

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KHOA HỌC VÀ KỸ THUẬT THÔNG TIN



UIT
TRƯỜNG ĐẠI HỌC
CÔNG NGHỆ THÔNG TIN

BÁO CÁO ĐỒ ÁN

Phân tích, thiết kế CSDL cho một bài toán quản lý

Sinh viên thực hiện:

Nguyễn Cao Thắng - 22521329

Võ Anh Quân - 22521192

Chí Nhịt Phú - 22521097

Giảng viên:

TS. Nguyễn Tấn Cầm

Thành phố Hồ Chí Minh, tháng 5 năm 2024

LỜI MỞ ĐẦU

Đồ án này tập trung vào việc phân tích và thiết kế cơ sở dữ liệu (CSDL) cho một bài toán quản lý. Trong thế giới ngày càng số hóa, việc quản lý thông tin một cách hiệu quả thông qua CSDL đã trở thành một yếu tố quan trọng đối với sự thành công của các tổ chức và doanh nghiệp.

CSDL không chỉ giúp tổ chức dữ liệu một cách có hệ thống mà còn cung cấp khả năng truy cập và phân tích dữ liệu một cách nhanh chóng và hiệu quả. Điều này đặc biệt quan trọng trong việc quản lý, nơi mà việc có thể truy cập nhanh vào thông tin chính xác có thể tạo ra sự khác biệt giữa sự thành công và thất bại.

Đồ án này sẽ đi sâu vào việc phân tích yêu cầu của bài toán quản lý, xác định các thực thể, thuộc tính và mối quan hệ trong CSDL. Sau đó, chúng tôi sẽ thiết kế một mô hình CSDL phù hợp, đảm bảo rằng nó có thể hỗ trợ hiệu quả cho các hoạt động quản lý.

Chúng tôi hy vọng rằng, thông qua việc hoàn thành đồ án này, chúng tôi không chỉ có thể cung cấp một giải pháp CSDL hiệu quả cho bài toán quản lý mà còn mở rộng kiến thức và kỹ năng của mình trong lĩnh vực phân tích và thiết kế CSDL.

LỜI CẢM ƠN

Lời đầu tiên, chúng em xin gửi lời cảm ơn chân thành đến giảng viên bộ môn - TS. Nguyễn Tấn Cầm, người đã dìu dắt, truyền đạt kiến thức và định hướng cho em trong suốt quá trình học tập. Nhờ sự quan tâm, chỉ bảo tận tình của thầy, chúng em đã hoàn thành được đồ án này.

Do chưa có nhiều kinh nghiệm làm đề tài cũng như những hạn chế về kiến thức, trong đồ án chắc chắn sẽ không tránh khỏi những thiếu sót. Rất mong nhận được sự nhận xét, ý kiến đóng góp từ phía thầy để đồ án chúng em được hoàn thiện hơn.

Xin trân trọng cảm ơn!

MỤC LỤC

LỜI MỞ ĐẦU	0
LỜI CẢM ƠN.....	1
MỤC LỤC	2
DANH MỤC HÌNH ẢNH	3
Chương 1: Cơ sở lý thuyết.....	4
1.1 Cơ sở dữ liệu	4
1.1.1 Khái niệm.....	4
1.1.2 Ưu nhược điểm của lưu trữ bằng cơ sở dữ liệu.....	4
1.1.3 Những xu hướng công nghệ hiện nay	4
1.2 Thiết kế cơ sở dữ liệu	5
1.2.1 Tổng quan về thiết kế cơ sở dữ liệu	5
1.2.2 Lợi ích của thiết kế cơ sở dữ liệu.....	5
1.2.3 Nguyên tắc thuyết kế CSDL	5
1.2.4 Các bước thiết kế CSDL	6
Chương 2: Phân tích thiết kế CSDL cho phần mềm dự báo thời tiết.....	8
2.1. Tổng quan.....	8
2.2 Mục tiêu	8
2.3 Thực hiện.....	8
2.3.1 Tạo các Table và các khóa ngoại	8
2.3.2 Trigger.....	14
2.3.3 Procedure	17
2.3.4 VIEW	19
2.3.5 Function	19
2.3.6 Phân quyền.....	21
TÀI LIỆU THAM KHẢO.....	26

DANH MỤC HÌNH ẢNH

Hình 1. ER Diagram của CSDL quản lý ứng dụng dự báo thời tiết.....	14
--	----

Chương 1: Cơ sở lý thuyết

1.1 Cơ sở dữ liệu [1]

Trong thời đại ngày nay, cơ sở dữ liệu dần là một phần không thể thiếu trong các phần mềm quản lý, từ việc lưu trữ thông tin cho đến việc quản lý, truy xuất dữ liệu của các công ty hàng đầu

1.1.1 Khái niệm

Cơ sở dữ liệu là một hệ thống các thông tin có cấu trúc và được lưu trữ trên các thiết bị lưu trữ thông tin. Cơ sở dữ liệu có thể giúp người dùng khai thác thông tin được lưu trữ một cách nhanh chóng với nhiều mục đích sử dụng khác nhau.

Với việc các dữ liệu đang ngày càng phát triển thì để đáp ứng được nhu cầu về việc quản lý cũng như truy xuất thông tin một cách nhanh chóng và chính xác thì cơ sở dữ liệu đóng một vai trò quan trọng.

1.1.2 Ưu nhược điểm của lưu trữ bằng cơ sở dữ liệu

Ưu điểm:

Giảm độ trùng lặp thông tin xuống mức thấp nhất và đảm bảo tính nhất quán, toàn vẹn của dữ liệu, cũng như có thể sắp xếp dữ liệu một cách hợp lý

Có thể dùng các công cụ nhất định để có thể truy xuất thông tin nhanh chóng và chính xác như SQL,...

Có thể chia sẻ thông tin đến những người khác cũng như các ứng dụng khác

Nhược điểm:

Bảo mật dữ liệu và những rủi ro về an ninh thông tin

Tranh chấp dữ liệu, tính chủ quyền dữ liệu

1.1.3 Những xu hướng công nghệ hiện nay

Cloud: Ngày càng nhiều tổ chức, công ty, doanh nghiệp lớn hướng đến việc sử dụng cơ sở dữ liệu đám mây vì tính linh hoạt cũng như chi phí và khả năng lưu trữ của chúng. Các tập đoàn lớn như Google (Google Cloud), Microsoft (Microsoft Azure), Amazon (AWS),...

Big Data: với sự phát triển nhanh chóng thì hiện nay số lượng dữ liệu được tạo ra và được lưu trữ đang gia tăng với tốc độ nhanh chóng và đặt ra thử thách cho các cách lưu trữ dữ liệu truyền thống. Vì vậy mà các công nghệ cơ sở dữ liệu với những cách xử lý

lượng dữ liệu hợp lý và lưu trữ một cách tương đối hiệu quả và được những công ty công nghệ lớn sử dụng

Machine Learning và AI: Trong những năm gần đây thì AI và Machine Learning đang trở thành một xu hướng công nghệ được nhiều người quan tâm đến. Việc lưu trữ dữ liệu để có thể được AI tự động hóa các tác vụ cũng như cải thiện và phân tích dữ liệu một cách tự động và nâng cao hiệu quả của dữ liệu

1.2 Thiết kế cơ sở dữ liệu

1.2.1 Tổng quan về thiết kế cơ sở dữ liệu [2]

Thiết kế cơ sở dữ liệu là quá trình tạo ra kế hoạch có cấu trúc để tổ chức, lưu trữ và quản lý dữ liệu nhằm đảm bảo tính toàn vẹn, nhất quán và hiệu quả của dữ liệu. Thiết kế cơ sở dữ liệu là quá trình quan trọng để chọn lọc, phân nhóm và tổ chức dữ liệu trên hệ thống. Công việc chính của quá trình là chọn lọc dữ liệu để lưu trữ và quyết định tính tương quan giữa chúng. Thiết kế cơ sở dữ liệu giúp hệ thống dễ dàng hiểu, quản lý, truy xuất dữ liệu, tìm kiếm thông tin dễ dàng, tiết kiệm được chi phí lưu trữ.

