*Biểu mẫu BTL01*

TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP

**KHOA ĐIỆN TỬ**

**Bộ môn: Công nghệ Thông tin**.

**BÀI TẬP KẾT THÚC MÔN HỌC**

MÔN HỌC

**KHOA HỌC DỮ LIỆU**

Sinh viên: CAO ĐỨC THÀNH. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

Lớp: K57-KMT. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

Giáo viên GIẢNG DẠY: NGUYỄN VĂN HUY. . . . . . . . . . . . . . . . . . . .

Link GitHub: <https://github.com/thanh123ab/KhoaHocDuLieu> . . . . . . . .

**Thái Nguyên – 2025**

*Biểu mẫu BTL02*

|  |  |
| --- | --- |
| **TRƯỜNG ĐHKTCN** | **CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM** |
| **KHOA ĐIỆN TỬ** | ***Độc lập - Tự do - Hạnh phúc*** |

**BÀI TẬP KẾT THÚC MÔN HỌC**

**MÔN HỌC: KHOA HỌC DỮ LIỆU**

BỘ MÔN : CÔNG NGHỆ THÔNG TIN

*Sinh viên: Cao Đức Thành*

*Lớp*: K57KMT ……………………………………………. *Ngành:Kỹ Thuật Máy Tính*

*Giáo viên hướng dẫn:Nguyễn Văn Huy*

*Ngày giao đề :20/05/2025*  *Ngày hoàn thành :26/05/2025*

*Tên đề tài : Xây dựng web app phân tích và dự báo dân số theo vùng địa lý.*

*Yêu cầu :*

***Đầu vào:***

* *Dữ liệu dân số từ* [*World Population Data*](https://www.kaggle.com/datasets/iamsouravbanerjee/world-population-dataset)

***Đầu ra:***

* *Dự báo dân số và biểu đồ tăng trưởng dân số theo năm.*

***Các tính năng:***

* *Dự báo dân số bằng Time Series*
* *Visualization dân số theo quốc gia*

|  |
| --- |
| **GIÁO VIÊN HƯỚNG DẪN** |
| *(Ký và ghi rõ họ tên)* |

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

### Thái Nguyên, ngày….tháng…..năm 20....

## GIÁO VIÊN HƯỚNG DẪN

*(Ký ghi rõ họ tên)*

MỤC LỤC

[Chương I. Giới thiệu đầu bài 5](#_Toc199293470)

[1.1. Mô tả chương trình 5](#_Toc199293471)

[Chương II. Cơ sở lý thuyết 6](#_Toc199293472)

[**2.1. Pandas – Thư viện xử lý dữ liệu** 6](#_Toc199293473)

[**2.2. ARIMA – Mô hình dự báo chuỗi thời gian** 6](#_Toc199293474)

[**2.3. Flask – Xây dựng ứng dụng web** 7](#_Toc199293475)

[**2.4. Matplotlib – Trực quan hóa dữ liệu** 8](#_Toc199293476)

[Chương III. Thiết kế và xây dựng chương trình 9](#_Toc199293477)

[**3.1. Sơ đồ khối hệ thống** 9](#_Toc199293478)

[**3.2. Sơ đồ khố các thuật toán chính** 10](#_Toc199293479)

[Chương IV: Thực nghiệm và Kết luận 14](#_Toc199293480)

[**4.1. Thực nghiệm** 14](#_Toc199293481)

[**4.2. Kết luận** 16](#_Toc199293482)

[LỜI CẢM ƠN 18](#_Toc199293483)

# **Chương I. Giới thiệu đầu bài**

* 1. **Mô tả chương trình**

Đề tài này là một ứng dụng web phục vụ cho việc phân tích và dự báo dân số theo vùng địa lý, với mục tiêu giúp người dùng có cái nhìn trực quan và chính xác về xu hướng phát triển dân số của các quốc gia hoặc khu vực trong các năm tiếp theo. Chương trình sử dụng dữ liệu dân số từ World Population Data để thực hiện phân tích, dự báo và trực quan hóa sự tăng trưởng dân số.

