

# Time-Series Analysis and Forecasting with Spark and Kafka for Weather Data



## **Thành viên nhóm:**

- 1. Triệu Vũ Hoàn - 22022654**
- 2. Hoàng Đình Hưng - 22022662**
- 3. Lê Thành Đạt - 22022627**
- 4. Hoàng Bùi Tuân Anh - 22022611**
- 5. Lê Hoàng Anh - 22022563**

# Nội dung

- Giới thiệu.
- Thiết kế và triển khai hệ thống.
- Kết luận.



# Phần I: Giới thiệu

- Tổng quan về dự án
- Các công nghệ được sử dụng
- Tầm quan trọng và ứng dụng thực tế

# Phần 1: Giới thiệu

## 1. Lý do thực hiện

- **Thách thức:** Biến đổi khí hậu và hiện tượng thời tiết cực đoan đang gây ảnh hưởng nghiêm trọng đến nông nghiệp, giao thông, và công tác quản lý khẩn hoảng. Vì vậy, việc dự báo thời tiết chính xác và nhanh chóng trở thành một nhu cầu cấp thiết.
- **Vai trò của dự báo thời tiết:**
  - Giảm thiểu các rủi ro tiềm tàng.
  - Tối ưu hóa việc sử dụng nguồn lực.
  - Cung cấp thông tin kịp thời cho hoạt động cứu trợ.
  - Hỗ trợ đưa ra những quyết định đúng lúc.

# Phần 1: Giới thiệu

## 2. Mục Tiêu Dự Án

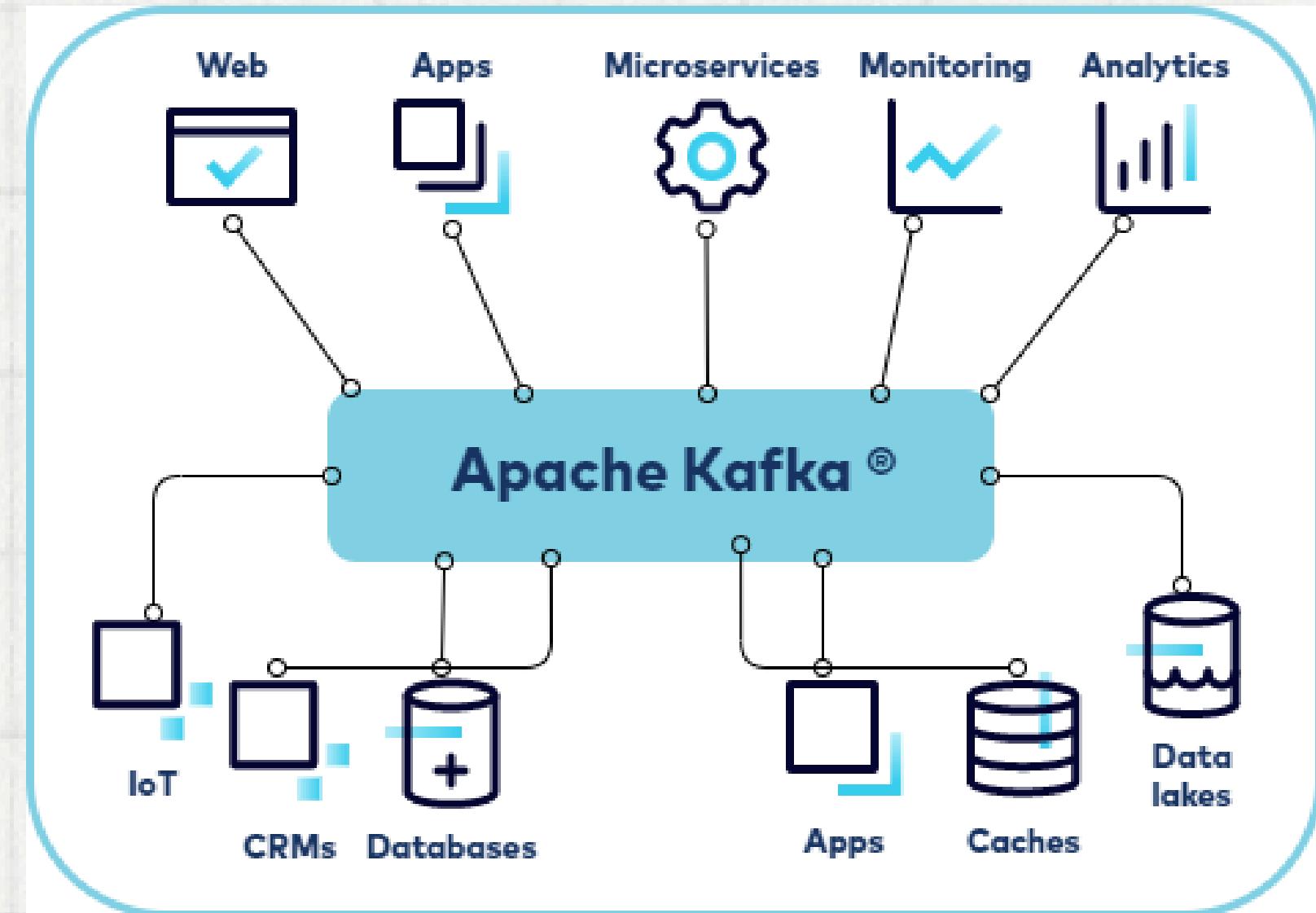
- Xây dựng hệ thống thu thập, phân tích và dự báo thời tiết theo thời gian thực.
- Tích hợp các công nghệ:
  - Apache Spark, Kafka, Power BI, Delta Lake, Docker.
- Ứng dụng các mô hình học máy tiên tiến để nâng cao độ chính xác.
- Đảm bảo hệ thống hoạt động hiệu quả, xử lý tốt khối lượng dữ liệu lớn.
- Xây dựng hệ thống hữu ích, hỗ trợ cải thiện việc ra quyết định dựa trên dữ liệu thời tiết.

# Phần 1: Giới thiệu

## 3. Các Công Nghệ Được Sử Dụng

### a. Apache Kafka

- Nền tảng phân tán chuyên dụng để truyền tải, lưu trữ và xử lý dữ liệu thời gian thực.
- Hoạt động như một hệ thống log phân tán hoặc hàng đợi tin nhắn.
- Đáp ứng tốt các ứng dụng cần xử lý dữ liệu lớn với độ trễ thấp.

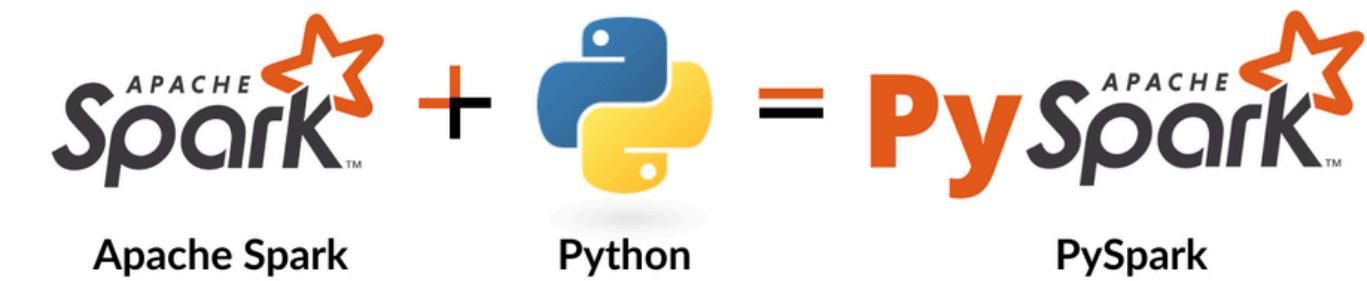
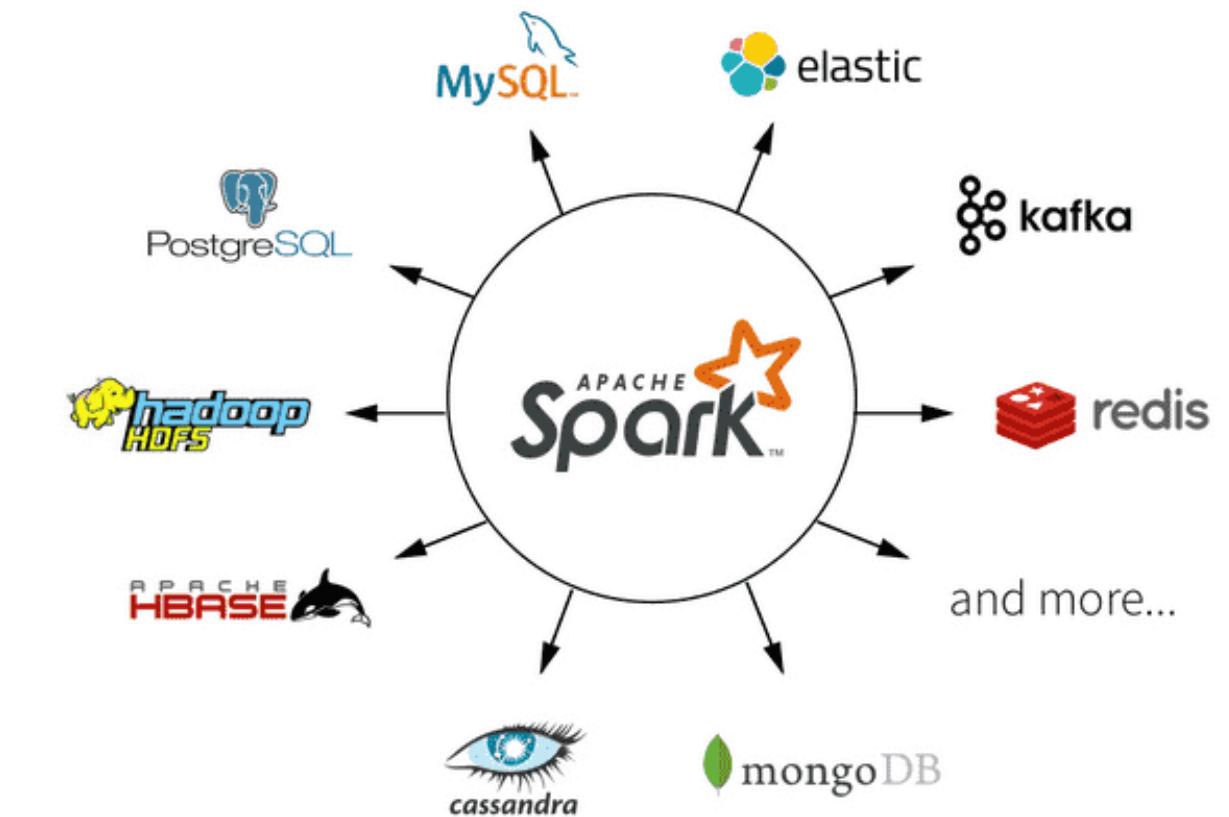


# Phần 1: Giới thiệu

## 3. Các công nghệ được sử dụng

### b. PySpark:

- Là một API dành cho Python, phát triển trên nền tảng Spark.
- Cho phép dùng Python để xử lý và phân tích dữ liệu lớn trên các cụm máy tính.