1.2.2 Lợi ích của thiết kế cơ sở dữ liệu [2]

Thực hiện quá trình thiết kế cơ sở dữ liệu sẽ mang lại nhiều lợi ích cho việc quản lý và sử dụng dữ liệu:

- Việc lưu trữ, truy xuất, tìm kiếm và sử dụng dữ liệu trở nên dễ dàng.
- Dễ dàng quản lý và đảm bảo việc bảo mật thông tin. Các dữ liệu được tổ chức logic và có hệ thống giúp việc chia sẻ CSDL dễ dàng hơn.
- Xác định các quy tắc và ràng buộc dữ liệu trong quá trình thiết kế, đảm bảo dữ liệu được duy trì với tính nhất quán, tránh nguy cơ dữ liệu bị trùng lặp và không đồng nhất.

1.2.3 Nguyên tắc thuyết kế CSDL [3]

- Chính xác
- Tránh trùng lặp
- Dễ hiểu
- Chọn đúng thuộc tính và kiểu thuộc tính
- Chọn đúng mối quan hệ

1.2.4 Các bước thiết kế CSDL [2]

i. Xác định mục đích, yêu cầu:

Xác định mục đích, yêu cầu một cách đầy đủ, cụ thể sẽ giúp cho việc thiết kế CSDL trở lên dễ dàng hơn. Thu thập yêu cầu của người sử dụng, các bên liên quan; bao gồm các loại dữ liệu cần được thu thập và cách sử dụng dữ liệu đó. Việc thu thập thông tin có thể tiến hành qua phỏng vấn, khảo sát, thực tập, qua quan sát các nghiệp vụ, quá trình, qua các báo cáo, tài liệu.... Trên cơ sở đó sẽ phân tích yêu cầu để nắm bắt được chức năng cần thiết của cơ sở dữ liệu.

ii. Xây dựng mô hình thực thể-mối kết hợp (Entity-Relationship Diagram - ERD)

ERD là đồ thị biểu diễn các tập thực thể, thuộc tính và mối quan hệ. Các yếu tố cơ bản gồm: Thực thể (Entity); Mối kết hợp (Relationship); Bản số mối kết hợp (Cardinate).

Đỉnh là tên tập thực thể, tên thuộc tính và tên quan hệ.

Cạnh là đường nối giữa: Thực thể - Thuộc tính; Mối quan hệ - Thuộc tính và Thực thể - Mối quan hệ.

- Thực thể:
 - Nhằm mô tả một đối tượng dữ liệu trong hệ thống thông tin.
 - Thuộc tính là những tính chất đặc trưng của loại thực thể cần lưu trữ. Trong các thuộc tính của thực thể, có 1 (hay nhiều) thuộc tính chọn là khóa chính (primary key_PK)
- Mối kết hợp: là sự kết hợp giữa hai hay nhiều loại thực thể, mô tả sự liên quan có thực giữa các thực thể trong 1 ứng dụng. Giữa hai thực thể có thể: không có mối kết hợp nào hoặc có tồn tại nhiều mối kết hợp.
- Bản số: loại mối kết hợp thể hiện liên kết giữa các thực thể nhằm mô tả định lượng mối quan hệ giữa các thực thể. Giá trị định lượng thể hiện 2 giá trị (a,b) tương ứng với bản số tối thiểu (min) và bản số tối đa (max).

Các bước tạo ra ERD

- Bước 1: Xác định loại thực thể, thuộc tính
- Bước 2: Xác định mối kết hợp, thuộc tính
- Bước 3: Xác định bản số.
- Bước 4: Vẽ mô hình ERD

iii. Xây dựng mô hình quan hệ

Khi đã có mô hình thực thể-mối kết hợp (ERD) ta sẽ cần ánh xạ qua mô hình quan hệ để cung cấp các khái niệm cho người dùng một cách gần gũi, thể hiện dữ liệu trong cơ sở dữ liệu theo cách mà hệ quản trị cơ sở dữ liệu (DBMS) có thể hiểu và xử lý được.

- Bước 1: Chuyển đổi các tập thực thể thành các lược đồ quan hệ:
Các tập thực thể chuyển thành các quan hệ có cùng tên và tập thuộc tính
- Bước 2: Chuyển đổi các mối quan hệ
 - + Một - một: thêm vào quan hệ này thuộc tính khóa của quan hệ kia hoặc thêm thuộc tính khóa vào cả 2 quan hệ.
 - + Một - nhiều: Thêm vào quan-hệ-một thuộc tính khóa của quan-hệ-nhiều
 - + Nhiều - nhiều: tạo một quan hệ mới có tên quan hệ là tên của mỗi quan, thuộc tính là những thuộc tính khóa của các tập t
- Bước 3: Chuyển đổi các tập thực thể yếu thành các quan hệ.
 - + Có cùng tên với thực thể yếu
 - + Thêm vào thuộc tính khóa của quan hệ liên quan
- Bước 4: Chuyển đổi thuộc tính đa trị thành một quan hệ.
 - + Có cùng tên với thuộc tính đa trị
 - + Thuộc tính khóa của quan hệ này là khóa ngoài của quan hệ chứa thuộc tính đa trị
- Bước 5: Chuyển đổi mỗi quan hệ đa ngôi thành một quan hệ.
 - + Có cùng tên với tên mỗi liên kết đa ngôi
 - + Khóa chính là tổ hợp các khóa của tập các thực thể tham gia liên kết

iv. Tạo database trong hệ quản trị cơ sở dữ liệu

Thiết kế cơ sở dữ liệu theo mô hình dữ liệu ở mức vật lý (Mô hình dữ liệu ở mức vật lý: mô hình lưu trữ và xử lý bên trong máy tính) trên hệ quản trị cơ sở dữ liệu nhất định qua việc xây dựng các tập tin dữ liệu, các chỉ mục, cấu trúc của chúng, sắp xếp trong các thiết bị lưu giữ. Các bảng trong CSDL quan hệ sẽ quyết định cấu trúc thực tế của các bảng lưu trữ trong mô hình quan hệ.

Chương 2: Phân tích thiết kế CSDL cho phần mềm dự báo thời tiết

2.1. Tổng quan

Ngày nay, với việc thời tiết thay đổi thất thường cũng như những tác động của biến đổi khí hậu thì việc xem dự báo thời tiết để chuẩn bị công việc trở nên cần thiết hơn. Bằng việc thường xuyên thêm vào các dữ liệu cũng như cập nhật các dữ liệu liên quan thì chúng ta có thể dự báo được sơ bộ về thời tiết trong một thời gian cụ thể sắp tới. Người dùng có thể đăng nhập để sử dụng tài khoản của mình. Từ các thông tin được người dùng cung cấp hoặc được phần mềm thu thập thì sẽ tổng hợp và có được dự báo thời tiết sơ bộ về thời gian sắp tới tại vị trí của người dùng và thông báo cho người dùng nếu có thời tiết xấu.