**Tính năng của chương trình :**

**Dự báo dân số bằng Time Series:** Sử dụng các kỹ thuật phân tích chuỗi thời gian (Time Series) để dự báo dân số trong tương lai, với mô hình ARIMA

**Trực quan hóa dân số theo quốc gia:** Cung cấp các biểu đồ trực quan cho phép người dùng theo dõi sự tăng trưởng dân số qua các năm, phân chia theo các quốc gia hoặc vùng lãnh thổ.

**Hiển thị kết quả dự báo dân số:** Sau khi dự báo, người dùng có thể thấy dân số dự báo cho các năm tiếp theo cùng với biểu đồ tăng trưởng.

**Thách thức trong việc thực hiện chương trình:**

**Ứng dụng ARIMA cho dự báo:** Việc chọn các tham số thích hợp cho mô hình ARIMA để có được kết quả chính xác nhất là một thử thách.

**Tối ưu hóa và tạo giao diện người dùng dễ hiểu:** Mặc dù có tính toán phức tạp phía sau, ứng dụng cần được thiết kế để người dùng có thể dễ dàng tương tác và sử dụng.  
**Kiến thức và kỹ năng vận dụng:**

**Python và thư viện Pandas:** Sử dụng để xử lý và phân tích dữ liệu dân số.

**ARIMA và Time Series:** Áp dụng các mô hình dự báo chuỗi thời gian để đưa ra dự đoán về dân số trong tương lai.

**Matplotlib:** Dùng để trực quan hóa dữ liệu và kết quả dự báo dưới dạng các biểu đồ.

**Phát triển Web:** Xây dựng ứng dụng web đơn giản để người dùng có thể tương tác với hệ thống.

# **Chương II. Cơ sở lý thuyết**

## **2.1. Pandas – Thư viện xử lý dữ liệu**

Pandas là một thư viện mạnh mẽ trong Python giúp xử lý và phân tích dữ liệu dạng bảng. Trong bài toán này, Pandas sẽ giúp chúng ta thực hiện các công việc sau:

**Đọc và xử lý dữ liệu**: Đọc dữ liệu từ các file CSV chứa thông tin dân số của các quốc gia.

**Làm sạch dữ liệu**: Xử lý các giá trị thiếu hoặc không hợp lệ trong dữ liệu, giúp đảm bảo chất lượng của dữ liệu đầu vào.

**Tạo bảng dữ liệu**: Tạo các bảng dữ liệu dạng DataFrame để chứa thông tin dân số của các quốc gia theo năm.

**Trích xuất thông tin**: Tìm kiếm, lọc và trích xuất thông tin dân số từ các bảng dữ liệu theo yêu cầu.

Dữ liệu về dân số được cung cấp trong file CSV từ nguồn World Population Data. Chúng ta sử dụng Pandas để đọc và xử lý dữ liệu này trước khi thực hiện việc dự báo.

## **2.2. ARIMA – Mô hình dự báo chuỗi thời gian**

ARIMA (AutoRegressive Integrated Moving Average) là một mô hình thống kê được sử dụng để phân tích và dự báo chuỗi thời gian. Trong bài toán dự báo dân số này, chúng ta sử dụng mô hình ARIMA để dự đoán dân số trong tương lai dựa trên các giá trị quá khứ.

Mô hình ARIMA có ba thành phần chính:

**AR (AutoRegressive)**: Mô hình hồi quy tự động, mô hình hóa sự phụ thuộc giữa giá trị hiện tại và các giá trị trong quá khứ.

**I (Integrated)**: Phần tích hợp giúp làm cho chuỗi thời gian trở thành dừng (stationary). Điều này rất quan trọng để mô hình có thể đưa ra dự báo chính xác.

**MA (Moving Average)**: Mô hình trung bình trượt, mô hình hóa các sai số của dự báo trong quá khứ.

## **2.3. Flask – Xây dựng ứng dụng web**

Flask là một micro-framework phổ biến trong Python để xây dựng các ứng dụng web. Nó nhẹ nhàng, dễ sử dụng và cực kỳ linh hoạt. Flask giúp chúng ta xây dựng giao diện web để người dùng có thể nhập liệu và nhận kết quả dự báo dân số từ mô hình ARIMA.