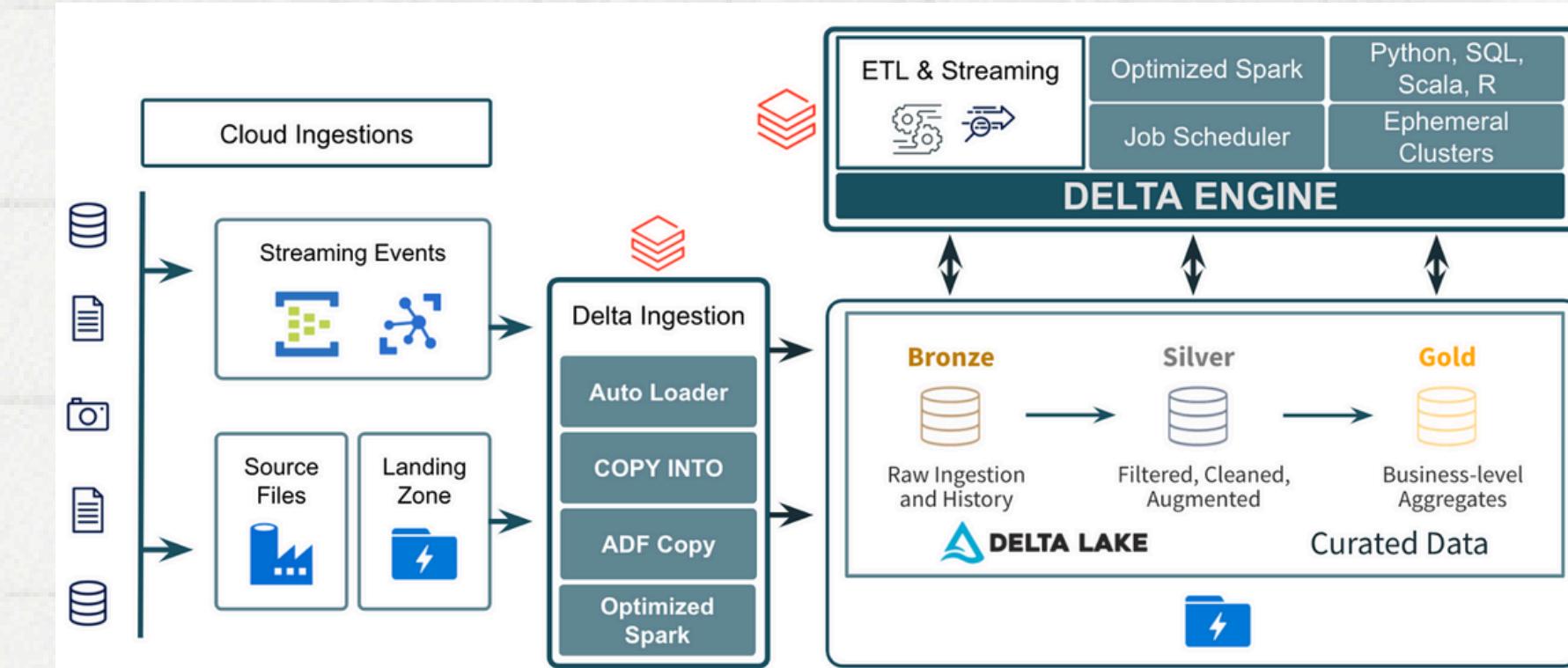


# Phần 1: Giới thiệu

## 3. Các công nghệ được sử dụng

### c. Delta Lake

- Là lớp lưu trữ mã nguồn được xây dựng trên Apache Spark.
- Cung cấp khả năng lưu trữ dữ liệu đáng tin cậy, hiệu quả và nhất quán.

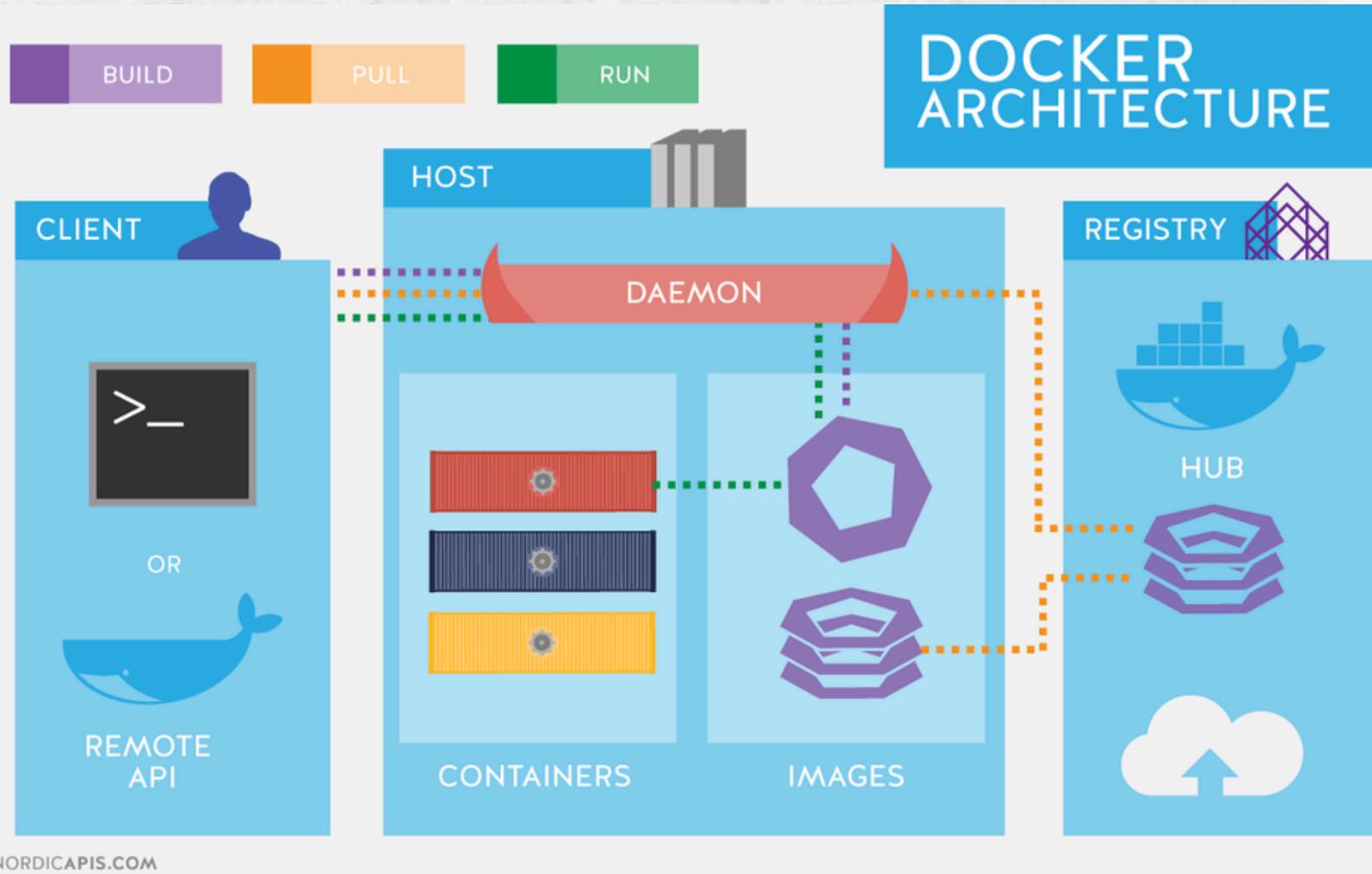


# Phần 1: Giới thiệu

## 3. Các công nghệ được sử dụng

### d. Docker

- Là công cụ tạo container, giúp đóng gói mã nguồn, thư viện và môi trường cần thiết vào một khố duy nhất.
- Đảm bảo tính linh hoạt, khả năng mở rộng và quản lý dễ dàng.
- Dễ dàng triển khai hệ thống lên cloud hoặc server.



# Phần 1: Giới thiệu

## 3. Các công nghệ được sử dụng

### e. Power BI

- Là công cụ phân tích dữ liệu mạnh mẽ phát triển bởi Microsoft, cho phép trực quan hóa và chia sẻ dữ liệu dưới dạng biểu đồ, báo cáo.
- Power BI giúp kết nối, xử lý, và tích hợp dữ liệu từ nhiều nguồn khác nhau.



# **Phần II: Thiết kế và triển khai hệ thống**

- Mục tiêu và mô tả hệ thống
- Quy trình hoạt động
- Phân tích dữ liệu với Power BI
- Triển khai và đánh giá mô hình dự đoán nhiệt độ
- Triển khai và đánh giá mô hình dự đoán thời tiết

# Phần 2: Thiết kế và triển khai hệ thống

## Mục Tiêu

Hệ thống dự báo thời tiết được thiết kế để thu thập, xử lý và lưu trữ dữ liệu thời tiết thời gian thực từ API thông qua Delta Lake. Đồng thời, hệ thống còn có khả năng lưu trữ và phân tích dữ liệu lịch sử, hỗ trợ các hoạt động dự đoán và lập báo cáo.

## Mô Tả Hệ Thống

- Docker: Cấu hình hệ thống được định nghĩa trong tệp docker-compose.yml.
- Producer: Sử dụng Python để truy vấn dữ liệu thời tiết từ API.
- Apache Kafka: Đóng vai trò trung gian truyền tải dữ liệu. Các thông điệp được gửi lên weatherkafkatopic sẽ được Kafka quản lý và chờ Consumer xử lý.
- Consumer: Sử dụng Spark Streaming để xử lý dữ liệu từ Kafka.
- Delta Lake: Lưu trữ dữ liệu đã xử lý để phục vụ phân tích và dự đoán.
- Spark: Tích hợp PySpark và Spark MLlib để phân tích, xử lý dữ liệu và triển khai mô hình dự báo.



# Phần 2: Thiết kế và triển khai hệ thống

## Mô Tả Hệ Thống

- **Quy trình phân tích trong Power BI:**

- Sau khi kết nối với Delta Lake, dữ liệu thời tiết được nhập vào Power BI để xây dựng các dashboard và báo cáo tương tác.