2.2 Mục tiêu

- Có thể dự báo thời tiết và cảnh báo người dùng nếu có thời tiết xấu.
- Lưu trữ và quản lý thông tin về thời tiết các khoảng thời gian trước để người dùng có thể truy xuất.
- Cho phép người dùng ghi chú và phản hồi về phần mềm nếu dự báo đúng hoặc sai để có thể cải thiện dữ liệu và đưa ra những dự báo chính xác hơn
- Phần mềm đơn giản, dễ sử dụng.
- Được cập nhật dữ liệu thường xuyên và đáp ứng được nhu cầu của người dùng.

2.3 Thực hiện

2.3.1 Tạo các Table và các khóa ngoại

Bảng KhuVuc

Tên	Kiểu dữ liệu	Mô tả	Diễn giải
IDKhuVuc	INT	Mã khu vực (khóa chính)	IDKhuVuc
TenKhuVuc	VARCHAR(255)	Tên khu vực	TenKhuVuc
ViDo	DECIMAL(9,6)	Vĩ độ	ViDo
KinhDo	DECIMAL(9,6)	Kinh độ	KinhDo

Bảng thời tiết

Cột	Kiểu dữ liệu	Mô tả	Diễn giải
IDThoiTiet	INT	Mã thời tiết (khóa chính)	Mã duy nhất để xác định mỗi bản ghi dữ liệu thời tiết
IDKhuVuc	INT	Mã khu vực	Mã ngoại lai liên kết với bảng KhuVuc để xác định khu vực thu thập dữ liệu thời tiết
NgayGio	datetime	Ngày giờ	Thời điểm ghi nhận dữ liệu thời tiết
NhietDo	DECIMAL	Nhiệt độ	Giá trị nhiệt độ được đo bằng độ C
DoAm	DECIMAL	Độ ẩm	Tỷ lệ phần trăm hơi nước trong không khí
TocDoGio	DECIMAL	Tốc độ gió	Tốc độ gió được đo bằng m/s
HuongGio	nvarchar(255)	Hướng gió	Hướng mà gió thổi, được mô tả bằng la bàn (ví dụ: Bắc, Đông Bắc, v.v.)
LuongMua	DECIMAL	Lượng mưa	Lượng mưa được đo bằng mm

Bảng dự báo

Cột	Kiểu dữ liệu	Mô tả	Diễn giải
IDDuBao	INT	Mã dự báo (khóa chính)	Mã duy nhất để xác định mỗi bản ghi dự báo thời tiết
IDKhuVuc	INT	Mã khu vực	Mã ngoại lai liên kết với bảng KhuVuc để xác định khu vực áp dụng dự báo
NgayGioDuBao	DATETIME	Ngày giờ dự báo	Thời điểm mà dự báo thời tiết được đưa ra
NhietDoDuBao	DECIMAL	Nhiệt độ dự báo	Giá trị nhiệt độ dự kiến được đo bằng độ C
DoAmDuBao	DECIMAL	Độ ẩm dự báo	Tỷ lệ phần trăm hơi nước trong không khí dự kiến
TocDoGioDuBao	DECIMAL	Tốc độ gió dự báo	Tốc độ gió dự kiến được đo bằng m/s
HuongGioDuBao	VARCHAR(255)	Hướng gió dự báo	Hướng mà gió thổi dự kiến, được mô tả bằng la bàn (ví dụ: Bắc, Đông Bắc, v.v.)
LuongMuaDuBao	DECIMAL	Lượng mưa dự báo	Lượng mưa dự kiến được đo bằng mm

Bảng loại thời tiết

Tên cột	Kiểu dữ liệu	Mô tả	Diễn giải
IDThoiTiet	INT	Mã thời tiết (khóa ngoại)	Liên kết với bảng ThoiTiet, tham chiếu đến một bản ghi dữ liệu thời tiết cụ thể
IDLoaiThoiTiet	INT	Mã loại thời tiết (khóa ngoại)	Liên kết với bảng LoaiThoiTiet, tham chiếu đến một loại thời tiết cụ thể cho bản ghi dữ liệu thời tiết

Bảng ThoiTiet_LoaiThoiTiet

Tên cột	Kiểu dữ liệu	Mô tả	Diễn giải
IDThoiTiet	INT	Mã thời tiết (khóa ngoại)	Liên kết với bảng ThoiTiet, tham chiếu đến một bản ghi dữ liệu thời tiết cụ thể

Bảng NguoiDung

Tên cột	Kiểu dữ liệu	Mô tả	Diễn giải
IDNguoiDung	INT	Mã người dùng (khóa chính)	Mã duy nhất để xác định từng người dùng
TenNguoiDung	VARCHAR(255)	Tên người dùng	Tên của người dùng
Email	VARCHAR(255)	Email	Địa chỉ email của người dùng
SoDienThoai	VARCHAR(15)	Số điện thoại	Số điện thoại của người dùng (tối đa 15 ký tự)

Bảng Login

Tên cột	Kiểu dữ liệu	Mô tả	Diễn giải
IDNguoiDung	INT	Mã người dùng (khóa chính)	Liên kết với bảng NguoiDung, tham chiếu đến một người dùng cụ thể
Password	VARCHAR(255)	Mật khẩu	Mật khẩu được lưu trữ an toàn (khuyến khích sử dụng hàm băm mật khẩu)

Bảng ThietLap

Tên cột	Kiểu dữ liệu	Mô tả	Diễn giải
IDNguoiDung	INT	Mã người dùng (khóa chính)	Liên kết với bảng NguoiDung, tham chiếu đến một người dùng cụ thể
NgonNgu	VARCHAR(255)	Ngôn ngữ	Ngôn ngữ giao diện ưa thích của người dùng
CheDoXem	VARCHAR(255)	Chế độ xem	Chế độ hiển thị thông tin thời tiết ưa thích của người dùng

Bảng QuanSat

Tên cột	Kiểu dữ liệu	Mô tả	Diễn giải
IDQuanSat	INT	Mã quan sát (khóa chính)	Mã duy nhất để xác định mỗi quan sát thời tiết của người dùng
IDNguoiDung	INT	Mã người dùng (khóa ngoại)	Liên kết với bảng NguoiDung, tham chiếu đến người dùng thực hiện quan sát
IDThoiTiet	INT	Mã thời tiết (khóa ngoại)	Liên kết với bảng ThoiTiet, tham chiếu đến bản ghi dữ liệu thời tiết được quan sát
NgayGioQuanSat	DATETIME	Ngày giờ quan sát	Thời điểm người dùng thực hiện quan sát thời tiết
GhiChu	TEXT	Ghi chú	Ghi chú của người dùng về quan sát thời tiết

Bảng Thông Báo

Tên cột	Kiểu dữ liệu	Mô tả	Diễn giải
IDThongBao	INT	Mã thông báo (khóa chính)	Mã duy nhất để xác định mỗi thông báo thời tiết
IDKhuVuc	INT	Mã khu vực (khóa ngoại)	Liên kết với bảng KhuVuc, tham chiếu đến khu vực mà thông báo áp dụng
NgayGioThongBao	DATETIME	Ngày giờ thông báo	Thời điểm ban hành thông báo thời tiết
NoiDungThongBao	TEXT	Nội dung thông báo	Nội dung chi tiết của thông báo thời tiết