Trong chương trình này, Flask sẽ được sử dụng để:

**Xây dựng giao diện người dùng**: Tạo các trang web cho phép người dùng chọn quốc gia và số năm cần dự báo.

**Nhận dữ liệu từ người dùng**: Thu thập dữ liệu nhập từ người dùng, như quốc gia và số năm cần dự báo.

**Hiển thị kết quả**: Hiển thị biểu đồ dự báo dân số và các dữ liệu dự báo trong bảng.

Cấu trúc của ứng dụng web bao gồm các phần sau:

**Giao diện HTML**: Sử dụng Jinja để render các trang HTML và hiển thị các kết quả.

**Route**: Xử lý các yêu cầu GET và POST từ người dùng và gửi dữ liệu vào mô hình ARIMA để tính toán kết quả.

## **2.4. Matplotlib – Trực quan hóa dữ liệu**

Matplotlib là một thư viện phổ biến trong Python để tạo ra các biểu đồ và hình ảnh trực quan. Trong bài toán này, Matplotlib được sử dụng để vẽ biểu đồ tăng trưởng dân số và dự báo dân số cho các quốc gia.

Các chức năng chính của Matplotlib trong chương trình bao gồm:

**Vẽ biểu đồ**: Vẽ các biểu đồ đường để hiển thị dân số thực tế qua các năm và dự báo dân số trong tương lai.

**Tạo ảnh trực quan**: Chuyển đổi biểu đồ thành hình ảnh PNG và nhúng vào trong giao diện web dưới dạng hình ảnh Base64.

# **Chương III. Thiết kế và xây dựng chương trình**

## **3.1. Sơ đồ khối hệ thống**

Hệ thống của chúng ta bao gồm các module chính sau:

**Module Nhập liệu (Input Module)**:

Chức năng: Đọc và nạp dữ liệu dân số từ file CSV (dữ liệu dân số từ World Population Data).

Sử dụng: Thư viện **Pandas** để xử lý file CSV và chuyển đổi dữ liệu vào các cấu trúc dữ liệu như DataFrame.

**Module Xử lý dữ liệu (Data Processing Module)**:

Chức năng: Làm sạch dữ liệu, xử lý dữ liệu thiếu, và chuẩn bị dữ liệu cho mô hình dự báo.

Sử dụng: **Pandas** để xử lý các dữ liệu dân số từ các quốc gia.

**Module Dự báo (Forecasting Module)**:

Chức năng: Xây dựng mô hình ARIMA để dự báo dân số của quốc gia trong tương lai dựa trên dữ liệu quá khứ.

Sử dụng: **ARIMA** từ thư viện **statsmodels** để dự báo chuỗi thời gian.

**Module Trực quan hóa (Visualization Module)**:

Chức năng: Trực quan hóa dữ liệu dân số qua các năm và dự báo dân số trong tương lai dưới dạng biểu đồ.

Sử dụng: **Matplotlib** để vẽ biểu đồ tăng trưởng dân số và dự báo.

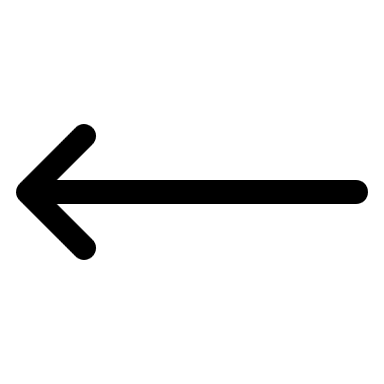
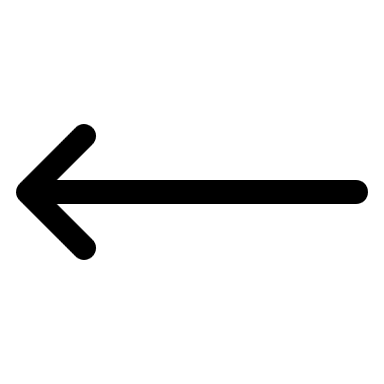
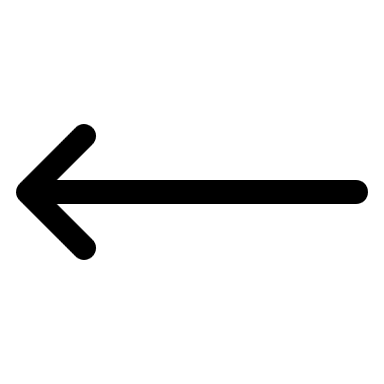
**Module Giao diện web (Web Interface Module)**:

Chức năng: Xây dựng giao diện người dùng để nhập dữ liệu (quốc gia và số năm cần dự báo), hiển thị biểu đồ và bảng dữ liệu.

