamentos y disponibilidad o tratamientos en el menor tiempo

posible sin tener que cubrir demasiada información porque no requiere de expertos para proporcionar información correcta a los usuarios.

MANUAL DE USUARIO

1. Ingresar a la aplicación

Para ingresar a la aplicación chatmedic se debe ejecutar el archivo

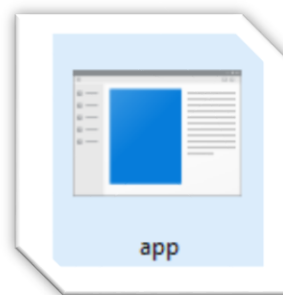


Ilustración 1 Ejecutable app.exe ChatMedic

2. Interacción con ChatMedic menú principal

Inicialmente ChatMedic solicitará el nombre del usuario, una vez ingresado el nombre mostrará un menú de 3 opciones en las cuales se puede acceder a la opción de consulta de medicamentos, a la solicitud de medicamentos y una finalización de chat en caso de que el usuario no quiera continuar.

```
Buen día, por favor ingrese su nombre
Daniel

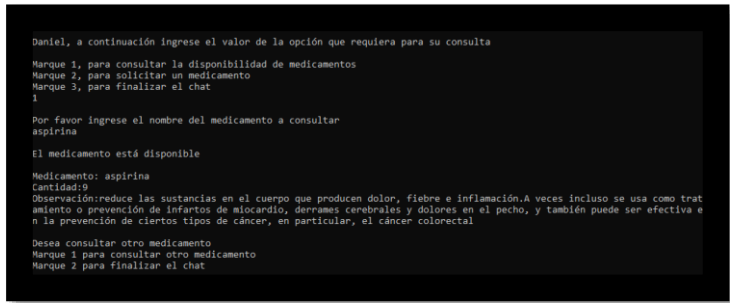
Daniel, a continuación ingrese el valor de la opción que requiera para su consulta

Marque 1, para consultar la disponibilidad de medicamentos
Marque 2, para solicitar un medicamento
Marque 3, para finalizar el chat
```

Ilustración 2 Menú Principal

3. Consulta de medicamentos

El chatbot solicitará al usuario que ingrese el nombre del medicamento que requiere consultar para obtener información relevante del mismo y validar su disponibilidad, en caso de que no exista el medicamento, el chat solicitará si se requiere consultar nuevamente otro medicamento o se desea finalizar el chat, en caso que exista se mostrará la información específica del medicamento consultado y a su vez se le pedirá al usuario que indique si desea hacer



otra consulta o finalización del chat

Ilustración 3 Consulta de medicamento

4. Solicitud de medicamento

Para la solicitud de medicamento, chatmedic le mostrará en un listado de opciones al usuario, aquellos medicamentos disponibles en stock.

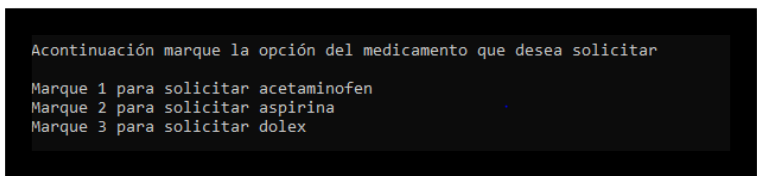


Ilustración 4 Solicitud de medicamento disponibles

5. Captura de datos para finalizar la solicitud

Para finalizar con la solicitud de medicamento chatmedic le indicará al usuario que ingrese unos datos personales para así hacer la recepción de los mismos, posteriormente, almacenarlos en la base de datos y actualizar el stock de medicamentos según el medicamento solicitado por el usuario. Adicional chatmedic una vez capture los datos

enviará un correo electrónico con la confirmación de la solicitud realizada por parte del usuario.