Bảng PhanHoiCongDong

Cột	Kiểu dữ liệu	Mô tả	Diễn giải
IDPhanHoi	INT	Mã phản hồi (khóa chính)	Mã duy nhất để xác định mỗi phản hồi cộng đồng về dữ liệu thời tiết
IDNguoiDung	INT	Mã người dùng (khóa ngoại)	Liên kết với bảng NguoiDung, tham chiếu đến người dùng thực hiện phản hồi
IDThoiTiet	INT	Mã thời tiết (khóa ngoại)	Liên kết với bảng ThoiTiet, tham chiếu đến bản ghi dữ liệu thời tiết được phản hồi
NoiDungPhanHoi	TEXT	Nội dung phản hồi	Ý kiến, đánh giá hoặc góp ý của người dùng về dữ liệu thời tiết
ThoiGianPhanHoi	datetime	Thời gian phản hồi	Thời điểm người dùng thực hiện phản hồi

Tạo bảng bằng câu lệnh SQL

```

create database quanlythoitet123
Use QuanLyThoiTiet123
Go
CREATE TABLE KhuVuc (
    IDKhuVuc INT PRIMARY KEY,
    TenKhuVuc VARCHAR(255),
    ViDo DECIMAL(9,6),
    KinhDo DECIMAL(9,6)
);
CREATE TABLE ThoiTiet (
    IDThoiTiet INT PRIMARY KEY,
    IDKhuVuc INT,
    NgayGio DATETIME,
    NhietDo DECIMAL,
    DoAm DECIMAL,
    TocDoGio DECIMAL,
    HuongGio VARCHAR(255),
    LuongMua DECIMAL,
    FOREIGN KEY (IDKhuVuc) REFERENCES KhuVuc(IDKhuVuc)
);

```

```

CREATE TABLE DuBao (
    IDDuBao INT PRIMARY KEY,
    IDKhuVuc INT,
    NgayGioDuBao DATETIME,
    NhietDoDuBao DECIMAL,
    DoAmDuBao DECIMAL,
    TocDoGioDuBao DECIMAL,
    HuongGioDuBao VARCHAR(255),
    LuongMuaDuBao DECIMAL,
    FOREIGN KEY (IDKhuVuc) REFERENCES KhuVuc(IDKhuVuc)
);

CREATE TABLE LoaiThoiTiet (
    IDLoaiThoiTiet INT PRIMARY KEY,
    TenLoaiThoiTiet VARCHAR(255)
);

CREATE TABLE ThoiTiet_LoaiThoiTiet (
    IDThoiTiet INT,
    IDLoaiThoiTiet INT,
    PRIMARY KEY (IDThoiTiet, IDLoaiThoiTiet),
    FOREIGN KEY (IDThoiTiet) REFERENCES ThoiTiet(IDThoiTiet),
    FOREIGN KEY (IDLoaiThoiTiet) REFERENCES LoaiThoiTiet(IDLoaiThoiTiet)
);

CREATE TABLE NguoiDung (
    IDNguoiDung INT PRIMARY KEY,
    TenNguoiDung VARCHAR(255),
    Email VARCHAR(255),
    SoDienThoai VARCHAR(15)
);

CREATE TABLE LogIn (
    IDNguoiDung INT PRIMARY KEY,
    Password VARCHAR(255) ,
    FOREIGN KEY (IDNguoiDung) REFERENCES NguoiDung(IDNguoiDung)
);

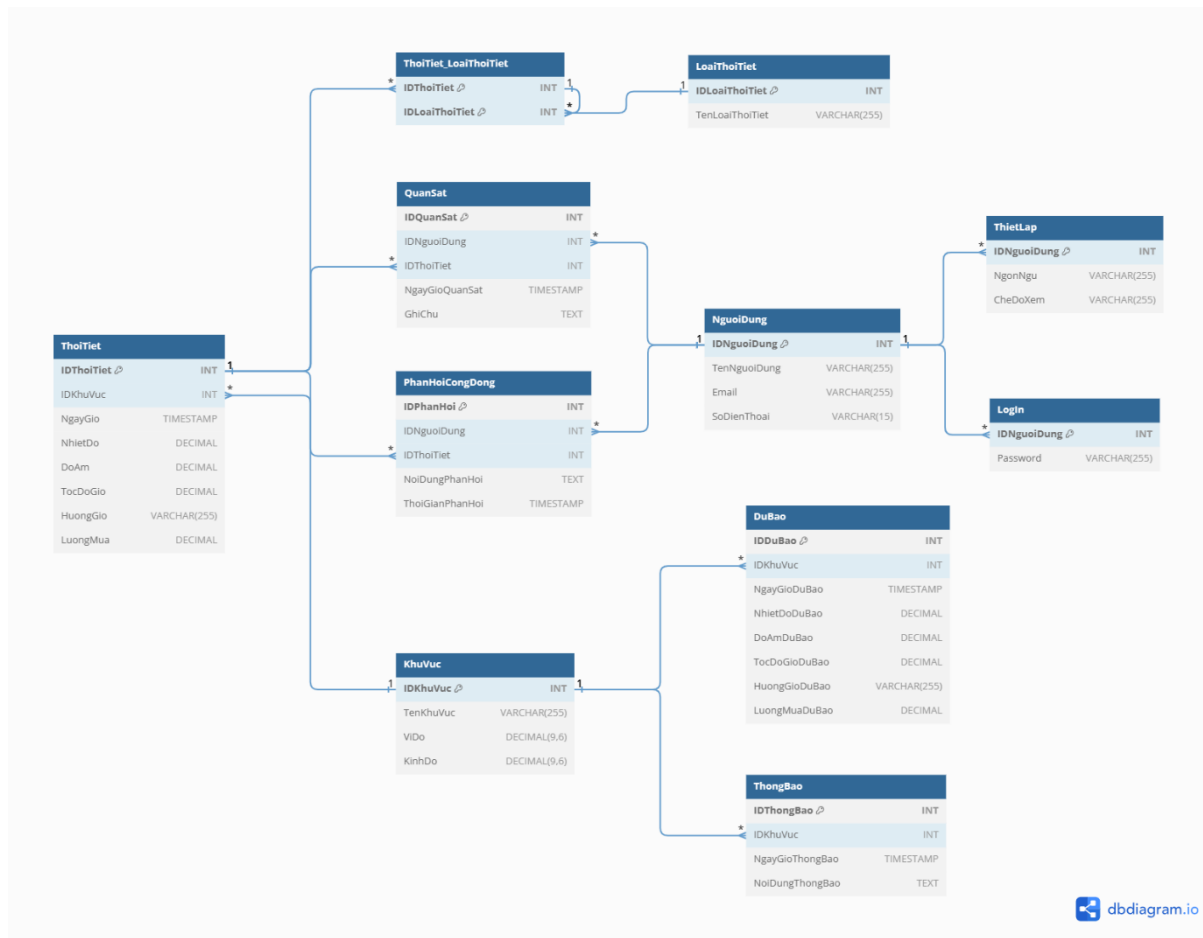
CREATE TABLE ThietLap (
    IDNguoiDung INT PRIMARY KEY,
    NgonNgu VARCHAR(255),
    CheDoXem VARCHAR(255),
    FOREIGN KEY (IDNguoiDung) REFERENCES NguoiDung(IDNguoiDung)

