

# DOCTOR FINDER

## 1. ABSTRACT:

This project involves the development of a Streamlit application designed to scrape data from the “Practo” website using BeautifulSoup. The application allows users to input two key fields: location and specialist. Based on these inputs, it retrieves and displays relevant information, including the number of available doctors, their names, clinic names, years of experience, and patient recommendations. The streamlined interface of the Streamlit application ensures that users can easily access and view the scraped data, providing a valuable tool for those seeking medical professionals in specific areas and specialties.

## 2. INTRODUCTION:

In today’s digital age, accessing reliable healthcare information is crucial for making informed decisions. With the vast amount of data available online, finding the right healthcare provider can be overwhelming. This project addresses this challenge by developing a Streamlit application that scrapes data from the Practo website using BeautifulSoup.

The application is designed with user-friendliness in mind, allowing individuals to input their location and the type of specialist they are seeking. Upon receiving these inputs, the application efficiently retrieves and displays comprehensive details about available doctors. This includes their names, clinic names, years of experience, and patient recommendations.

By leveraging web scraping techniques, the application ensures that users have access to up-to-date and relevant information. This tool not only simplifies the search process but also empowers users with valuable insights, enabling them to make well-informed healthcare decisions. Whether you are looking for a specialist in a new city or seeking the best-reviewed doctor in your area, this application serves as a reliable resource to connect patients with healthcare providers.

Ultimately, this project aims to enhance the accessibility and transparency of healthcare information, making it easier for individuals to find the right medical professionals to meet their needs.

## 3. OBJECTIVES:

- **To develop a user-friendly Streamlit application** that allows users to search for doctors based on location and specialty.
- **To scrape data from the Practo website using BeautifulSoup**, ensuring the retrieval of accurate and up-to-date information.
- **To display comprehensive details** about doctors, including their names, clinic names, years of experience, and patient recommendations.
- **To enhance the accessibility and transparency** of healthcare information, making it easier for individuals to find the right medical professionals.

## 4. TECHNOLOGIES USED:

- **Streamlit:** A framework for creating interactive web applications in Python.
- **Beautiful Soup:** A Python library for parsing HTML and XML documents and extracting data.
- **Requests:** A Python library for making HTTP requests to fetch web pages.
- **Pillow:** A Python library for handling images that are used in the streamlit application.
- **Practo:** The target website from which data is scraped.

## 5. METHODOLOGY:

### 5.1. Data Collection:

- **Web Scraping:** Using Beautiful soup and Requests to scrape data from the HTML page.
- **User input:** Collecting location and specialist from the user through the streamlit application.

### 5.2. Data Preprocessing:

- **Parsing HTML:** Using Beautiful soup to parse the HTML content of the practo website.
- **Extracting information:** Identifying and extracting the relevant information such as doctor names, clinic names, years of experience, and patient recommendations.
- **Data Cleaning:** Cleaning and organizing the extracted data for display.

### 5.3. Data Display:

- **Streamlit Interface:** Designing an user-friendly interface to gather inputs from the user and to display the scraped data.
- **Interactive Elements:** Adding interactive elements such as buttons and input boxes to enhance the user experience.

## 6. IMPLEMENTATION:

### 6.1. Setting up the environment:

- **Installing the necessary libraries:** Streamlit, Beautiful Soup, Requests, Pillow and Time.
- Setting up a virtual environment for the project.

### 6.2. Building a streamlit application:

- **Creating the user interface:** Designing the layouts and interactive elements of the streamlit application.
- **Handing the user input:** Writing functions to capture and process the input data.
- **Scraping data:** Implementing web scraping logic using Beautiful soup and Requests.
- **Displaying the data:** Formatting and displaying the scraped data in the streamlit application.

## 7. CODES:

### Main.py:

# Importing the necessary libraries:

```
import streamlit as st
from PIL import Image
import requests
from bs4 import BeautifulSoup
import time
```

# Creating the streamlit application:

```
st.set_page_config(page_title="DoctorFinder",layout="wide",
initial_sidebar_state="expanded")
st.markdown("# Welcome to Doctor Finder!  ")
st.subheader("We help you to find doctors around the world.")
```

```
st.markdown(
    """
    <style>
    .main{
        background-color: white;
    }
    </style>
    """,
    unsafe_allow_html=True
)
```