- **Trực quan hóa:**

- Tạo biểu đồ và đồ thị tương tác như biểu đồ đường (line chart) để thể hiện xu hướng nhiệt độ theo thời gian, hoặc biểu đồ thanh (bar chart) để so sánh lượng mưa giữa các khu vực.
  - Sử dụng bản đồ nhiệt (heatmap) để phân tích mức độ ảnh hưởng thời tiết tại từng địa điểm cụ thể.

- **Cập nhật dữ liệu thời gian thực:**

- Power BI có thể được cấu hình để tự động làm mới dữ liệu từ Delta Lake ở các khoảng thời gian định kỳ, đảm bảo dashboard luôn hiển thị thông tin mới nhất.

# Phần 2: Thiết kế và triển khai hệ thống

## Quy trình triển khai

- Chuẩn bị Docker Compose
- Khởi chạy các container: Sử dụng lệnh docker-compose up -d
- Kiểm tra trạng thái container: docker ps. Đảm bảo hai container zookeeper và broker hoạt động bình thường.
- Kết nối Kafka từ ứng dụng
  - Producer và Consumer kết nối với Kafka thông qua địa chỉ localhost:9092
  - Nếu ứng dụng chạy trong container khác trong cùng mạng Docker, sử dụng địa chỉ broker:29092

# Phần 2: Thiết kế và triển khai hệ thống

## Quy trình hoạt động

Hệ thống thu thập dữ liệu thời tiết từ Open-Meteo API, một nền tảng miễn phí cho phép truy xuất dữ liệu linh hoạt.

### Các thông tin chính gồm:

- Thời gian: Định dạng ISO 8601.
- Nhiệt độ: Đơn vị °C.
- Lượng mưa: Đơn vị mm.
- Tốc độ gió: Đơn vị km/h hoặc m/s.
- Tình trạng thời tiết: Nắng, mưa, gió mạnh, v.v.

### Quy Trình Thu Thập Dữ Liệu

- Kết nối API: Sử dụng Python (thư viện requests) để lấy dữ liệu.
- Khung thời gian: Thu thập dữ liệu theo chu kỳ giờ, ngày hoặc tháng.
- Lưu trữ: Lưu dưới định dạng bảng Delta Lake để phục vụ phân tích.

### Hệ Thống Thu Thập Dữ Liệu Thời Gian Thực

- Thu thập: Producer định kỳ truy vấn API để lấy dữ liệu.
- Gửi dữ liệu: Dữ liệu được gửi tới Kafka topic dưới dạng JSON.
- Xử lý: Consumer (spark streaming) đọc và chuẩn hóa các thông tin (nhiệt độ, độ ẩm, lượng mưa, v.v.).
- Lưu trữ: Dữ liệu được lưu vào Delta Lake trong chế độ streaming, hỗ trợ phân tích và truy vấn hiệu quả.

# Phần 2: Thiết kế và triển khai hệ thống

## Quy trình hoạt động

### Ưu điểm hệ thống:

- Thời gian thực: Xử lý ngay khi API tạo ra dữ liệu.
- Mở rộng: Tích hợp Apache Kafka và PySpark.
- Bền vững: Lưu trữ với Delta Lake, hỗ trợ ACID và lịch sử thay đổi.

### Dữ liệu lịch sử:

- Gồm hơn 130.000 bản ghi từ năm 2010-2024 tại Hà Nội, Việt Nam.

# Phần 2: Thiết kế và triển khai hệ thống

## Phân tích dữ liệu với Power BI

Dữ liệu bao gồm các thông tin về:

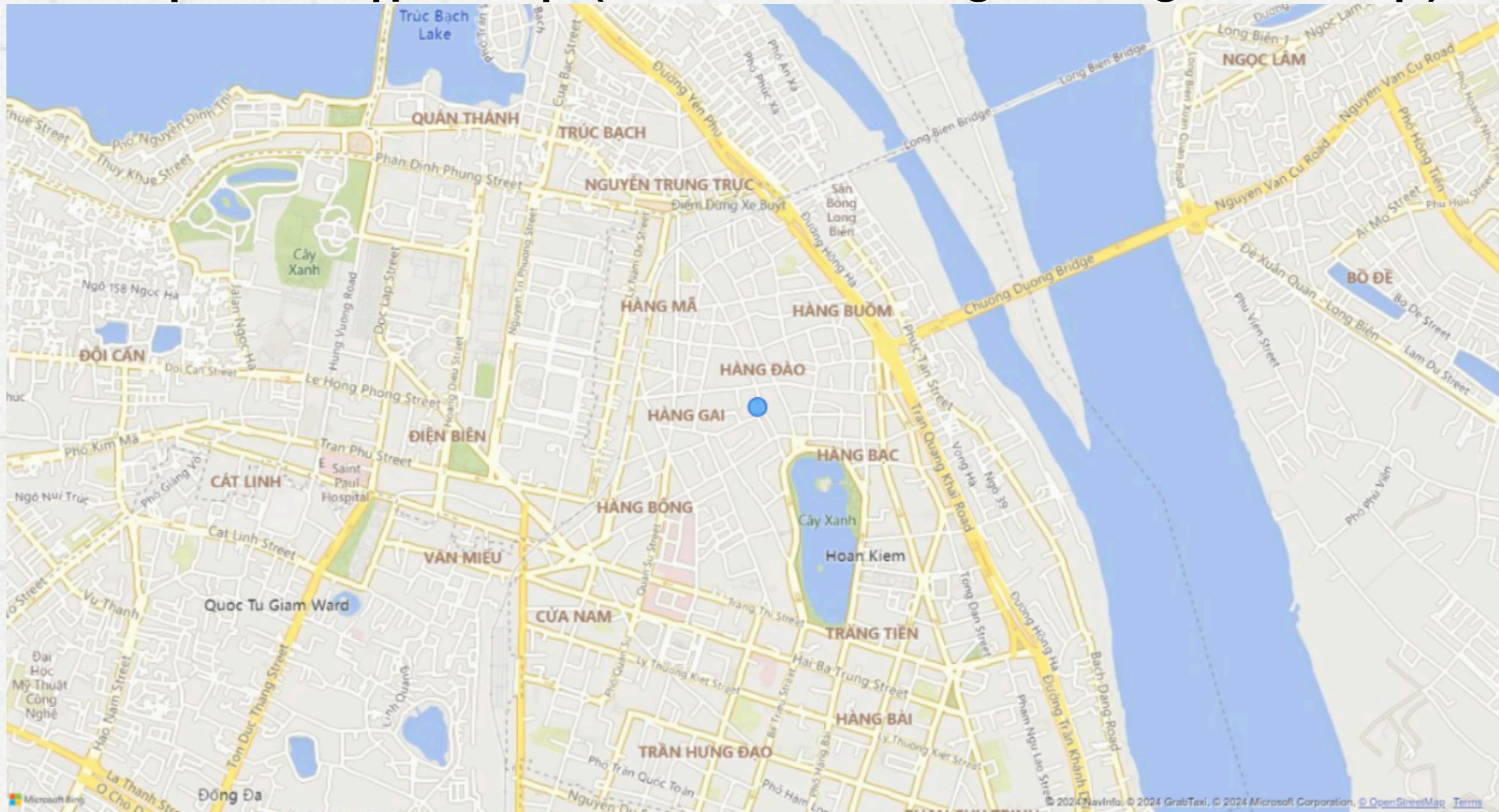
- Datetime: dữ liệu thời gian(ngày/tháng/năm giờ)
- Name: địa điểm thu thập dữ liệu( Hà Nội)
- Country: đất nước (Việt Nam)
- Latitude và longitude(vị trí trên bản đồ)
- Temp\_c: nhiệt độ
- Wind\_mph : tốc độ gió
- Humidity: độ ẩm
- Precip\_mm: lượng mưa
- Condition: trình trạng thời tiết tại thời điểm đó

datetime	name	country	latitude	longitude	temp_c	wind_mph	humidity	precip_mm	condition
29/10/2024 5:00:00 PM	Hanoi	Vietnam	21.0333	105.85	22.5	4.9	72	0	Partly cloudy
29/10/2024 6:00:00 PM	Hanoi	Vietnam	21.0333	105.85	22.1	5.4	73	0	Partly cloudy
29/10/2024 7:00:00 PM	Hanoi	Vietnam	21.0333	105.85	21.6	5.1	75	0	Partly cloudy
29/10/2024 8:00:00 PM	Hanoi	Vietnam	21.0333	105.85	21.2	5.6	77	0	Partly cloudy
29/10/2024 9:00:00 PM	Hanoi	Vietnam	21.0333	105.85	21	5.6	78	0	Partly cloudy
29/10/2024 10:00:00 PM	Hanoi	Vietnam	21.0333	105.85	21	5.4	78	0	Partly cloudy
29/10/2024 11:00:00 PM	Hanoi	Vietnam	21.0333	105.85	20.9	5.4	78	0	Partly cloudy
30/10/2024 12:00:00 AM	Hanoi	Vietnam	21.0333	105.85	21.6	5.8	76	0	Partly cloudy
30/10/2024 4:00:00 AM	Hanoi	Vietnam	21.0333	105.85	26.1	4.3	59	0	Partly cloudy
30/10/2024 5:00:00 AM	Hanoi	Vietnam	21.0333	105.85	27	5.1	55	0	Partly cloudy
30/10/2024 6:00:00 AM	Hanoi	Vietnam	21.0333	105.85	27.6	6.5	54	0	Partly cloudy
30/10/2024 7:00:00 AM	Hanoi	Vietnam	21.0333	105.85	27.9	6.7	52	0	Partly cloudy
30/10/2024 8:00:00 AM	Hanoi	Vietnam	21.0333	105.85	27.7	6	52	0	Partly cloudy
30/10/2024 9:00:00 AM	Hanoi	Vietnam	21.0333	105.85	27.3	4.9	53	0	Partly cloudy
30/10/2024 10:00:00 AM	Hanoi	Vietnam	21.0333	105.85	26.2	3.4	59	0	Partly cloudy
30/10/2024 11:00:00 AM	Hanoi	Vietnam	21.0333	105.85	25.3	4	60	0	Partly cloudy
30/10/2024 12:00:00 PM	Hanoi	Vietnam	21.0333	105.85	24.9	3.6	62	0	Partly cloudy
30/10/2024 1:00:00 PM	Hanoi	Vietnam	21.0333	105.85	24.5	3.4	64	0	Partly cloudy
30/10/2024 2:00:00 PM	Hanoi	Vietnam	21.0333	105.85	24.2	3.4	66	0	Partly cloudy
30/10/2024 3:00:00 PM	Hanoi	Vietnam	21.0333	105.85	23.9	3.6	67	0	Partly cloudy
30/10/2024 4:00:00 PM	Hanoi	Vietnam	21.0333	105.85	23.6	2.7	68	0	Partly cloudy
30/10/2024 5:00:00 PM	Hanoi	Vietnam	21.0333	105.85	23.4	2.2	69	0	Partly cloudy
30/10/2024 6:00:00 PM	Hanoi	Vietnam	21.0333	105.85	23.1	1.3	70	0	Partly cloudy
30/10/2024 7:00:00 PM	Hanoi	Vietnam	21.0333	105.85	22.9	2.7	70	0	Partly cloudy
30/10/2024 8:00:00 PM	Hanoi	Vietnam	21.0333	105.85	22.6	2.7	71	0	Partly cloudy
30/10/2024 9:00:00 PM	Hanoi	Vietnam	21.0333	105.85	22.4	2.2	72	0	Partly cloudy
30/10/2024 10:00:00 PM	Hanoi	Vietnam	21.0333	105.85	22.2	2.5	72	0	Partly cloudy