```

```

);
CREATE TABLE QuanSat (
    IDQuanSat INT PRIMARY KEY,
    IDNguoiDung INT,
    IDThoiTiet INT,
    NgayGioQuanSat DATETIME,
    GhiChu TEXT,
    FOREIGN KEY (IDNguoiDung) REFERENCES NguoiDung(IDNguoiDung),
    FOREIGN KEY (IDThoiTiet) REFERENCES ThoiTiet(IDThoiTiet)
);
CREATE TABLE ThongBao (
    IDThongBao INT PRIMARY KEY,
    IDKhuVuc INT,
    NgayGioThongBao DATETIME,
    NoiDungThongBao TEXT,
    FOREIGN KEY (IDKhuVuc) REFERENCES KhuVuc(IDKhuVuc)
);
CREATE TABLE GhiChu (
    IDGhiChu INT PRIMARY KEY,
    IDNguoiDung INT,
    IDThoiTiet INT,
    NgayGioGhiChu DATETIME,
    NoiDungGhiChu TEXT,
    FOREIGN KEY (IDNguoiDung) REFERENCES NguoiDung(IDNguoiDung),
    FOREIGN KEY (IDThoiTiet) REFERENCES ThoiTiet(IDThoiTiet)
);
CREATE TABLE PhanHoiCongDong (
    IDPhanHoi INT PRIMARY KEY,
    IDNguoiDung INT,
    IDThoiTiet INT,
    NoiDungPhanHoi TEXT,
    ThoiGianPhanHoi DATETIME,
    FOREIGN KEY (IDNguoiDung) REFERENCES NguoiDung(IDNguoiDung),
    FOREIGN KEY (IDThoiTiet) REFERENCES ThoiTiet(IDThoiTiet)
);

```



Hình 1. ER Diagram của CSDL quản lý ứng dụng dự báo thời tiết

2.3.2 Trigger

STT	Tên trigger	Bảng ảnh hưởng	Sự kiện	Nội dung
1	IDnguoidung	Nguoidung	insert, update	Nếu ID đã tồn tại thì yêu cầu nhập ID mới
2	Khuvukinhdovido	Khuvuc	insert, update	Các khu vực phải có kinh độ hoặc vĩ độ khác nhau
3	Checktemperature	Thoitiet	Insert	Kiểm tra khi thêm mới dữ liệu vào bảng thì nhiệt độ phải nằm trong khoảng hợp lý
4	CheckForecastTemperature	DuBao	Insert	khi thêm mới dữ liệu vào bảng DuBao, nhiệt độ dự báo cũng phải nằm trong khoảng hợp lý
5	CheckAreaBeforeDelete	KhuVuc	Delete	khi xóa dữ liệu từ bảng KhuVuc, không cho

				phép xóa nếu khu vực đang có dữ liệu trong bảng ThoiTiet hoặc DuBao:
6	CheckSettingsBeforeDelete	KhuVuc	Delete	khi xóa dữ liệu từ bảng ThietLap, không cho phép xóa nếu người dùng đang có dữ liệu trong bảng Login
7	CheckWeatherTypeBeforeDelete	Loaithoitiet	Delete	kiểm tra khi xóa dữ liệu từ bảng LoaiThoiTiet, không cho phép xóa nếu loại thời tiết đang có dữ liệu trong bảng ThoiTiet_LoaiThoiTiet

a) IDnguoidung

Nội dung: nếu ID đã tồn tại thì yêu cầu nhập ID mới

```
create trigger idnguoidung
on nguoidung
for insert,update
as
begin
    declare @id int
    select @id=IDNguoiDung from inserted
    if exists (select * from NguoiDung where @id=IDNguoiDung)
        begin
            print'id nguoi dung da ton tai, nhap lai'
            rollback transaction
        end
    else
        begin
            print'them nguoi dung thanh cong';
        end
end
```

b) khuvuckinhdovido

Nội dung: các khu vực có kinh độ vĩ độ khác nhau

```
create trigger khuvuckdvd
on khuvuc
for insert,update
as
begin
    declare @kinhdo decimal(9,6),@vido decimal(9,6)
    select @kinhdo=KinhDo, @vido=ViDo from inserted
    if exists(select * from KhuVuc where @kinhdo=KinhDo and @vido=ViDo)
        begin
            print'vi tri nay da ton tai khu vuc khac'
            rollback transaction
        end
    else
        begin
            print'thanh cong';
        end
end
End
```

c) Checktemperature

Nội dung: kiểm tra khi thêm mới dữ liệu vào bảng ThoiTiet, nhiệt độ phải nằm trong khoảng hợp lý

```
CREATE TRIGGER CheckTemperature
ON ThoiTiet
FOR INSERT
AS
BEGIN
    IF EXISTS (SELECT * FROM inserted WHERE NhietDo < -20 OR NhietDo > 50)
    BEGIN
        RAISERROR ('Nhiệt độ phải nằm trong khoảng từ -20 đến 50 độ C.', 16, 1);
        ROLLBACK TRANSACTION;
        RETURN;
    END
    ELSE BEGIN
        PRINT 'Thêm dữ liệu thành công'
    END
END;
```

d) CheckForecastTemperature

Nội dung: khi thêm mới dữ liệu vào bảng DuBao, nhiệt độ dự báo cũng phải nằm trong khoảng hợp lý

```
CREATE TRIGGER CheckForecastTemperature
ON DuBao
FOR INSERT
AS
BEGIN
    IF EXISTS (SELECT * FROM inserted WHERE NhietDoDuBao < -20 OR NhietDoDuBao > 50)
    BEGIN
        RAISERROR ('Nhiệt độ dự báo phải nằm trong khoảng từ -20 đến 50 độ C.', 16,
1);
        ROLLBACK TRANSACTION;
        RETURN;
    END
    ELSE BEGIN
        PRINT 'Thêm dữ liệu thành công'
    END
END;
```

e) CheckAreaBeforeDelete

Nội dung: khi xóa dữ liệu từ bảng KhuVuc, không cho phép xóa nếu khu vực đang có dữ liệu trong bảng ThoiTiet hoặc DuBao

```
CREATE TRIGGER CheckAreaBeforeDelete
ON KhuVuc
FOR DELETE
AS
BEGIN
    IF EXISTS (SELECT * FROM deleted JOIN ThoiTiet ON deleted.IDKhuVuc =
ThoiTiet.IDKhuVuc) OR EXISTS (SELECT * FROM deleted JOIN DuBao ON deleted.IDKhuVuc =
DuBao.IDKhuVuc)
    BEGIN
        RAISERROR ('Không thể xóa khu vực này vì nó đang có dữ liệu trong bảng
ThoiTiet hoặc DuBao.', 16, 1);
        ROLLBACK TRANSACTION;
    END
END;
```

```

        RETURN;
    END
END;

```

f) CheckSettingsBeforeDelete

Nội dung: khi xóa dữ liệu từ bảng ThietLap, không cho phép xóa nếu người dùng đang có dữ liệu trong bảng LogIn

```

CREATE TRIGGER CheckSettingsBeforeDelete
ON ThietLap
FOR DELETE
AS
BEGIN
    IF EXISTS (SELECT * FROM deleted JOIN LogIn ON deleted.IDNguoiDung =
LogIn.IDNguoiDung)
    BEGIN
        RAISERROR ('Không thể xóa thiết lập này vì người dùng đang có dữ liệu trong
bảng LogIn.', 16, 1);
        ROLLBACK TRANSACTION;
        RETURN;
    END
END;