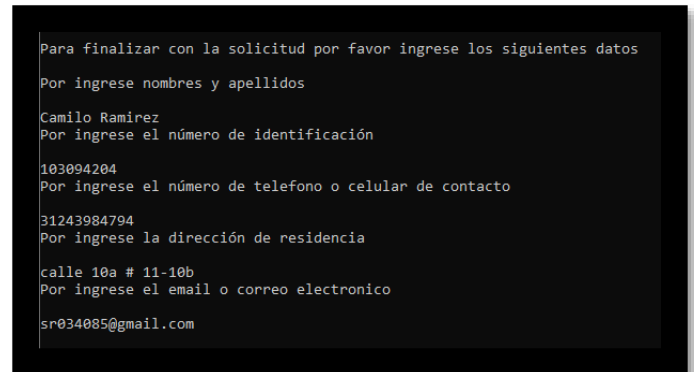


Ilustración 5 Captura de datos para finalizar la solicitud

6. Finalización Proceso Solicitud

El sistema una vez envíe el correo electrónico, notificará al usuario y finalizará automáticamente el chat con el usuario

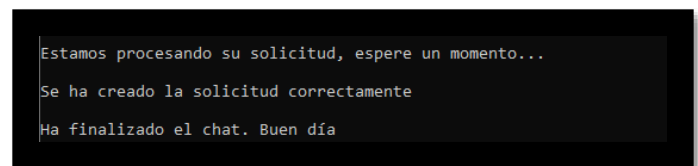


Ilustración 6 Finalización Solicitud



Ilustración 7 Email Confirmación Solicitud

MANUAL DEL PROGRAMADOR

- Link Repositorio Chatmedic:
✓ <https://github.com/sebastianR05/ChatMedic>
- Link Repositorio Api Chatmedic:
✓ <https://github.com/sebastianR05/APIChatMedic>

mensajes se recomienda realizarlo en este archivo para así centralizar el tema de mensajes en un solo lugar.

1. Contenido archivo *app.py*

El archivo cuenta con la importación de una clase la cual maneja los diferentes métodos para la interacción del proceso del chatbot de medicamentos tanto para la consulta como la solicitud de los mismos.

```
import clases.proceso as proceso

opcion_inicio = proceso.Proceso.inicio()
if(opcion_inicio == "1"):
    proceso.Proceso.consultarMedicamentos()
elif(opcion_inicio == "2"):
    proceso.Proceso.solicitarMedicamentos()
```

2. Contenido archivo *funciones.py*

Este archivo Python, permite la creación de diferentes métodos o funciones lógicas para validaciones, casteos que se requieran dentro del proceso para una reutilización optima del código.

```
def validaNumeros(valor):
    if(valor.isdigit()):
        return True
    else:
        return False

def convertMayuscula(texto):
    return texto.upper()
```

3. Contenido archivo *mensajes.py*

Este archivo Python, permite interactuar con un método que realiza la impresión de mensajes por consola, sin embargo, en caso que se requiera agregar una funcionalidad diferente para la visualización de

```
def mostrarMensaje(msj):
    print(msj)
```

4. Contenido archivo *callApi.py*

Este archivo Python, busca centralizar el consumo de los diferentes contratos o métodos api que se requieren para la interacción del proceso. Inicialmente se encontrará la importación de las librerías mínimas requeridas para realizar una petición a una API.

```
import requests
import json
```

5. Contenido archivo *proceso.py*

Se definió como una clase, la cual contiene los distintos métodos accesibles para el proceso interactivo del chatbot. En su inicio se encuentra los respectivos imports de los utils de mensajes, funciones y llamados a las apis

```
import utils.mensajes as mensajes
import utils.funciones as funciones
import utils.callApi as callapi
from datetime import datetime
```

```
class Proceso:
    def inicio():...
    def consultarMedicamentos():...
    def validarNuevamenteConsulta():...
    def mostrarInformacionMedicamento(response):...
    def solicitarMedicamentos():...
    def mostrarMedicamentosDisponibles(response):...
    def capturarDatosSolicitudMedicamento(opciones_medimento, opcion_escogida):...
```