# Phần 2: Thiết kế và triển khai hệ thống

## Phân tích dữ liệu với Power BI

Khu vực thu thập dữ liệu(21.0333 105.05, gần trung tâm hà nội)



# Phần 2: Thiết kế và triển khai hệ thống

## Phân tích dữ liệu với Power BI

- Biểu đồ độ ẩm, nhiệt độ, tốc độ gió, lượng mưa trung bình theo tháng
- Tháng có nhiệt độ thấp nhất là tháng 1(tới 8.9 C)
- Tháng có nhiệt độ cao nhất là tháng 4(tới 43.8 C)
- Các tháng có nhiệt độ trung bình là 28 C, standard deviation là 5.1

hour	January	February	March	April	May	June	July	August	September	October	November	December
0	14.35	16.03	19.29	22.69	25.91	27.58	27.22	26.47	25.37	22.62	19.89	14.77
1	15.11	16.76	20.09	23.68	26.97	28.70	28.28	27.46	26.39	23.90	21.08	15.89
2	16.05	17.66	20.98	24.70	28.01	29.75	29.32	28.44	27.37	25.05	22.26	17.01
3	16.97	18.69	21.92	25.74	29.02	30.69	30.28	29.31	28.28	26.05	23.29	18.01
4	17.81	19.64	22.87	26.70	29.84	31.50	31.01	30.06	29.03	26.82	24.14	18.87
5	18.53	20.46	23.73	27.52	30.49	32.08	31.53	30.59	29.58	27.38	24.77	19.56
6	19.06	21.09	24.36	28.11	30.87	32.36	31.80	30.93	29.84	27.64	25.17	20.04
7	19.26	21.34	24.52	28.21	31.04	32.49	31.91	31.01	29.86	27.77	25.32	20.30
8	19.36	21.55	24.66	28.32	31.04	32.48	31.82	30.87	29.67	27.65	25.25	20.38
9	19.24	21.45	24.52	28.06	30.80	32.19	31.50	30.45	29.32	27.24	24.86	20.13
10	18.70	20.91	24.00	27.40	30.14	31.58	30.91	29.86	28.75	26.35	23.80	19.25
11	17.67	19.73	22.91	26.24	29.08	30.61	30.06	29.05	27.77	25.24	22.80	18.18
12	16.95	18.83	21.92	25.15	28.13	29.62	29.23	28.37	27.27	24.65	22.22	17.48
13	16.53	18.29	21.32	24.50	27.54	29.07	28.68	27.89	26.88	24.27	21.77	17.04
14	16.06	17.74	20.78	23.93	27.01	28.62	28.23	27.50	26.48	23.83	21.30	16.54
15	15.65	17.31	20.37	23.49	26.62	28.26	27.88	27.16	26.11	23.42	20.91	16.12
16	15.32	16.97	20.06	23.14	26.29	27.94	27.53	26.86	25.80	23.07	20.56	15.76
17	15.07	16.70	19.83	22.84	25.99	27.64	27.24	26.59	25.52	22.77	20.27	15.46
18	14.87	16.50	19.62	22.60	25.72	27.37	26.98	26.32	25.26	22.50	20.04	15.24
19	14.72	16.33	19.45	22.41	25.55	27.16	26.81	26.20	25.12	22.35	19.91	15.07
20	14.60	16.19	19.31	22.27	25.36	26.93	26.61	26.01	24.93	22.15	19.72	14.84
21	14.46	16.08	19.18	22.13	25.20	26.72	26.43	25.83	24.78	21.97	19.57	14.66
22	14.33	16.00	19.07	22.03	25.04	26.54	26.26	25.68	24.65	21.81	19.47	14.51
23	14.24	15.93	18.99	22.03	25.17	26.69	26.34	25.67	24.57	21.69	19.35	14.37

January  
30.10 Max of temp\_c 16.45 Average of temp\_c  
4.40 Min of temp\_c

February  
34.50 Max of temp\_c 18.26 Average of temp\_c  
6.20 Min of temp\_c

March  
36.10 Max of temp\_c 21.41 Average of temp\_c  
9.40 Min of temp\_c

April  
41.00 Max of temp\_c 24.75 Average of temp\_c  
12.10 Min of temp\_c

May  
39.00 Max of temp\_c 27.78 Average of temp\_c  
17.50 Min of temp\_c

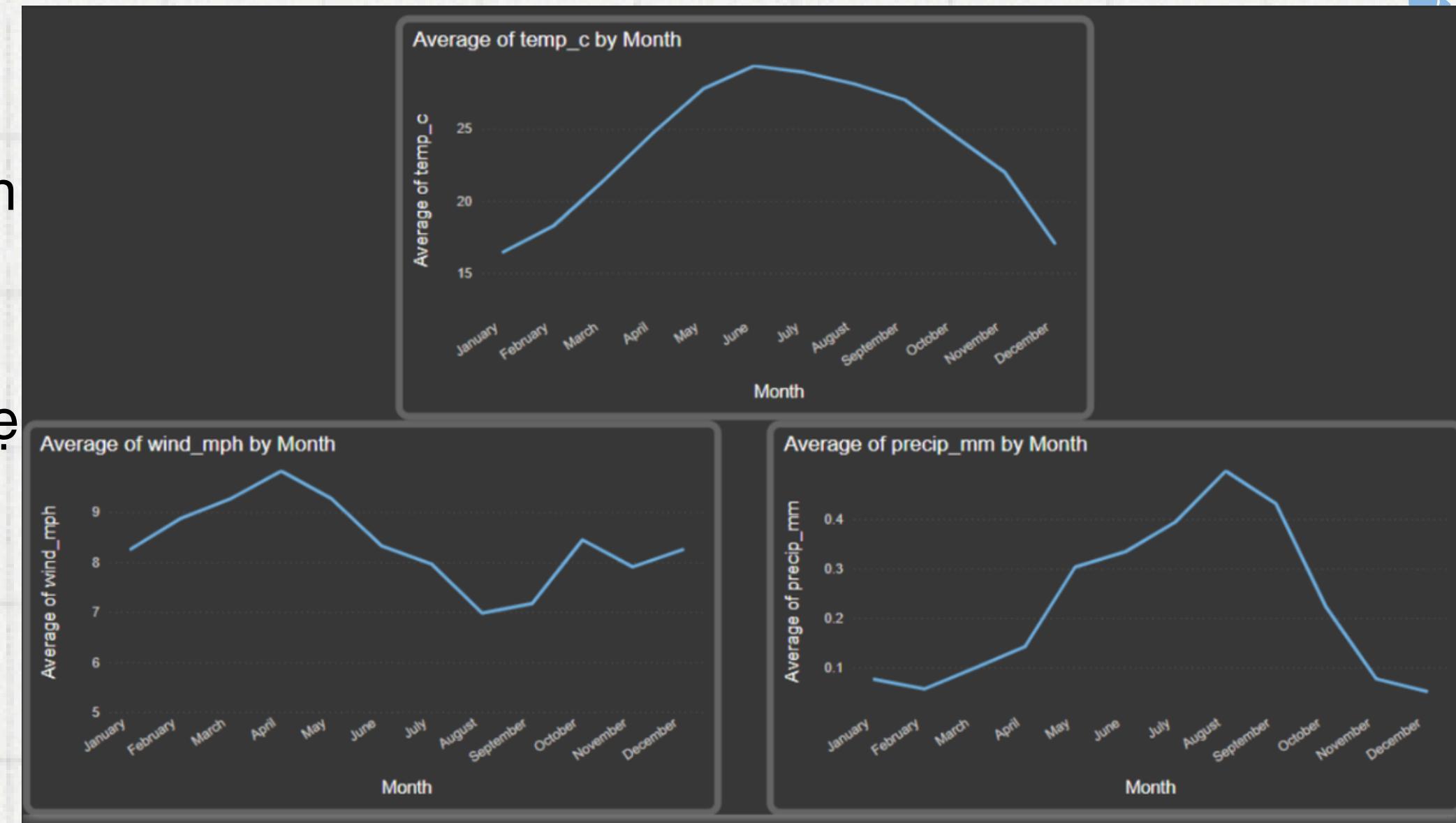
June  
38.50 Max of temp\_c 29.36 Average of temp\_c  
19.40 Min of temp\_c

July

# Phần 2: Thiết kế và triển khai hệ thống

## Phân tích dữ liệu với Power BI

- Nhiệt độ: Tăng dần từ tháng 1 và đạt đỉnh vào tháng 6, sau đó giảm dần đến cuối năm.
- Tốc độ gió: Tốc độ trung bình giao động mạnh, đạt đỉnh vào tháng 4 và thấp nhất vào tháng 7, sau đó tăng nhẹ vào cuối năm.
- Lượng mưa: Thấp nhất vào đầu năm và tăng đáng kể vào tháng 7 và đạt đỉnh vào tháng 9