Sử dụng: **Flask** để xây dựng ứng dụng web, render HTML và nhận các yêu cầu POST từ người dùng.

## **3.2. Sơ đồ khố các thuật toán chính**

Dưới đây là sơ đồ khối thể hiện các thuật toán chính trong chương trình và mối quan hệ :



Trực quan hóa, vẽ biểu đồ và hiển thị kết quả

Dự báo ARIMA

(dự báo dân số từ dữ liệu CSV)

Xử lý dữ liệu

(làm sạch và chuẩn bị dữ liệu đầu vào để dễ dàng thống kê, chuẩn hóa CSV)

Nhập, đọc dữ liệu từ CSV rồi chuyển thành Dataframe

**Mô tả các thuật toán chính**:

**Nhập liệu**:

Đọc file CSV chứa dữ liệu dân số.

Sử dụng Pandas để chuyển đổi dữ liệu thành DataFrame và loại bỏ các dữ liệu thiếu hoặc không hợp lệ.

**Xử lý dữ liệu**:

Làm sạch dữ liệu: Loại bỏ các giá trị thiếu, xử lý dữ liệu ngoại lai.

Trích xuất và chuẩn hóa dữ liệu từ các năm và quốc gia.

**Dự báo ARIMA**:

Áp dụng mô hình ARIMA cho dữ liệu dân số quá khứ của quốc gia đã chọn.

Sử dụng thư viện **statsmodels** để huấn luyện mô hình và thực hiện dự báo cho các năm tiếp theo.

**Trực quan hóa**:

Vẽ biểu đồ tăng trưởng dân số trong quá khứ và biểu đồ dự báo dân số trong tương lai.

Chuyển đổi biểu đồ thành hình ảnh Base64 để nhúng vào giao diện web.

**3.3. Cấu trúc Dữ liệu**

Chương trình sử dụng cấu trúc dữ liệu chính là DataFrame trong thư viện Pandas để lưu trữ và xử lý dữ liệu dân số. Cấu trúc dữ liệu được chia thành các phần sau:

**1. DataFrame chứa dữ liệu dân số**

**Country**: Đây là tên của các quốc gia hoặc vùng lãnh thổ có trong dữ liệu.

**Year**: Các năm có dữ liệu dân số, từ năm đầu tiên có sẵn trong dữ liệu đến năm cuối cùng (ví dụ từ năm 1970 đến 2020).

**Population**: Đây là các cột chứa dân số của quốc gia cho từng năm cụ thể.

**Cấu trúc dữ liệu** trong bảng sẽ bao gồm các cột như sau:

**Country**: Tên quốc gia.

**Year Population**: Dân số của quốc gia đó trong các năm tương ứng (ví dụ: "Year 2000 Population", "Year 2001 Population",...).

**Ví dụ dữ liệu**:

Quốc gia: **Vietnam**

Cột "Year 2000 Population" sẽ chứa giá trị dân số của Việt Nam năm 2000.

Cột "Year 2001 Population" sẽ chứa giá trị dân số của Việt Nam năm 2001.

Dữ liệu này sẽ được tiếp tục cho đến năm cuối cùng có sẵn trong dữ liệu.

**2. Dữ liệu đầu vào của mô hình ARIMA**

Mô hình **ARIMA** (AutoRegressive Integrated Moving Average) được sử dụng để dự báo dân số trong tương lai. Dữ liệu đầu vào cho mô hình ARIMA sẽ bao gồm:

**Years**: Một danh sách các năm có dữ liệu dân số trong quá khứ.

**Population**: Dân số của các quốc gia trong các năm đó.

**Ví dụ về dữ liệu đầu vào**:

**Years** = [2000, 2001, 2002, ..., 2020]

**Population** = [78,831,500, 79,954,300, 81,100,000, ..., 97,300,000]

Dữ liệu này sẽ được sử dụng để huấn luyện mô hình ARIMA nhằm tìm ra xu hướng và mẫu tăng trưởng dân số.