```

g) CheckWeatherTypeBeforeDelete

Nội dung: kiểm tra khi xóa dữ liệu từ bảng LoaiThoiTiet, không cho phép xóa nếu loại thời tiết đang có dữ liệu trong bảng ThoiTiet_LoaiThoiTiet

```

CREATE TRIGGER CheckWeatherTypeBeforeDelete
ON LoaiThoiTiet
FOR DELETE
AS
BEGIN
    IF EXISTS (SELECT * FROM deleted JOIN ThoiTiet_LoaiThoiTiet ON
deleted.IDLoaiThoiTiet = ThoiTiet_LoaiThoiTiet.IDLoaiThoiTiet)
    BEGIN
        RAISERROR ('Không thể xóa loại thời tiết này vì nó đang có dữ liệu trong bảng
ThoiTiet_LoaiThoiTiet.', 16, 1);
        ROLLBACK TRANSACTION;
        RETURN;
    END
END;

```

2.3.3 Procedure

STT	Tên Procedure	Nội dung
1	Thoietietkhuvuc	Procedure đưa vào ID khu vực, ngày tháng xuất ra dự báo thời tiết khu vực đó
2	GetCommunityResponses	Procedure này nhận vào tên người dùng và trả về các phản hồi cộng đồng của họ.
3	GetUserObservations	Procedure để lấy thông tin chi tiết về các quan sát của một người dùng cụ thể

a) Procedure thời tiết khu vực

Nội dung: Procedure đưa vào ID khu vực, ngày tháng xuất ra dự báo thời tiết khu vực đó

```
create procedure thoitietkhuvuc (
    @idkhuvuc int,
    @ngay timestamp,
    @dubao varchar(255) output
)
as
begin
    select @dubao = TenLoaiThoiTiet
    from LoaiThoiTiet,ThoiTiet,ThoiTiet_LoaiThoiTiet
    where ThoiTiet.IDThoiTiet=ThoiTiet_LoaiThoiTiet.IDThoiTiet and
    ThoiTiet_LoaiThoiTiet.IDLoaiThoiTiet=LoaiThoiTiet.IDLoaiThoiTiet
    and @idkhuvuc=IDKhuVuc and NgayGio=@ngay
    if(@dubao is null)
    set @dubao='chua co du lieu'
    return @dubao;
end
```

b) Procedure GetCommunityResponses

Nội dung: Procedure này nhận vào tên người dùng và trả về các phản hồi cộng đồng của họ.

```
CREATE PROCEDURE GetCommunityResponses
    @TenNguoiDung NVARCHAR(255)
AS
BEGIN
    IF EXISTS (SELECT 1 FROM PhanHoiCongDong JOIN NguoiDung ON
    PhanHoiCongDong.IDNguoiDung = NguoiDung.IDNguoiDung WHERE NguoiDung.TenNguoiDung =
    @TenNguoiDung)
    BEGIN
        SELECT PhanHoiCongDong.NoiDungPhanHoi, PhanHoiCongDong.ThoiGianPhanHoi
        FROM PhanHoiCongDong
        JOIN NguoiDung ON PhanHoiCongDong.IDNguoiDung = NguoiDung.IDNguoiDung
        WHERE NguoiDung.TenNguoiDung = @TenNguoiDung;
    END
    ELSE
    BEGIN
        SELECT 'Không có dữ liệu' AS ThongBao;
    END
END;
EXEC GetCommunityResponses @TenNguoiDung = N'Tên người dùng';
```

c) Procedure GetUserObservations

Nội dung: Procedure để lấy thông tin chi tiết về các quan sát của một người dùng cụ thể:

```
CREATE PROCEDURE GetUserObservations
    @TenNguoiDung NVARCHAR(255)
AS
BEGIN
    BEGIN
        IF EXISTS (SELECT 1 FROM QuanSat JOIN NguoiDung ON QuanSat.IDNguoiDung =
    NguoiDung.IDNguoiDung WHERE NguoiDung.TenNguoiDung = @TenNguoiDung)
```

```

BEGIN
    SELECT QuanSat.*
    FROM QuanSat
    JOIN NguoiDung ON QuanSat.IDNguoiDung = NguoiDung.IDNguoiDung
    WHERE NguoiDung.TenNguoiDung = @TenNguoiDung;
END
ELSE
BEGIN
    SELECT 'Không có dữ liệu' AS ThôngBao;
END
END
END;

```

2.3.4 VIEW

STT	Tên	Nội dung
1	ThoiTiet_KhuVuc	hiển thị thông tin về thời tiết hiện tại của khu vực
2	DuBao_ThoiTiet_Hour	hiển thị thông tin dự báo thời tiết của khu vực theo từng giờ
3	DuBao_ThoiTiet_Day	hiển thị thông tin dự báo thời tiết của khu vực theo từng ngày

a) ThoiTiet_KhuVuc

```

CREATE VIEW ThoiTiet_KhuVuc as
SELECT TT.IDThoiTiet, KV.IDKhuVuc, KV.TenKhuVuc, NgayGio, NhiệtĐo, DoAm, TocDoGio,
HuongGio, LuongMua, TenLoaiThoiTiet
FROM ThoiTiet as TT INNER JOIN KhuVuc as KV on TT.IDKhuVuc = KV.IDKhuVuc
INNER JOIN ThoiTiet_LoaiThoiTiet as TT_LTT on
TT.IDThoiTiet = TT_LTT.IDThoiTiet
INNER JOIN LoaiThoiTiet as LTT on
TT_LTT.IDLoaiThoiTiet = LTT.IDLoaiThoiTiet
WHERE KV.TenKhuVuc = '...' and NgayGio = CURRENT_datetime

```

b) DuBao_ThoiTiet_Hour

```

CREATE VIEW DuBao_ThoiTiet_Hour as
SELECT DB.IDDuBao, KV.IDKhuVuc, KV.TenKhuVuc, DATEPART(hour, NgayGioDuBao) as
GioDuBao, NhiệtĐoDuBao, DoAmDuBao, TocDoGioDuBao, HuongGioDuBao, LuongMuaDuBao,
NoiDungThôngBao
FROM DuBao as DB INNER JOIN KhuVuc as KV on DB.IDKhuVuc = KV.IDKhuVuc
INNER JOIN ThôngBao as TB on KV.IDKhuVuc = TB.IDKhuVuc
WHERE KV.TenKhuVuc = '...' and NgayGioDuBao > CURRENT_datetime

```

c) DuBao_ThoiTiet_Day

```

CREATE VIEW DuBao_ThoiTiet_Day as
SELECT DB.IDDuBao, KV.IDKhuVuc, KV.TenKhuVuc, DATEPART(day, NgayGioDuBao) as
NgayDuBao, NhiệtĐoDuBao, DoAmDuBao, TocDoGioDuBao, HuongGioDuBao, LuongMuaDuBao,
NoiDungThôngBao
FROM DuBao as DB INNER JOIN KhuVuc as KV on DB.IDKhuVuc = KV.IDKhuVuc
INNER JOIN ThôngBao as TB on KV.IDKhuVuc = TB.IDKhuVuc
WHERE KV.TenKhuVuc = '...' and NgayGioDuBao > CURRENT_datetime
SELECT * FROM DuBao_ThoiTiet_Day

```

2.3.5 Function

STT	Tên	Nội Dung
1	GetAverageWeatherAndForecast	Function để tính trung bình nhiệt độ, độ ẩm của một khu vực từ dữ liệu của bảng THOITTIET và DUBAO