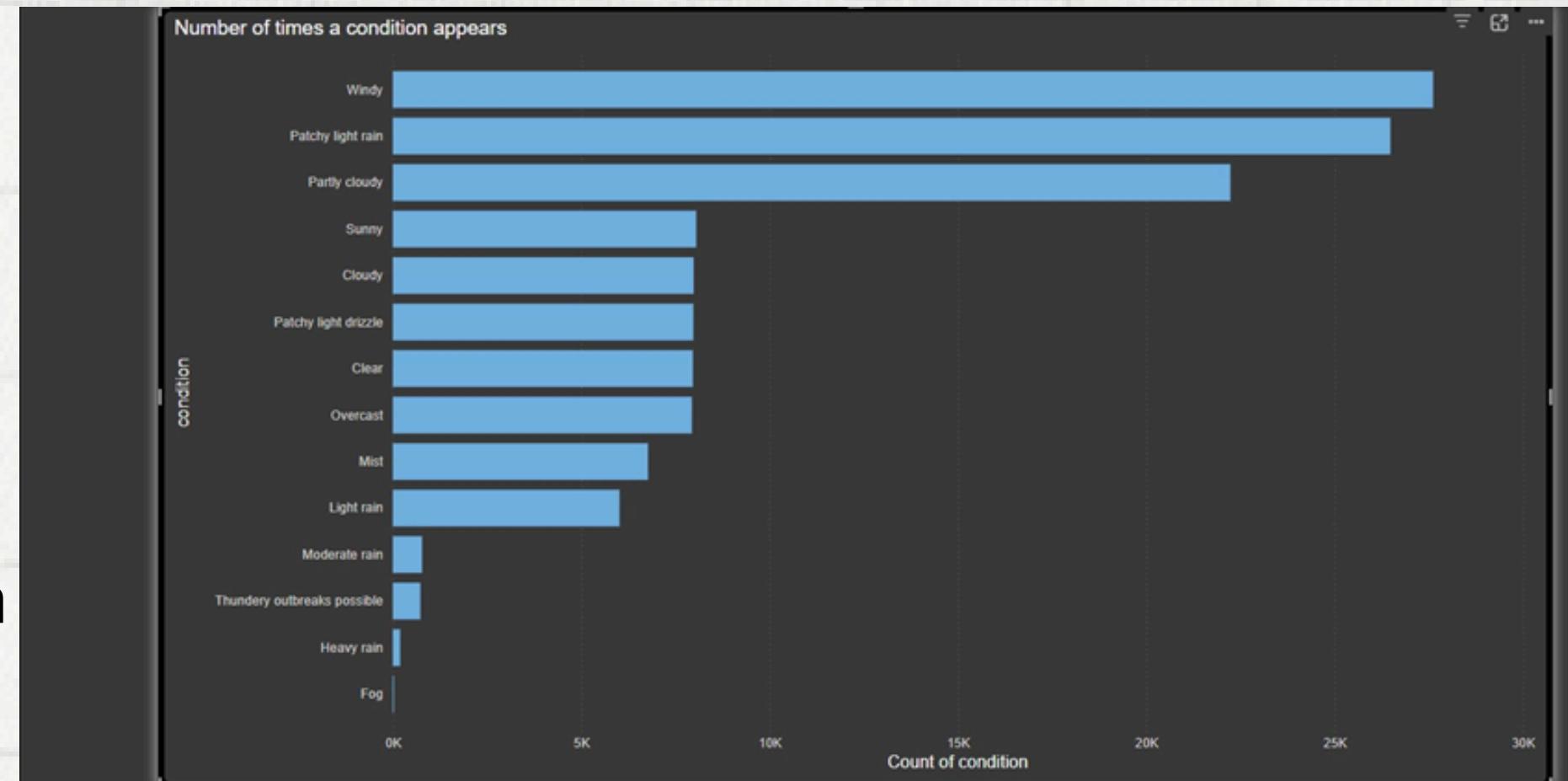


# Phần 2: Thiết kế và triển khai hệ thống

## Phân tích dữ liệu với Power BI

Dữ liệu classification(các tình trạng thời tiết - weather conditions)

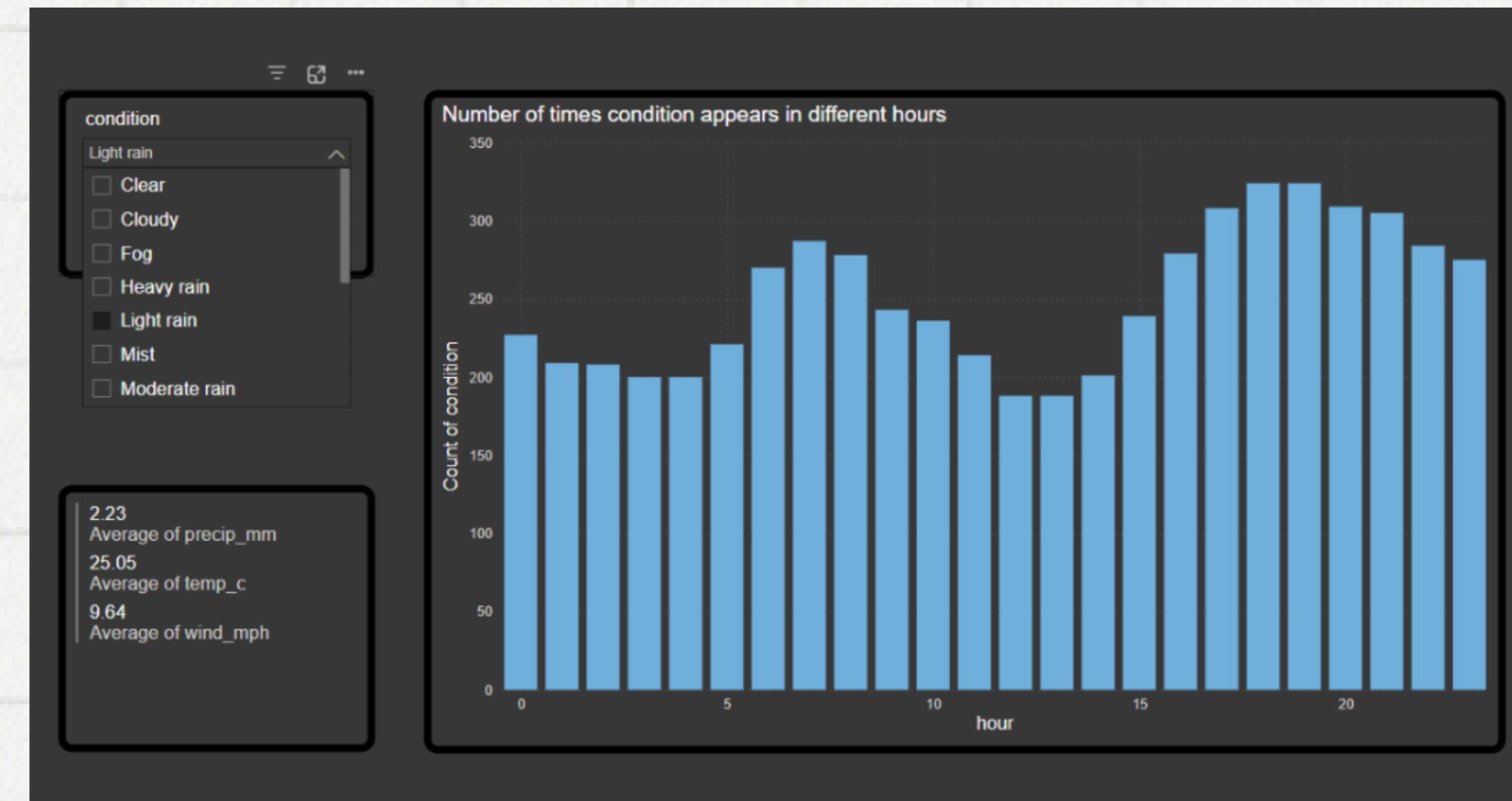
- Thời tiết chủ yếu ôn hòa với các điều kiện nhẹ như Windy, Patchy rain (có mưa rải rác), Partly cloudy (trời có mây), Overcast (âm u) và Clear( trời quang) chiếm đa số.
- Các điều kiện cực đoan như Heavy rain (mưa lớn) hay thunderstorms(sấm sét) xảy ra ít, cho thấy khu vực có thời tiết tương đối ổn định, không thường xuyên xảy ra hiện tượng khắc nghiệt.



# Phần 2: Thiết kế và triển khai hệ thống

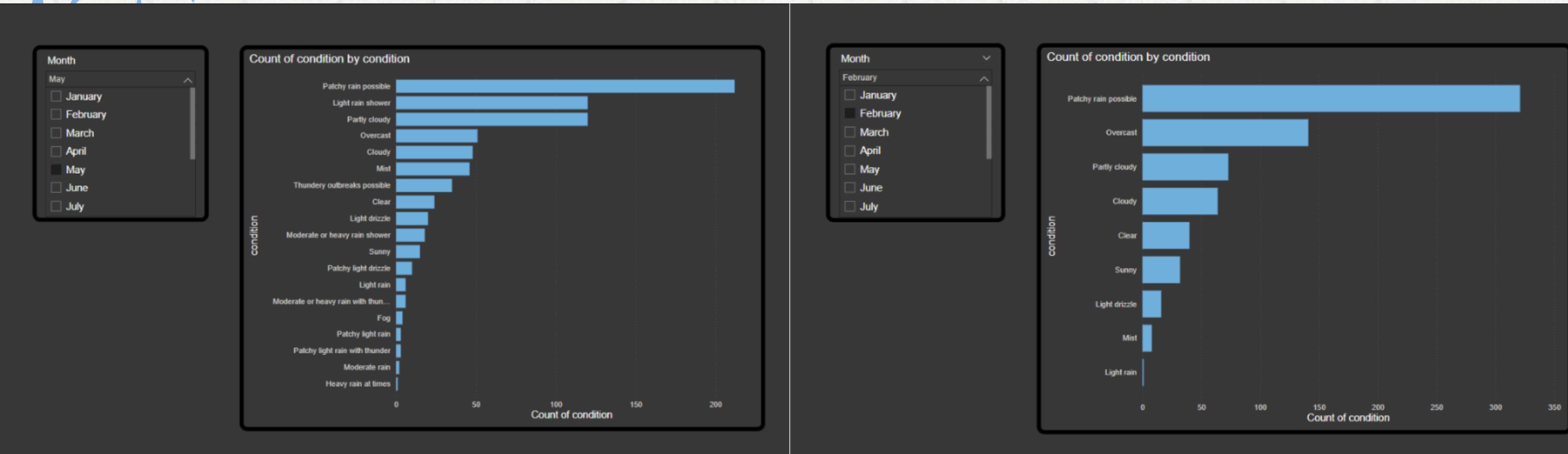
## Phân tích dữ liệu với Power BI

- Mỗi tình trạng thời tiết đều có các đặc điểm dữ liệu khác nhau
- Mô hình phân tích dữ liệu đặc trưng để phân loại các condition
- Ví dụ ở đây với condition là cloudy, thấy trung bình temp là 20.27 C, tb gió là 6.03, không có precipitation nào