**3. Dữ liệu đầu ra từ ARIMA**

Sau khi huấn luyện mô hình ARIMA, chương trình sẽ sử dụng nó để dự báo dân số cho các năm tiếp theo. Kết quả đầu ra sẽ bao gồm:

**Forecasted Years**: Các năm được dự báo, tính từ năm cuối cùng trong dữ liệu đầu vào.

**Forecasted Population**: Dự báo dân số cho các năm trong tương lai.

**Ví dụ về kết quả đầu ra**:

**Forecasted Years** = [2021, 2022, ..., 2030]

**Forecasted Population** = [100,000,000, 102,000,000, ..., 120,000,000]

**3.4. Chương trình**

Chương trình được xây dựng từ các hàm xử lý chính sau:

**1. Hàm format\_population(value)**

**Chức năng**: Hàm này định dạng dân số theo đơn vị triệu (M) nếu dân số lớn hơn 1 triệu, hoặc giữ nguyên nếu dân số nhỏ hơn 1 triệu.

**Đầu vào**: Một giá trị dân số.

**Đầu ra**: Dân số đã được định dạng dưới dạng chuỗi, ví dụ: "12.5M" (triệu) hoặc "1,250,000" (nguyên bản).

**2. Hàm get\_country\_list()**

**Chức năng**: Trả về danh sách tất cả các quốc gia có trong dữ liệu dân số.

**Đầu vào**: Không có.

**Đầu ra**: Một danh sách các quốc gia.

Hàm này sử dụng dữ liệu từ file CSV để trích xuất và trả về danh sách các quốc gia đã được sắp xếp theo thứ tự bảng chữ cái.

**3. Hàm forecast\_population\_arima(country, years\_ahead=10)**

**Chức năng**: Đây là hàm chính thực hiện dự báo dân số cho một quốc gia nhất định và số năm dự báo.

**Đầu vào**: Tên quốc gia và số năm cần dự báo (mặc định là 10 năm).

**Đầu ra**: Biểu đồ dự báo dân số (dưới dạng Base64 để hiển thị trên web), bảng dữ liệu dự báo, hoặc thông báo lỗi nếu dữ liệu không đủ để thực hiện dự báo.

Hàm này sử dụng mô hình ARIMA để dự báo dân số cho các năm tiếp theo, đồng thời trả về biểu đồ dự báo để hiển thị trên giao diện web.

**4. Hàm index() (Flask App)**

**Chức năng**: Đây là hàm xử lý các yêu cầu từ người dùng trên giao diện web. Khi người dùng chọn quốc gia và số năm dự báo, hàm sẽ gọi các hàm xử lý dự báo dân số và hiển thị kết quả trên trang web.

**Đầu vào**: Dữ liệu từ form HTML (quốc gia và số năm).

**Đầu ra**: Giao diện web với biểu đồ và bảng dữ liệu dự báo dân số.

Hàm này sẽ lấy thông tin từ người dùng thông qua form HTML, gọi hàm dự báo dân số, và sau đó render lại kết quả trên trang web cho người dùng thấy.

# **Chương IV: Thực nghiệm và Kết luận**

## **4.1. Thực nghiệm**

Sau khi hoàn thiện các thành phần chính của ứng dụng dự báo dân số, em đã tiến hành chạy thử nghiệm thực tế để kiểm tra tính chính xác của mô hình và khả năng hiển thị giao diện người dùng.