2	GetPopularWeatherType	Function để xuất ra tên kiểu thời tiết phổ biến nhất tại một khu vực
---	-----------------------	--

a) Function GetAverageWeatherAndForecast

Nội dung: function để tính trung bình nhiệt độ, độ ẩm của một khu vực từ dữ liệu của bảng THOITIET và DUBAO

```
CREATE FUNCTION GetAverageWeatherAndForecast(@TenKhuVuc NVARCHAR(255))
RETURNS @Result TABLE
(
    Source NVARCHAR(255),
    AvgTemperature DECIMAL(18, 2),
    AvgHumidity DECIMAL(18, 2)
)
AS
BEGIN
    DECLARE @IDKhuVuc INT;
    SELECT @IDKhuVuc = IDKhuVuc FROM KhuVuc WHERE TenKhuVuc = @TenKhuVuc;

    DECLARE @AvgTempThoiTiet DECIMAL(18, 2), @AvgHumidityThoiTiet DECIMAL(18, 2);
    DECLARE @AvgTempDuBao DECIMAL(18, 2), @AvgHumidityDuBao DECIMAL(18, 2);

    SELECT @AvgTempThoiTiet = AVG(NhietDo), @AvgHumidityThoiTiet = AVG(DoAm)
    FROM ThoiTiet
    WHERE IDKhuVuc = @IDKhuVuc;

    SELECT @AvgTempDuBao = AVG(NhietDoDuBao), @AvgHumidityDuBao = AVG(DoAmDuBao)
    FROM DuBao
    WHERE IDKhuVuc = @IDKhuVuc;

    INSERT INTO @Result
    VALUES ('Trung bình nhiệt độ, độ ẩm:', @AvgTempThoiTiet, @AvgHumidityThoiTiet),
           ('Trung bình nhiệt độ, độ ẩm dự báo:', @AvgTempDuBao, @AvgHumidityDuBao);

    RETURN;
END;
SELECT * FROM GetAverageWeatherAndForecast(N'Tên khu vực');
```

b) GetPopularWeatherType

Nội dung: function để xuất ra tên kiểu thời tiết phổ biến nhất tại một khu vực

```
CREATE FUNCTION GetPopularWeatherType(@TenKhuVuc NVARCHAR(255))
RETURNS NVARCHAR(255)
AS
BEGIN
    DECLARE @IDKhuVuc INT, @TenLoaiThoiTiet NVARCHAR(255);

    SELECT @IDKhuVuc = IDKhuVuc FROM KhuVuc WHERE TenKhuVuc = @TenKhuVuc;

    SELECT TOP 1 @TenLoaiThoiTiet = LoaiThoiTiet.TenLoaiThoiTiet
    FROM ThoiTiet
    JOIN ThoiTiet_LoaiThoiTiet ON ThoiTiet.IDThoiTiet =
    ThoiTiet_LoaiThoiTiet.IDThoiTiet
    JOIN LoaiThoiTiet ON ThoiTiet_LoaiThoiTiet.IDLoaiThoiTiet =
    LoaiThoiTiet.IDLoaiThoiTiet
    WHERE ThoiTiet.IDKhuVuc = @IDKhuVuc
    GROUP BY LoaiThoiTiet.TenLoaiThoiTiet
    ORDER BY COUNT(*) DESC;
```

```

RETURN CONCAT(@TenKhuVuc, ' hay ', @TenLoaiThoiTiet, '.');
END;
SELECT dbo.GetPopularWeatherType(N'Tên khu vực') AS PopularWeatherType;

```

2.3.6 Phân quyền

1. Tạo role

Tạo các role để định nghĩa các nhóm người dùng với các quyền hạn cụ thể, trong trường hợp này, có hai loại role chính: admin_role (dành cho admin), user_role (dành cho người dùng).

```

-- Create role for admin
CREATE ROLE admin_role;
-- Create role for user
CREATE ROLE user_role;

```

2. Tạo user

Tạo các người dùng admin@; user@ và phân nhóm người dùng

```

--          Tạo          các          người          dùng          admin@;          user@
CREATE LOGIN Login1 WITH PASSWORD = 'admin@123'
CREATE USER admin@ FOR LOGIN Login1

CREATE LOGIN Login2 WITH PASSWORD = 'user@123'
CREATE USER user@ FOR LOGIN Login2

-- Tạo nhóm: admin@ thuộc admin_role; user@ thuộc user_role
EXEC sp_addrolemember admin_role, admin@
EXEC sp_addrolemember user_role, user@

```

3. Xác định quyền hạn cho từng role

a) Role admin

- Danh sách các bảng được cấp quyền trực tiếp của role

STT	Tên bảng	Quyền được cấp
1	KhuVuc	SELECT, INSERT, UPDATE, DELETE
2	ThoiTiet	SELECT, INSERT, UPDATE, DELETE
3	DuBao	SELECT, INSERT, UPDATE, DELETE
4	LoaiThoiTiet	SELECT, INSERT, UPDATE, DELETE
5	ThoiTiet_LoaiThoiTiet	SELECT, INSERT, UPDATE, DELETE

6	NguoiDung	SELECT, INSERT, UPDATE, DELETE
7	ThietLap	SELECT, INSERT, UPDATE, DELETE
8	QuanSat	SELECT, INSERT, UPDATE, DELETE
9	ThongBao	SELECT, INSERT, UPDATE, DELETE
10	PhanHoiCongDong	SELECT, INSERT, UPDATE, DELETE
11	Login	SELECT, INSERT, UPDATE, DELETE

- Danh sách các stored procedure được cấp quyền thực hiện của role:

STT	Tên	Quyền được cấp
1	thoitiethuvuc	EXECUTE
2	GetCommunityResponses	EXECUTE
3	GetUserObservations	EXECUTE

- Câu lệnh SQL:

```
-- Tables
GRANT SELECT, INSERT, UPDATE, DELETE ON KhuVuc TO admin_role
GRANT SELECT, INSERT, UPDATE, DELETE ON ThoiTiet TO admin_role
GRANT SELECT, INSERT, UPDATE, DELETE ON DuBao TO admin_role
GRANT SELECT, INSERT, UPDATE, DELETE ON LoaiThoiTiet TO admin_role
GRANT SELECT, INSERT, UPDATE, DELETE ON ThoiTiet_LoaiThoiTiet TO admin_role
GRANT SELECT, INSERT, UPDATE, DELETE ON NguoiDung TO admin_role
GRANT SELECT, INSERT, UPDATE, DELETE ON ThietLap TO admin_role
GRANT SELECT, INSERT, UPDATE, DELETE ON QuanSat TO admin_role
GRANT SELECT, INSERT, UPDATE, DELETE ON ThongBao TO admin_role
GRANT SELECT, INSERT, UPDATE, DELETE ON PhanHoiCongDong TO admin_role
GRANT SELECT, INSERT, UPDATE, DELETE ON Login TO admin_role

-- Stored procedure
GRANT EXECUTE ON thoitiethuvuc TO admin_role WITH GRANT OPTION
GRANT EXECUTE ON GetCommunityResponses TO admin_role WITH GRANT OPTION
GRANT EXECUTE ON GetUserObservations TO admin_role WITH GRANT OPTION
```

b) Role user

- Danh sách các bảng được cấp quyền trực tiếp của role

STT	Tên bảng	Quyền được cấp
1	KhuVuc	SELECT
2	ThoiTiet	SELECT
3	DuBao	SELECT
4	LoaiThoiTiet	SELECT
5	ThoiTiet_LoaiThoiTiet	SELECT
6	NguoiDung	SELECT
7	ThietLap	SELECT
8	QuanSat	INSERT, UPDATE
9	ThongBao	SELECT
10	PhanHoiCongDong	SELECT
11	Login	SELECT