# Phần 2: Thiết kế và triển khai hệ thống

## Phân tích dữ liệu với Power BI

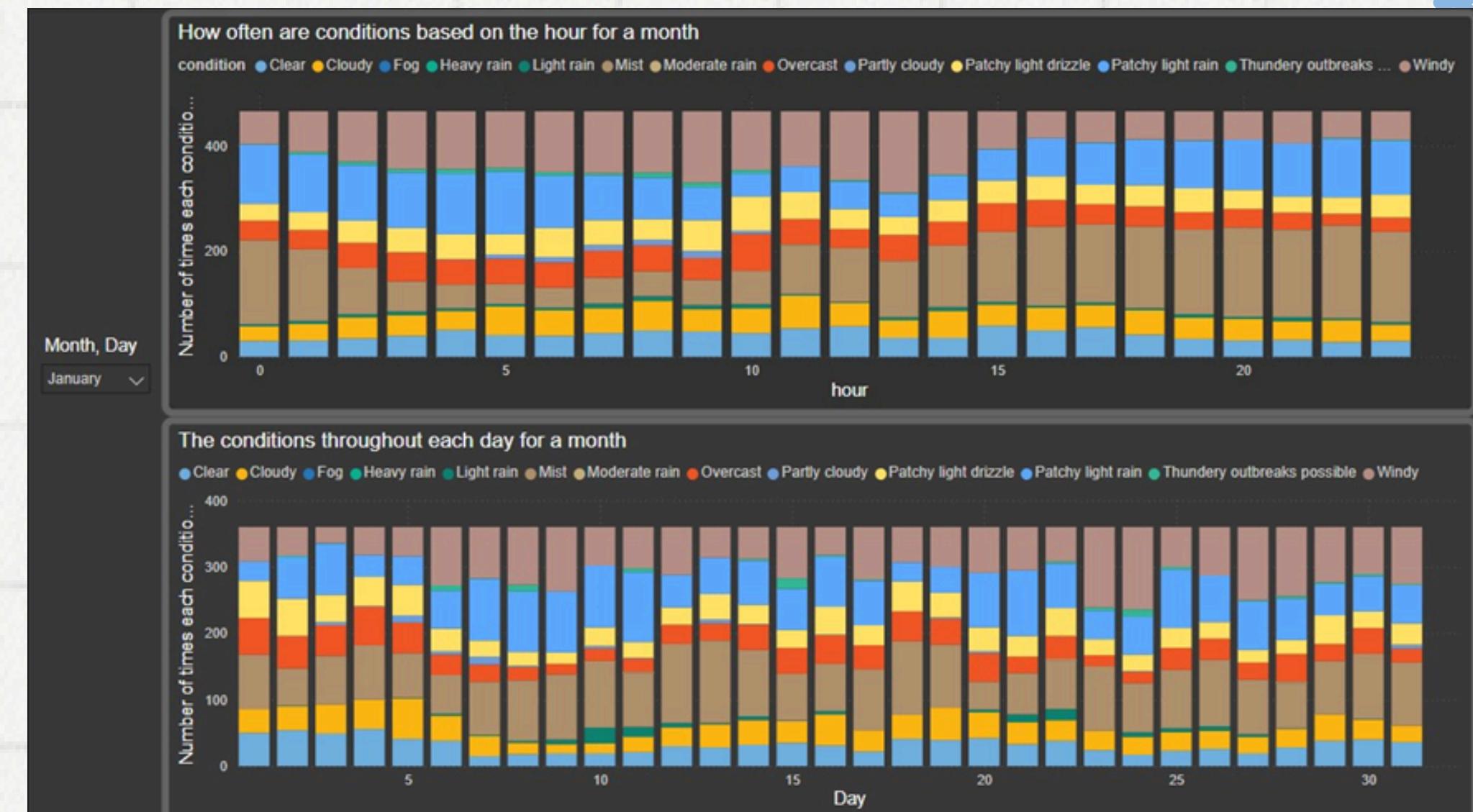


- Tùy vào từng tháng các weather conditions cụ thể có thể xảy ra thường xuyên hoặc ít thường xuyên hơn, ví dụ như ở trên ta có thể thấy tháng 2 và tháng 10 mưa nhiều

# Phần 2: Thiết kế và triển khai hệ thống

## Phân tích dữ liệu với Power BI

- Ta có thể thấy phân bố conditions của mỗi ngày đều khác nhau, và các conditions thường có khung giờ cụ thể xuất hiện



# Phần 2: Thiết kế và triển khai hệ thống

## Phân tích dữ liệu với Power BI

Tháng 1	Tháng 2	Tháng 3	Tháng 4	Tháng 5	Tháng 6	Tháng 7	Tháng 8	Tháng 9	Tháng 10	Tháng 11	Tháng 12
TOP 5											
Cloudy	Clear	Cloudy	Clear	Cloudy	Clear	Clear	Clear	Clear	Clear	Clear	Cloudy
Light drizzle	Cloudy	Overcast	Partly cloudy	Light rain shower	Light rain shower	Light drizzle					
Overcast	Overcast	Partly cloudy	Patchy rain possible	Overcast	Partly cloudy	Partly cloudy	Overcast				
Partly cloudy	Partly cloudy	Patchy rain possible	Sunny	Partly cloudy	Patchy rain possible	Patchy rain possible	Partly cloudy				
Patchy rain possible	Patchy rain possible	Sunny	Thundery outbreak...	Partly cloudy	Thundery outbreak...	Thundery outbreak...	Patchy rain possible				
BOTTOM 5											
Clear	Clear	Light rain	Light drizzle	Fog	Mist	Light rain	Light rain	Heavy rain	Moderate or heavy...	Light drizzle	Clear
Light rain	Light drizzle	Moderate rain	Light rain	Light rain shower	Moderate rain	Moderate rain	Moderate rain	Moderate rain	Moderate or heavy...	Light rain	Light rain
Light rain shower	Light rain	Moderate rain at ti...	Mist	Moderate or heavy...	Moderate rain	Moderate rain at ti...	Moderate rain at ti...	Moderate rain at ti...	Moderate or heavy...	Light rain shower	Light rain shower
Moderate rain	Mist	Patchy light rain	Patchy light rain	Moderate rain at ti...	Patchy light rain	Patchy light rain	Patchy light rain	Overcast	Moderate rain at ti...	Overcast	Moderate rain
Patchy light drizzle	Sunny	Thundery outbreak...	Thundery outbreak...	Patchy light rain wi...	Patchy light drizzle	Patchy light drizzle					

- Ta có thể thấy từ tháng 5 đến tháng 10 là 'light rain shower' xuất hiện nhiều
- Clear và cloudy là 2 conditions xuất hiện thường xuyên nhất, tháng nào cũng top 3 (trừ tháng 12 và tháng 1).

# Phần 2: Thiết kế và triển khai hệ thống

## Triển khai mô hình

### I. Mô hình hóa dữ liệu

#### 1.1 Mô hình dự đoán nhiệt độ

- Mục tiêu: Dự đoán nhiệt độ ( $^{\circ}\text{C}$ ) sau 1 giờ dựa trên thời gian.
- Dữ liệu sử dụng:
  - Thông tin thời tiết: Nhiệt độ, độ ẩm, tốc độ gió, lượng mưa, điều kiện thời tiết.
  - Đặc trưng thời gian: Giờ, ngày, tháng, quý, tuần, ngày trong năm.
- Mô hình ban đầu: Chỉ sử dụng đặc trưng thời gian.

#### 1.2 Tiền xử lý dữ liệu

- Chuyển đổi cột datetime thành các đặc trưng:
  - hour, week, month, quarter, day\_of\_week, day\_of\_month, day\_of\_year.
- Chuẩn hóa đặc trưng thời gian để sử dụng trong mô hình.

$$\text{Normalized Value} = \frac{\text{Value} - \text{Min}}{\text{Max} - \text{Min}}$$

#### 1.3 Phân chia dữ liệu

- Dữ liệu huấn luyện: Trước 2022-10-01 12:00:00.
- Dữ liệu kiểm tra: Từ 2022-10-01 12:00:00 trở đi.