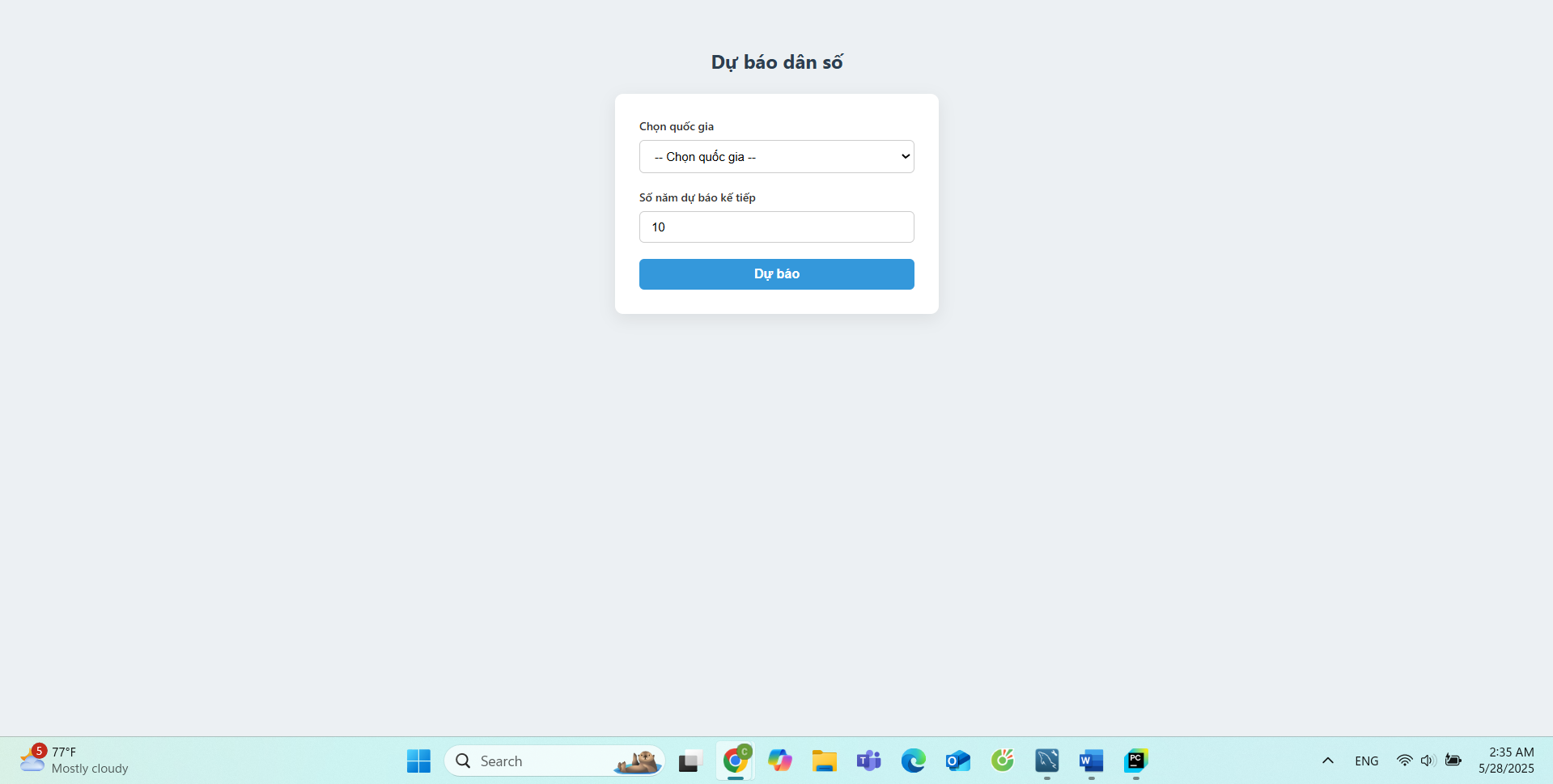
**Tính năng được kiểm tra:**

Chọn quốc gia từ danh sách sẵn có.