with st.sidebar:

```
img = Image.open('images/Designer.png')
st.image(img, width=450)
st.markdown("Simply enter your location and discover a variety of doctors near you, "
            "each with their own specialties. Whether you need a general practitioner, a specialist, or a surgeon, "
            "our platform connects you with the right healthcare professionals in your area. Start your "
            "search today and take a step towards better health!")
```

```
location = st.text_input("Enter any city", help="Use only city names")
speciality = st.selectbox('Speciality:', ['Dentist', 'Gynecologist/obstetrician', 'General Physician',
                                           'Dermatologist', 'Ear-nose-throat (ent) Specialist',
                                           'Homeopath', 'Ayurveda'])
```

```

# Function to scrape the data from the practo website:

def scrape_doctors(location, speciality):
    page = 1
    doc_list = []
    values = []
    recommendation_list = []
    hospital_list = []
    area_list = []
    while True:
        response = requests.get(f"https://www.practo.com/search/doctors?results_type=doctor&q=%5B%7B%22word%22%3A%22{speciality}%22%2C%22autocompleted%22%3Atrue%2C%22category%22%3A%22subspeciality%22%7D%5D&city={location}&page={page}")
        time.sleep(5)
        practo_data = response.text
        soup = BeautifulSoup(practo_data, 'html.parser')
        doctors = soup.find_all(name='h2', class_='doctor-name')
        reccom = soup.find_all('span', {'data-qa-id': 'doctor_recommendation'})
        hospitals = soup.find_all(name='span', class_='u-c-pointer u-t-hover-underline')
        areas = soup.find_all(name='div', class_='u-bold u-d-inlineblock u-valign--middle')
        if not doctors:
            break
        for doctor in doctors:
            doc_list.append(doctor.getText())
            values.extend([div.text.strip() for div in soup.select('div.info-section > div.u-grey_3-text > div.uv2-spacer--xs-top > div')])
            recommendation_list.extend(span.text for span in reccom)
        for area in areas:
            a = area.find('a')
            area_list.append(a.getText())
        for hospital in hospitals:
            hospital_list.append(hospital.getText())

        time.sleep(2)
        page += 1
    return doc_list, values, recommendation_list, hospital_list, area_list


if st.button("SCRAPE"):
    with st.spinner("Scraping data .... "):
        doc_list, exp_list, reccom_list, hos_lists, area_list = scrape_doctors(location, speciality)
        st.success(f"Found {len(doc_list)} doctors:")
        cleaned_hos = [clinic for clinic in hos_lists if clinic not in ['\xa0+ 1 more', '\xa0+ 2 more']]

```

# Displaying the data in the streamlit application:

```
if doc_list:
    st.write("### Doctors List")
    for i in range(len(doc_list)):
        with st.expander(f"{i+1} - {doc_list[i]}"):
            st.text(f'Clinic: {cleaned_hos[i]}\n'
                    f'Location: {area_list[i]}\n'
                    f'{exp_list[i]}\n'
                    f'Patient recommendation: {reccom_list[i]}')
```

## 8. OUTPUTS:



Simply enter your location and discover a variety of doctors near you, each with their own specialties. Whether you need a general practitioner, a specialist, or a surgeon, our platform connects you with the right healthcare professionals in your area. Start your search today and take a step towards better health!

Deploy

### Welcome to Doctor Finder! 🎉


We help you to find doctors around the world.

Enter any city

Speciality:  
Dentist

SCRAPE

After scraping,



Simply enter your location and discover a variety of doctors near you, each with their own specialties. Whether you need a general practitioner, a specialist, or a surgeon, our platform connects you with the right healthcare professionals in your area. Start your search today and take a step towards better health!

Deploy

Ayurveda

SCRAPE

Found 12 doctors:

### Doctors List

1 - Dr. M Vishnupriya  
Clinic: Aditti Ayurveda And Panchakarma Clinic  
Location: Selayur, Chennai  
16 years experience overall  
Patient recommendation: 100%

2 - Dr. Rajesh Verma  
Clinic: Bliss Medicare Centre  
Location: Vadapalani, Chennai  
29 years experience overall  
Patient recommendation: 95%

3 - Dr. Poonam Verma

## 9. CONCLUSION:

This project exemplifies the power of combining web scraping techniques with a user-friendly interface to enhance the accessibility of healthcare information. By developing a Streamlit application that scrapes data from the Practo website using BeautifulSoup, we have created a tool that simplifies the process of finding medical professionals based on specific criteria such as location and specialty.

The application not only retrieves and displays essential details about doctors, including their names, clinic names, years of experience, and patient recommendations, but also ensures that this information is presented in an organized and easily accessible manner. This empowers users to make well-informed decisions about their healthcare needs, ultimately bridging the gap between patients and healthcare providers.

Moreover, the project highlights the importance of accurate and up-to-date information in the healthcare sector. By leveraging web scraping, we can continuously provide users with the latest data, ensuring that they have access to the most relevant information. This is particularly valuable in a rapidly changing field like healthcare, where new practitioners and clinics are constantly emerging.

Looking ahead, there are numerous opportunities for enhancing this application. Adding more filters, such as ratings, fees, and availability, could further refine the search process. Automating updates to the scraped data would ensure that users always have access to the most current information. Additionally, improving the design and usability of the Streamlit interface could make the application even more intuitive and engaging.

In conclusion, this project not only demonstrates the technical feasibility of web scraping and data presentation but also underscores the potential impact of such tools on improving healthcare accessibility. By making it easier for individuals to find the right medical professionals, we can contribute to better health outcomes and a more informed patient community. This Streamlit application serves as a testament to the innovative ways in which technology can be harnessed to address real-world challenges and improve lives.