# Phần 2: Thiết kế và triển khai hệ thống

## Mô hình dự đoán nhiệt độ-P1

### 1. Random Forest Regressor

#### a. Huấn luyện mô hình

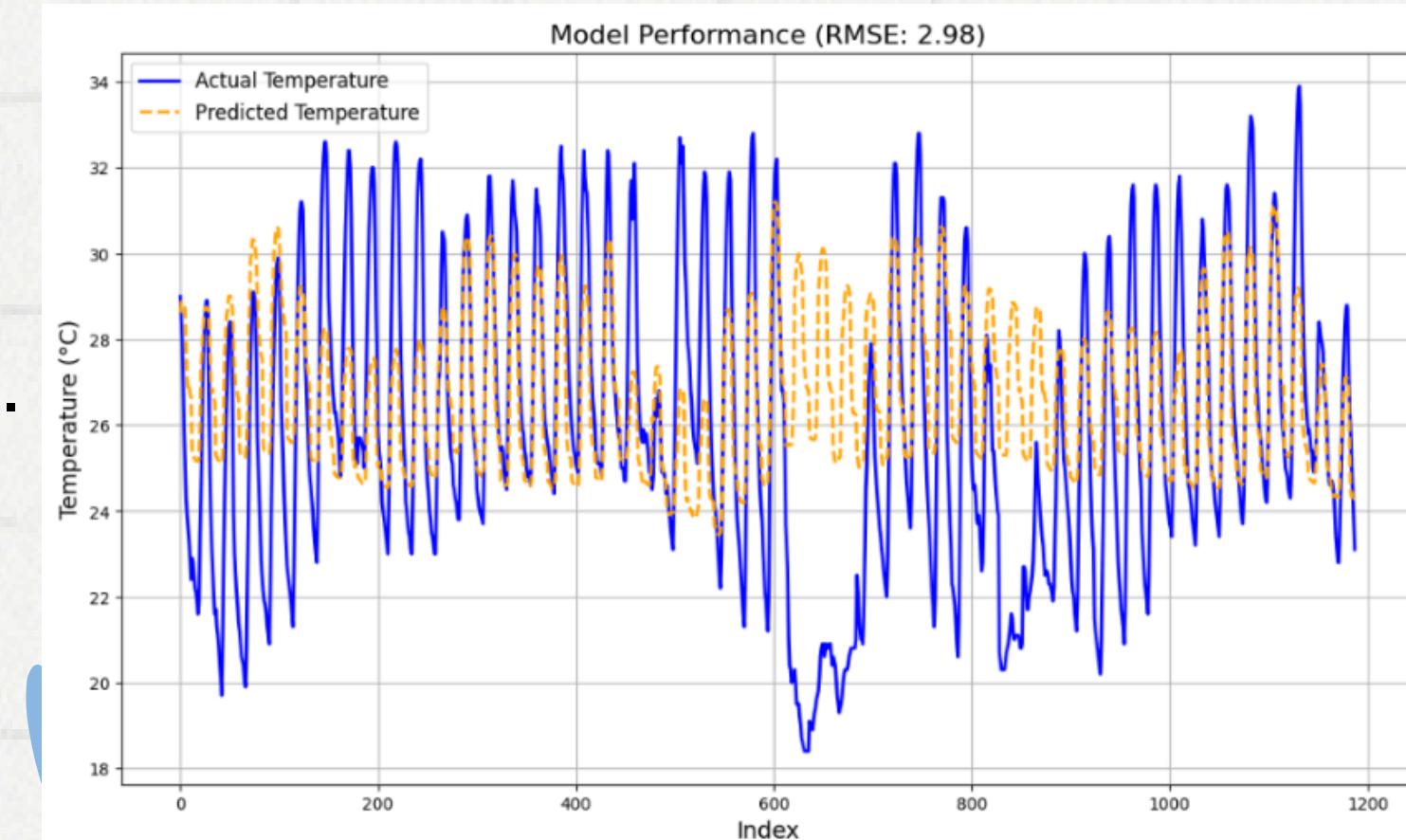
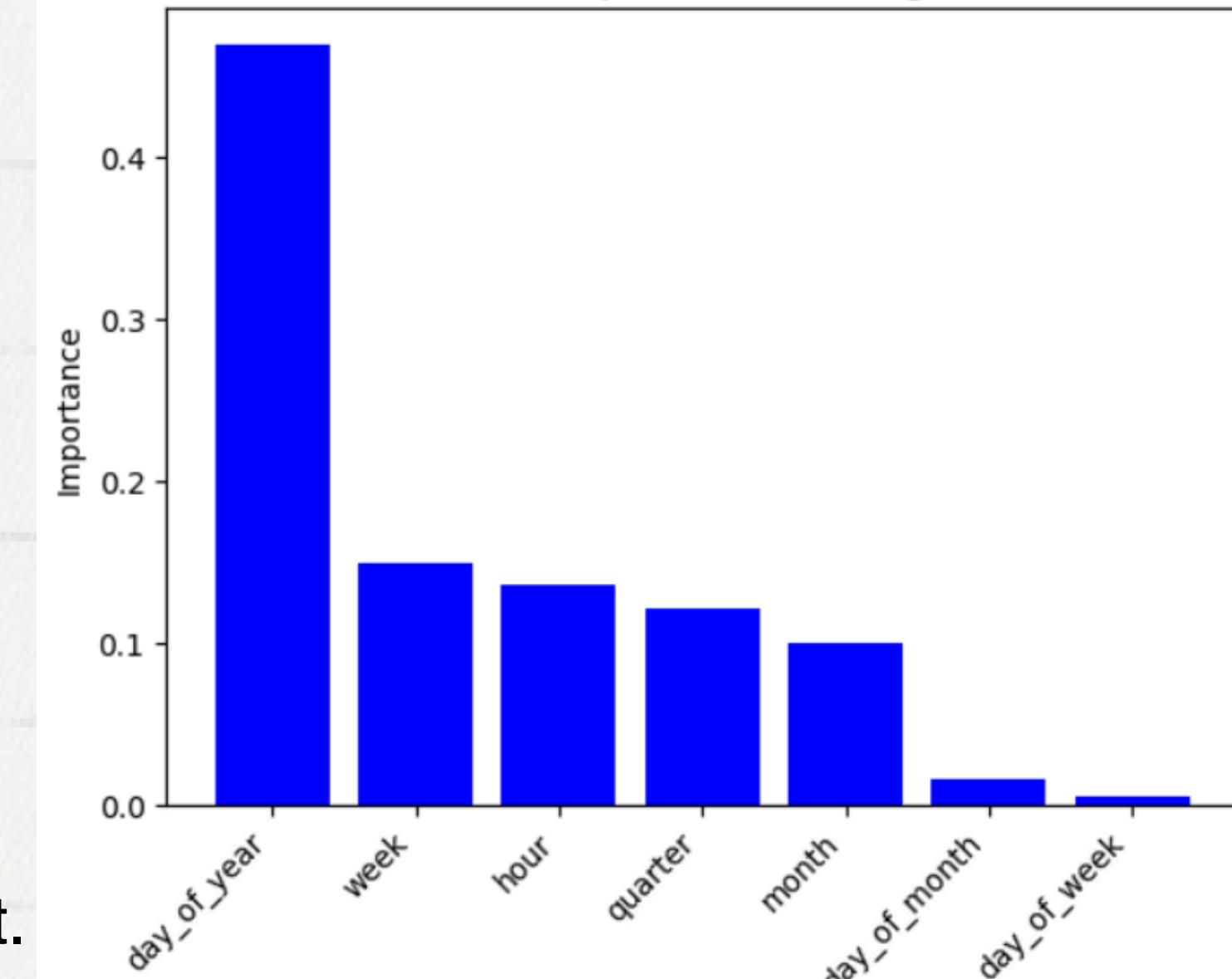
- Sử dụng Grid Search để tối ưu tham số:
  - numTrees: 10, 20, 50, 100
  - maxDepth: 5, 7, 10
- Áp dụng Cross-Validation (3-fold) để tìm tham số tốt nhất.

#### b. Hiệu suất mô hình

- RMSE: 2.98
- MSE: 8.88
- MAE: 2.4

#### c. Đặc trưng quan trọng

- Các đặc trưng quan trọng nhất: day\_of\_year, week, hour.
- Sai số lớn do chỉ sử dụng đặc trưng thời gian.



# Phần 2: Thiết kế và triển khai hệ thống

## Mô hình dự đoán nhiệt độ-P2

### 2. Cải Thiện Mô Hình

#### a. Mô Hình Sử Dụng

- Random Forest Regression
- Gradient Boosted Tree Regression

#### b. Tiền Xử Lý Dữ Liệu

- Mã hóa điều kiện thời tiết: Chuyển đổi các chuỗi ký tự thành giá trị số.
- Đặc trưng thời gian:
  - Thêm giá trị trễ (lag features) cho các thông số: nhiệt độ, độ ẩm, tốc độ gió, lượng mưa (4 mốc thời gian trước).
  - Tích hợp các đặc trưng bổ sung: giờ, ngày, tháng, quý, và tuần.
- Xử lý dữ liệu thiếu: Loại bỏ các bản ghi không đầy đủ bằng phương pháp dropna.

#### c. Kết Quả Cải Thiện

Mô hình tận dụng tối đa dữ liệu, bao gồm cả thông tin thời tiết lịch sử, để nâng cao độ chính xác.

# Phần 2: Thiết kế và triển khai hệ thống

## Phương pháp và đánh giá mô hình dự đoán nhiệt độ

### 1. Phương pháp sử dụng

#### a. Random Forest Regression

- Tham số tối ưu:
  - numTrees: 70
  - maxDepth: 5

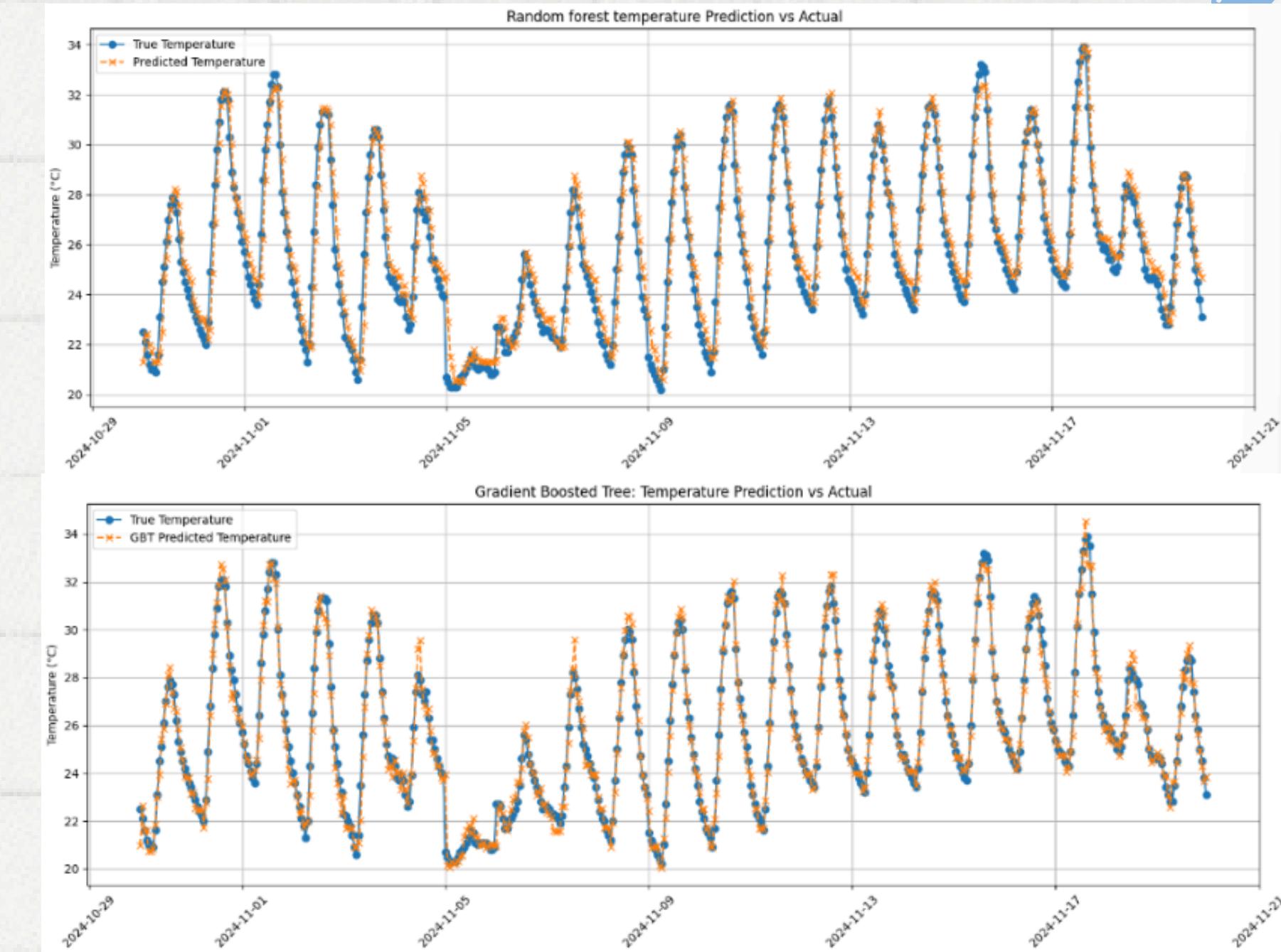
#### b. Gradient Boosted Tree Regression

- Tham số tối ưu:
  - maxIter: 50
  - maxDepth: 5

### 2. Đánh giá mô hình

- Tiêu chí: Root Mean Squared Error (RMSE)
  - Random Forest: RMSE = 0.89
  - Gradient Boosted Tree: RMSE = 0.63

=> Gradient Boosted Tree cho kết quả tốt hơn với RMSE thấp hơn, dự đoán chính xác hơn.