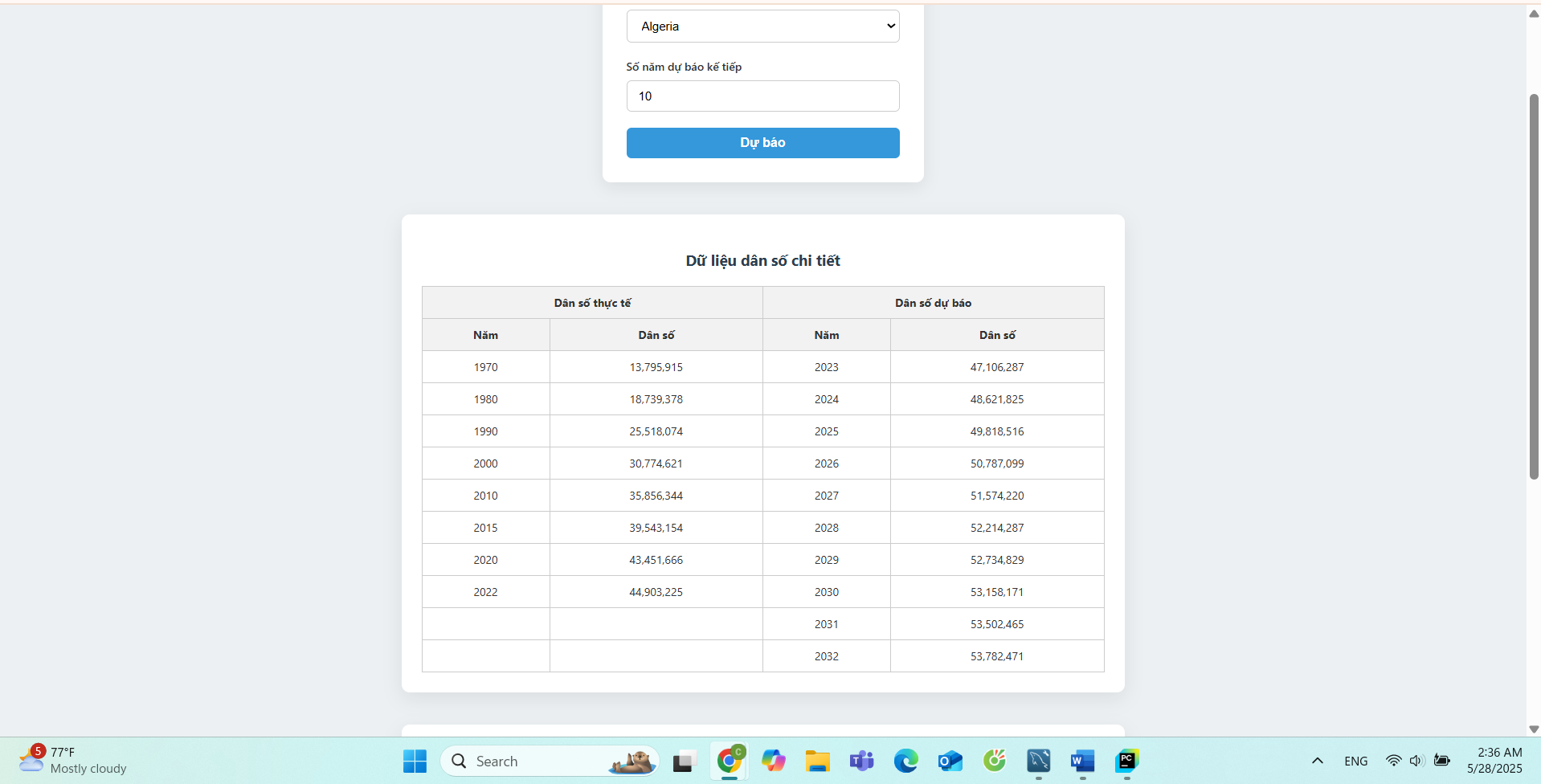
Nhập số năm cần dự báo tiếp theo.

Hiển thị bảng dân số thực tế và dân số dự báo.