- Danh sách các stored procedure được cấp quyền thực hiện của role:

STT	Tên	Quyền được cấp
1	thoietietkhuvuc	EXECUTE
2	GetCommunityResponses	EXECUTE
3	GetUserObservations	EXECUTE

- Câu lệnh SQL:

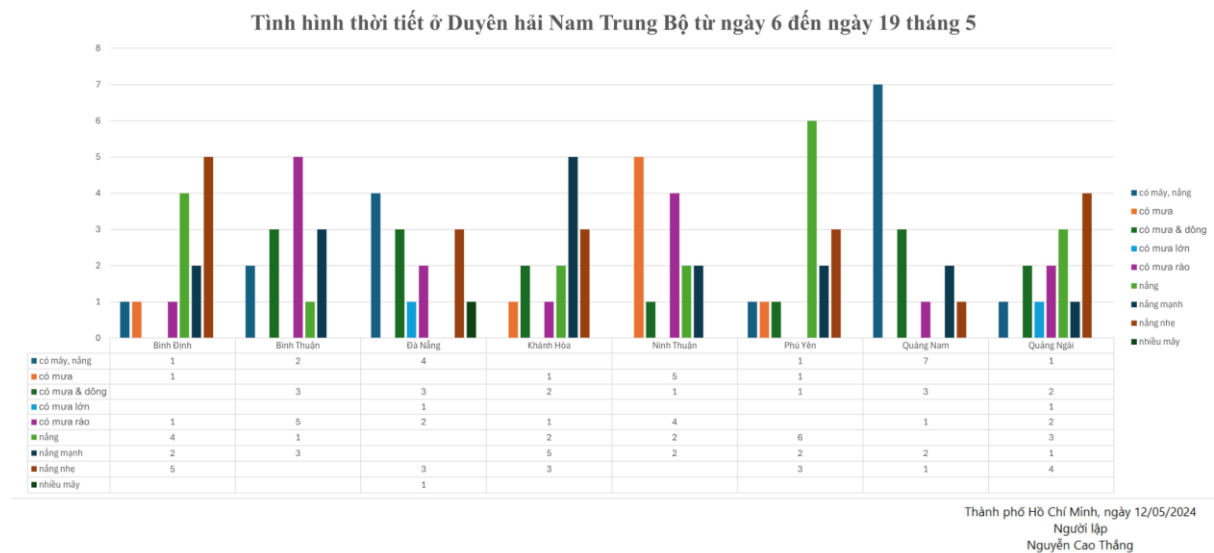
```
-- Tables:
GRANT INSERT, UPDATE ON QuanSat TO user_role
GRANT SELECT ON KhuVuc TO user_role
GRANT SELECT ON ThoiTiet TO user_role
GRANT SELECT ON LoaiThoiTiet TO user_role
GRANT SELECT ON ThoiTiet_LoaiThoiTiet TO user_role
GRANT SELECT ON NguoiDung TO user_role
GRANT SELECT ON ThietLap TO user_role
GRANT SELECT ON QuanSat TO user_role
GRANT SELECT ON ThongBao TO user_role
GRANT SELECT ON PhanHoiCongDong TO user_role
GRANT SELECT ON Login TO user_role

-- Stored procedure
```

```
GRANT EXECUTE ON thoitietkhuvuc TO user_role
GRANT EXECUTE ON GetCommunityResponses TO user_role
GRANT EXECUTE ON GetUserObservations TO user_role
```

```
-- VIEW để ẩn mật khẩu người dùng
CREATE VIEW Nguoidung_View as
SELECT NguoiDung.IDNguoiDung, TenNguoiDung, SoDienThoai, Email
FROM NguoiDung, Login
WHERE NguoiDung.IDNguoiDung = Login.IDNguoiDung
-- Cấp quyền truy cập cho VIEW thay vì cấp quyền truy cập trực tiếp vào bảng
REVOKE SELECT ON NguoiDung FROM admin_role
REVOKE SELECT ON Login FROM admin_role
GRANT SELECT ON Nguoidung_View TO admin_role
```

2.3.7 Report



Hình 2 Tình hình thời tiết ở Duyên hải Nam Trung Bộ

Report trên bao gồm một bảng và một biểu đồ thống kê về tình hình thời tiết ở Duyên hải Nam Trung Bộ từ ngày 12 đến ngày 19 tháng 5 năm 2024. Biểu đồ được chia thành 7 miền, mỗi miền đại diện cho một tỉnh thành phố: Bình Định, Bình Thuận, Đà Nẵng, Khánh Hòa, Ninh Thuận, Phước Yên, và Quảng Ngãi.

Mỗi vùng được chia thành 8 cột, mỗi cột đại diện cho tần suất xuất hiện một kiểu thời tiết trong khoảng thời gian từ ngày 6 đến ngày 19 tháng 5. Report này thể hiện kiểu thời tiết nào chiếm ưu thế trong khoảng thời gian 2 tuần trên, giúp phân tích tổng quát về tình hình thời tiết khu vực (đang trong mùa khô, nắng nhiều,...)

Chương 3: Kết Luận

1) Kết quả đạt được

- Xác định được rõ ràng và giải quyết các yêu cầu đặt ra của bài toán liên quan đến cơ sở dữ liệu
- Xây dựng và thiết kế được cơ bản về cơ sở dữ liệu cho bài toán đặt ra
- Ứng dụng các lý thuyết đã được học để hoàn thành bài toán đặt ra
- Phân chia thời gian và công việc trong nhóm một cách hiệu quả

2) Khó khăn gặp phải

- Thiếu kinh nghiệm trong việc xây dựng phần mềm và dẫn đến những sai sót trong bài làm và cũng chưa đưa ra được phần mềm
- Các procedure hay các trigger cần được cải thiện để đảm bảo tính hiệu quả
- Thiết kế vẫn đang ở mức độ cơ bản

3) Hướng phát triển tiếp theo

- Tiếp tục cải thiện các chức năng hay tích hợp thêm các tính năng mới để cải thiện hiệu suất và tối ưu hóa sản phẩm
- Cải thiện kỹ năng cũng như kiến thức để đưa ra được sản phẩm hoàn chỉnh nhất

TÀI LIỆU THAM KHẢO

- [1] T. B. Trần, “Bài giảng môn Cơ sở dữ liệu”, Trường Đại học Công nghệ Thông tin, Khoa Hệ thống thông tin, 11 Tháng Chín 2023.
- [2] “Database Design là gì? Tổng quan khái niệm, lợi ích và các bước thực hiện”. Truy cập: 21 Tháng Năm 2024. [Online]. Available at: <https://som.edu.vn/database-design-la-gi-khai-niem-loi-ich-cach-trien-khai/>
- [3] “Làm thế nào để thiết kế một cơ sở dữ liệu [Phần 1]”. Truy cập: 21 Tháng Năm 2024. [Online]. Available at: <https://viblo.asia/p/lam-the-nao-de-thiet-ke-mot-co-so-du-lieu-phan-1-rYvGwavgKVw>