## Phần 2: Thiết kế và triển khai hệ thống

### Phương pháp và đánh giá mô hình dự đoán nhiệt độ

#### 3. Kết luận

- **Gradient Boosted Tree** là phương pháp tối ưu cho bài toán nhờ khả năng xử lý các mối quan hệ phi tuyến tính và đặc trưng phức tạp, mang lại độ chính xác cao.
- **Ứng dụng thực tiễn:** Kết quả mô hình có thể được áp dụng để dự báo thời tiết, cung cấp thông tin kịp thời và chính xác cho các hoạt động như nông nghiệp, giao thông và phòng chống thiên tai.

# Phần 2: Thiết kế và triển khai hệ thống

## Mô hình dự đoán tình trạng thời tiết (condition)

### Mục Tiêu

Dự đoán tình trạng thời tiết (Nắng, Mưa, Sương mù,...) trước 1 giờ dựa trên:

- Dữ liệu lịch sử: Điều kiện thời tiết, nhiệt độ, độ ẩm, tốc độ gió, và lượng mưa.
- Đặc trưng thời gian: Giờ, ngày, tháng, và quý.

### Dữ Liệu

Các cột dữ liệu bao gồm:

- Thời gian: datetime
- Nhiệt độ: temp\_c
- Độ ẩm: humidity
- Tốc độ gió: wind\_mph
- Lượng mưa: precip\_mm
- Điều kiện thời tiết: condition (được mã hóa thành condition\_index).

# Phần 2: Thiết kế và triển khai hệ thống

## Phương pháp và Mô hình

### Phương pháp

#### 1. Mã hóa nhãn:

- Dùng từ điển để mã hóa condition thành số nguyên.
- Sử dụng UDF (User Defined Function) trong PySpark.

#### 2. Tạo đặc trưng:

- Sử dụng dữ liệu từ 1-4 bước thời gian trước.
- Thêm đặc trưng thời gian: Giờ, ngày, tháng, tuần, quý.

#### 3. Chia dữ liệu:

- Tập huấn luyện: Trước ngày 30/10/2024.
- Tập kiểm tra: Sau ngày 30/10/2024.

### Mô hình thử nghiệm

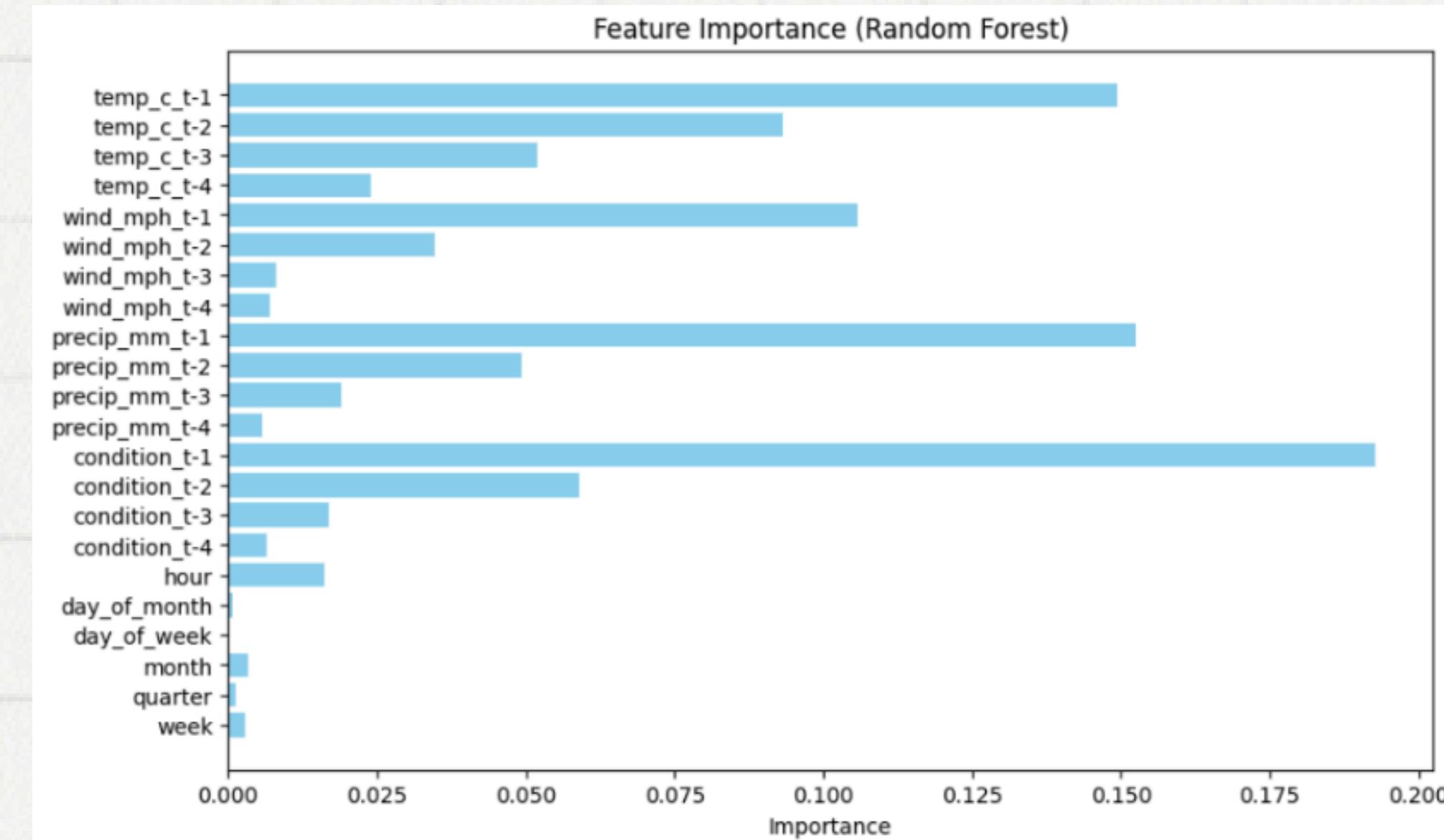
- Random Forest Classifier
- Logistic Regression
- Naive Bayes

# Phần 2: Thiết kế và triển khai hệ thống

## Kết quả

### Kết quả đánh giá

- Độ chính xác:
  - Logistic Regression: 37.5%
  - Naive Bayes: 20%
  - Random Forest: 61.5%
- **Random Forest** cho kết quả tốt nhất, các đặc trưng quan trọng nhất gồm:
  - Nhiệt độ, độ ẩm, tốc độ gió, lượng mưa, các đặc trưng thời gian.



# Phần 2: Thiết kế và triển khai hệ thống

## Kết luận

### Kết Luận

- **Random Forest** nổi bật nhờ khả năng xử lý các mối quan hệ phi tuyến tính và tương tác phức tạp giữa các đặc trưng.
- **Logistic Regression** và **Naive Bayes** không phù hợp vì:
  - **Logistic Regression**: Hạn chế trong việc xử lý các bài toán phân loại đa lớp phức tạp.
  - **Naive Bayes**: Giả định các đặc trưng độc lập, điều này không đúng trong thực tế.
- **Ví Dụ Hạn Chế**
  - Các đặc trưng như tốc độ gió và độ ẩm thường phụ thuộc vào nhiệt độ và điều kiện thời tiết.
  - Nếu không mô hình hóa đúng các mối quan hệ này, có thể dẫn đến sai lệch trong dự đoán.

# Phần 2: Thiết kế và triển khai hệ thống

## Thử nghiệm thực tế

### Pipeline Kafka và PySpark

- **Kafka:**

- Producer: Nhập dữ liệu từ API, CSV, hoặc database.
- Topic: Làm kênh trung gian cho dữ liệu thô.
- Consumer: Trích xuất dữ liệu và lưu vào Delta Lake.

- **PySpark:**

- Tạo Spark Session và Spark Streaming để xử lý dữ liệu từ Kafka.
- Tiền xử lý:
  - Chuẩn hóa đặc trưng.
  - Xóa dữ liệu khuyết thiếu.
- Đánh giá dữ liệu: Streaming và Batch.

### Kết quả thực tế

- Triển khai: Hoạt động ổn định, nhận dữ liệu thời gian thực từ API.
- Hiệu suất cao, dữ liệu được xử lý liên tục mà gần như không có độ trễ

# Phần III: Kết luận

- Kết luận
- Hạn chế của dự án
- Đề xuất hướng phát triển tiếp theo

# Phần 3: Kết Luận Và Đề Xuất

## Kết luận

- Sử dụng **Apache Kafka, Delta lake, PySpark** và **Power BI** để phân tích và xử lý dữ liệu thời gian thực là phương pháp hiệu quả.
- **Gradient Boosted Tree Regression** cho hiệu quả dự đoán nhiệt độ tốt nhất, trong khi sử dụng **Random Forest** để phân loại tình trạng thời tiết cho kết quả hơi thấp do thời tiết có yếu tố rất phức tạp.

# Phần 3: Kết Luận Và Đề Xuất

## Hạn chế của dự án

- **Dữ liệu còn hạn chế:**

- Dữ liệu hiện tại chưa đủ phong phú và đa dạng, cần mở rộng thu thập từ nhiều địa điểm trên toàn cầu để tăng độ chính xác và tổng quát.

- **Yêu cầu hạ tầng tính toán cao:**

- Hệ thống cần tài nguyên tính toán mạnh mẽ để xử lý dữ liệu lớn trong thời gian thực, dẫn đến chi phí triển khai cao khi mở rộng.

# Phần 3: Kết Luận Và Đề Xuất

## Đề xuất hướng phát triển tiếp theo

- **Mở rộng dữ liệu:** Thu thập dữ liệu thời tiết từ nhiều khu vực hơn và làm phong phú dữ liệu để cải thiện độ chính xác.
- **Nâng cấp hạ tầng:** Sử dụng hạ tầng Spark trên đám mây để xử lý lưu lượng lớn và mở rộng quy mô.
- **Tích hợp Deep Learning:** Áp dụng các mô hình AI tiên tiến như LSTM hoặc Transformer cho bài toán chuỗi thời gian.

**Thank you  
very much!**