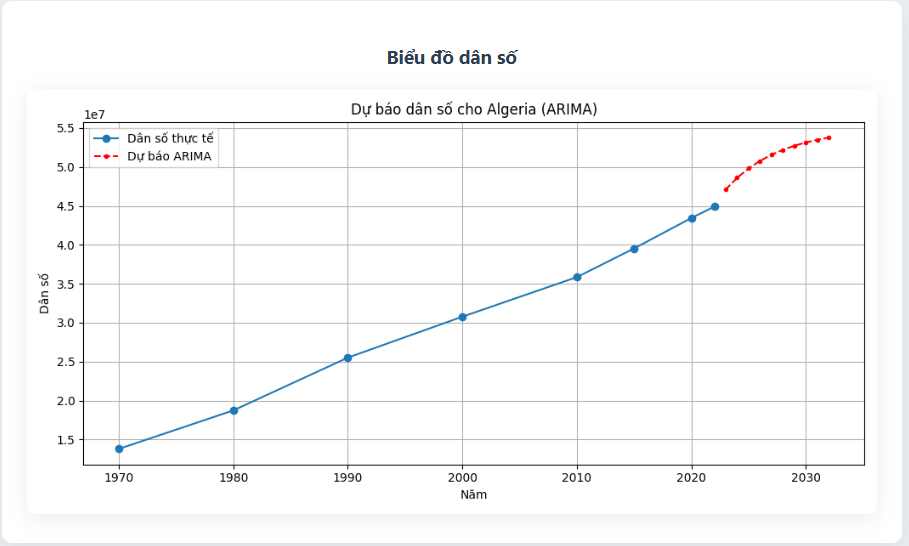
Hiển thị biểu đồ trực quan hóa quá trình dự báo bằng mô hình ARIMA  
Giao diện và kết quả thực nghiệm:



Hình 4.1.1. Giao diện Web



Hình 4.1.2. Bảng kết quả hiển thị dữ liệu dân số thực tế và dân số được dự báo cho các năm tiếp theo.



Hình 4.1.3. Biểu đồ thể hiện dân số qua các năm và phần dân số được dự báo bằng đường nét đứt màu đỏ.

**Đánh giá kết quả:**

Giao diện hoạt động ổn định, hiển thị chính xác dữ liệu đầu vào và đầu ra.

Mô hình ARIMA cho kết quả khá phù hợp với xu hướng dân số trong dữ liệu thực tế.

Biểu đồ trực quan giúp người dùng dễ dàng nắm bắt được xu hướng biến đổi dân số.

## **4.2. Kết luận**

**1. Sản phẩm đã làm được**

Ứng dụng web có khả năng:

Lấy dữ liệu dân số từ tập tin.

Cho phép người dùng chọn quốc gia và số năm dự báo.

Thực hiện dự báo dân số bằng mô hình ARIMA.

Hiển thị kết quả dưới dạng bảng và biểu đồ trên giao diện web.

Giao diện được xây dựng bằng Flask, thân thiện và dễ sử dụng.

**2. Kiến thức học được**

Ứng dụng Flask vào xây dựng web app phân tích dữ liệu.

Làm việc với dữ liệu thời gian bằng thư viện Pandas.

Dự báo chuỗi thời gian bằng mô hình ARIMA.

Tạo biểu đồ với matplotlib và nhúng vào website.

Kết nối dữ liệu và xử lý tương tác giữa front-end và back-end.

**3. Hướng cải tiến trong tương lai**

**Cải Thiện trang web**: tạo ra một trang web bắt mắt và dễ sử dụng hơn

**Mở rộng mô hình**: So sánh nhiều mô hình dự báo khác nhau như Linear Regression, Prophet, hoặc LSTM để nâng cao độ chính xác.

**Thêm tính năng người dùng**: Cho phép lưu lịch sử dự báo theo từng tài khoản.

**Cải thiện giao diện người dùng (UI/UX)**: Tối ưu trên thiết bị di động, thêm biểu đồ tương tác.

**Bổ sung dữ liệu**: Ngoài dân số có thể thêm dữ liệu về tỷ lệ sinh, tử, di cư, mật độ dân số,...

# LỜI CẢM ƠN

Em xin gửi lời cảm ơn chân thành đến Thầy Nguyễn Văn Huy người đã tận tình hướng dẫn và chia sẻ những kiến thức quý báu trong suốt quá trình thực hiện đề tài này. Những lời khuyên và sự động viên của Thầy đã giúp em vượt qua được không ít khó khăn và hoàn thiện bài báo cáo một cách tốt nhất.

Em cũng rất biết ơn các bạn đồng nghiệp và cộng sự, những người đã luôn sẵn sàng hỗ trợ và chia sẻ kinh nghiệm với em. Nhờ có sự giúp đỡ của mọi người, em mới có thể triển khai và hoàn thành dự án này một cách suôn sẻ.